ON THE TRADEOFF BETWEEN PRIVACY AND UTILITY IN GENOMIC STUDIES: DIFFERENTIAL PRIVACY UNDER DEPENDENT TUPLES

A DISSERTATION SUBMITTED TO

THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR

THE DEGREE OF

DOCTOR OF PHILOSOPHY

IN

COMPUTER ENGINEERING

By

Nour M. N. Alserr

August 2020

ON THE TRADEOFF BETWEEN PRIVACY AND UTILITY IN GENOMIC STUDIES: DIFFERENTIAL PRIVACY UNDER DEPENDENT TUPLES

By Nour M. N. Alserr

August 2020

We certify that we have read this dissertation and that in our opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of Doctor of Philosophy.

Özgür Ulusoy(Advisor)

Erman Ayday(Co-Advisor)

Abdullah Ercüment Çiçek

Can Alkan

Murat Cenk

Ali Aydın Selçuk

Approved for the Graduate School of Engineering and Science:

Ezhan Karaşan
Director of the Graduate School

ABSTRACT

ON THE TRADEOFF BETWEEN PRIVACY AND UTILITY IN GENOMIC STUDIES: DIFFERENTIAL PRIVACY UNDER DEPENDENT TUPLES

Nour M. N. Alserr
Ph.D. in Computer Engineering
Advisor: Özgür Ulusoy
Co-Advisor: Erman Ayday
August 2020

The rapid progress in genome sequencing and the decrease in the sequencing costs have led to the high availability of genomic data. Studying these data can greatly help answer the key questions about disease associations and our evolution. However, due to growing privacy concerns about the sensitive information of participants, accessing key results and data of genomic studies (such as genome-wide association studies - GWAS) is restricted to only trusted individuals. On the other hand, paving the way to biomedical breakthroughs and discoveries requires granting open access to genomic datasets. Privacy-preserving mechanisms can be a solution for granting wider access to such data while protecting their owners. In particular, there has been growing interest in applying the concept of differential privacy (DP) while sharing summary statistics about genomic data. DP provides a mathematically rigorous approach to prevent the risk of membership inference while sharing statistical information about a dataset. However, DP has a known drawback as it does not take into account the correlation between dataset tuples, which is a common situation for genomic datasets due to the inherent correlations between the genomes of family members. This may degrade the privacy guarantees offered by the DP. In this Thesis, focusing on static and dynamic genomic datasets, we show this drawback of the DP and we propose techniques to mitigate it. First, using a real-world genomic dataset, we demonstrate the feasibility of an attribute inference attack on differentially private query results by utilizing the correlations between the entries in the dataset. We show the privacy loss in count, minor allele frequency (MAF), and chi-square queries. The results explain the scale of vulnerability when we have dependent tuples in the dataset. Our results demonstrate that the adversary can infer sensitive genomic data about a user from the differentially private results of a sum query by exploiting the correlations between the genomes of family members. Our results also show that

using the results of differentially-private MAF queries on static and dynamic genomic datasets and utilizing the dependency between tuples, an adversary can reveal up to 50% more sensitive information about the genome of a target (compared to original privacy guarantees of standard DP-based mechanisms), while differentially-privacy chi-square queries can reveal up to 40% more sensitive information. Furthermore, we show that the adversary can use the inferred genomic data obtained from the attribute inference attack to infer the membership of a target in another genomic dataset (e.g., associated with a sensitive trait). Using a log-likelihood-ratio (LLR) test, our results also show that the inference power of the adversary can be significantly high in such an attack even by using inferred (and hence partially incorrect) genomes. Finally, we propose a mechanism for privacy-preserving sharing of statistics from genomic datasets to attain privacy guarantees while taking into consideration the dependence between tuples. By evaluating our mechanism on different genomic datasets, we empirically demonstrate that our proposed mechanism can achieve up to 50% better privacy than traditional DP-based solutions.

Keywords: Genomic datasets, differential privacy, inference attacks.

ÖZET

GENOMİK ÇALIŞMALARDA GİZLİLİK VE VERİNİN İŞE YARARLILIĞI ÜZERİNE ANALİZ: BAĞIMLI ELEMANLAR ALTINDA DİFERANSİYEL GİZLİLİK

Nour M. N. Alserr
Bilgisayar Mühendisliği, Doktora
Tez Danışmanı: Özgür Ulusoy
İkinci Tez Danışmanı: Erman Ayday
Ağustos 2020

Genom dizilemesindeki hızlı ilerleme ve dizilim maliyetlerindeki azalma, genomik verilerin yüksek kullanılabilirliğine yol açmıştır. Bu verilerin incelenmesi, hastalık çağrışımları ve evrimimiz hakkındaki kilit soruların cevaplanmasına büyük ölçüde yardımcı olabilir. Bununla birlikte, katılımcıların hassas bilgileri hakkındaki artan gizlilik endişeleri nedeniyle, önemli sonuçlara ve genomik çalışmaların verilerine (genom çapında ilişki çalışmaları - GWAS gibi) erişmek yalnızca güvenilir kişilerle sınırlıdır. Öte yandan, biyomedikal atılımlara ve keşiflere giden yolu açmak, genomik veri kümelerine açık erişim verilmesini gerektirir. Gizlilik koruma mekanizmaları, kişilerin genetik verilerinin gizliliğini korurken bu tür verilere daha geniş erişim sağlamak için bir çözüm olabilir. Özellikle, genomik veriler hakkındaki özet istatistikleri paylaşırken, diferansiyel gizlilik (DP) kavramının uygulanmasına yönelik ilgi artmaktadır. DP, veri seti hakkında istatistiksel bilgiler paylaşırken veri bankasına üyelik çıkarım riskini önlemek için matematiksel bir yaklaşım sağlar. Öte yandan, DP, genomik veri kümeleri için ortak bir durum olan veri seti elemanları arasındaki korelasyonu (aile üyelerinin genomları arasındaki doğal korelasyonlar) dikkate almadığı için bilinen bir dezavantaja sahiptir. Bu DP'nin sunduğu gizlilik garantilerini bozabilir. Statik ve dinamik genomik veri kümelerine odaklanan bu tezde DP'nin bu dezavantajını gösteriyor ve onu hafifletmek için teknikler öneriyoruz. İlk olarak, gerçek hayattaki bir genomik veri kümesi kullanarak, veri kümesindeki girdiler arasındaki korelasyonları kullanarak, farklı özel sorgu sonuçlarına bir özellik çıkarsama saldırısının fizibilitesini gösteriyoruz. Sayıda gizlilik kaybını, küçük alel frekansını (MAF) ve ki-kare sorgularını gösteriyoruz. Sonuçlar, veri kümesinde birbirine bağımlı elemanlar olduğunda güvenlik açığı ölçeğini gösteriyor. Elde ettiğimiz sonuçlar, saldırganın, kullanıcı üyelerinin genomları

arasındaki korelasyonları kullanarak bir toplam sorgusunun farklı sonuçlarından bir kullanıcı hakkındaki hassas genomik verileri çıkarabildiğini göstermektedir. Sonuçlarımız ayrıca statik ve dinamik genomik veri kümelerinde farklı MAF sorgularının sonuçlarını kullanarak ve elemanlar arasındaki bağımlılığı kullanarak, bir saldırganın bir hedefin genomu hakkında (orijinal gizlilik garantilerine kıyasla) %50'ye kadar daha hassas bilgi ortaya çıkarabildiğini göstermektedir. Ayrıca, saldırganın, bir hedefin başka bir genomik veri kümesine (örn., hassas bir özellik ile ilişkili) üyeliğini çıkarmak için nitelik çıkarım saldırısından elde edilen çıkarımsal genomik verileri kullanabileceğini gösteriyoruz. Bir log olabilirlik oranı (LLR) testi kullanarak, sonuçlarımız, saldırganın çıkarım gücünün, böyle bir saldırıda, çıkarımsal (ve dolayısıyla kısmen yanlış) genomlar kullanarak bile önemli ölçüde yüksek olabileceğini göstermektedir. Son olarak, elemanlar arasındaki bağımlılığı göz önünde bulundurarak gizlilik garantilerini elde etmek için genomik veri kümelerinden istatistiklerin gizliliğin korunması için bir mekanizma öneriyoruz. Mekanizmamızı farklı genomik veri kümeleri üzerinde değerlendirerek, önerilen mekanizmamızın geleneksel DP tabanlı çözümlerden %50'ye kadar daha iyi gizlilik sağlayabildiğini ampirik olarak gösteriyoruz.

Anahtar sözcükler: Genomik veri kümesi, diferansiyel gizlilik, çıkarsama saldırısı.

Acknowledgement

First and foremost, I would like to thank Allah the Almighty for granting me the patience, strength, and guidance to accomplish this work.

I am so grateful for meeting many inspiring and knowledgeable professors, colleagues, and friends, during the last five years in my PhD journey at Bilkent University. I was able to experience an enormous amount of educational, professional, and personal growth due to the support and encouragement of several people. For that, I would like to seize this opportunity to acknowledge all of them.

I would like to express my sincere gratitude to my advisor Prof. Özgür Ulusoy and my co-advisor Prof. Erman Ayday for guiding me, supporting me, and giving me valuable feedback throughout this challenging period. Their dedication to research has inspired me and has significantly influenced my research. I am very proud to be their PhD student. I am also grateful to the members of my thesis committee Prof. Ercüment Çiçek, Prof. Can Alkan, Prof. Murat Cenk, and Prof. Ali Aydın Selçuk for their valuable feedback and discussions.

My PhD is an enjoyable journey with my lovely husband, Mohammed Alser and my sons, Hasan and Adam. Their love, care, and never ending support always give me the internal strength to move on whenever I feel like giving up or whenever things become too hard.

To my father Mohammed Almadhoun, my mother Nariman who raised me up with their love, care and great endless support. To my brothers Ahmed Nemer, Ezz Eldeen, Mo'tasem, Mahmoud and my beautiful sisters Rania, Nermeen, Eman and their lovely families, I would like to extend my appreciation, love and faithful thank to them for always believing in me and being by my side. Thanks to my parents-in-law, Hasan Alser and Itaf Abdelhadi, my brothers-in-law Ayman, Hani, Sohaib, Osaid, Moath, and Ayham and my lovely sister-in-law Deema for their love and support.

I feel very grateful to all my inspiring friends who made Bilkent a happy home for me and my family, especially Amirah Ahmed, Maha Sharei, Nabeel Abu Baker, Zahra Ghanem, Muaz Draz, Elif Doğan Dar, Salman Dar, Fatma Kahveci, Gülfem Demir, Shady Zahed, Mohammed Tareq and his mother, Abdelrahman Teskyeh,

Obada and many others. I would also thank all my lovely friends especially Mrwa Elbatsh, Ola Murtaga, Heba Sallouha, Eman Almadhoun, Heba Kadry, Olfat Eltoukhy, Amaal Bashir, Tasneem Hamza, Nisreen Qazaqzeh, Heba Sa'adah, Hiba Hab, my relatives and those who have directly or indirectly assisted me.

Finally, I gratefully acknowledge the Computer Engineering Department at Bilkent University for supporting my PhD work, providing generous financial support, and funding my academic visits.

Contents

1 Introduction **2**

1.1 Motivation . . . 3
1.2 Research Problem . . . 3
1.3 Thesis Statement . . . 5
1.4 Contributions . . . 5
1.5 Outline . . . 7

2 Background **8**

2.1 Overview . . . 8
2.2 Genomics . . . 8
2.2.1 Genome-Wide Association Studies (GWAS) . . . 11
2.3 Genetic Privacy Breaching Strategies . . . 11
2.3.1 Identity Tracing by Meta-Data and Side-Channel Leaks . . . 12
2.3.2 Identity Tracing by Genealogical Triangulation . . . 13
2.3.3 Identity Tracing by Phenotypic Prediction . . . 14
2.3.4 Completion Attacks . . . 14
2.3.5 Attribute Disclosure Attacks via DNA (ADAD) . . . 15
2.3.6 Membership Inference Attacks Against Statistical Genomic Datasets . . . 17
2.4 Privacy of Genomic Data . . . 17
2.4.1 Identity Tracing by Meta-Data and Side-Channel Leaks . . . 18
2.4.2 Identity Tracing by Genealogical Triangulation . . . 18
2.4.3 Identity Tracing by Phenotypic Prediction . . . 19
2.4.4 Completion Attacks . . . 19

2.4.5 Attribute Disclosure Attacks via DNA (ADAD) and Membership Attacks . . . 20
2.5 Differential Privacy . . . 20
2.5.1 Overview . . . 20
2.5.2 Laplace Perturbation Mechanism (LPM) . . . 21
2.5.3 DP-Based Privacy Mechanisms . . . 21
2.5.4 Inference Attacks Against DP-Based Mechanisms . . . 22
2.5.5 DP for Privacy-Preserving Release of GWAS Results . . . 22
2.5.6 Handling dependent tuples for DP . . . 23
2.6 Summary . . . 24

3 Attribute Inference Attack For Count Query 25
3.1 Overview . . . 25
3.2 Threat Model . . . 25
3.3 Dataset Description . . . 27
3.3.1 1000 Genomes Phase 3 data . . . 28
3.3.2 CEPH/Utah Pedigree 1463 . . . 28
3.3.3 Manuel Corpas (MC) Family Pedigree . . . 29
3.4 Differential Privacy under Dependent Tuples . . . 30
3.5 Inference Evaluation Algorithm . . . 32
3.6 Evaluation . . . 35
3.7 Experimental Results . . . 36
3.7.1 Estimation Error Experiment . . . 36
3.7.2 Leaked Information Experiment . . . 39
3.7.3 Estimation Error Experiment for Stronger Adversary . . . 42
3.7.4 Leaked Information Experiment for Stronger Adversary . . . 45
3.8 Summary . . . 46

4 Attribute Inference Attack for Complex Queries 48
4.1 Overview . . . 48
4.2 Threat Model . . . 48
4.3 Attribute Inference Attack for Complex Queries . . . 51
4.4 Inference Evaluation Algorithm . . . 52
4.5 Evaluation . . . 53

4.6 Experimental Results 54
4.6.1 MAF Queries 54
4.6.2 Chi-Square Queries 60
4.6.3 Comparison with the Sum Query in [1] 64
4.7 Summary 65

5 Membership Inference Attack 66
5.1 Overview 66
5.2 Threat Model 66
5.3 Membership Inference Attack 67
5.4 Membership Inference Attack Evaluation Algorithm 68
5.5 Evaluation 69
5.6 Experimental Results 70
5.7 Summary 72

6 Attribute Inference Attacks against Dynamic Genomic Datasets 73
6.1 Overview 73
6.2 Dynamic Datasets 73
6.3 Threat Model 76
6.4 Evaluation 78
6.5 Experimental Results 80
6.5.1 Independent Tuples Assumption Experiments 80
6.5.2 Dependent Tuples Assumption Experiments 82
6.6 Summary 91

7 Countermeasures 92
7.1 Overview 92
7.2 Detailed Discussion of Experimental Results 92
7.3 Countermeasures 94
7.4 Methodology for Countermeasures 97
7.5 Evaluation of Countermeasures 99
7.6 Comparison with Existing Work 102
7.7 Summary 104

8 Conclusion and Future Work 105

8.1 Future Research Directions 106

A Data and Code **118**

List of Figures

2.1 Mendelian inheritance 9
2.2 A possible route for identity tracing using both metadata/side-channel leaks and phenotypic prediction. 13
2.3 A possible route for identity tracing using genealogical triangulation. 14
2.4 A possible route for identity de-anonymization using a completion attack. 15
2.5 Attribute disclosure attacks via DNA. 16

3.1 Threat model for the sum query 26
3.2 CEPH/Utah Family Tree 29
3.3 Manuel Corpas Family Tree 30
3.4 The effect of including non-relatives or/and relatives on the probability of the adversary's correctness 37
3.5 The effect of different values of the privacy budget and the number of non-relatives or/and relatives on the probability of the adversary's correctness 38
3.6 The effect of different values of the privacy budget and the number of non-relatives or/and relatives on the target's leaked SNPs the adversary can infer 40
3.7 The effect of different values of the privacy budget, ϵ, and the number of (a)(b) family relatives in set $\mathbf{F}$ ($|\mathbf{F}| = f$) and (c)(d) non-relatives in set $\mathbf{U}$ ($|\mathbf{U}| = u$) on adversary's success in inferring the targeted SNPs. 41

3.8 The effect of including non-relatives or/and relatives on the adversary's correctness when the adversary has access to $K = 50\%$ of other d family members' genomes 43
3.9 The effect of different values of the privacy budget and the number of non-relatives or/and relatives on the probability of the adversary's correctness when the adversary has access to $K = 50\%$ of other d family members' genomes 44
3.10 The effect of different values of the privacy budget and the number of non-relatives or/and relatives on the leaked SNPs the adversary can infer when the adversary has access to $K = 50\%$ of other d family members' genomes 46

4.1 Threat model for the complex queries 49
4.2 The effect of different values of the privacy budget, ϵ, and the number of (a) family members in set $\mathbf{F}$ ($|\mathbf{F}| = f$) and (b) 2 first-degree relatives (father and mother) in set $\mathbf{F}$ along with different numbers of non-relatives in set $\mathbf{U}$ ($|\mathbf{U}| = u$) on adversary's correctness (1 - estimation error) in inferring the targeted SNPs from the noisy results of MAF statistics. (w/ Dep) represents the scenario in which the adversary considers the data dependency and (w/o Dep) represents the opposite. 56
4.3 The effect of including (a) only family members in $\mathbf{F}$ ($|\mathbf{F}| = f$) and (b) 2 first-degree relatives (father and mother) with different numbers of non-relatives in $\mathbf{U}$ ($|\mathbf{U}| = u$) on the leaked information (i.e., number of leaked SNPs of target j) using the noisy results of MAF statistics. (w/ Dep) represents the scenario in which the adversary considers the data dependency and (w/o Dep) represents the opposite. 58
4.4 The effect of including (a) only family members in $\mathbf{F}$ ($|\mathbf{F}| = f$) and (b) 2 first-degree relatives (father and mother) with different numbers of non-relatives in $\mathbf{U}$ ($|\mathbf{U}| = u$) on the leaked information by only considering the rare SNPs of the target for MAF queries. 59

4.5 The effect of different values of the privacy budget, ϵ, and the number of (a) family members in $\mathbf{F}$ ($|\mathbf{F}| = f$) and (b) 2 first-degree relatives (father and mother) with different numbers of non-relatives in $\mathbf{U}$ ($|\mathbf{U}| = u$) on adversary's correctness (1 - estimation error) in inferring the targeted SNPs, using the noisy results of χ^2 statistics. (w/ Dep) represents the scenario in which the adversary considers the data dependency and (w/o Dep) represents the opposite. . . . 61

4.6 The effect of including (a) only family members in $\mathbf{F}$ ($|\mathbf{F}| = f$) and (b) 2 first-degree relatives (father and mother) with different numbers of non-relatives in $\mathbf{U}$ ($|\mathbf{U}| = u$) on the leaked information (i.e., number of leaked SNPs of target j) using the noisy results of χ^2 statistics. (w/ Dep) represents the scenario in which the adversary considers the data dependency and (w/o Dep) represents the opposite. 62

4.7 The effect of different values of the privacy budget, ϵ on the leaked information by only considering the rare SNPs of the target for χ^2 queries. The query results include: (i) 3 family members (father, mother, and sister) in $\mathbf{F}$ and (ii) 2 family members (father and mother) in $\mathbf{F}$ with different numbers of non-relatives in $\mathbf{U}$ ($|\mathbf{U}| = u$). 63

4.8 The effect of different values of the privacy budget, ϵ, when the query results include (i) 3 family members (father mother and sister) in set $\mathbf{F}$ and (ii) 2 family members (father and mother) in set $\mathbf{F}$ along with 10 non-relatives in set $\mathbf{U}$ ($|\mathbf{U}| = 10$) on adversary's correctness (1-estimation error) in inferring the targeted SNPs. The adversary exploits the noisy results of 3 different queries: sum, MAF, and χ^2 statistics. χ^2 statistics include 3 different cases since we apply 3 existing algorithms to generate the query results. . . . 65

5.1 Power of the adversary for the membership inference attack for different number of MAF queries over dataset T_3. (a) shows the power when the adversary uses the actual (correct) SNPs of target j. (b) shows the power when the adversary uses the inferred SNPs of the target as a result of the attribute inference attack. In (b), ϵ values are the ones used in the attribute inference attack; for membership inference, the values in M_S are shared (in a differentially-private way) with the adversary using $\epsilon = 5$ for all cases. 71

6.1 Threat model for dynamic datasets 77
6.2 The effect of including different numbers of unrelated individuals in the queries results on the probability of the adversary's correctness in inferring the targeted SNPs 81
6.3 The effect of including different numbers of unrelated individuals in the queries results on the target's leaked SNPs the adversary can infer 82
6.4 The effect of including different numbers of unrelated or/and related individuals in the queries results on the probability of the adversary's correctness in inferring the targeted SNPs 85
6.5 The effect of including different numbers of unrelated or/and related individuals in the queries results on the leaked information (i.e., number of leaked SNPs of target j) using the noisy results of MAF statistics 87
6.6 The effect of including different numbers of unrelated or/and related individuals in the queries results on the probability of the adversary's correctness (1 - estimation error) in inferring the targeted SNPs 89
6.7 The effect of including different numbers of unrelated or/and related individuals in the queries results on the leaked information using the noisy results of MAF statistics 90

7.1 Dependence between tuples violate the privacy guarantees of DP. For instance, including up to 9 family members of the target j in the query results, degrades the privacy guarantees from ϵ to 1.68ϵ. 93

7.2 (a) σ values for computing the sensitivity when the query results contain the dependent tuples only (b) σ values for computing the sensitivity when the query results contain the dependent tuples and other unrelated tuples. 99

7.3 The effect of applying our proposed countermeasure for different values of the privacy budget, ϵ. "DP" lines stand for applying differential privacy mechanism (over 3 different sets of family members in the dataset) and the other three lines show the leaked SNPs when our proposed mechanism is applied. 100

7.4 (a)(b) The effect of applying our privacy model using different values of the privacy budget ϵ, and different number of family relatives starting from 1 until 7 first degree family members over UTAH family dataset. 101

7.5 (a) The amount of Laplace noise added for different values of privacy budget ϵ. (b) The privacy performance of different mechanisms which guarantee the (α, β)-usefulness. Here, the noisy output of the query should deviate by at most α from the real value (in terms of L1-norm) with probability $(1 - \beta)$. 103

List of Tables

2.1 Mendelian inheritance probabilities for a child's SNP value given his/her parents genotypes (left), and Mendelian inheritance probabilities for a father's SNP value given the genotypes of mother and child (right). "B" represents a major allele and "b" represents a minor allele. 10

2.2 GWAS genotype distribution for a 2×3 contingency table (left) and a 2×2 contingency table (right). 11

Notation	Explanation
A	A mechanism produces outputs with noise drawn from a suitable Laplace distribution.
Range(A)	The domain of the output under mechanism A, $O \in Range(A)$
Q	The Query function
ϵ	Parameter expressing the privacy budget
T	The actual genomic statistical dataset
T'	Neighboring genomic statistical dataset
Individual j	The target
h	A tuple in T
n	Number of participants in the dataset
SNP	The set of SNPs IDs
m	The number of SNPs
X	An m×n matrix that stores the values of the SNPs for all members
X_j	The set of SNPs for an individual j
X'_j	The inferred genomic record of target j
$\tilde{T^i_{pj}}$	The noisy query results
δ	The added Laplace noise to the query results, δ for sum query $= Lap(2/\ \epsilon)$
P	The set of all participants included in the query result (except for the target j) ($\|\mathbf{P}\| = p$)
T^i_j	Represents x_{ij}, the SNP value i for individual j
T^i_p	The sum of the SNP i values for p participants
d	The size of the tuples which are correlated with the individual j
D	$D = p$ if $p \leq 2$, $D = 2p$ if $p > 2$
y	$y \in [-1, 1]$ for $p \leq 1$, and $y \in [0, 1]$ for $p > 1$
k	The adversary prior knowledge about T^i_p
$R_{j,h}$	The dependence relationship between two tuples j and h
E	The expected estimation error
L	The leaked information

Notation	Explanation
$Dist$	The distance between the true value of the SNP and the adversary estimated value
v	The set of different epsilon values
σ	The variable used to get the new value of ϵ to be applied to the query results over a correlated genomic data
ς	The sensitivity for releasing the query results over a correlated genomic data
L_0	The leaked information any adversary can get without dependency assumption
L_1	The leaked information any adversary can get with dependency assumption
T_1	A statistical genomic dataset
T_2	A case-control genomic statistical dataset
F	The set of all the family members of target j ($\|\mathbf{F}\| = f$)
U	The set of other unrelated individuals ($\|\mathbf{U}\| = u$)
$\tilde{M^i_{pj}}$	Differentially-private MAF query result for all participants included in the query result
Q^i_{pj}	The number of minor alleles for all participants included in the query result
τ	The total number of alleles for all participants included in the query result
Q^i_p	The number of minor alleles in $\mathbf{P}$
τ_1	The total number of alleles in $\mathbf{P}$ for SNP it ($\tau = 2p + 2$)
M^i_p	The MAF values due to individuals in $\mathbf{P}$
M^i_j	The MAF values for target j
y_1	A kinship coefficient that satisfies the Mendel's law. y_1 is in $\left[\frac{-2}{\tau}, \frac{2}{\tau}\right]$
$\widetilde{\chi^2_i}$	Differentially-private chi square query result
y_2	A kinship coefficient that satisfies the Mendel's law. Thus, y_2 is in $[-2, 2]$
T_3	A statistical case-control dataset

Notation	Explanation
M_C	The MAF values of SNPs of individuals in the control group
M_P	The MAF values of the SNPs for the entire dataset population
M_S	The MAF values for the SNPs of individuals in the case group
S	The set of individuals in the case group where $\|\mathbf{S}\| = s$)
C	The set of individuals in the control group where $\|\mathbf{C}\| = c$)
$\widetilde{M_S^i}$	The LPM-based noisy query result about the MAF value of a SNP i for individuals in the case group ($\mathbf{S}$)
β	The maximum achievable power
α	The false-positive rate
Z_α	The 100(1-α) percentile of the standard normal distribution
th	The threshold from the null hypothesis with $\alpha = 0.1$
M'_S	The adversary estimation for the MAF values of the SNPs in the case group
T_4	A dynamic statistical genomic dataset
t *and* $t + \varphi$	times where $(0 \leq t < t + \varphi)$

Chapter 1

Introduction

Today's high-throughput sequencing platforms (HTS) are capable of generating a tremendous amount of sequencing data. These technologies allow sequencing the full human genome for as little as few hundred dollars [2]. As a result, production of genomic information for research, clinical care, and recreational purposes at a rapid pace is no longer impossible from a technical point of view. One of the most prominent uses of genomic data is for research purposes and to make such research initiatives successful, researchers need individuals donate their genomic data. Several studies report the attitudes of public in different countries (including USA, Sweden, Japan, and Singapore) towards genomic research and their willingness to donate genomic samples [3, 4, 5, 6, 7, 8, 9, 10]. Although the majority of respondents show positive attitude towards genomic research and participating in such studies, the overwhelming majority of them have ranked privacy of sensitive information as one of their top concerns. Therefore, proper and privacy-preserving management of the personal information is necessary in order to attain public support for genomic studies. In addition, transparency of the research aim and proper management of genomic data utilization should be also maintained in order to not utilize the data beyond the donor's intention [11]. Thus, while information sharing is imperative for enabling better utility in many applications, statistics and academic research; protecting the privacy of users whose data is being shared is also significant.

1.1 Motivation

The availability of human genomic banks provides an adequate basis for several important applications and studies [12]. GWAS is considered as one of the most widely conducted genomic studies. These studies help scientists uncover associations between differences in the human genomes called single nucleotide polymorphisms (SNPs) and disorders that are passed from one generation to the next. We provide a brief background on genomics in Chapter 2. Since the first GWAS study in 2005 [13], researchers have assumed that it is safe to publish aggregate statistics about the SNPs that they found relevant to particular diseases and its associated phenotypes. A typical GWAS compares the genomes of individuals that carry a disease (cases) with genomes of healthy individuals (controls). Because the reported aggregate statistics were pooled from thousands of individuals, researchers believed that their release would not compromise the participants' privacy. However, such belief was challenged when [14] demonstrated that, under certain conditions, given an individual's genotype, one only needs the minor allele frequencies (MAFs) of the SNPs used in the study and other publicly available information in order to determine whether the individual is in the case group of a GWAS study. After this attack, the NIH restricted the access to key results and data of GWAS to only trusted individuals. The purpose of this access policy is mainly due to the growing privacy concern about the participants in any genomic studies and their sensitive information such as their health status. However, accelerating the pace of biomedical breakthroughs and discoveries necessitates not only collecting millions of genomic samples, but also granting an open access to the genomic banks and datasets [15].

1.2 Research Problem

There has been a growing interest in applying different privacy-preserving techniques to the GWAS results in order to grant access to genomic datasets. Many works in the literature propose utilizing the differential privacy (DP) notion [16]

to provide formal privacy guarantees for the participants of genomic studies. In a nutshell, DP guarantees that the distribution of query results change only slightly with the addition or removal of a single individual's data in the dataset. Although DP mechanism provides formal guarantees to preserve privacy [16], it does not consider the dependency of the data tuples in the dataset. In reality, data from different users in the datasets may be dependent according to social, behavioral, and genomic interactions between them [17, 18, 19]. For example, in social network datasets, "friendship" relation between two users may imply similar interests [20]. Moreover, one can infer the locations of an individual from the friends' locations since they are likely to visit the same places [19, 21]. Similarly, in medical studies, an adversary may infer the susceptibility of an individual to a hereditary disease by using the correlation between genomes of family members [22, 23]. These facts about the effect of correlation between tuples and data privacy was first observed by [22]. Later, other researchers [18, 19, 24] show that one can take advantage from dependencies between users to predict the users' sensitive information from the differentially private query results. However, this privacy risk has not yet been studied for statistical genomic dataset. Therefore, privacy guarantees of DP-based techniques may degrade if a genomic datasets includes dependent tuples (e.g., individuals from the same family). Hence, the genomic studies participant's sensitive information may be threaten when correlations exist within the dataset tuples.

Attribute inference not only may result in genomic discrimination, but it may also be utilized in membership inference attacks. The outcome of the attribute inference can be utilized in a membership inference attack. Possibility of membership inference attacks against genomic datasets was first shown by [14]. Later, [25] exploit the correlations between SNPs to perform the membership inference [14]. [26] showed the effectiveness of the membership inference attack using a log-likelihood-ratio (LLR) test [27].

Furthermore, many real-world statistical genomic datasets are dynamic, in which dataset size may change over time since new tuples (individuals) are inserted in or deleted from the genomic dataset. Most existing privacy-preserved statistical genomic models have been applied on static datasets. Maintaining

the privacy of genomic dynamic datasets is a challenging task since any adversary can exploit the difference between the released statistics results in different times to infer more sensitive information. Thus, a systematic analysis of these vulnerabilities for statistical static and dynamic genomic datasets is crucial.

Moreover, existing proposed mechanisms for maintaining the privacy of the statistical datasets fail to provide high data utility, which is a crucial requirement when releasing data from statistical genomic datasets. Medical researchers need highly accurate information for high quality and effective research outcome. Therefore, it is also crucial to develop utility-preserving countermeasures for this privacy risk.

1.3 Thesis Statement

Our goal in this thesis is to develop an effective perturbation mechanism to achieve the privacy guarantees in DP for static and dynamic statistical genomic datasets with dependent tuples, while maintaining the data utility.

Our thesis statement is:

A better tradeoff between privacy guarantees and medical utility can be gained in the static/dynamic statistically genomic datasets with dependent tuples by considering practically significant adversary models.

1.4 Contributions

The overarching contribution of this thesis is that we formalize the DP concept to handle probabilistic dependence relationships between tuples in genomic datasets. We develop an effective perturbation mechanism to achieve the privacy guarantees in DP for datasets with dependent tuples. Our mechanism uses a carefully

computed dependence coefficient that quantifies the probabilistic dependence between tuples in a fine-grained manner. The contributions of our thesis are as follows:

- We demonstrate the feasibility of an inference attack on differentially private query results by exploiting the dependence between tuples in a static real-world genomic dataset. We assume that the goal of the adversary is to infer the genomic data of a target individual using query results from a statistical genomic dataset. We also assume that the dataset includes correlated individuals (i.e., family members of the target individual). We show that the adversary can infer significantly more genomic data about the target from the results of queries by only exploiting the correlations between the genomes of family members. Moreover, we show that a stronger adversary with partial prior information about the genomic data of family members can infer even more sensitive data. Published in Bioinformatics [1].

- We demonstrate the scale of attribute inference attacks using differentially-private results of two complex and real-life queries over statistical genomic datasets (compared to the simple sum query considered in [1]). As opposed to [19], which only considers pairwise correlation between the tuples, we consider interdependent correlations between dataset participants. We also show how an adversary performs a successful membership inference attack using the inferred genomic data as a result of the attribute inference attacks. Published in Bioinformatics [28], and presented at ISMB2020.

- We formalize the notion of ϵ-differential privacy for genomic datasets with dependent tuples to avoid inference of sensitive information by any adversary with prior knowledge about the dependency between tuples. Our proposed mechanism computes the "adjusted" ϵ value that provides privacy guarantees in the existence of dependent tuples in the dataset. That is, according to the number of dependent tuples in the dataset and their relationships, our mechanism allows accurate computation of the ϵ values for dependent data to preserve the privacy of the dataset participants while maintaining the utility of the data. Published in Bioinformatics [1].

- We evaluate our mechanisms over two different real-world genomic datasets. We demonstrate that it can be applied to any genomic statistics dataset with dependent tuples. Applying the proposed mechanism can provide better privacy and utility guarantees compared to other state of the art DP-based mechanisms. Published in Bioinformatics [1].

- We demonstrate the feasibility of an inference attack on two differentially private query results shared in a differentially-private way at time t and $t + \varphi$ in an external dynamic real-world genomic dataset. We have two assumptions for the dynamic dataset: first it only includes independent tuples. Second, it has correlated tuples (i.e., family members of the target individual). We show that the adversary can infer significantly more genomic data about the target by exploiting the two queries results or the change on these results. This work is yet to be published.

1.5 Outline

This thesis is organized into 8 chapters. Chapter 2 describes the necessary background on genomics privacy and related prior works on differential privacy mechanisms for genomic datasets. Chapter 3 explores the privacy threats from an attribute inference attack using count query over genomic datasets. Chapter 4 explores the privacy threats from an attribute inference attack over genomic datasets using complex queries. Chapter 5 presents the membership inference attack using the results of an attribute inference attack over genomic statistical datasets. Chapter 6 presents attribute inference attacks against dynamic genomic datasets. Chapter 7 presents countermeasures for the privacy issues arrised from the inference attacks. Chapter 8 presents conclusions and future research directions that are enabled by this thesis. Finally, Appendix A presents the online sources for the used datasets and algorithms codes.

Chapter 2

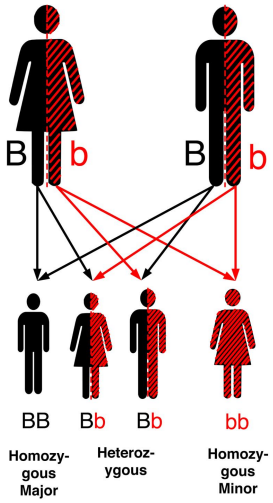
Background

2.1 Overview

In this chapter, we provide the necessary background on genomics and differential privacy. We then provide an extensive literature review on the prior, existing, and recent inference attacks and differential privacy mechanisms. We also highlight the differences of our work from the existing work.

2.2 Genomics

The human genome consists of about 3.2 billion base pairs, where each base pair is composed of a nucleotide from the DNA alphabet $\{A, C, G, T\}$ and its complementary base. Only around 0.5% of these base pairs differ between any two individuals and this value is even smaller for closely related people [29]. In the following, we briefly introduce the genomic concepts we use throughout this paper.



(a)

Mother \ Father	BB	Bb	bb
BB	(1,0,0)	(0.5,0.5,0)	(0,1,0)
Bb	(0.5,0.5,0)	(0.25,0.5,0.25)	(0,0.5,0.5)
bb	(0,1,0)	(0,0.5,0.5)	(0,0,1)

(b)

Mother \ Child	BB	Bb	bb
BB	(0.5,0.5,0)	(0,0.5,0.5)	N/A
Bb	(0.5,0.5,0)	(0.33,0.33,0.33)	(0,0.5,0.5)
bb	N/A	(0.5,0.5,0)	(0,0.5,0.5)

(c)

Figure 2.1: (a) Mendelian inheritance for a child. (b) Mendelian inheritance probabilities for a SNP given all the genotypes of the parents. The probabilities of the child's genotype to be (BB, Bb, or bb) are represented in parentheses. (c) Mendelian inheritance probabilities for the father's SNP given all the genotypes of the child and his mother. The probabilities of the father's genotype to be (BB, Bb, or bb) are represented in parentheses.

The most common genomic differences in the human genome are single base-pair differences. Scientists call these differences single-nucleotide polymorphisms

(SNPs). They are responsible for the differences in our phenotypes (i.e., observable physical characteristics, like height and eye color) and genotypes (i.e., our genes that are translated into proteins and cause our phenotypes). Researchers analyze these variations to discover their genomic contributions to many diseases such as cancer, autism, heart disease, and diabetes [30]. Two different nucleotides (called alleles) are observed for each SNP. One allele is inherited from the father and one inherited from the mother. Each inherited allele can be either (i) a major allele (which is the frequently observed nucleotide), or (ii) a minor allele (which is the rare nucleotide). The minor allele frequency (MAF) of a SNP denotes the frequency at which the rare nucleotide occurs at a particular SNP. The standard convention used to represent the value of a SNP is the number of minor alleles it contains: 0 if no minor allele is observed (homozygous-major genotype), 1 if one minor allele is observed (heterozygous genotype), and 2 if both nucleotides are minor alleles (homozygous-minor genotype). As discussed, SNPs are inherited by an offspring from his/her parents. As stated in the Mendel's first law, given the SNPs of both parents, their child's SNPs are independent from the other ancestors. Figure 2.1(a) shows all potential genotypes that can be inherited from a father and a mother. In the figure, "B" represents a major allele and "b" represents a minor allele. According to [31] the Mendelian inheritance probabilities are also shown as in Figure 2.1(b) and (c) or Table 2.1.

Table 2.1: Mendelian inheritance probabilities for a child's SNP value given his/her parents genotypes (left), and Mendelian inheritance probabilities for a father's SNP value given the genotypes of mother and child (right). "B" represents a major allele and "b" represents a minor allele.

	Father			**Son**		
Mother	**BB**	**Bb**	**bb**	**BB**	**Bb**	**bb**
BB	BB: 1 Bb: 0 bb: 0	BB: 0.5 Bb: 0.5 bb: 0	BB: 0 Bb: 1 bb: 0	BB: 0.5 Bb: 0.5 bb: 0	BB: 0 Bb: 0.5 bb: 0.5	N/A
Bb	BB: 0.5 Bb: 0.5 bb: 0	BB: 0.25 Bb: 0.5 bb: 0.25	BB: 0 Bb: 0.5 bb: 0.5	BB: 0.5 Bb: 0.5 bb: 0	BB: 0.33 Bb: 0.33 bb: 0.33	BB: 0 Bb: 0.5 bb: 0.5
bb	BB: 0 Bb: 1 bb: 0	BB: 0 Bb: 0.5 bb: 0.5	BB: 0 Bb: 0 bb: 1	N/A	BB: 0.5 Bb: 0.5 bb: 0	BB: 0 Bb: 0.5 bb: 0.5

2.2.1 Genome-Wide Association Studies (GWAS)

GWAS is the general name for case-control studies that focus on identifying genomic variations that are associated with a particular phenotype. On a broad scale, these studies help scientists uncover associations between individual SNPs and disorders that are passed from one generation to the next. A typical study compares the genomes of individuals that carry a disease or phenotype (cases) with the ones of healthy individuals (controls) to identify the functional impacts of certain SNPs on the corresponding disease. The SNP is causative or associated with the phenotype if there is a positive or negative correlation. In order to summarize the association information for each SNP, a 2×3 or 2×2 contingency table (as shown in Table 2.2) is used to show the number of cases and controls having a particular SNP with different values. The output of GWAS studies often consist of chi-square statistic, p-values, or MAFs for the most significant SNPs.

Table 2.2: GWAS genotype distribution for a 2×3 contingency table (left) and a 2×2 contingency table (right).

	Genotype			
	0	1	2	Total
Case	S_0	S_1	S_2	S
Control	C_0	C_1	C_2	C
Total	n_0	n_1	n_2	n

	Genotype		
	0	1	Total
Case	S_0	$S_1 + S_2$	S
Control	C_0	C_1+C_2	C
Total	n_0	n_1+n_2	n

2.3 Genetic Privacy Breaching Strategies

In this section, we survey a wide spectrum of privacy threats to human genomic data, as reported by prior research. In general, we assume the existence of a passive adversary who has bounded computational power. In all below threats, the adversary only has access to publicly available genetic databases and other publicly available resources on the Internet.

2.3.1 Identity Tracing by Meta-Data and Side-Channel Leaks

In such an attack, as illustrated in Figure 2.2, the hacker or curious party needs both human genomic data, which is already available online via a certain privacy-preserving mechanism (i.e., hiding the identity information of the owner), and additional metadata. Such an attack, once it succeeds, can cause serious implications, for instance genetic discrimination, financial loss, and blackmail. A real-life example of this threat was in 1997 when Sweeney [32] successfully identified the medical condition of William Weld, former governor of Massachusetts, using only his demographic data (i.e., date of birth, gender, and 5-digit ZIP code) appearing in the hospital records and voter registration forms that are available to everyone. In 2013, Sweeney [33] again showed that it is possible to utilize the demographic data to discover the real identities of the DNA donors even though their names are removed from the published genomic database. The approach was very similar to her previous attack, besides, in this work, she exploited the side-channel data in the downloaded genomic data files associated with anonymized PGP profiles. Even for some participants, once the downloaded file was uncompressed, the resulting file had a filename that included the actual name of participant.

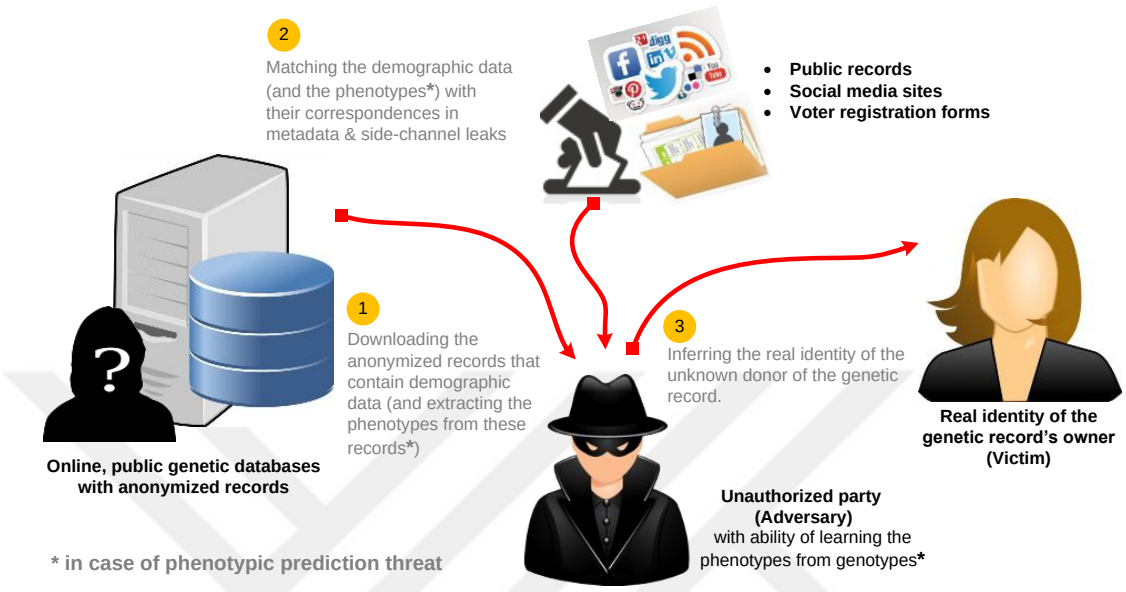


Figure 2.2: A possible route for identity tracing using both metadata/side-channel leaks and phenotypic prediction.

2.3.2 Identity Tracing by Genealogical Triangulation

In most human societies, surnames are paternally inherited, resulting a correlation with specific Y-chromosome haplotypes. Thus, there are several online public databases (e.g., Ysearch.org and SMGF.org) that collectively contain hundreds of thousands of surname-haplotype records, aiming at helping the public to identify their distant patrilineal relatives and the potential surnames of their biological fathers. However, these services can be exploited by an adversary towards learning the participant's identity, as illustrated in Figure 2.3. With the help of surname inferences in addition to the birth year and Zip code, the search results can be narrowed down the identity to few matches that can be investigated individually [34].

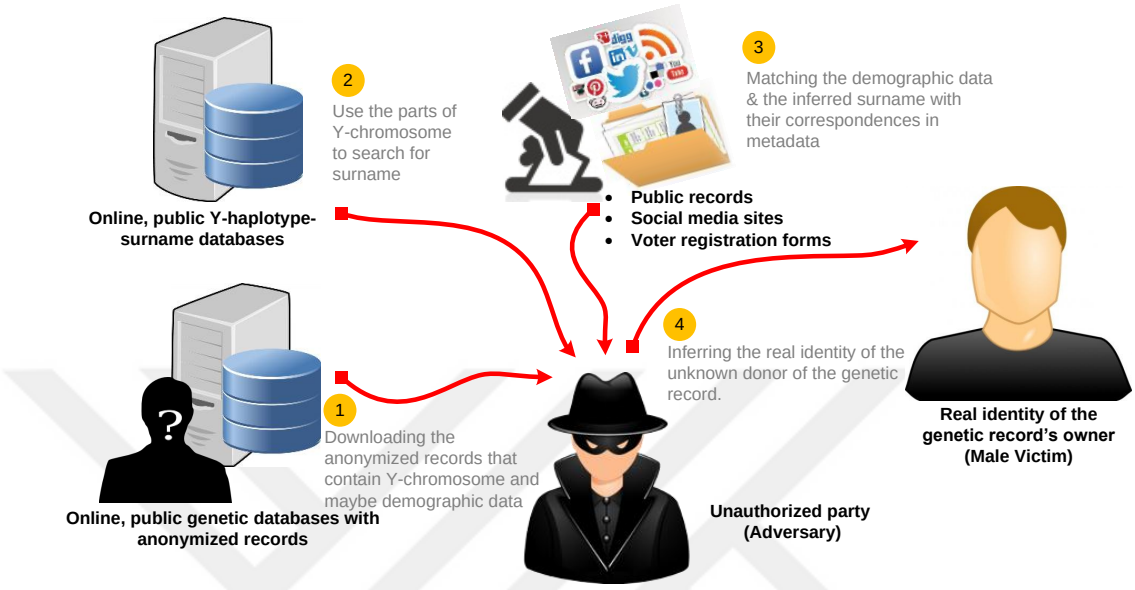


Figure 2.3: A possible route for identity tracing using genealogical triangulation.

2.3.3 Identity Tracing by Phenotypic Prediction

Visible phenotypes from genetic data could help in identity tracing. Such visible traits with high heritability that can be inferred from DNA include height, eye color, facial morphology, and age [35]. These traits can then be used as quasi-identifiers for decreasing the degree of uncertainty to infer the identity of an individual with the help of public records and social networks as explained in Figure 2.2. However, using only these quasi-identifiers for re-identification does not provide high accuracy; as the population-wide registries of these visible traits are not publicly accessible and searchable.

2.3.4 Completion Attacks

In genomics, genotype imputation is a well-studied task in which genetic information can be reconstructed from partial data by completing the missing genotype

values. A well-known example of a completion attack is the inference of Jim Watson's predisposition for Alzheimer's disease from his published genome, despite removing the ApoE locus gene (which is the indicator for Alzheimer's predisposition) from the published data [36]. Completion techniques can be used to predict the genomic information when there is no access to the DNA of a known individual, as shown in Figure 2.4.

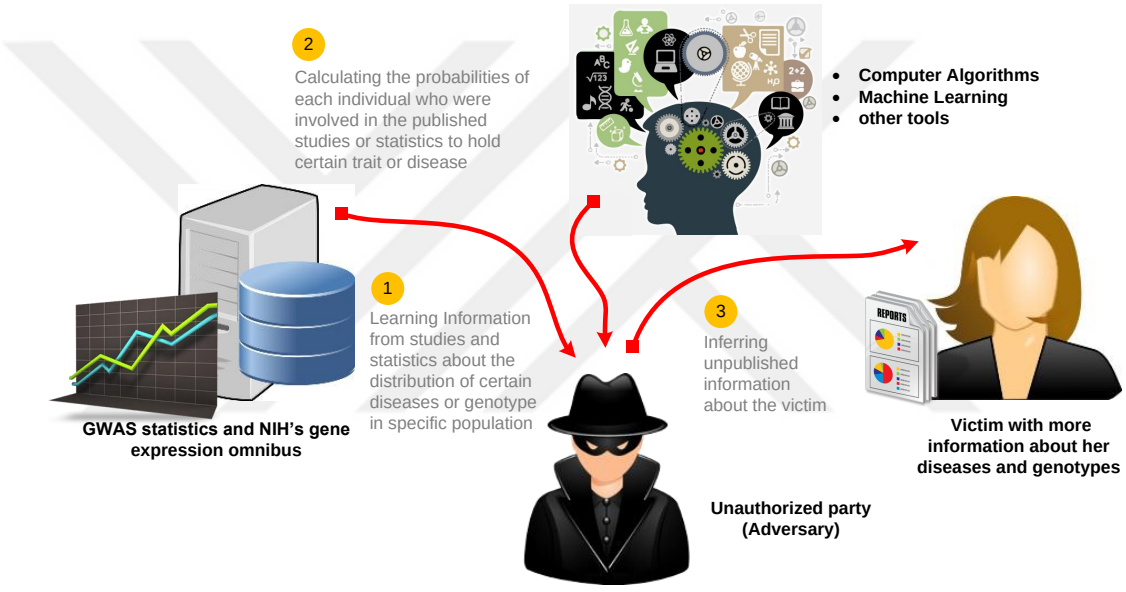


Figure 2.4: A possible route for identity de-anonymization using a completion attack.

2.3.5 Attribute Disclosure Attacks via DNA (ADAD)

The main concept of ADAD is when the adversary gains access to the DNA sample of the target. Using the identified DNA, the adversary can search genetic databases with sensitive attributes (e.g., drug abuse) as shown in Figure 2.5. Finding the identified DNA in the database reveals the link between the person and the sensitive attribute. Based on [37], three scenarios are identified to illustrate the attribute disclosure attacks: the n=1 scenario, the summary statistic scenario, and the gene expression scenario. The n=1 scenario is the simplest scenario of ADAD. By acquiring a chosen set of 45 autosomal SNPs. As we mention

in Section 2.2, SNPs are the main cause for variations in the human genome. They are also responsible for the differences in our phenotypes/traits and genotypes. The adversary can simply match the genotype data that is associated with the identity of the individual with the genotype data that is associated with the attribute [38]. Thus, GWAS stores individual genotypes and phenotypes in restricted access area, while the statistics of allele frequencies [1] are stored in the public access area. In spite of the separation, GWAS datasets with allele frequencies of the participants have been exploited by the ADAD's summary statistic scenario [14] as follows: The allele frequencies are positively biased towards the target genotypes in the case group compared to the allele frequencies of the general population. Moreover, the analyzed common variations can be exploited to conduct ADAD by integrating the biases in the allele frequencies over a large number of SNPs in GWAS. Therefore, the performance of ADAD is a function of the size of the study and the adversary's prior knowledge. Apart from GWAS, the NIH's Gene Expression Omnibus (GEO) databases are also vulnerable to the ADAD's gene expression scenario [39]. The GEO database holds hundreds of thousands of human gene expression profiles and their linked medical attributes.

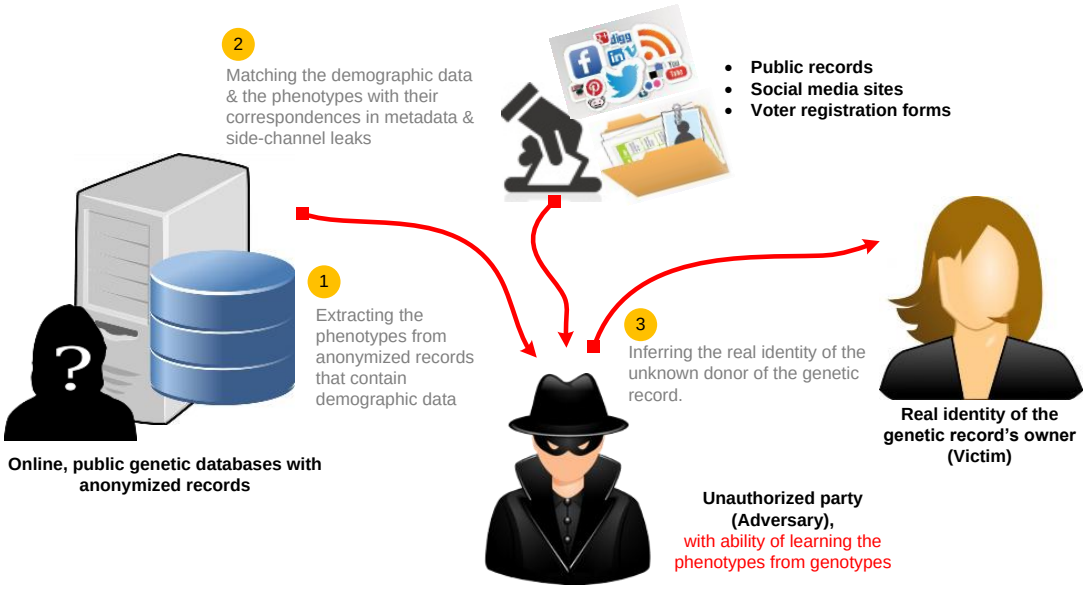


Figure 2.5: Attribute disclosure attacks via DNA.

[1]The allele frequency represents the incidence of a gene variant at a given gene location in a population gene pool.

2.3.6 Membership Inference Attacks Against Statistical Genomic Datasets

As we discuss in Section 2.3.5, possibility of membership inference attacks against genomic datasets was first shown by [14]. They show that using the distances between the MAF values of SNPs (released as a result of a genomic study, such as GWAS) and an individual's genotype, one can infer the involvement of an individual in the corresponding study. Later, [25] exploit the correlations between SNPs to perform the membership inference and showed that such an approach needs significantly less MAF values compared to [14]. [40] analyzed the theoretical complexity of membership inference attacks on genomic datasets. Furthermore, based on theoretical and empirical results, [26] showed that using log-likelihood-ratio test results in more powerful membership attack over statistical genomic datasets than the attack proposed by [14]. Recently, [41] showed the membership inference risk for datasets including miRNA expression data. Several solutions have been proposed to protect the privacy of statistical genomic datasets considering the identified vulnerabilities [42].

2.4 Privacy of Genomic Data

Data privacy is a critical important issue, which motivates many researchers to propose privacy-preserving mechanisms, such as k-anonymity [32], l-diversity [43], t-closeness [44], and differential privacy [16]. Privacy of genomic data has recently been a trending research topic [37]. Several solutions have been proposed to protect the privacy of genomic information considering the identified vulnerabilities [42]. Some researchers propose crypto-based techniques, such as[45, 46, 47], in which genomic data of individuals is processed using secure multi-party computation techniques. Differential privacy (DP) [16] has been widely applied for privacy-preserving release of statistical summaries for various genomic studies, such as GWAS [48, 49, 50].

In this section, we survey in details a wide spectrum of known privacy-preserving techniques against each aforementioned threat and make suggestions to prevent such threats. Here, we focus on the scenario, in which genomic data or the results of GWAS are made publicly available. There are also crypto-based mitigation techniques in which genomic data of individuals is stored in a database in encrypted form, and hence it is not publicly available on the Internet. Once other parties (e.g., medical centers) want to do operations on the data, they apply privacy-preserving techniques and they only obtain the result of the operation without having access to whole data. In this line of research, [46] proposed privacy-preserving techniques for medical tests and personalized medicine methods [46]. [51] make use of both medical and cryptographic tools for privacy-preserving paternity tests, personalized medicine, and genetic compatibility tests [51]. Also [52] developed a technique for privacy-compliant processing of raw genomic data [52].

2.4.1 Identity Tracing by Meta-Data and Side-Channel Leaks

As discussed in this threat model, metadata can be used for inferring the identities of involved individuals. Hence, any metadata that may decrease the level of privacy, should either be removed from datasets or strictly follow the 2002 Health Insurance Portability and Accountability Act (HIPAA) Privacy Rule. Data covered under HIPAA should follow certain strict formats; dates (e.g. birth, admittance, and discharge dates) would only contain the year, the ZIP code would only have the first 2 digits if the population in the ZIP code is less than 20,000 people, and no explicit identifiers (e.g. Social Security numbers) would be present.

2.4.2 Identity Tracing by Genealogical Triangulation

The first step towards protecting against this attack depends on the purpose of the genetic database. If the database provides services for descendants of

anonymous sperm donors to identify the surnames of their potential biological father and distant patrilineal relatives, then it should be an access-controlled database. Otherwise, the surname should be removed or replaced with the given name in haplotype records in order to decrease the ability of connecting surname to unknown's genome [34]. Reconstruction attacks based on available online datasets should be performed to measure the connection of surname or other unique identifier with genomic data.

2.4.3 Identity Tracing by Phenotypic Prediction

To prevent this threat, data about visible traits of individuals in public genomic databases as well as other public sources should be restricted (only to qualified researchers or close connections) or removed whenever applicable in order to preserve privacy. Nonetheless, predicting a victim's phenotypes is not only based on the revealed information through genetic databases; online social networks can also be a rich source of public sensitive data, and hence privacy risk will be amplified.

2.4.4 Completion Attacks

For this attack that relies on reconstructing genetic information based on partial data, one must consider all available data of each individual that is publicly shared (either by himself, his family members, or genomic researchers). If with existing completion techniques, one can predict the missing genomic information then specific parts of genomic data should be removed from datasets. Another solution is using dedicated cryptographic techniques, which enable researchers to access only some parts of the genome by requesting the decryption key from the owner. Such solutions can be merged with the reconstruction attack model from [23] to infer the amount of risk that occurs with releasing new portions of data.

2.4.5 Attribute Disclosure Attacks via DNA (ADAD) and Membership Attacks

To address this threat, data perturbation techniques (e.g., differential privacy) can be used for adding noise to the result of a query (on a genomic database) before releasing it publicly. In this way, the reported result will not be much different than original result, but an adversary will not understand if a given individual is in the database or not. Assuming the genomic database includes individuals with a given sensitive attribute, an adversary with prior knowledge can never be sure if that sensitive attribute belongs to a specific individual, as similar results will be given when the individual is included in the database or not. However, the added noise should be carefully considered as it will affect the accuracy and the utility of the data at the expense of privacy. Existing work mainly consider DP as a protective measure against the inference attack discovered by [14]. In the following section, we discuss the DP mechanisms and it's limitations.

2.5 Differential Privacy

2.5.1 Overview

Differential privacy provides formal guarantees that the distribution of query results change only slightly with the addition or removal of a single tuple in the dataset [53]. In other words, for any two neighboring input datasets T and T', using a probabilistic mechanism A will induce output distributions $A\ (T)$ and $A(T')$ with probabilities differing by a bounded multiplicative factor e^{ϵ}.

Definition 2.5.1. ϵ-Differential Privacy [19, 54] A randomized algorithm A gives ϵ-differential privacy if for any pair of neighboring datasets T and T', and any $O \subseteq Range\,(A)$, $Pr\,[A\,(T) \in O] \leq e^{\epsilon} Pr\,[A\,(T') \in O]$.

Based on the condition of these two neighboring datasets, DP is categorized as

bounded and unbounded. In bounded DP (BDP), all datasets T and T' have a fixed size, while in unbounded DP (UDP), they differ in size by one tuple. In order to achieve DP, there are different general general approaches.The most common three approaches are: First, by adding Laplace noise proportional to the query's global sensitivity [54]. Second, by adding noise related to the smooth bound of the query's local sensitivity [55]. Finally, by using the exponential mechanism to select a result among all possible results [56].

2.5.2 Laplace Perturbation Mechanism (LPM)

If a mechanism A produces outputs, the way to achieve ε-DP is by perturbing the query results with noise drawn from a suitable Laplace distribution before their release. Let A be a mechanism computing a query function Q. Then, if on dataset T, A outputs $Q(T) + \alpha$, where α is drawn from a Laplace distribution with mean 0 and scale $\Delta Q/\varepsilon$ then A satisfies ε-DP, where ΔQ is the global sensitivity computed for the issued query function Q [54].

2.5.3 DP-Based Privacy Mechanisms

Many techniques are proposed to achieve DP for many data types [16]. Mainly due to the utility trade-off of DP, alternative privacy relaxations of differential privacy were proposed to achieve higher utility. Many previous works have attempted to propose privacy-preserving mechanisms, such as k-anonymity, l-diversity, t-closeness, and differential privacy.

With an adversarial model that assumes the adversary to have full knowledge of all entities in the datasets except one (t) and a partial knowledge of that unknown entity t, a popular line of work provides relaxed privacy notions such as (ϵ,δ) differential-privacy [53], ϵ-privacy [57], differential identifiability [58], differential-privacy under sampling [59], membership privacy [60], differential privacy with bounded priors [61], crowd-blending privacy [62], coupled-worlds privacy [63], and

outlier privacy [64].

2.5.4 Inference Attacks Against DP-Based Mechanisms

The auxiliary information the adversary may learn from other channels is a big challenge. For instance, [65] use differentially private query results to infer a patient's genomic marker by utilizing additional information about the patient demographic information.

The strong dependence between the tuples in the real-world datasets introduces many privacy inference attacks. [22] were the first to criticize the independent tuples assumption of DP. [19] consider predicting the user location from the differentially private clustering query results by utilizing pairwise dependencies between users using Gowalla dataset [19].

2.5.5 DP for Privacy-Preserving Release of GWAS Results

[48, 49] developed many differentially-private algorithms that can be applied to release the statistical results for genomic studies, such as GWAS. For instance, according to [48, 49], Laplace noise with scale $2/\epsilon$ can be applied in order to get differentially-private cell counts from genomic datasets. [48, 49, 50] also developed algorithms that release differentially-private MAF and χ^2 results. In a case-control dataset with n individuals and m SNPs, under the assumption of equal number of cases $s = \frac{n}{2}$ and controls $c = \frac{n}{2}$, [48] computed the sensitivity for privacy-preserving release of MAFs as $\frac{2m}{n}$ and χ^2 statistics as $\frac{4n}{(n+2)}$ (based on 2×3 contingency tables). [50] claimed that adding Laplace noise with scale $\frac{2}{\epsilon}$ to the counts of any 2×2 contingency tables results in accurate χ^2 statistics or p-values. [49] assumed that the adversary can know the complete information of the individuals in the control group using the publicly available datasets. Hence, the sensitivity for privacy-preserving release of the χ^2 statistics is computed as

$\frac{n^2}{SC}\ \frac{C_{max}}{(C_{max}+1)}$, where $C_{max} = max(C_0, C_1, C_2)$ (based on 2×3 contingency tables).

In general, these works develop algorithms try to achieve DP when releasing statistics about genomic datasets or studies. However, they do not consider the correlation between the dataset tuples, and hence their privacy guarantees weaken when such correlations exists within the dataset.

2.5.6 Handling dependent tuples for DP

Handling dependent tuples is a significant challenge to guarantee privacy. [66] propose the Pufferfish framework as a generalization of DP to provide rigorous privacy guarantees against adversaries with access to any auxiliary background information and have a belief about the relationships between data tuples. However, no perturbation algorithm is proposed to handle the tuple dependencies. Blowfish [67] is a subclass of Pufferfish, considering the data correlations and adversarial prior knowledge specified by the users in the form of deterministic constraints. [67] provide perturbation mechanisms to handle these constraints. [68] handle the correlation in network data using DP by multiplying the original sensitivity of the query with the number of correlated records. This approach result in deteriorating the utility performance of the shared query results since an excessive amount of noise is added to the dataset. Bayesian DP [69] uses a modification of Pufferfish. In [69], they propose a perturbation mechanisms which consider the adversary's prior information and the correlations between data tuples. They only focus on the data correlations which can be modeled by Gaussian Markov Random Fields. To quantify the privacy loss when applying the traditional DP for continuous aggregate data release, [70] consider the temporal correlation which can be also modeled by a Markov Chain.

[19] define the dependent differential privacy (DDP) to protect the privacy of an individual's location information in a correlated dataset. They propose a Laplace mechanism to tackle the pairwise correlations in the dataset by computing the distance between any two tuples. Recently, [24] concretized the Pufferfish privacy. They propose the Wasserstein mechanism. The definition of the ϵ-DP

for correlated data in [24] is the same as in [19]. To satisfy that definition, the Wasserstein mechanism offers a weaker privacy budget. [18] improve the prior work of [19] by presenting a new definition of DDP. The privacy guarantees of DDP address any adversary with arbitrary correlation knowledge. They propose using the Laplace mechanism to handle the numeric queries and exponential mechanism to handle the non-numeric ones. However, these studies [18, 24, 19] provide less privacy and utility than our mechanism as we show in Section 7.6.

2.6 Summary

We survey in this chapter the existing inference attacks threaten the genomic privacy and the key directions that aim at providing privacy guarantees for statistical genomic studies. We analyze these approaches and provide the pros and cons of each direction.

Chapter 3

Attribute Inference Attack For Count Query

3.1 Overview

Two major threats against statistical datasets are membership inference and attribute inference. In this chapter, we do not consider membership inference attacks, and we focus on the attribute inference attacks using a simple count query over two real-world statistical genomic datasets.

3.2 Threat Model

Based on the noise added to the query results, the DP mechanism probabilistically guarantees that users' sensitive data are protected regardless of adversary's prior knowledge about the dataset. However, the privacy guarantees provided by the existing DP mechanisms do not account for the dependence between the data tuples. They assume that the dataset tuples are independent. In fact, this assumption can degrade the privacy of the data from different users as they can

be dependent due to various interactions.

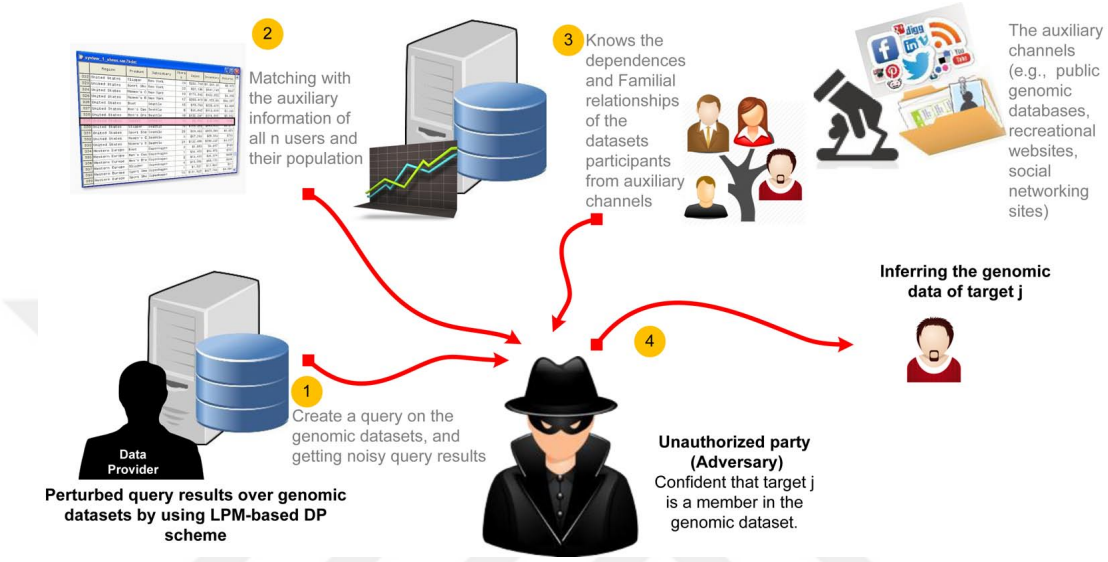


Figure 3.1: The threat model. The adversary does not have any prior knowledge about the genomic data of target j, but the adversary may have partial prior knowledge K for other members' genomic data. First, the adversary sends a query to the data provider. The data provider sends back the results with added noise using Laplace perturbation mechanism. Second, the adversary identifies the individuals that are used to generate the query result using the metadata that is released along with the dataset (e.g., population). That is, the adversary identifies how many of the target's family members and unrelated individuals are used to generate the query result. Next, the adversary uses other auxiliary channels to learn the familial relationship of target j with his family members that are (i) in the dataset and (ii) used to generate the query result. Finally, using the noisy query results along with the auxiliary information and the probabilistic dependence between tuples, the adversary infers the genomic record of target j.

An adversary can use auxiliary information channels to learn about such dependencies in the dataset and exploit the vulnerabilities in DP mechanisms as illustrated by [19]. The goal of the adversary in our model is to infer genomic data of a target individual.

We follow the same attack model in [19]. We assume that the adversary has

access to the membership of all participants in the dataset of n individuals. This may be possible by using the metadata that is released along with the dataset (e.g., in 1000 Genomes release, metadata includes the populations of the dataset members). However, the adversary in our threat model is more powerful than the DP adversary since he can also access auxiliary channels to estimate the relationship (or dependency) between tuples. The DP adversary do not consider the correlations between tuples in the dataset. To attain his goal, the adversary in our model will exploit the presence of target's family members in the same dataset and apply Mendelian inheritance rules to estimate the SNP values of the target. We explain all Mendelian inheritance probabilities, in Section 2.2. With this adversary model, we first perform an inference attack on the Laplace perturbation mechanism (LPM)-based differentially private data release to demonstrate that a powerful adversary can extract more information than that guaranteed by DP.

In our attack scenario (Figure 3.1), the adversary is confident that the target j is a member of the dataset and some of his family members are also in the dataset. Also, the adversary may have some prior knowledge about the genomic data of target's family members. We represent the amount of such information as (i.e., K represents the fraction of prior information of the adversary about the genomic data of target's family members). The adversary combines the released noisy query results (that are compliant with DP) with knowledge of the existing dependence relations to infer the genomic data of the target (which is not available to the adversary before the attack).

3.3 Dataset Description

For the evaluation, we use the genomic data of the family members from two datasets. Then, to get the unrelated members' genomic data, we use another dataset. Finally, we combined the family genomic data with the others genomic data. Hence, our final two datasets contain the partial DNA sequences from three sources:

- 1000 Genomes Phase 3 data
- CEPH/Utah Pedigree 1463
- Manuel Corpas Family Pedigree

3.3.1 1000 Genomes Phase 3 data

We use data from 1000 Genomes Phase 3 with 2504 individuals from 26 populations. We extract the genotypes from chromosome 1 and chromosome 22 using the Beagle genetic analysis package [71] to convert the values of genotypes to 0, 1, or 2 according to the minor alleles on each SNP. We use this data to include more participants to our dataset from the same or different population of the target and his family members. The main objective here is to test if the adversary can infer more sensitive information about the target even if the query results contain more unrelated participants.

3.3.2 CEPH/Utah Pedigree 1463

We use CEPH/Utah Pedigree 1463 with the partial DNA sequences of 10 family members [56]. In our inference attack, we consider the parent to be our target (Par 1 in Figure 3.2). We only focus on first degree relatives, and hence we use the genomic records of 1 parent, 2 grandparents, and 7 children.[1] We obtain the SNPs data for 10 individuals from the variant call format (VCF) file. The family bonds between these 10 individuals are illustrated in Figure 3.2. To evaluate our inference algorithm, we construct our dataset by extracting the common SNPs from all family members. We identify 259K common SNPs between family members for the 240M positions in chromosome 1. Then, we identify 9.8K common SNPs (between position 10K and position 23M) between 1000 Genomes members

[1]The original CEPH/Utah Pedigree 1463 includes data for 11 children, we randomly select 7 of them for our evaluation.

and UTAH family members. From these common SNPs, we select 100 SNPs to apply our inference algorithm.

3.3.3 Manuel Corpas (MC) Family Pedigree

A scientist named Manuel Corpas decided to release his family DNA dataset for research purposes. The dataset contains the DNA sequences in variant call format (VCF) for the father, mother, son (Manuel Corpas), daughter, and aunt. The family tree of the individuals in this dataset is illustrated in Figure 3.3(a). We choose the son to be our target and we used the genomic records of his first degree family members (father, mother, and sister) as in Figure 3.3(b). Similar to the Utah family dataset, we extract the common SNPs in all MC family members. This results in 8.057K common SNPs between the 4 MC members in chromosome 22. Then, we extract the common SNPs between 1000 Genomes members and MC family members and we select 100 of such common SNPs for the evaluation of our inference algorithm.

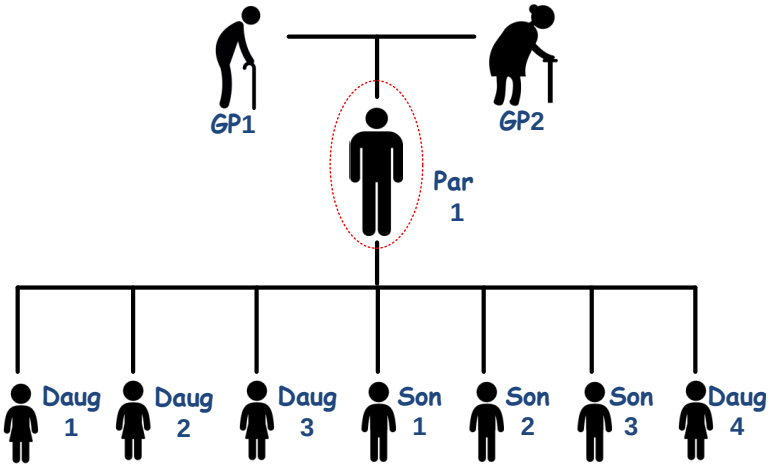


Figure 3.2: CEPH/Utah Family Tree, where the target individual is "Par 1".

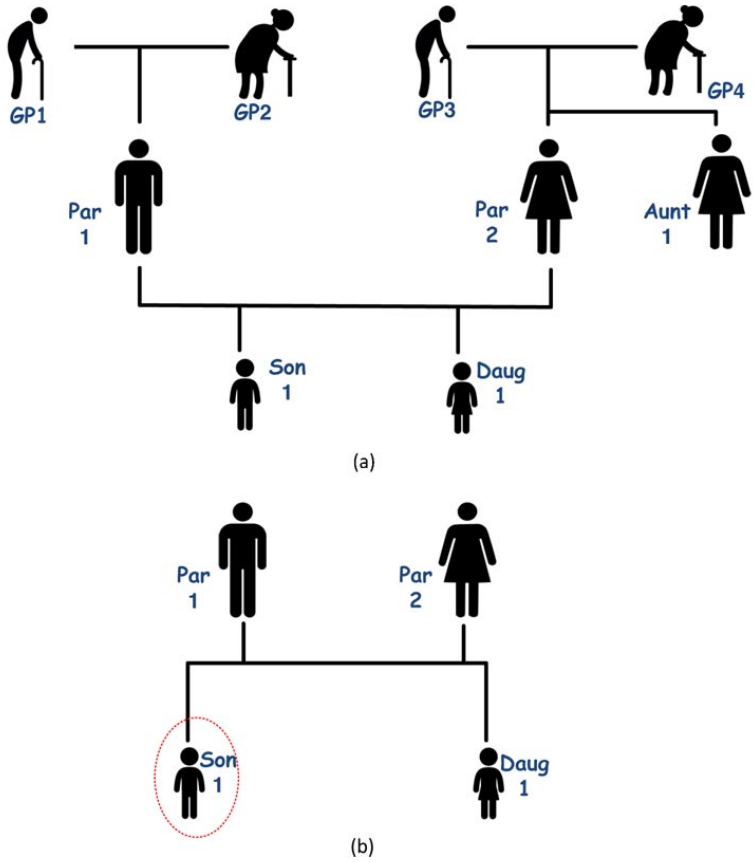


Figure 3.3: Manuel Corpas Family Tree. (a) The full tree of the family consisting of a son and a daughter, their father and mother, 4 grandparents, and an aunt. (b) The 4 family members (the son, his sister, father, and mother) we consider in our inference attack, where the target individual is "Son 1".

3.4 Differential Privacy under Dependent Tuples

As we discussed in Section 3.2, DP mechanism does not account for the dependency of the data tuples in the dataset. On the other hand, family members' genotypes are inherently correlated and this correlation is stronger between close family members. Thus, existence of individuals from a target individual's family may provide an important source for any adversary to infer the target's genomic

data even though the unavailability of their genomic records. This privacy breach has been proven by [23].

In our scenario, the adversary sent his query asking about the total number of a specific SNP i for participants sharing the same demographic data, such as location or age. The adversary gets the noisy result of his query, $\tilde{T^i_{pj}} = (T^i_p + T^i_j) + \delta$, for (p) participants included in the query results and individual j. δ represents the added Laplace noise with parameter $2/\epsilon$, T^i_j represents the SNP value for individual j, and T^i_p is the sum of the SNP values for other (p) participants. According to the query statement, the query results may include only the target j's related family members or also other unrelated individuals. Hence, the probabilistic dependence can be considered as:

$$T^i_p = T^i_j + Dy, \tag{3.1}$$

where $D = p$ if $p \leq 2$ and $D = 2p$ if $p > 2$ (p is the number of all individuals included in the query result except the target j).

Also, y is a a kinship coefficient that satisfies the Mendelian inheritance laws. y is in $[-1, 1]$ for $p \leq 1$, and y is in $[0, 1]$ for $p > 1$. We derive Equation 3.1 from all the Mendelian inheritance law discussed in Section 2.2. As a simple example, assume we have a family of 3 members in the dataset (father, mother, and son). Consider an inference attack, in which an adversary targets the son and issues a sum query Q(T) for a particular SNP position in the genome. Also, assume that the query result is computed only over these 3 family members and the result of the query is $Q(T) = T^i_{pj} = (T^i_p + T^i_j) = 3$, where T^i_j is the SNP value of the son (target) and T^i_p is the sum of SNP values of father and mother. The adversary uses the query results to infer T^i_j. Therefore, a powerful adversary can exploit the dependence between these $p+1$ tuples where $p = 2$ (father and mother of target (j)). Using the probabilistic dependence between T_p and T_j as in Equation 3.1, $D = 2p = 4$. When we consider the 3 possibilities (0, 1, and 2) for the value of T_j, we end up having the below cases for the value of T_p (since Tp+Tj = 3):

(i) $2 = 1 + 4(0.25)$. The father's and mother's SNP values = 2, the target's SNP value = 1.

(ii) $1 = 2 + 4(-0.25)$. The father's and mother's SNP values = 1, the target's SNP value = 2.

(iii) $3 = 0 + 4(0.75)$. The father's and mother's SNP values = 3, the target's SNP value = 0.

According to Mendel's law, (ii) and (iii) are not possible. Since if the sum of father's and mother's SNP values = 1, their son's SNP value can only be 0 or 1. Similarly, if the sum of father's and mother's SNP values = 3, their son's SNP value can only be 1 or 2; not 0. Hence, the adversary only considers (i). Using Mendel's law for different possibilities for the target's SNP value, we formulate Equation 3.1 and we give the range of y as a kinship coefficient, as well as D value for different p values.

3.5 Inference Evaluation Algorithm

We assume that the adversary can query the dataset based on the demographics of the dataset participants. As a result of his query, the adversary obtains the differentially private sum of genotype values $\tilde{T^i_{pj}} = (T^i_p + T^i_j)$ for different cases, e.g.:

- Total value of a SNP for the people from the same location area, or address.
- Total value of a SNP for the people with the same age.

The adversary has access to auxiliary information about the membership of each participant including the target j, and also to the familial relationship between the target and other individuals in the dataset. Hence, the adversary can infer the value of T^i_j for target j using the number of dependent people related to that member in the dataset. We use two metrics to quantify the success of the attacks: correctness and leaked information. Correctness quantifies the distance *Dist* between the true value of the SNP and the inferred value by the adversary.

The leaked information quantifies the change in the adversary's prior information after the inference attack. To measure the correctness we use the expected estimation error as follows:

$$E = \sum_{i=1}^{m} P\left(x_{ij} \mid T_j^i\right) \left|Dist\left(x_{ij}, x'_{ij}\right)\right|, \tag{3.2}$$

To measure the leaked information we use the following equation:

$$L = \sum_{i=1}^{m} 1 - |sgn(Dist(x_{ij}, x'_{ij}))|, \tag{3.3}$$

where m is the number of targeted SNPs, and *sgn* denotes the sign function, which extract the sign of any real number. *sgn* gives the value of 1 for all positive real numbers, 0 for number 0, and -1 for all negative real numbers. Hence in Equation 3.3, if there is any difference between x_{ij} which is the true value of SNP i for the target individual i and x'_{ij} which is the estimated value of SNP i for the target individual i, it means the adversary could not infer the correct value of the SNP and the SNP information is not leaked. We use Algorithm 1 for evaluating the correctness and leaked information.

First, the adversary forms the queries that cover the members of the same family (e.g., by forming a query based on age - location - street level - city level - state level, etc.) to get $\tilde{T^i_{pj}}$. Then, the adversary uses the coin change algorithm [72] to get all the possible partitions of a specific number. The coin change algorithm extracts the list of all partitions of a number. In our scenario this number is the total sum of the SNP values $\tilde{T^i_{pj}}$. Considering each SNP can get a value from the set {0, 1, 2} for the noisy sum $\tilde{T^i_{pj}}$, the coin change algorithm generates the $\tilde{T^i_{pj}}$ as an integer combination of the set {0, 1, 2} as described in Algorithm 1 (line 1). Each partition should only include $\leq (p+1)$ individuals. After that, the algorithm finds all the unique permutations for each valid partition. If the adversary considers the dependency assumption ($Dep = 1$), the algorithm checks the validity of each permutation using Mendel's law. Once validated, the algorithm computes the probability of each valid permutation from Mendel's law by considering potential values of SNP i (0, 1, and 2) for individual j (line 2 to line 15). Finally, the algorithm compares the inferred SNP value with the real value and computes the error using Equation 3.2 (line 16).

For the independent assumption ($Dep = 0$), the adversary does not utilize the Mendel's law for his inference. The adversary uses his prior information about the number of individuals included in the query results to estimate the values of all SNPs for j. The adversary divides each permutation by $(p+1)$ individuals, then computes the probability of each permutation, by considering potential values of SNP i for individual j. Algorithm 1 can also be used for evaluating the number of leaked information.

We assume that the adversary knows the ϵ value used to provide the noisy query results. For instance, if $\epsilon > 4$, the adversary expects Laplace distribution noise of +/- 2 over the $\tilde{T^i_{pj}}$. Then, for each possible value of $\tilde{T^i_{pj}}$, the adversary finds the possible partitions, and the permutations for each partition, validates the permutations using Mendel's tables, and computes the probabilities for each permutation (line 1 through line 15). Subsequently, he chooses the SNP value with the largest probability as individual j's inferred SNP value.

Algorithm 1 Evaluations

Input: $sum\ \tilde{T}^i_{pj},\ \ p,\ \ R,\ \ Dep$

Output: The expected estimation error (E), Leaked Information (L)

```
pList ← Partitions(sum, [0, 1, 2])                    ▷ Coin Change Problem
for i ← 1  to  pList.size() do
    if ∑(pList(i)) > P + 1 then
        delete(pList(i))
    end if
end for
for i ← 1  to  pList.size() do
    V ← perms(pList(i))
    B ← unique(V)
    for j ← 1  to  B.size() do
        if Dep ← 1 then checkMendel(B(i))
            if checkMendel(B(i)) ← False then
                delete(B(i))
            end if
        end if
        findProb(B(i))
    end for
end for
Apply Equation 3.2 or 3.3
```

3.6 Evaluation

As discussed in Section 3.3, we use two datasets to evaluate the proposed attack model. We define both datasets as T that include n individuals (n= 2514 for the first dataset and n= 2508 for the second one). *SNP* is the set of SNP IDs on chromosome 1 and chromosome 22, and m is the number of SNPs for each individual (m = 100 for each dataset). To infer the values of these m SNPs, 100 queries are performed for each dataset. T^i_j represents the value of a SNP

i ($i \in SNP$) for individual j ($j \in T$). The set $\mathbf{F}$ represent the family members ($|\mathbf{F}| = f$) and the set $\mathbf{U}$ represent other unrelated individuals($|\mathbf{U}| = u$).

In the proposed inference attacks we assume the differentially private query results that are computed including individuals for different cases as follows:

- Case 1: Individual j with a direct family member in set $\mathbf{F}$ ($|\mathbf{F}| = 1$).
- Case 2: Individual j with multiple family members in set $\mathbf{F}$ ($|\mathbf{F}| = f$).
- Case 3: Individual j with multiple family members in set $\mathbf{F}$ ($|\mathbf{F}| = f$), and other unrelated individuals in set $\mathbf{U}$ ($|\mathbf{U}| = u$).

We evaluate the performance of the attack for these cases considering two different types of attacks: (i) the adversary assumes that there is no correlation between individuals, and (ii) the adversary utilizes the genomic association between individuals to do genome reconstruction and infer genomic data. We use the algorithm described in Section 3.5 to quantify the success of the attacks by evaluating the two metrics: correctness and leaked information.

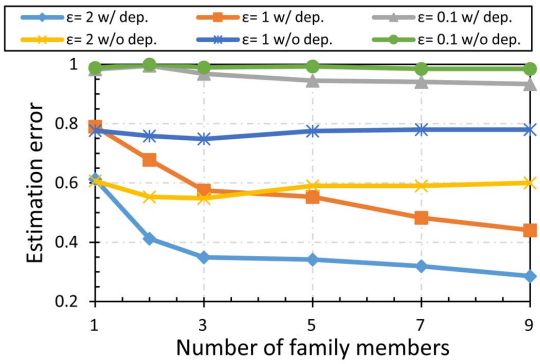
3.7 Experimental Results

The adversary aims to construct individual j's genomic record, while the adversary only knows the membership of the individual and his family members in the dataset. We compare the dependent and independent assumptions to show the vulnerability of independent assumption and to come up with countermeasures for dependent cases.

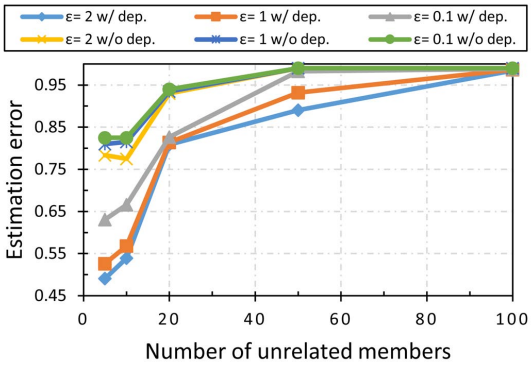
3.7.1 Estimation Error Experiment

In Figure 3.4, we examine the effect of the number of relatives and non-relatives included in the result of a query to the target dataset on adversary's success in

terms of his correctness in inferring SNPs of individual j.



(a)



(b)

Figure 3.4: The effect of (a) including only first degree relatives in set $\mathbf{F}$ and (b) including 9 first degree relatives ($|\mathbf{F}| = 9$) with different numbers of non-relatives in set $\mathbf{U}$ in the query results, on the probability of the adversary's correctness (1-estimation error) in inferring the targeted SNPs.

We make three key observations: (i) In Figure 3.4(a), the adversary is able to infer the targeted SNPs ($m = 100$) more accurately as the number of family members included in the query computation increases. We start with 1 first degree relative which can be the father or the mother. Then, we gradually include the sons of individual j together with his father and mother. We observe that if the query results include data for more than 4 first degree relatives of the same family, then the correctness of the adversary converges for $\epsilon \leq 2$. (ii) In Figure 3.4(a), based on the correctness metric, we observe that if the adversary has the knowledge that the data of relatives (i.e., dependent tuples) exist in the target dataset, then the adversary's observation of the targeted SNPs is up to 2 times (depending

on the value of ϵ) more accurate compared to not having this knowledge. As expected, we also observe that the difference between the correctness of the inferred SNPs with and without the knowledge of the data dependency increases as the value of the privacy budget, ϵ, increases. (iii) In Figure 3.4(b), we observe that including the 9 first degree relatives and increasing the number of non-relatives included in results of the queries from 5 to 100 decreases significantly the ability of the adversary to infer the actual value of the targeted SNPs by about 20-50%, even if the adversary has the knowledge of the data dependency. Increasing the number of non-relatives beyond 100 members leads to the mitigation (with a probability of 0.99) of the leakage of SNP information of the participants. These heuristic results show the estimated scale of vulnerability that occurs when we have dependent tuples in a dataset that responds to queries based on DP.

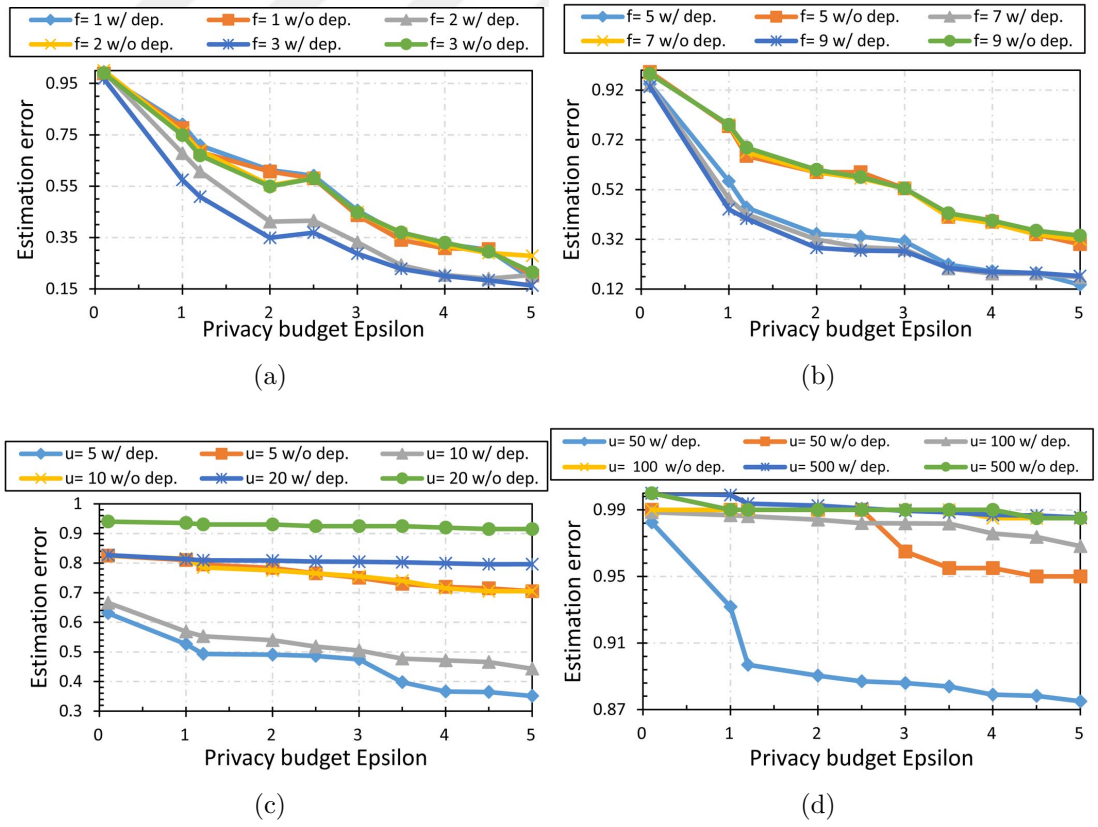


Figure 3.5: The effect of different values of the privacy budget, ϵ, and the number of (a)(b) family relatives in set $\mathbf{F}$ ($|\mathbf{F}| = f$) and (c)(d) non-relatives in set $\mathbf{U}$ ($|\mathbf{U}| = u$) on adversary correctness(1-estimation error) in inferring the targeted SNPs.

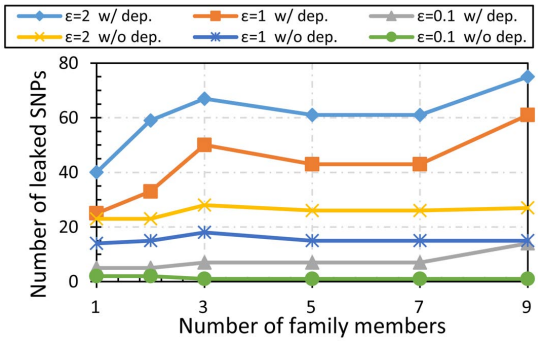
Next, in Figure 5.1, we evaluate the effect of different values of the privacy budget, ϵ, on the adversary's correctness in inferring the targeted SNPs. We make two key observations: (i) The correctness of the adversary for the actual values of the targeted SNPs decreases considerably (by 70% to 90%) as the budget privacy, ϵ, increases from 0.1 to 5 when three members of the same family are included in the query results, as shown in Figure 5.1(a). We also observe a similar trend in Figure 5.1(b), where the query results contains data of 9 family members. However, in Figure 5.1(b), we observe that the difference between the correctness of the inferred SNPs with and without the knowledge of the data dependency is about 2 times larger than when the query results include data for at most 3 members of the same family (Figure 5.1(a)).

(ii) The probability of inferring the wrong value of the targeted SNPs slightly decreases as the value of the privacy budget, ϵ, increases from 0.1 to 5, in both Figures S5.1(c) and (d).

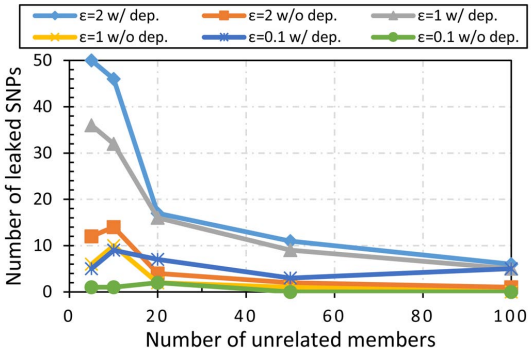
3.7.2 Leaked Information Experiment

We show in Figure 3.6 the number of leaked SNPs the adversary can infer about individual j with the presence of a different number of first degree relatives and non-relatives included in the query results. We make two key observations: (i) In Figure 3.6(a), if the adversary has the knowledge about the tuples correlation due to the familial relationship in the dataset, then the number of leaked SNPs increases by 10%-40% (depending on the value of the ϵ used) more accurate compared to not having this knowledge, as the number of family members included in dataset increases from 1 member to 9 members. We start with 1 first degree relative which can be the father or the mother. Then, we gradually include the sons of individual j together with his father and mother. We also observe that if the adversary does not have the knowledge about the data dependency, then increasing the number of family relatives up to 9 members increases the number of leaked SNPs by up to only 5%.

(ii) In Figure 3.6(b), based on the leaked information metric, if the adversary



(a)



(b)

Figure 3.6: The effect of (a) including only first degree relatives in set **F** and (b) including 9 first degree relatives in set **F** ($|\mathbf{F}| = 9$) with different numbers of non-relatives in set **U** in the query results, on the target's leaked SNPs the adversary can infer.

has the knowledge that the data of relatives (i.e., dependent tuples) exist in the dataset, then increasing the number of non-relatives covered in the dataset from 5 to 100 with including the 9 first degree relatives, decreases significantly the number of actual values of the targeted SNPs the adversary can infer by about 45%. The number of the leaked targeted SNPs decreases by up to 14% as the number of non-relatives increases from 5 to 100 and the adversary has no knowledge about the data dependency.

Increasing the number of non-relatives beyond 100 members leads to the mitigation (with a probability of 0.95) of the leakage of the targeted SNPs information, since only about 5 leaked SNPs of individual j can be inferred, even if the adversary has the knowledge of the data dependency. Giving that the adversary

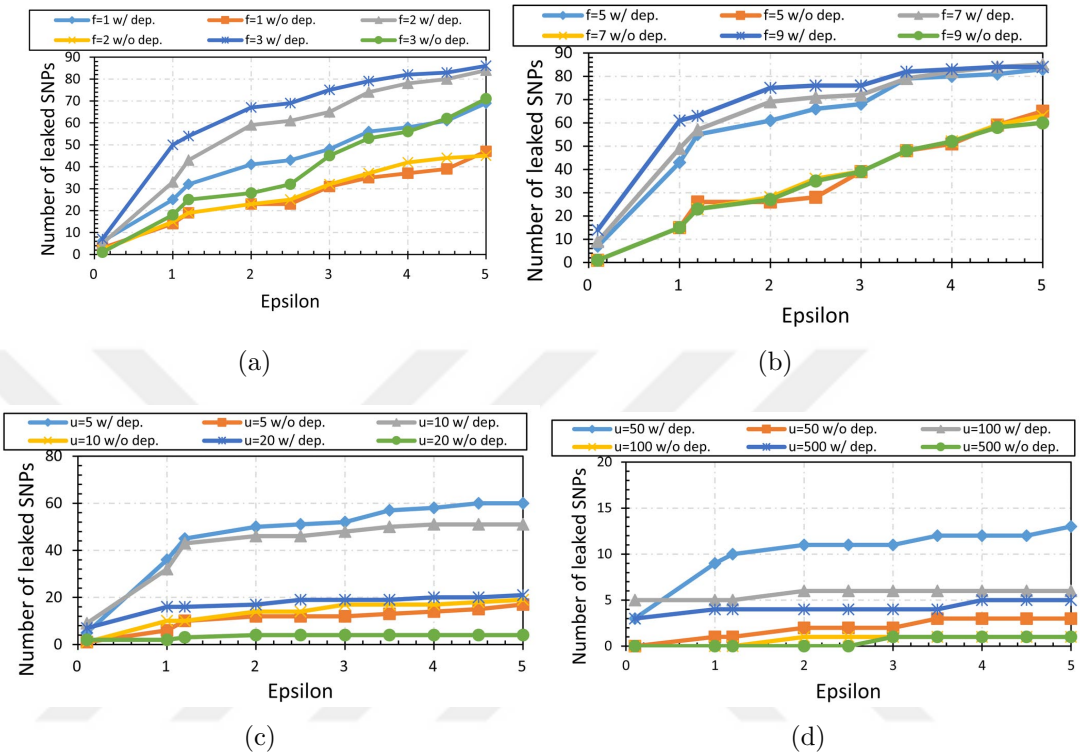


Figure 3.7: The effect of different values of the privacy budget, ϵ, and the number of (a)(b) family relatives in set $\mathbf{F}$ ($|\mathbf{F}| = f$) and (c)(d) non-relatives in set $\mathbf{U}$ ($|\mathbf{U}| = u$) on adversary's success in inferring the targeted SNPs.

has no knowledge about the data dependency, The number of leaked SNPs the adversary can infer drops to only 2 SNPs with the presence of data for more than 100 non-relative members.

Next, in Figure 3.7, we evaluate the effect of different values of the privacy budget, ϵ, on the number of the target's leaked SNPs the adversary can infer. We make two key observations: (i) Increasing the privacy budget ϵ from 0.1 to 5 allows for 4 times to 8 times (depending on the number of relatives and whether or not the adversary has the knowledge of the data dependency) more SNPs to be revealed by the adversary with the existence of data of up to 3 members of the same family, as shown in Figure 3.7(a).

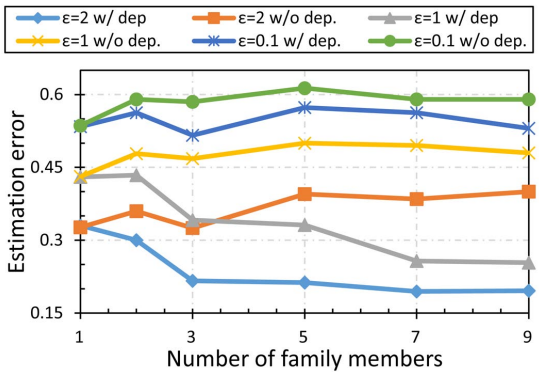
However, increasing the number of relatives to 9 family members (Figure 3.7(b)) provides a similar trend like that we observe in including only 3

family members (Figure 3.7(a)). The number of leaked SNPs increases by up to 7 times as the privacy budget ϵ increases from 0.1 to 5 and the adversary has the knowledge of the dependency between tuples, as we show in (Figure 3.7(b)). We also observe that if the adversary does not have the knowledge about the data dependency, then the number of leaked SNPs increases by up to 6 times as the privacy budget increases from 0.1 to 5, as we show in (Figure 3.7(b)). We also observe that varying the number of family relatives from 5 to 9 has a slight effect on the number of leaked SNPs, if the adversary has the data dependency knowledge. Otherwise, it almost has no effect on the number of leaked SNPs. This observation is in accordance with our previous observations in Figure 3.6.(ii) Increasing the privacy budget ϵ from 0.1 to 5 increases the number of leaked SNPs by only up to 2 times if the adversary does not have the knowledge of the data dependency and regardless the number of the non-relatives, as we provide in both Figures S3.7(c) and (d). If the adversary has the knowledge of the data dependency, then increasing the privacy budget ϵ from 0.1 to 5 increases the number of leaked SNPs significantly (up to 5 times in Figure 3.7(c) and up to 6 times in Figure 3.7(d)). However, increasing the number of non-relatives covered in the dataset to 500 members allows for only up to 5 SNPs to be inferred by the adversary, even if the adversary has the knowledge of the data dependency, as we show in Figure 3.7(d).

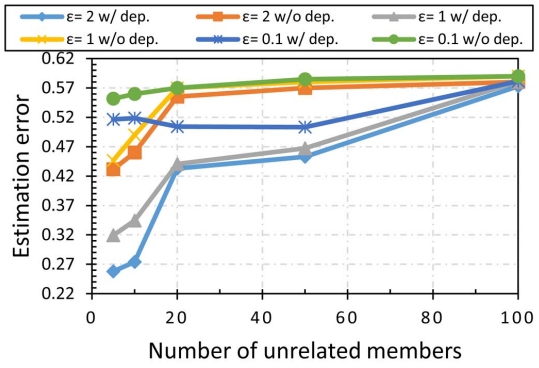
3.7.3 Estimation Error Experiment for Stronger Adversary

Finally, We consider a stronger adversary who has access to partial information, e.g., $K = 50\%$ of other d members' genomes included in the query results.

Figure 3.8 shows that the adversary who considers the familial relationship between tuples in any genomic dataset can infer more sensitive information than the DP adversary who ignores the dependence between tuples. If the query results include many uncorrelated individuals with the target's relatives, it is more difficult for the adversary to infer the genomic record of the target even though



(a)



(b)

Figure 3.8: The effect of the number of (a) family relatives in set **F** and (b) non-relatives in set **U** included in the query results (computed on the target dataset) on the adversary's correctness (1-estimation error) when the adversary has access to $K = 50\%$ of other d family members' genomes included in the query results.

he knows the dependence between tuples (Figure 3.8(b)). Moreover, Figure 3.8 shows that for any adversary with prior partial information $K = 50\%$, correctness of the targeted SNPs is considerably less (about 50%) than the correctness of the same targeted SNPs for any adversary without any prior information $K = 0\%$.

Next, Figure 3.9 shows the effect of different values of the privacy budget, ϵ, on adversary's success in terms of his correctness in inferring SNPs of individual j. We make two key observations: (i) The correctness of the actual values of the targeted SNPs decreases considerably (by 50% to 70%) as the budget privacy, ϵ, increases from 0.1 to 5 when up to 3 members of the same family are included in the query results, as shown in Figure 3.9(a).

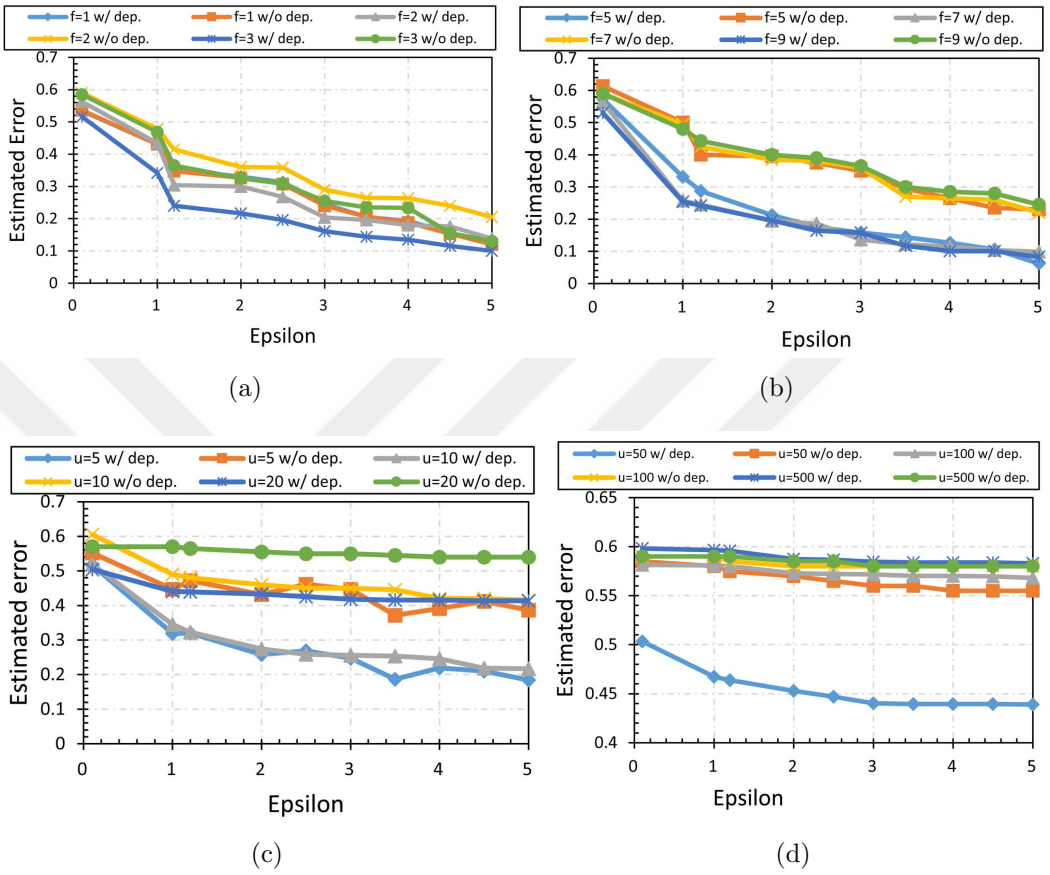


Figure 3.9: The effect of different values of the privacy budget, ϵ, and the number of (a)(b) family relatives in set $\mathbf{F}$ ($|\mathbf{F}| = f$) and (c)(d) non-relatives in set $\mathbf{U}$ ($|\mathbf{U}| = u$) on adversary's correctness (1-estimation error) when the adversary has access to $K = 50\%$ of other d family members' genomes included in the query results.

We also observe a similar trend in Figure 3.9(b), where the query results contains data of 9 family members. However, in Figure 3.9(b), we observe that the difference between the correctness of the inferred SNPs with and without the knowledge of the data dependency is about 2 times larger than when the query results include data for at most 3 members of the same family (Figure 3.9(a)).

(ii) The probability of inferring the wrong value of the targeted SNPs slightly decreases (mostly 2%-20% depending on the knowledge of the adversary and the number of non-relative individuals in the query results) as the value of the privacy budget, ϵ, increases from 0.1 to 5, in both Figures S3.9(c) and (d). Figures S3.9(a)

and (b) approve that the adversary who considers the familial relationship between tuples in any genomic dataset can infer more sensitive information than the DP adversary who ignores the dependence between tuples. Figures S3.9(c) and (d) also approve that if the query results include many uncorrelated individuals with the target's relatives, it will be more difficult for the adversary to infer the genomic record of the target even though he knows the dependence between tuples. Furthermore, Figure 3.9 shows that for any adversary with prior partial information $K = 50\%$, the correctness of the targeted SNPs is considerably less (about 50%) than the correctness of the same targeted SNPs for any adversary without any prior information $K = 0\%$.

3.7.4 Leaked Information Experiment for Stronger Adversary

Figure 3.10 shows the effect of different values of privacy budget, ϵ, and different numbers of family relatives and non-relatives included in the result of a query, on adversary's success in terms of the number of leaked information about individual j. We make two key observations: (i) Increasing the privacy budget ϵ from 0.1 to 5 allows for 2 times to 3 times (depending on the number of relatives and whether or not the adversary has the knowledge of the data dependency) more SNPs to be revealed by the adversary with the existence of data of 3 members of the same family, as shown in Figure 3.10(a).

Moreover, increasing the number of relatives to 9 family members (Figure 3.10(b)) can increase the number of leaked SNPs by up to $2\times$ as the privacy budget ϵ increases from 0.1 to 5. We also observe that varying the number of family relatives from 5 to 9 has a slight effect on the number of leaked SNPs, if the adversary has the data dependency knowledge. (ii) Increasing the privacy budget ϵ from 0.1 to 5 increases the number of leaked SNPs by only up to 2 times if the adversary does not have the knowledge of the data dependency and regardless the number of the non-relatives, as we provide in both Figures S3.10(c) and (d). If the adversary has the knowledge of the data dependency, then increasing the

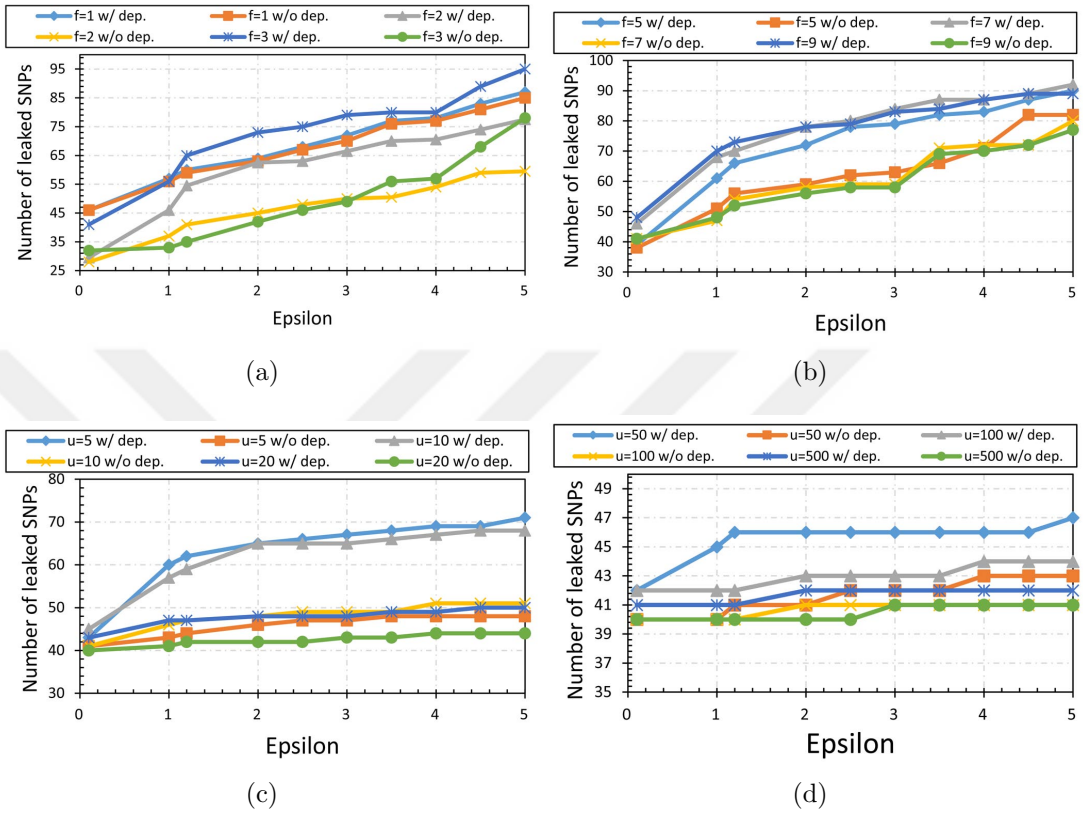


Figure 3.10: The effect of different values of the privacy budget, ϵ, and the number of (a)(b) family relatives in set $\mathbf{F}$ ($|\mathbf{F}| = f$) and (c)(d) non-relatives in set $\mathbf{U}$ ($|\mathbf{U}| = u$) on the leaked SNPs the adversary can infer when the adversary has access to $K = 50\%$ of other d family members' genomes included in the query results.

privacy budget ϵ from 0.1 to 5 increases the number of leaked SNPs significantly (up to 5 times in Figure 3.10(c) and up to 4 times in Figure 3.10(d)). However, increasing the number of non-relatives covered in the dataset to 500 members allows for only up to 5 SNPs to be inferred by the adversary, even if the adversary has the knowledge of the data dependency, as we show in Figure 3.10(d).

3.8 Summary

We introduce the first attribute inference attack using the DP-noisy results of a count query over two statistical genomic datasets. We explain the threat model,

describe the genomic datasets, develop the inference algorithm, show the evaluation and the experimental results. The results show that the adversary who considers the familial relationship between tuples in any genomic dataset can infer more sensitive information than the DP adversary who ignores the dependence between tuples. If the query results include many uncorrelated individuals with the target's relatives, it is more difficult for the adversary to infer the genomic record of the target even though he knows the dependence between tuples. Moreover, for any adversary with prior partial information $K = 50\%$, the correctness of the targeted SNPs is considerably less (about 50%) than the correctness of the same targeted SNPs for any adversary without any prior information $K = 0\%$.

Chapter 4

Attribute Inference Attack for Complex Queries

4.1 Overview

In this chapter, we focus on the attribute inference attacks using two complex queries (MAF and chi square) over two real-world statistical genomic datasets.

4.2 Threat Model

We consider two major threats against genomic datasets: attribute inference and membership inference. We notably aim to show how the outcome of attribute inference attack (that includes complex queries to a dataset) can be utilized in a membership inference attack. In this chapter we consider the attribute inference attack. Hence, in our scenario, the first goal of the adversary is (i) to infer sensitive genomic information about a target (e.g., target's rare SNPs) by sending queries to a dataset (for which the adversary knows the membership of a target and his family members).

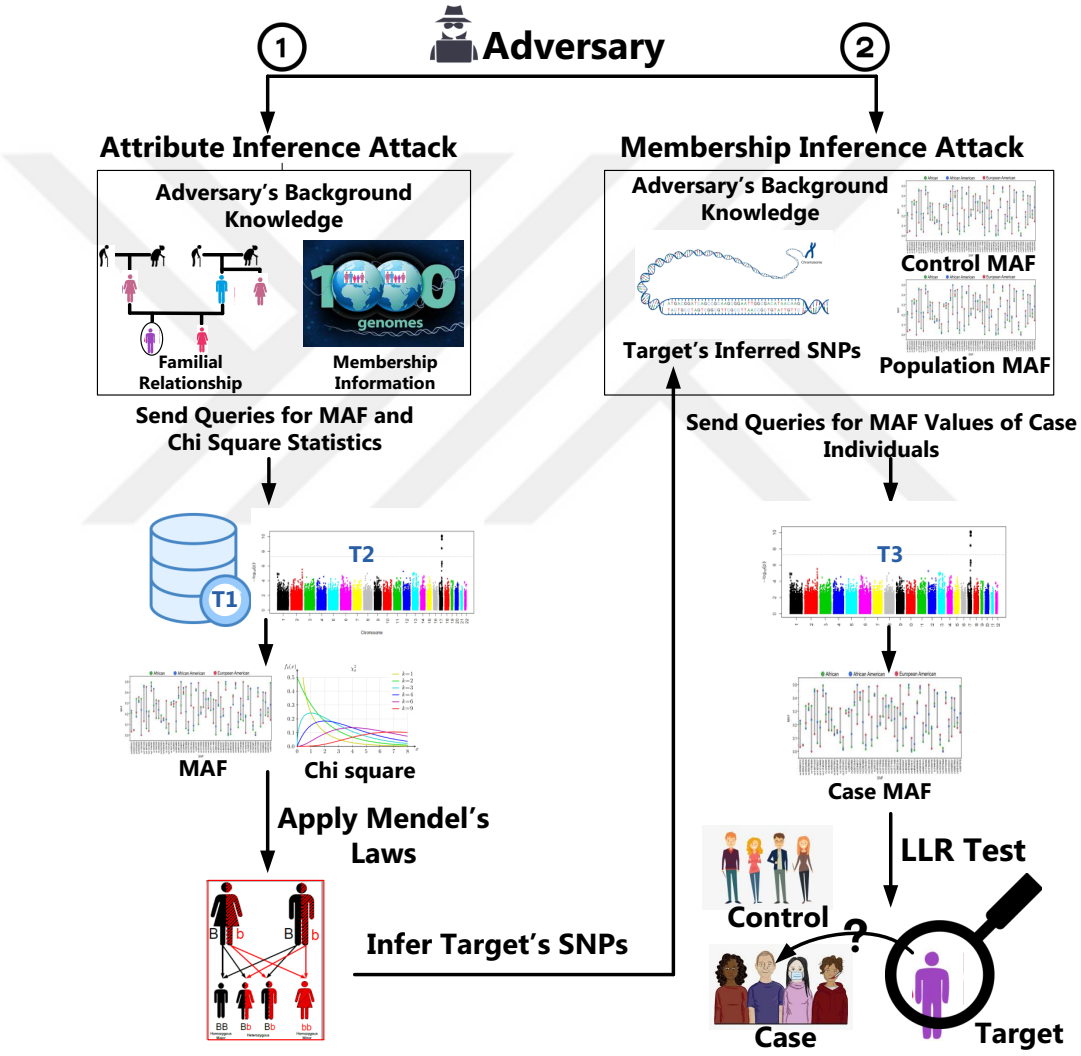


Figure 4.1: Considered threat model. The adversary first runs the attribute inference attack against the target by using (i) results of differentially-private MAF queries (from dataset T_1) or chi-square statistics (from dataset T_2), (ii) dependency between the target and target's family members that are in the dataset, and (iii) Mendel's laws. Using the inferred SNPs of the target, the adversary runs a membership inference attack to infer the membership of the target in the case population in dataset T_3 (via a log-likelihood-ratio - LLR test).

Then later in chapter 5 we show how the adversary can (ii) infer the target's membership to another genomic dataset (for which the membership information is not publicly available). The membership of an individual in a dataset means that the corresponding individual is included in the dataset. We also show this threat model in Figure 4.1. DP mechanism provides strong guarantees for protecting sensitive data of dataset members even if the adversary has prior information about the dataset. The amount of random perturbation added to the aggregate query results determines the privacy and accuracy levels. However, standard DP mechanisms do not consider the inherent correlations (or dependency) between the data tuples (e.g., correlations between genomes of family members in a genomic dataset), and hence their privacy guarantees degrade if such correlations are used by the adversary. An adversary can use auxiliary channels to learn about such dependencies in the dataset and exploit this vulnerability in DP mechanism as shown in [19, 1]. For the attribute inference, we follow the same attack model in [1] for the count query discussed in Chapter 3. We assume a stronger adversary than the standard DP adversary with the following assumptions: (i) the adversary has access to the membership information of all n members in the dataset. This is possible by using the publicly available metadata associated to the dataset (e.g., population information about dataset members are published along with the 1000 Genomes dataset). And, (ii) the adversary can estimate the dependency between the dataset tuples (e.g. familial relationships) using auxiliary channels. Using prior information about the familial relationships between a target j and his family members in a genomic dataset along with the released (noisy) query results, the adversary uses the Mendelian inheritance rules to infer the genomic records of target j (X_j). These Mendelian inheritance probabilities are shown in Table 2.1 and Figure 2.1 in Chapter 2. We study the attribute inference attack on the Laplace perturbation mechanism (LPM)-based differentially-private query results including two queries: minor allele frequency (MAF) and chi-square (χ^2).

4.3 Attribute Inference Attack for Complex Queries

In Chapter 3 [1], we show the attribute inference attack for datasets with dependent tuples by considering a simple count (sum) query, in which the adversary asks about the total number of a specific SNP i's value among a subset of the dataset participants (determined based on demographic data, such as location or age). Here, we analyze the attribute inference risk for more complex and realistic query types that are used in actual genomic studies (e.g., GWAS). In particular, we consider queries that compute minor allele frequency (MAF) and chi-square (χ^2) statistics of SNPs.

For a statistical dataset T_1, we represent a SNP i's (noisy or differentially-private) MAF value that is computed over target j and other dataset members in set $\mathbf{P}$ ($|\mathbf{P}| = p$) sharing some demographic information with j as $\frac{\widetilde{Q^i_{pj}}}{\tau} = \frac{Q^i_p}{\tau_1} + \frac{x^i_j}{2} + \delta$. Here, δ is the added Laplace noise suggested by [48] and x^i_j is the value of SNP i for target j. $\frac{Q^i_p}{\tau_1}$ is the MAF value of SNP i due to individuals in $\mathbf{P}$, where Q^i_p represents the number of minor alleles in $\mathbf{P}$ and τ_1 is the total number of alleles in $\mathbf{P}$ for SNP i (since each SNP carries 2 alleles, $\tau_1 = 2p$). Also, τ is the total number of alleles for all participants included in the query result ($\tau = 2p + 2$). Individuals in set $\mathbf{P}$ (except for the target) can be (i) the family members of target j, which we represent as a set $\mathbf{F}$ ($|\mathbf{F}| = f$) or (ii) other unrelated individuals in set $\mathbf{U}$ ($|\mathbf{U}| = u$). Hence, the probabilistic dependence for the MAF statistics can be shown as:

$$M^i_p = M^i_j + (p+1)y_1, \tag{4.1}$$

where M^i_p and M^i_j are the MAF values due to individuals in $\mathbf{P}$ and target j, respectively. $(p+1)$ is the number of all participants included in the query result ($\mathbf{P}$ and target j). Also, y_1 is a kinship coefficient that satisfies the Mendel's law. y_1 is in $\left[\frac{-2}{\tau}, \frac{2}{\tau}\right]$.

For a case-control dataset T_2, (noisy) χ^2 statistics for a SNP i when the query

results include target j and other dataset members in set $\mathbf{P}$ sharing some demographic data with j is represented as $\widetilde{\chi_i^2} = \chi_i^2 + \delta$. Here, δ represents the added Laplace noise suggested by [48, 49, 50]. Then, the probabilistic dependence for the χ^2 statistics can be considered as:

$$\chi_i^2 = T_{pj}^i + 2(p+1)y_2, \tag{4.2}$$

where T_{pj}^i is the sum of the SNP values for $(p+1)$ participants included in the query (dataset members in set $\mathbf{P}$ and target j). The probabilistic dependence equation for the sum of the SNP values T_{pj}^i is shown in Chapter 3. Similar to the MAF case, y_2 is a kinship coefficient that satisfies the Mendel's law. Thus, y_2 is in $[-2, 2]$.

4.4 Inference Evaluation Algorithm

The adversary generates its queries that include the members of the same family (e.g., by forming a query based on age - location - street level - city level - state level, etc.) and receives the differentially-private MAF ($\tilde{M}_{pj}^i$) or the differentially-private chi-square ($\widetilde{\chi_i^2}$) values. As we discussed in Section 4.2, the adversary has full knowledge about the membership of the dataset participants using auxiliary channels. The adversary is also aware about (i) the dependency between target j and other family members (in set $\mathbf{F}$) that are also in the dataset and (ii) Mendel's law to formulate this dependency. Hence, using the (noisy) query result about a SNP i along with the knowledge of familial relationships, the adversary can infer the value of x_j^i, which represents the value of SNP i for target j. More specifically, from the MAF query results, the adversary can estimate the total number of minor alleles Q_{pj}^i for $(p+1)$ individuals in the query results. Then, it uses the coin change algorithm [72] to obtain all the possible partitions of Q_{pj}^i (each partition will include $\leq (p+1)$ individuals). Next, for each valid partition (validity of the partition is determined using Mendel's law when family members of the target are in the query results), the adversary computes its the probability using Mendel's law by considering potential values of SNP i (0, 1, and 2) for target j. For the χ_i^2 query results, the adversary utilizes the valid partitions for

different Q_{pj}^i values to guess the actual value of χ_i^2 (before the added noise by the DP-based mechanism) for different number of cases and controls. Then, the adversary compares $\widetilde{\chi_i^2}$ with χ_i^2 for the same number of individuals included in the query result.

We use two metrics as the count query metrics in Chapter 3 to quantify the success of the identified attacks: correctness and leaked information. We define the correctness of the adversary in terms of its estimation error (Equation 3.2), which is the distance between the true value of the target's actual SNP x_j^i and the value inferred by the adversary x'^i_j. Thus, correctness of the adversary can be expressed as 1 minus its estimation error. To quantify the difference between the adversary's prior and posterior information after the attack for m SNPs of the target, we use the leaked information metric (Equation 3.3).

4.5 Evaluation

To evaluate the identified vulnerability, we use (and customize) real genomic datasets from two sources: (i) Manuel Corpas Family Pedigree and (ii) 1000 Genomes Phase 3 data. The family structures are discussed in Section 3.3. We extracted chromosome 22 genotypes for the European population from 1000 Genomes Phase 3 data using the Beagle genetic analysis package [71]. We identified the common SNPs between the extracted genomic data of the 1000 Genomes participants and the MC family members to build datasets that include members from MC family and other unrelated participants. Eventually, we created two different datasets using 1000 Genomes phase 3 data: (i) a statistical dataset T_1 (to evaluate the attribute inference attack using MAF queries) with n individuals ($n = 164$), (ii) a case-control (defined in Section 2.2.1) dataset T_2 (to evaluate attribute inference attack using χ^2 queries). Dataset T_2 includes ($s = \frac{n}{2} = 82$) cases and ($c = \frac{n}{2} = 82$) controls. **SNP** is the set of SNP IDs for target j. For both queries, the adversary aims to infer m SNPs $\in$ set **SNP** on chromosome 22 for target j using $m = 100$ queries over datasets T_1 or T_2. Out of the m SNPs we have m =11 rare variants which carry allele with low frequencies in populations

($MAF < 0.05$).

To study different cases for individuals that are included in the differentially-private query results, we consider the following 2 scenarios: (i) query results are computed over target j and multiple first and second-degree family members in $\mathbf{F}$; and (ii) results are computed over target j, multiple family members in $\mathbf{F}$, and multiple other unrelated members (non-relatives) in $\mathbf{U}$. Furthermore, to show the vulnerability due to considering dependence between tuples in the query results, we evaluate the attribute inference attack considering two types of adversaries. The first adversary exploits the familial relationships (i.e., dependency) between the dataset members to reconstruct target j's genomic record. The second adversary considers that there is no dependency between the dataset members. Using the correctness and leaked information metrics described in Section 4.4, we evaluate the success of these attacks. Any extra leaked information the first adversary can infer by considering the familial relationships (dependence between the tuples) is considered as leakage that violates the standard DP guarantees.

4.6 Experimental Results

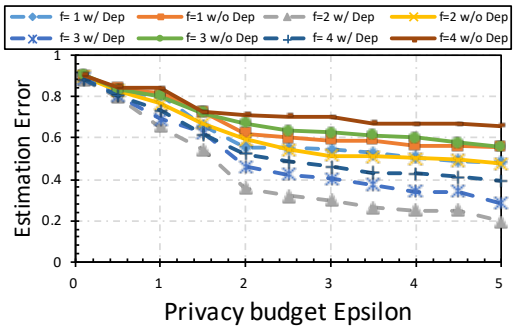
Here, we show the results of the attribute inference attack for the MAF and chi-square queries.

4.6.1 MAF Queries

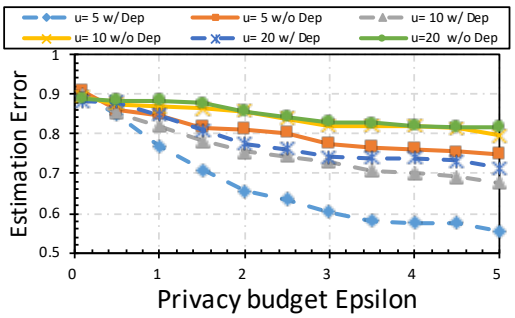
In Figure 4.2, we show the estimation error of the adversary in inferring target j's m SNPs ($m = 100$). Figure 4.2(a) shows the effect of different sets of family members to the estimation error of the adversary. We start including 1 first-degree relative (which can be the father or the mother of the target) to the query results. Second, we include both the father and the mother in the query results. Third, we include the sister of target j together with his father and mother. Last, we consider a second-degree relative (aunt of target j) in the query results along

with the rest of the family members (we follow the same strategy for the other MAF experiments as well). Furthermore, Figure 4.2(b) shows the effect of the number of non-relatives included in the result of a query on adversary's success in terms of its correctness in inferring SNPs of target j.

Using the results of MAF queries over the statistical dataset T_1, we make the following key observations: (i) In Figure 4.2, the most accurate inference of the adversary is when the query computation includes 2 first-degree family members along with target j. This is unlike the results in [1], in which the adversary obtains more accurate results as the number of family members included in the query computation increases. This is because here, we consider both the first-degree and second-degree relatives in a different familial dataset; and the query types we consider are more complex than the one in [1]. Including a second or third degree family member can enlarge the range of possible SNP values for the target, and hence make it more difficult to accurately infer the correct SNP value with a high probability. (ii) The estimation error of the adversary to infer the actual values of the targeted SNPs decreases (i.e., its correctness increases) considerably (by 70% with considering data dependency and by 40% without considering data dependency) as the budget privacy, ϵ, increases from 0.1 to 5, as shown in Figure 4.2(a). In a similar trend, in Figure 4.2(b), the probability of inferring the correct values for the targeted SNPs slightly increases as the value of the privacy budget, ϵ, increases from 0.1 to 5. (iii) The estimation error of the adversary with the knowledge of the data dependency is about 30% less compared to the case in which the adversary do not consider the data dependency in the query results (Figure 4.2(a)). And, (iv) in accordance with the results of [1], including more non-relatives in the query results (e.g., increasing the size of $\mathbf{U}$ from 5 to 20) results in a significant increase in the estimation error of the adversary (even if the adversary has the knowledge of the data dependency), as shown Figure 4.2(b). Moreover, increasing the number of non-relatives in the query results beyond 20 leads to a natural countermeasure (with a probability of 0.9) against the leakage of SNP information of the target.



(a)

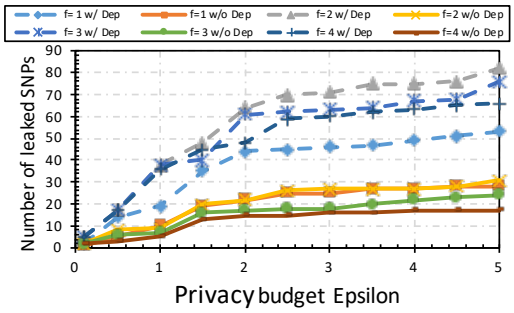


(b)

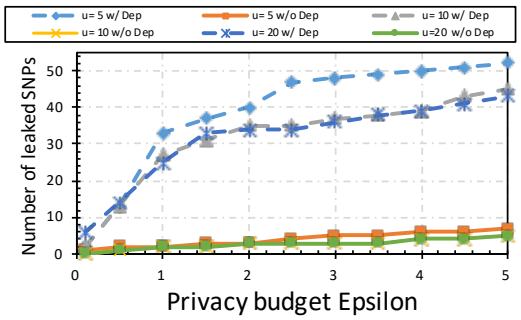
Figure 4.2: The effect of different values of the privacy budget, ϵ, and the number of (a) family members in set $\mathbf{F}$ ($|\mathbf{F}| = f$) and (b) 2 first-degree relatives (father and mother) in set $\mathbf{F}$ along with different numbers of non-relatives in set $\mathbf{U}$ ($|\mathbf{U}| = u$) on adversary's correctness (1 - estimation error) in inferring the targeted SNPs from the noisy results of MAF statistics. (w/ Dep) represents the scenario in which the adversary considers the data dependency and (w/o Dep) represents the opposite.

Next, in Figure 4.3, we evaluate the effect of different values of the privacy budget, ϵ, on the number of the target's leaked SNPs with different numbers of relatives (in set $\mathbf{F}$) and non-relatives (in set $\mathbf{U}$) included in the query results. The results we obtain are consistent with the results of the correctness (in Figure 4.2). We make four key observations: (i) The adversary, using the knowledge of the data dependency, can infer up to 50% more SNPs of the target compared to the case in which it do not consider the data dependency in the query results. (ii) Increasing

the privacy budget ϵ from 0.1 to 5 results in 2 to 8 times (depending on the number of relatives in $\mathbf{F}$ and whether or not the adversary considers data dependency) more SNPs to be inferred by the adversary, as shown in Figure 4.3(a). The adversary infers the maximum number of SNPs when we include 2 first-degree family members in $\mathbf{F}$ and the adversary considers the dependency between tuples. We observe that the inference power of the adversary decreases when we include a second-degree relative (along with target's first-degree family members) in $\mathbf{F}$. We also observe that if the adversary does not consider the data dependency, varying the number of family members in $\mathbf{F}$ from 1 to 4 has a slight effect on the number of leaked SNPs, as expected (Figure 4.3(a)). (iii) Increasing the privacy budget (ϵ) from 0.1 to 5 increases the number of leaked SNPs by only up to 3 times if the adversary does not consider the data dependency and regardless of the number of the non-relatives, as we show in Figure 4.3(b). If the adversary considers the data dependency, then increasing the privacy budget from 0.1 to 5 increases the number of leaked SNPs significantly (up to 6 times in Figure 4.3(b)). And, (iv) when the query results include only family members in $\mathbf{F}$, the adversary can infer about 2 times more SNPs when it considers the dependency compared to not considering dependency (Figure 4.3(a)). However, this increase in the adversary's inference power is 4 times when the query results include both family members and also non-related individuals (Figure 4.3(b)).



(a)

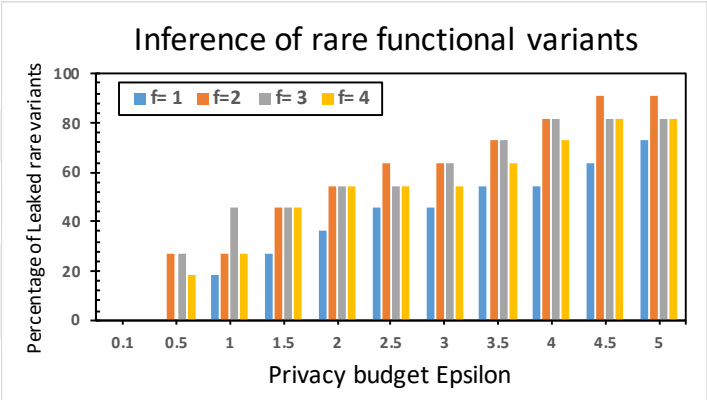


(b)

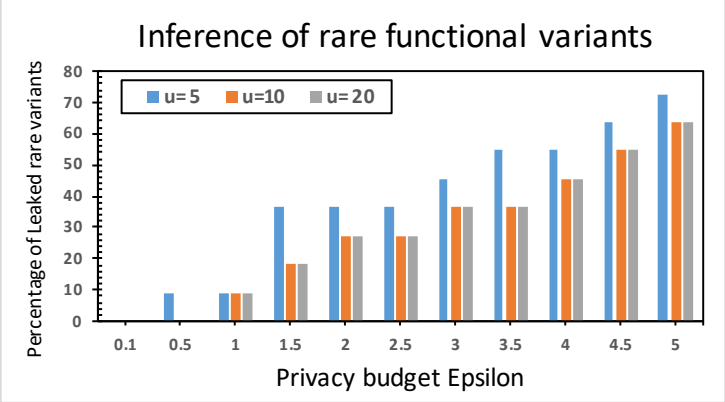
Figure 4.3: The effect of including (a) only family members in $\mathbf{F}$ ($|\mathbf{F}| = f$) and (b) 2 first-degree relatives (father and mother) with different numbers of non-relatives in $\mathbf{U}$ ($|\mathbf{U}| = u$) on the leaked information (i.e., number of leaked SNPs of target j) using the noisy results of MAF statistics. (w/ Dep) represents the scenario in which the adversary considers the data dependency and (w/o Dep) represents the opposite.

Finally, in Figure 4.4, based on the leaked information metric, we evaluate the effect of different values of the privacy budget, ϵ, on the number of the target's leaked rare SNPs. A SNP is considered rare when it carries an allele that have low frequencies in the population. Hence, rare SNPs provide sensitive information about predispositions of individuals for complex diseases [73]. Here, out of 100 targeted SNPs, we identified 11 rare variants, for which MAF < 0.05. Out of these 11 rare variants, the results show that the adversary that considers the dependency between the tuples can infer a significant portion of target's sensitive information. Consistent with the previous results, the adversary can infer more

rare variants when the query results includes the target and 2 first-degree family members in **F**, as we show in Figure 4.4(a). If the query results include many non-related individuals (in **U**) along with the target's relatives, it is more difficult for the adversary to infer the rare variants of the target even it considers the dependency between the tuples (Figure 4.4(b)).



(a)



(b)

Figure 4.4: The effect of including (a) only family members in **F** ($|\mathbf{F}| = f$) and (b) 2 first-degree relatives (father and mother) with different numbers of non-relatives in **U** ($|\mathbf{U}| = u$) on the leaked information by only considering the rare SNPs of the target for MAF queries.

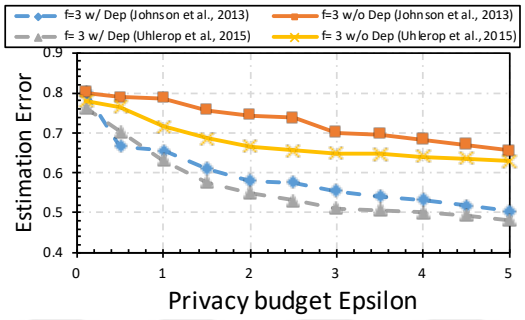
4.6.2 Chi-Square Queries

We use the techniques proposed in [48, 49, 50] (discussed in Section 2.5.5) for differentially-private release of χ^2 statistics over T_2. Similar to before, our goal is to show the how much the adversary's inference power increases when the dependencies in the dataset are utilized in the attack. Since [49] assumes that the adversary knows the complete information of the individuals in the control group, when using the technique in [49], we consider all the family members to be in the case group. Therefore, we use [49] only for the second scenario (described in 4.5), in which the query results are computed over target j, multiple family members, and multiple other unrelated members (non-relatives).

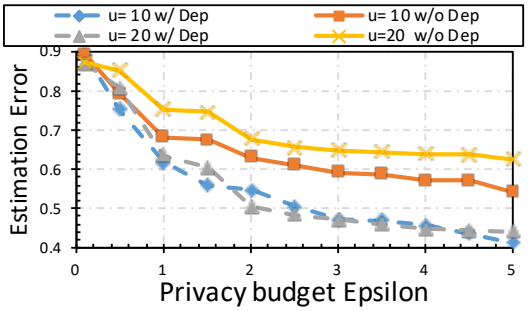
In Figure 4.5, we evaluate the effect of different values of the privacy budget, ϵ, on the adversary's correctness in inferring the targeted SNPs using (noisy) χ^2 statistics about the SNPs. Figure 4.5(a) shows the estimation error when χ^2 query results only include target j and his father, mother, and sister in $\mathbf{F}$ ($|\mathbf{F}| = 3$). Also, Figure 4.5(b) shows the estimation error when χ^2 query results include target j, both of his parents ($|\mathbf{F}| = 2$), and other unrelated individuals in $\mathbf{U}(|\mathbf{U}| = u)$.

From these results, we make three key observations: (i) The estimation error of the adversary with the knowledge of the data dependency is up to 20% less compared to the case in which the adversary do not consider the data dependency in the query results. (ii) The estimation error for the targeted SNPs is slightly less when we apply the algorithm in [48] (while releasing the noisy statistics) compared to the algorithm in [50]. And, (iii) in Figure 4.5(b), using the algorithm in [49] (for the release of noisy statistics) over (T_2), in which the data is represented as 2×3 case-control tables for each SNP, we observe that adding more non-relatives into the query computation does not significantly affect the correctness of the adversary compared to the results in Figure 4.5(a) (obtained using algorithms in [50] and [48]). This is due to the strong adversary assumption in [49] threat model. These results show that complex queries, such as χ^2, are also vulnerable to the dependency between the tuples in the dataset. Furthermore, we observe

that the results we obtain for the χ^2 query are consistent with the results of the MAF query in Section 4.6.1 and the count query in [1].

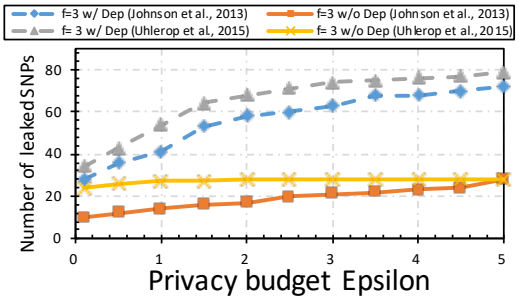


(a)

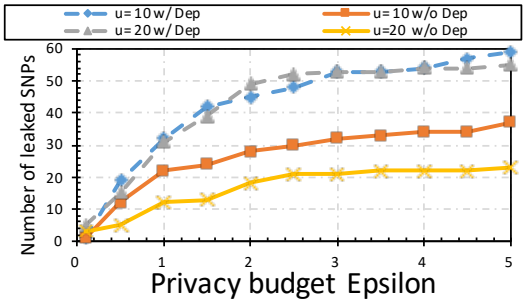


(b)

Figure 4.5: The effect of different values of the privacy budget, ϵ, and the number of (a) family members in $\mathbf{F}$ ($|\mathbf{F}| = f$) and (b) 2 first-degree relatives (father and mother) with different numbers of non-relatives in $\mathbf{U}$ ($|\mathbf{U}| = u$) on adversary's correctness (1 - estimation error) in inferring the targeted SNPs, using the noisy results of χ^2 statistics. (w/ Dep) represents the scenario in which the adversary considers the data dependency and (w/o Dep) represents the opposite.



(a)



(b)

Figure 4.6: The effect of including (a) only family members in $\mathbf{F}$ ($|\mathbf{F}| = f$) and (b) 2 first-degree relatives (father and mother) with different numbers of non-relatives in $\mathbf{U}$ ($|\mathbf{U}| = u$) on the leaked information (i.e., number of leaked SNPs of target j) using the noisy results of χ^2 statistics. (w/ Dep) represents the scenario in which the adversary considers the data dependency and (w/o Dep) represents the opposite.

Next, we evaluate the number of the target's leaked SNPs (leaked information) for different values of the privacy budget, ϵ, in Figure 4.6. We consider different numbers of relatives (in set $\mathbf{F}$) and non-relatives (in set $\mathbf{U}$) to be included in the query results. In-line with the results of the correctness (in Figure 4.5), we make the following key observations: (i) The adversary, using the knowledge of the data dependency, can infer up to 40% more SNPs of the target compared to the case in which it do not consider the data dependency in the query results. (ii) When the adversary considers the dependency in the data, increasing the privacy budget ϵ from 0.1 to 5 results in up to 4 times more SNPs to be inferred

by the adversary, as shown in Figure 4.6(a). (iii) applying different algorithms (to release the noisy statistics) has a slight effect (about 10%) on the number of leaked SNPs. And, (iv) when non-relatives from $\mathbf{U}$ are also included in the query results, the number of leaked SNPs increases by 2.5 times if the adversary considers the dependency in the data between the target j and his family relatives f compared to not considering this familial dependency (Figure 4.6(b)). And, (iv) as expected, Figure 4.6(b) shows that including more non-relatives from $\mathbf{U}$ in the query results results in a decrease in the inference power of the adversary.

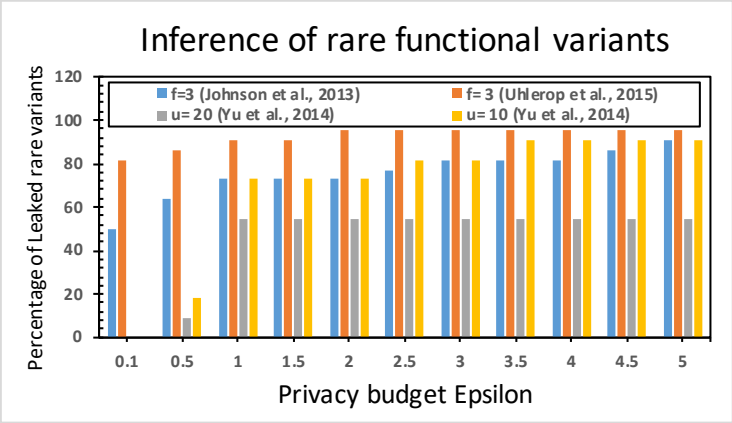


Figure 4.7: The effect of different values of the privacy budget, ϵ on the leaked information by only considering the rare SNPs of the target for χ^2 queries. The query results include: (i) 3 family members (father, mother, and sister) in $\mathbf{F}$ and (ii) 2 family members (father and mother) in $\mathbf{F}$ with different numbers of non-relatives in $\mathbf{U}$ ($|\mathbf{U}| = u$).

Finally, in Figure 4.7, we evaluate the adversary's power to infer the target's rare variants using the leaked information metric. For different values of the privacy budget, ϵ, we compare the results of applying the algorithms proposed by [48, 50, 49] (to release the noisy χ^2 statistics). Similar to the results of MAF query in Section 4.6.1, out of $m = 11$ rare variants, the results show that the adversary can exploit the dependency between the tuples to infer a significant portion of the target's rare SNPs. Consistent with the previous results, we observe that the technique proposed in [50] provides slightly better privacy compared

to the one proposed in [48] when the query results include 3 family members. Moreover, the adversary can still infer the target's rare SNPs, using the technique proposed in [49] when the query results include non-relatives.

4.6.3 Comparison with the Sum Query in [1]

In the following, to show how the vulnerability for attribute inference changes based on the type of the query, we compare the correctness of the adversary for MAF (in Section 4.6.1), χ^2 (in Section 4.6.2), and sum queries (Section 6.5 used in [1]). Figure 4.8 shows the effect of different values of the privacy budget, ϵ on the adversary's estimation error (1 - correctness) in inferring the SNPs for target j, when the adversary consider the familial relationship between tuples in the dataset. First, we consider that query results include 3 family members in set $\mathbf{F}$ (father, mother, and sister). Note that for this case, we used 2 different algorithms to compute differentially-private χ^2 statistics. Next, we consider that the query results include 2 family members in set $\mathbf{F}$ (father and mother) along with 10 non-relatives in set $\mathbf{U}$ ($|\mathbf{U}| = 10$).

We also observe that using the algorithm in [49] decreases the estimation error for the targeted SNPs even if we include non-relatives in the χ^2 query results. This is due to the strong adversary assumption in [49] (that the adversary has full information about the controls). Thus, the algorithm in [49] provides lower privacy compared to [48] and [50] for releasing χ^2 statistics. However, the algorithms in [48] and [50] provide less utility for the shared query results.

Overall, our results show that the existence of dependent tuples in a genomic dataset is the main reason for the identified privacy vulnerability. Hence, an action is required to filter out or distort the dependencies between tuples in statistical genomic datasets.

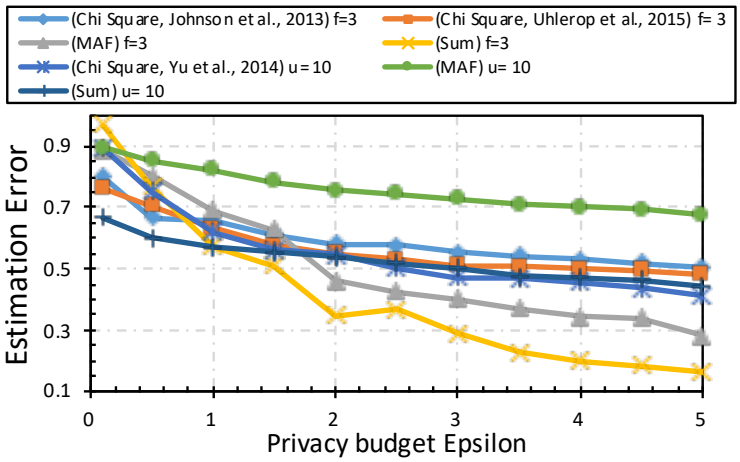


Figure 4.8: The effect of different values of the privacy budget, ϵ, when the query results include (i) 3 family members (father mother and sister) in set $\mathbf{F}$ and (ii) 2 family members (father and mother) in set $\mathbf{F}$ along with 10 non-relatives in set $\mathbf{U}$ ($|\mathbf{U}| = 10$) on adversary's correctness (1-estimation error) in inferring the targeted SNPs. The adversary exploits the noisy results of 3 different queries: sum, MAF, and χ^2 statistics. χ^2 statistics include 3 different cases since we apply 3 existing algorithms to generate the query results.

4.7 Summary

We introduce the second attribute inference attack using the DP-noisy results of two complex queries: MAF and chi square statistics over two statistical genomic datasets. We explain the threat model, describe the genomic datasets, develop the inference algorithm, show the evaluation and the experimental results. The results show that using the results of differentially-private MAF queries on static genomic datasets and utilizing the dependency between tuples, an adversary can reveal up to 50% more sensitive information about the genome of a target (compared to original privacy guarantees of standard DP-based mechanisms), while differentially-privacy chi-square queries can reveal up to 40% more sensitive information.

Chapter 5

Membership Inference Attack

5.1 Overview

In this chapter we focus on the membership inference attacks, where the adversary can use the inferred genomic data obtained from the attribute inference attack to infer the membership of a target in another genomic dataset (e.g., associated with a sensitive trait).

5.2 Threat Model

For the membership inference, we assume that the adversary uses the outcome of the attribute inference attack (in which it infers SNP data X'_j about a target j) and tries to infer the membership of the target to another statistical case-control dataset (e.g., that may be associated with a sensitive phenotype). This is a relatively harder task compared to existing membership inference attacks against statistical genomic datasets, since here, the adversary uses the inferred (and hence partially incorrect) data about the target for its attack. The adversary has access to the MAF values of SNPs of individuals in the control group (M_C) and the MAF

values of the SNPs for the entire dataset population (M_P). We assume the query type to be MAF for membership inference, and hence the adversary sends its queries to a case-control dataset asking about the MAF values for the SNPs of individuals in the case group (M_S). Using its prior information (M_C and M_P) along with the released LPM-based noisy query results about the values in M_S and the inferred genomic record X'_j of target j, the adversary's goal is to infer the membership of the target to the statistical genomic dataset. We quantify the success of this attack using a log-likelihood-ratio (LLR) test.

5.3 Membership Inference Attack

In Chapter 4, we show the attribute inference attack for two genomic datasets with dependent tuples by considering MAF and χ^2 queries. The adversary can infer target j's genomic data X_j exploiting the probabilistic dependence between the target and his family members in $\mathbf{F}$ that are included in the query results. Here, we show how the adversary can use the outcome of the attribute inference attack to infer the membership of the target in another dataset. For the membership inference, we assume that the goal of the adversary is to determine whether target j is in the case group (including individuals with a sensitive trait) of a statistical case-control dataset (T_3).

We assume T_3 includes 2×2 case-control tables for m SNPs and the adversary queries for the MAF values of the SNPs for the individuals in the case group in T_3. Similar to existing membership inference attacks on statistical genomic datasets, we assume that the adversary has access to (i) the set of MAF values of SNPs for individuals in the control group (M_C) and (ii) the set of MAF values of SNPs for a similar population as the entire dataset T_3, including both the case and control individuals (M_P). The adversary receives the LPM-based noisy query result about the MAF value of a SNP i for individuals in the case group ($\mathbf{S}$, where $|\mathbf{S}| = s$) as $\widetilde{M^i_S} = M^i_S + \delta$, where δ represents the added Laplace noise suggested by [48].

5.4 Membership Inference Attack Evaluation Algorithm

To determine the membership of target j in the case group, we evaluate the success of the attack using a log-likelihood-ratio test (LLR). According to [27], at a given false-positive level (α) in binary hypothesis test, the maximum power β can be achieved using the exact LLR test. [26] introduced an LLR test that predicts whether an individual is in the case group by using the query responses received from a statistical genomic dataset, and here, we follow the same model proposed in [26]. using the target's SNP values, allele frequencies of the SNPs in the population, and allele frequencies of the SNPs in the case group. Following the same model proposed in [26], we assume that the SNPs are independent. We assume that under the null hypothesis, target j is not a part of the case group and under the alternative hypothesis, target j is part of the case group $\mathbf{S}$. Accordingly, the overall log-likelihood-ratio (LLR) test can be computed as:

$$LLR = \sum_{i=1}^{m} x_j^i log\frac{M_S}{M_C} + (1 - x_j^i) log\frac{1 - M_S}{1 - M_C} \tag{5.1}$$

For a moderately large $\mathbf{S}$ ($|\mathbf{S}| = s$) and a moderately large number of SNPs with MAF > 0.05 (e.g., $m > 100$), [26] modelled the LLR as a Gaussian distribution to estimate the maximum achievable power β given the false-positive rate α as:

$$Z_\alpha + Z_{1-\beta} \simeq \sqrt{\frac{m}{s}}, \tag{5.2}$$

where Z_α is the 100(1-α) percentile of the standard normal distribution. To compare the practical LLR with the theoretical LLR, [26] suggested to replace $\frac{1}{s}$ in Equation 5.2 by $\frac{1}{s}(1 - \frac{s}{(s+c)})$. Moreover, [26] claimed that one can estimate the controls' MAF values M_C from the population MAFs M_P as follows:

$$M_P = \frac{s}{s+c} M_S + \frac{c}{s+c} M_C. \tag{5.3}$$

5.5 Evaluation

We use dataset (T_3) from two sources: (i) Manuel Corpas Family Pedigree and (ii) 1000 Genomes Phase 3 data, in which the data is represented as 2×2 case-control tables for each SNP, as discussed in Section 3.3. To evaluate the membership inference attack, we choose n individuals ($n = 80$) from European population including target j. we construct the case group (individuals carrying a sensitive phenotype) of size $s = \frac{n}{2} = 40$ and the control group of size $c = \frac{n}{2} = 40$. Out of the m SNPs in set **SNP**, we choose $m = 250$ independent SNPs after discarding the rare variants with MAF < 0.05. We discard the rare variants because in the association studies, usually the exposed SNPs have a pre-specified minimum MAF > 0.05. We empirically build the null hypothesis using the MAF values of the SNPs in the control group (M_C). We reject the null hypothesis and conclude that target j is more likely to be a part of the case group if LLR value is greater than a threshold value th. We find the threshold th from the null hypothesis with $\alpha = 0.1$. Here, α is the false-positive rate, describing when target j is not part of the case group and $LLR > th$. The power $1 - \beta$ is then the true positive rate, describing when target j is part of the case group and $LLR > th$.

Using the (noisy) differentially-private MAF values obtained for the case group (M_S), the adversary can compute the LLR value for target j. To estimate the number of queries needed to achieve a high power (which represents the adversary's confidence that target j is a member of the case group), we consider two types of adversaries: (i) The first adversary uses the actual (correct) SNP set X_j for target j; and (ii) the second adversary uses the inferred SNP set X'_j as a result of the attribute inference attack using the noisy MAF queries (in Section 4.3). We also let each adversary follow two different methods for the attack. In the first method, the adversary sends its queries for m SNPs by randomly choosing the SNPs from set **SNP**. In the second method, the adversary estimates the MAF values of the SNPs in the case group (M'_S) using MAF values of the population (M_P) and MAF values of the control group (M_C) by applying Equation 5.3 suggested by [26]. Then, the adversary identifies some "selected SNPs" to use in its

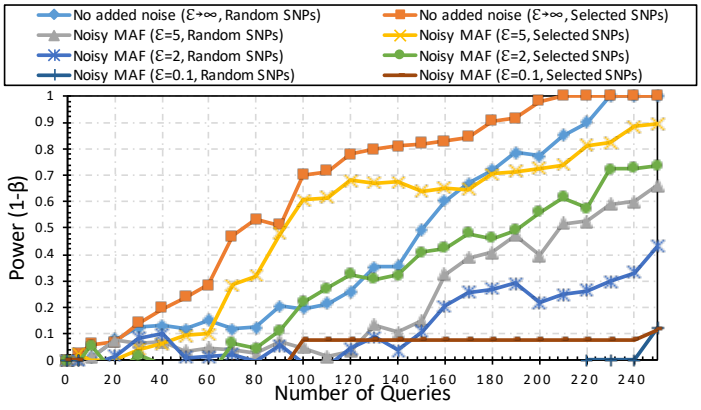
queries. A SNP i is categorized as a selected SNP as follows:

$$SelectedSNPi = \begin{cases} True & \text{if } M'_S \geq M_C \text{ and } x^i_j = 1 \\ True & \text{if } M'_S < M_C \text{ and } x^i_j = 0 \\ False & \text{otherwise} \end{cases}$$

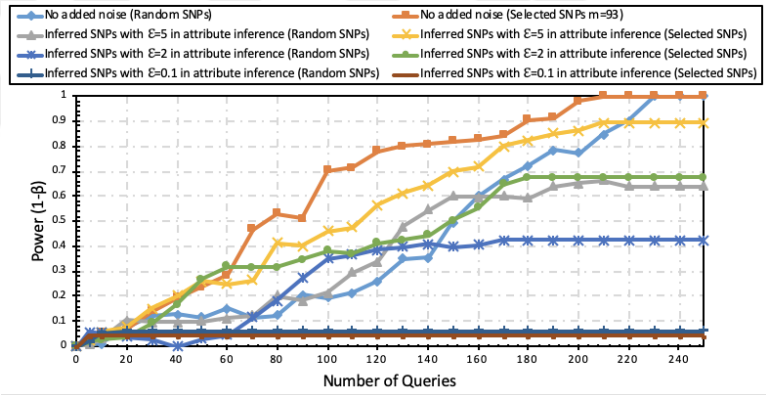
The adversary first sends its queries for the selected m SNPs. Then, (if needed) the adversary sends more queries for other $|\mathbf{SNP}| - m$ SNPs in **SNP**. Using the LLR metric described in Section 5.4, we evaluate the success of the membership inference attack.

5.6 Experimental Results

In Figure 5.1, we show the power curves for the membership inference attack considering the aforementioned two adversaries, each at 10% false positive rate (i.e., $\alpha = 0.1$). We start with the first adversary which uses the actual (correct) SNP values in set X_j for target j. Figure 5.1(a) shows the power for different numbers of queries over T_3 for randomly chosen SNPs and a selected set of SNPs. For each method, the values in M_S are shared (in a differentially-private way) with the adversary using 4 different ϵ (0.1, 2, 5, and when $\epsilon \to \infty$). Using the results of MAF queries over dataset T_3, we make the following key observations: (i) Using the actual (correct) SNP values in set X_j for target j and the correct values in M_S (when $\epsilon \to \infty$), it is possible to correctly determine the membership of the target (with power more than 0.8) by sending either 190 queries for random SNPs or 93 queries for selected SNPs followed by 27 queries for random SNPs. (ii) The adversary can achieve a strong power to infer the membership of the target if it starts to query for the selected SNPs rather than random ones. And, (iii) as expected, decreasing the privacy budget (ϵ) used for releasing the differentially-private values in M_S from 5 to 0.1 results in a 80% loss of power to infer the membership of target j in the case group.



(a)



(b)

Figure 5.1: Power of the adversary for the membership inference attack for different number of MAF queries over dataset T_3. (a) shows the power when the adversary uses the actual (correct) SNPs of target j. (b) shows the power when the adversary uses the inferred SNPs of the target as a result of the attribute inference attack. In (b), ϵ values are the ones used in the attribute inference attack; for membership inference, the values in M_S are shared (in a differentially-private way) with the adversary using $\epsilon = 5$ for all cases.

Figure 5.1(b) shows the power for different numbers of queries over T_3, when the adversary uses the target's inferred SNPs in X'_j. The inferred SNPs are computed using 4 different ϵ values (0.1, 2, 5, and when $\epsilon \to \infty$) over the MAF queries (as shown in Section 4.6.1). In this experiments, we assume that the

values in M_S are shared (in a differentially-private way) with the adversary using $\epsilon = 5$ for all cases. We make four key observations: (i) Here, the adversary has fewer queries to send because it may not infer the whole SNPs in X_j for the target (for some received MAF responses in the attribute inference attack, the adversary cannot make any inference). For instance, when $\epsilon = 5$ in the attribute inference attack, out of 250 SNPs, the adversary can infer 220 of them. In spite of this, it can still achieve a high power in the membership inference attack (1-β = 0.8) after sending 170 queries. (ii) First querying for the selected SNPs leads to a high power using less number of queries. (iii) The adversary can achieve a high power to detect the membership of target j in the case group even if part of the SNP values in X'_j are inferred wrong (as a result of the attribute inference attack). And, (iv) in-line with Figure 5.1(a), for smaller values of the privacy budget (ϵ) in the attribute attack, it becomes more challenging to determine the membership of target j in the case group. When the privacy budget is less than 2, power does not exceed 0.7 after 250 queries, and hence the adversary needs more inferred SNPs to infer the membership of the target.

5.7 Summary

We introduce the membership inference attack using the inferred genomic data of the target j obtained from the aforementioned attribute inference attacks. The adversary use the noisy MAF statistics over statistical genomic datasets to infer the membership of a target j in the dataset. We explain the threat model, describe the genomic datasets, develop the inference algorithm, show the evaluation and the experimental results. Using a log-likelihood-ratio (LLR) test, our results show that the inference power of the adversary can be significantly high in such an attack even by using inferred (and hence partially incorrect) genomes.

Chapter 6

Attribute Inference Attacks against Dynamic Genomic Datasets

6.1 Overview

In this chapter, we address the privacy risks for dynamic statistical genomic datasets, in which new individuals may join or existing individuals leave over time.

6.2 Dynamic Datasets

Most existing privacy-preserved statistical genomic models have been applied on static datasets, which have no updates and the released private statistics depends on one time updated dataset. However, many real-world statistical genomic datasets are dynamic (e.g., genomic data-sharing beacons). The size of these datasets are usually updated frequently, since new tuples (individuals) are

inserted in or deleted from the genomic dataset.

Due to the dynamic nature of the datasets, additional vulnerabilities arise. The datasets are constantly updated with new data. Let us assume that at time t, genomic statistics are released for such a dataset. Now, if the dataset is updated, then the previously computed statistics would change. Hence, maintaining the privacy of genomic dynamic datasets is a challenging task since any adversary can exploit the difference between the released statistics results to infer more sensitive information.

As in the case of static dataset, sharing statistics from dynamic genomic datasets while providing privacy guarantees enables many important medical breakthroughs and applications.

Dynamic notion was considered by many researchers for dataset publication. In real-world applications, datasets are anonymously republished many times and for different recipients in order to provide up-to-date information. Obviously, re-publication of the dynamic datasets leads to possible privacy attacks which was first identified by [74]. This open a new era of privacy research which is privacy preserving dynamic data publication (PPDDP) [75]. The goal of PPDDP research area is developing new privacy mechanisms for publishing dynamic datasets in a private mode without any data loss [76]. There are many PPDDP privacy mechanisms, such as m-invariance [77], t-safety [78], l-diversity [74, 79], k-linkability and k-diversity [80], and k-anonymity [81, 82]. Some PPDDP models are based on differential privacy (DP) notion [83, 84, 59, 85, 86]. However, DP remains useful for publishing the results of statistical queries rather than publishing the datasets [87].

According to [88], a dynamic dataset is defined as follows:

Definition 6.2.1. The dataset T with n tuples is a dynamic dataset iff it has differences in its data at different times t and $t + \varphi$.

where t and φ are any time slots ($0 \leq t < t+\varphi$). The differences in the dataset can occur due to internal update or/and external update. The internal update

considers the changes of the attribute values for the dataset tuples. While the external update considers the change in the dataset size n, which happens due to two specific updates: insertion and deletion. Based on that, the external update can be defined as:

Definition 6.2.2. A dataset is considered as dynamic if for any different times t and $t+\varphi$ $(0 \leq t < t+\varphi)$, if a tuple h in dataset T satisfies at least one of the following conditions:

- insertion: $h \notin T$ at time t and $h \in T$ at time $t+\varphi$.
- deletion: $h \in T$ at time t and $h \notin T$ at time $t+\varphi$.

Hence, if the dataset T has new tuples insertions or/and deletions while no change occurs in the attributes of each tuples between time t and $t+\varphi$, we can define T as external dynamic dataset:

Definition 6.2.3. A dataset T is external dynamic dataset, if for any different times t and $t+\varphi$ $(0 \leq t < t+\varphi)$, T has the following properties:

- $\exists$t,t+φ, T at time $t+\varphi$ is externally updated in contrast to T at time t.
- $\forall t, t+\varphi$, if $h \in T$ at time t and $h \in T$ at time $t+\varphi$, then the attributes of the tuple h at time t = the attributes of the tuple h at time $t+\varphi$.

For the aim of dynamic dataset publishing, [78] categorises the problems into (i) multiple release publishing, (ii) sequential release publishing, and (iii) continuous data publishing.

Publishing the dynamic datasets using PPDDP models is out of our scope. The goal of our work is to study the problem of inferring sensitive genomic data from dynamic datasets while guaranteeing ϵ-DP by using the results of differentially-private genomic queries at different times. Considering an external dynamic dataset T_4, as new individuals are added to or some individuals leaving from the

dataset, new statistics are shared in a differentially-private way at time t and $t + \varphi$.

Guarantees of DP-based solutions may degrade due to (i) the change in the queries responses due to newcomers (change in dataset size can be tracked from the metadata of the dataset) and (ii) the dependency of newcomers with the existing tuples. Thus, we aim to analyze this vulnerability for dynamic genomic dataset which may include dependent tuples.

6.3 Threat Model

The DP-based mechanisms probabilistically guarantee that sensitive data for any dataset participants are protected regardless of adversary's prior knowledge about the dataset. However, the DP privacy guarantees provided by adding noise to query results do not take into consideration neither the notion of dynamic datasets nor the dependency between dataset tuples. DP assumes a static dataset with independent tuples. This can degrade the privacy of the dataset's participants, since they can regularly join or leave the dataset anytime. They also can be dependent with other participants due to familial relationships.

We assume an adversary who knows exactly when the target joins to the dataset. According to that adversary send two queries for each SNP in two different times to exploit the change in query results caused by the external update in the dataset. Any extra leaked information the adversary can infer is considered as leakage that violates the standard DP guarantees. Focusing on the attribute inference attacks based on the Laplace perturbation mechanism (LPM)-based differentially private data release, the goal of the adversary in our model is to infer genomic data of a target individual. The adversary can also use the auxiliary information channels to learn about familial dependencies in the dataset and exploit the vulnerabilities in DP mechanisms. By applying Mendelian inheritance rules, the adversary can estimate the unknown SNP values of the target. Hence, it can extract more genomic sensitive information than that guaranteed by DP.

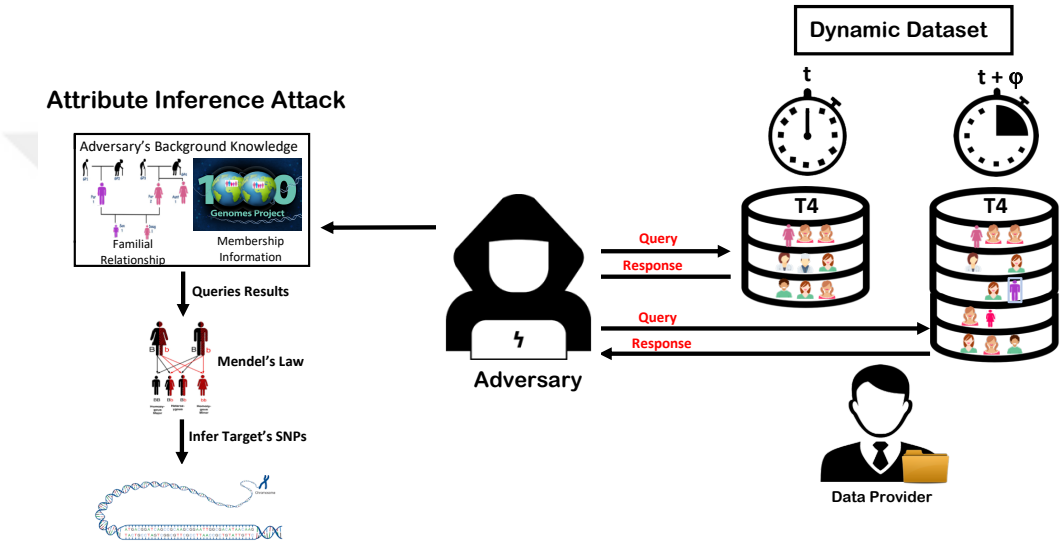


Figure 6.1: The threat model. The adversary does not have any prior knowledge about the genomic data of target j. First, the adversary sends a query to the data provider at different times t and $t + \varphi$. The target j can join any time between t and $t + \varphi$. The data provider sends back the results with added noise using Laplace perturbation mechanism. Second, the adversary identifies the individuals that are used to generate the query results t and $t + \varphi$ using the metadata that is released along with the dataset (e.g., population). That is, the adversary identifies how many of the target's family members and unrelated individuals are used to generate the query results. The adversary compute the differences between the queries results. Next, the adversary uses other auxiliary channels to learn if there is any familial relationship of target j with his family members that are (i) in the dataset and (ii) used to generate the query result. Finally, using the noisy query results along with the auxiliary information, change in the query results and the probabilistic dependence between tuples, the adversary applies Mendel's inheritance laws to infer the genomic record of target j

.

In summary, in our attack scenario (Figure 6.1), we assume that the adversary:

- has knowledge about the memberships of the individuals in the dataset.
- knows the number of dataset participants n at any time.
- uses auxiliary channels to estimate the familial relationships between individuals in the dataset.

Thus, for example, the adversary knows that the target is added in the dataset (with some family members) between times t and $t+\varphi$. In particular, if the target and some family members of the target are among the newly added individuals, we will show in the following sections how DP-guarantees that are provided by considering the entire dataset at time $t + \varphi$ degrade significantly (since the adversary considers the change in dataset responses only over the newly added individuals).

6.4 Evaluation

We use 1000 Genomes Phase 3 dataset for our evaluations. For genomes of family members, we use CEPH/Utah pedigree 1463 [56] and Manuel Corpas pedigree [89]. All details about the family structures are discussed in Section 3.3. We use two datasets to evaluate the identified vulnerability. In both datasets, we use data from 1000 Genomes Phase 3 with 2504 individuals. To add family members, in the first dataset, we use CEPH/Utah pedigree 1463 with the partial DNA sequences of 17 family members [56]. In the second dataset, we use Manuel Corpas family pedigree [89], which contains the DNA sequences in VCF format for 5 family members. We generate an external dynamic datasets which change between times t and $t + \varphi$ ($0 \leq t < t + \varphi$). We define both datasets as T that include n individuals (n= 2514 for the first dataset and n= 2508 for the second one). *SNP* is the set of SNP IDs on chromosome 1 and chromosome 22, and m is the number of SNPs for each individual ($m = 100$ for each dataset). To infer

the values of these m SNPs, 200 MAF queries are performed for each dataset. T_j^i represents the value of a SNP i ($i \in SNP$) for individual j ($j \in T$).

In the proposed inference attacks for external dynamic genomic dataset, we assume the differentially private two MAF query results that are computed at times t and $t + \varphi$ including individuals for different cases as follows:

- Case 1: Other unrelated individuals join at time t, then target j join at time $t + \varphi$ with other unrelated individuals.
- Case 2: Multiple family members of the target and other unrelated individuals join at time t, then target j join at time $t + \varphi$ with other family members and unrelated individuals.
- Case 3: Target j join at time t with other family members and unrelated individuals, then multiple family members of the target and other unrelated individuals join at time $t + \varphi$.

For cases 1 and 2, since the target j join at time $t + \varphi$, we evaluate the performance of the attack considering two different scenarios: (i) the adversary uses the second query result after the target j join at time $t + \varphi$, and (ii) the adversary utilizes the change in the two query results to do genome reconstruction and infer genomic data of j. For case 3, the target join at time $t + \varphi$, we have the following two attack scenarios (i) the adversary uses the first query result at time t and (ii) the adversary uses the second query result where the target, other family members, and unrelated individuals are included at time $t + \varphi$. For cases 2 and case 3, the adversary exploits the genomic association between individuals to infer more genomic information for target j.

To quantify the success of the attacks we use the two metrics: correctness and leaked information. All the details about the inference algorithm and metrics are discussed in Section 3.5.

6.5 Experimental Results

The adversary aims to construct individual j's genomic record, We first start with applying an attribute inference attack over a dynamic dataset assuming all the tuples are independent to show the vulnerability of DP mechanism when the dataset is externally dynamic. Next, for a stronger inference attack, we study the privacy risks by considering a dynamic statistical genomic datasets that include dependent tuples.

6.5.1 Independent Tuples Assumption Experiments

In Figure 6.2, we consider case 1, when the target join ant time between t and $t + \varphi$. We examine the effect of computing the difference between query results where target j and only non-relatives included in the result of the query on adversary's success in terms of his correctness in inferring SNPs of individual j. We show the estimation error of the adversary in inferring target j's m SNPs ($m = 100$). In Figure 6.2(a), two unrelated individuals are included in the first query result at time t, then target j and three unrelated individuals join after time t. Figure 6.2(b), shows the estimation error when two unrelated individuals are included in the first query result at time t, then target j and one unrelated individuals join after time t. In Figure 6.2(c), two unrelated individuals are included in the first query result at time t, then target j only join after time t.

We make three key observations: (i) In Figure 6.2(a)(b), the adversary is able to infer the targeted SNPs ($m = 100$) more accurately when it compute the difference between query 1 and query 2 results. This way, the change in the query results gave the noisy MAF for the target j and less number of unrelated individuals. As the number of unrelated individuals included in the query computation decreases, the estimation error decreases. (ii) In Figure 6.2(b), based on the correctness metric, we observe that if the adversary utilize the change in query results, then the adversary's observation of the targeted SNPs is up to 2 times (depending on the value of ϵ) more accurate compared to using the second query

result. (iii) As expected, if the adversary send its second query directly after target j join the dataset, using the difference between queries make the ability of the adversary to infer the actual value of the targeted SNPs increases by about 40-50% as shown in Figure 6.2(c).

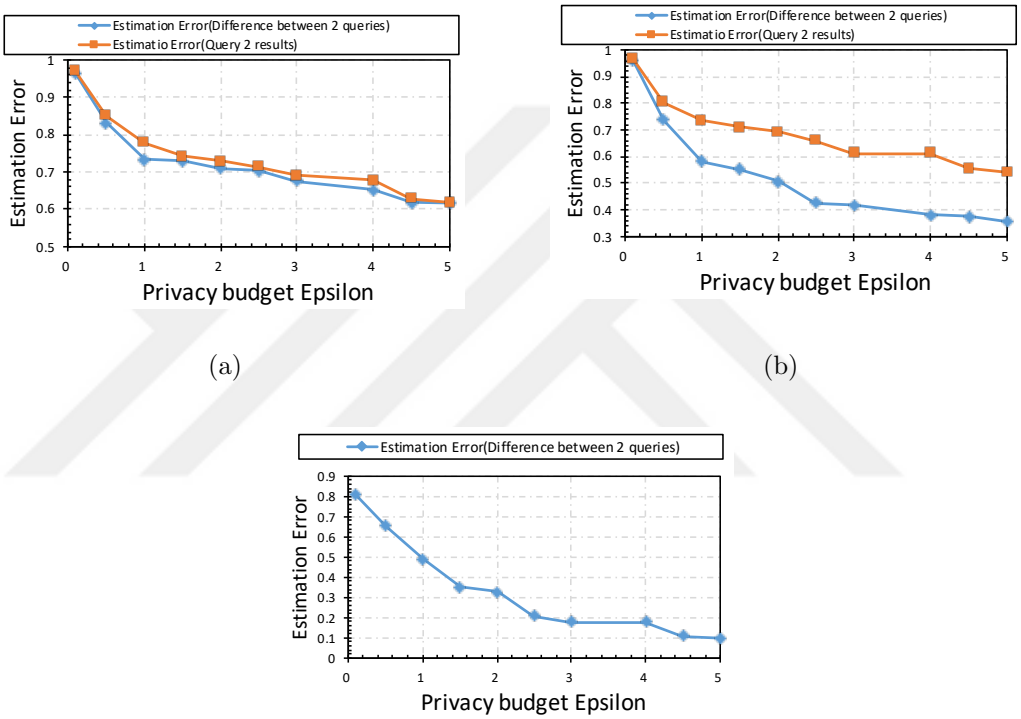


Figure 6.2: The effect of including (a) two unrelated individuals in the first query, target j and five unrelated individuals in the second query (b) two unrelated individuals in the first query, target j and three unrelated individuals in the second query (c) two unrelated individuals in the first query, target j and two unrelated individuals in the second query, on the probability of the adversary's correctness (1 - estimation error) in inferring the targeted SNPs.

Next, we show in Figure 6.3 the number of leaked SNPs the adversary can infer about individual j from the results of the second query or the variation between the two queries' results. We make two key observations: (i) In Figure 6.3(a)(b), using the variance in the queries results in the dataset, results in increasing the

number of leaked SNPs by 5%-10% (depending on the value of the ϵ used) more accurate compared to using the second query results. (ii) In Figure 6.3(c) we also observe that if the adversary choose the perfect time to send his second query, then the variance in the query results even without any dependency between the dataset tuples, increases the number of leaked SNPs by up to 40-50%.

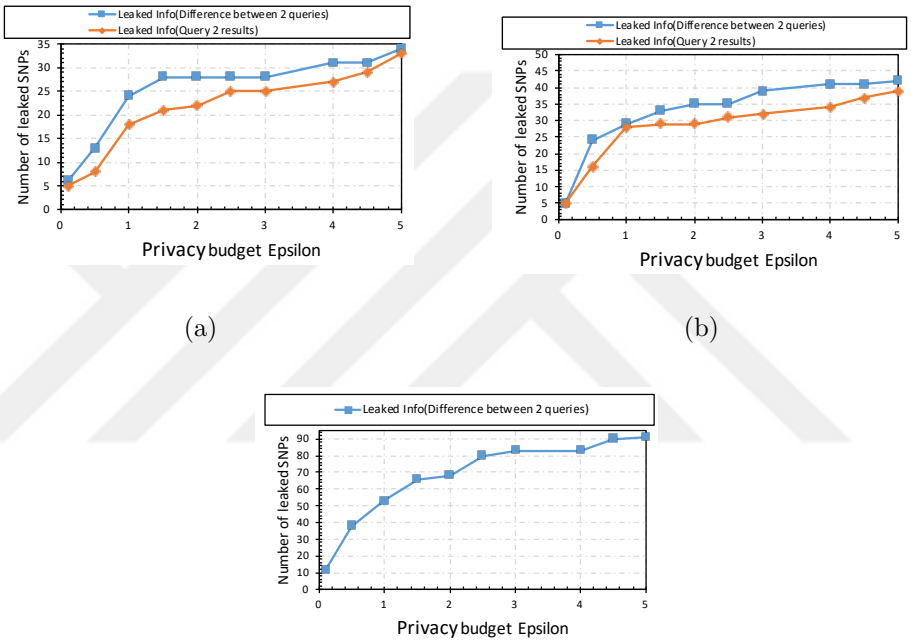


Figure 6.3: The effect of including (a) two unrelated individuals in the first query, target j and five unrelated individuals in the second query (b) two unrelated individuals in the first query, target j and five unrelated individuals in the second query (c) two unrelated individuals in the first query, target j and two unrelated individuals in the second query, on the target's leaked SNPs the adversary can infer.

6.5.2 Dependent Tuples Assumption Experiments

Using the results of MAF differentially-private queries over the statistical dataset T_3, we study the vulnerabilities of DP considering the second case when dependent

tuples can join the dynamic dataset T_3 with other unrelated individuals any time t ($0 \leq t < t + \varphi$). First, we assume that the target j join after time t (e.g. at time $t + \varphi$) with unrelated individuals, or family members or both of them. We study the following cases for our experimental results:

1. Ten unrelated individuals join at time t, target j and two family members join at time $t + \varphi$.
2. Ten unrelated individuals join at time t, target j, two unrelated individuals and two family members join at time $t + \varphi$.
3. Five family members join at time t, target j and two family members join at time $t + \varphi$.
4. Five family members join at time t, target j and two unrelated individuals join at time $t + \varphi$.
5. Two family members and seven unrelated individuals join at time t, target j and five unrelated individuals join at time $t + \varphi$.

We make the following key observations: (i) in Figure 6.4, the most accurate inference of the adversary is when the query computation includes only family members along with target j. The adversary exploits the difference between the noisy MAF query result at time $t + \varphi$ and the noisy MAF query result at time t as shown in Figure 6.4(a) and (b) to exclude the unrelated individuals MAF statistics. Hence, the estimation error of the adversary to infer the actual values of the targeted SNPs decreases (i.e., its correctness increases) considerably (by 80%) as the budget privacy, ϵ, increases from 0.1 to 10, as shown in Figure 6.4(a). In a similar trend, in Figure 6.4(b), the probability of inferring the correct values for the targeted SNPs slightly increases as the value of the privacy budget, ϵ, increases from 0.1 to 10 when the adversaty uses the difference between the two query results to eliminate the 10 unrelated individuals' MAF statistics released in the first query results as shown in Figure 6.4(b). (ii) In Figure 6.4(c), using the second query results, the estimation error of the adversary is about $10\% - 30\%$ less compared to the case in which the adversary use the difference between the

two queries. The reason for that is, after time t more family members join the dataset T_3 with the target j. Hence, the correctness of the adversary increases as the value of the privacy budget, ϵ, increases from 0.1 to 10 (Figure 6.4(c)).

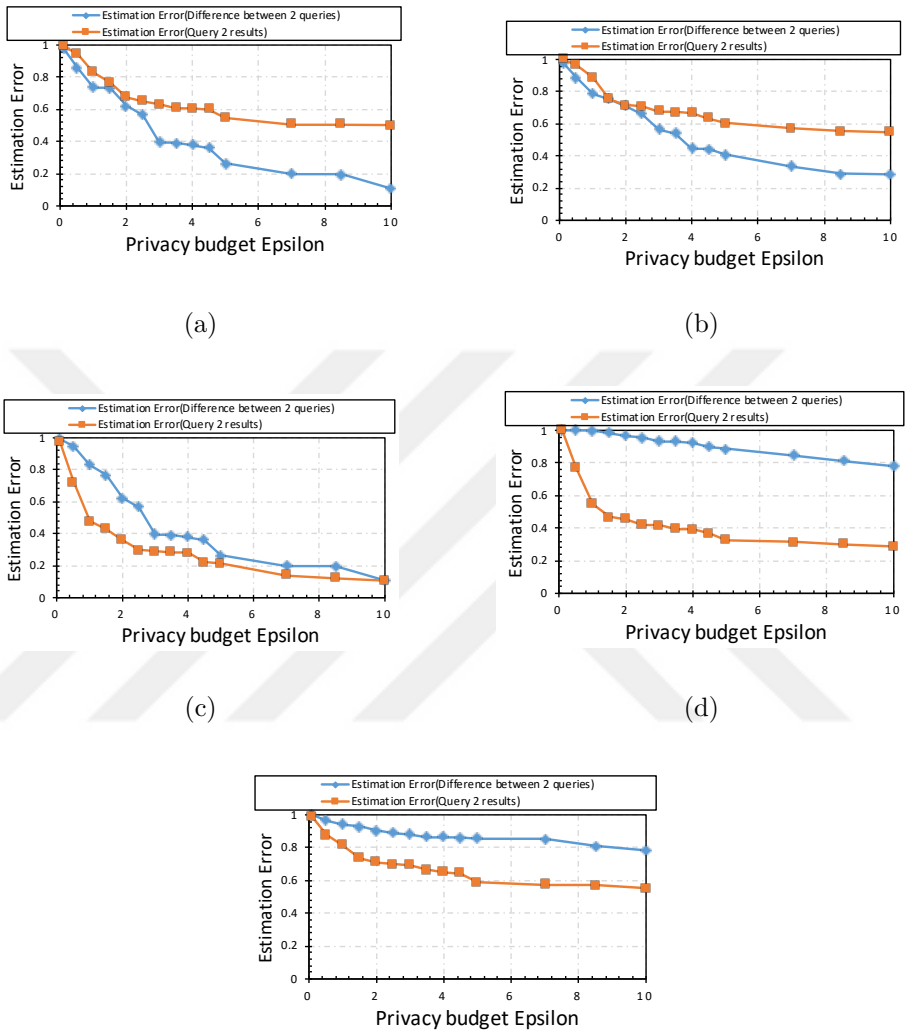


Figure 6.4: The effect of including (a) ten unrelated individuals in the first query, target j, two family members, and ten unrelated individuals in the second query (b) ten unrelated individuals in the first query, target j, twelve unrelated individuals and two family members in the second query,(c) five family members in the first query, target j and seven family members in the second query, (d) five family members in the first query, target j, two unrelated individuals and five family members in the second query, (e) two family members and seven unrelated individuals in the first query, target j, twelve unrelated individuals, and two family members in the second query, on the probability of the adversary's correctness (1 - estimation error) in inferring the targeted SNPs.

Also, in Figure 6.4(d) and (e), the estimation error of the adversary is about $20\% - 60\%$ less when the adversary uses the second query results, since the difference of the two queries excludes the MAF statistics of the family members included in the first query result. And, (iii) Generally, in accordance with the results of [1], including more non-relatives in the query results in a significant increase in the estimation error of the adversary, as shown Figure 6.4.

Next, in Figure 6.5, we evaluate the effect of different values of the privacy budget, ϵ, considering the second query results or the difference between the two queries, on the number of the target's leaked SNPs with different numbers of relatives (in set $\mathbf{F}$) and non-relatives (in set $\mathbf{U}$) included in the two query results. The results we obtain are consistent with the results of the correctness (in Figure 6.4). We make four key observations: (i) The adversary, using the variance of the two MAF queries results, can infer up to 40% more SNPs of the target when the family members join after time t with the target j and the adversary exclude the unrelated individuals included in the first query results (Figure 6.5(a) and (b)). (ii) When the second query results include only family members in $\mathbf{F}$ as in Figure 6.5(c), the adversary can infer about up to 40% more SNPs than when the second query results include unrelated individuals with the target j and his family members as in Figure 6.5(d) and (e). (iii) We also observe that varying the number of family members in $\mathbf{F}$ from five in the first query results to seven in the second query results has an effect on increasing the number of leaked SNPs, as expected (Figure 6.5(c)). (iv) Generally, increasing the privacy budget ϵ from 0.1 to 10 results in $30\% - 90\%$ (depending on the number of relatives in $\mathbf{F}$) more SNPs to be inferred by the adversary, as shown in Figure 6.5.

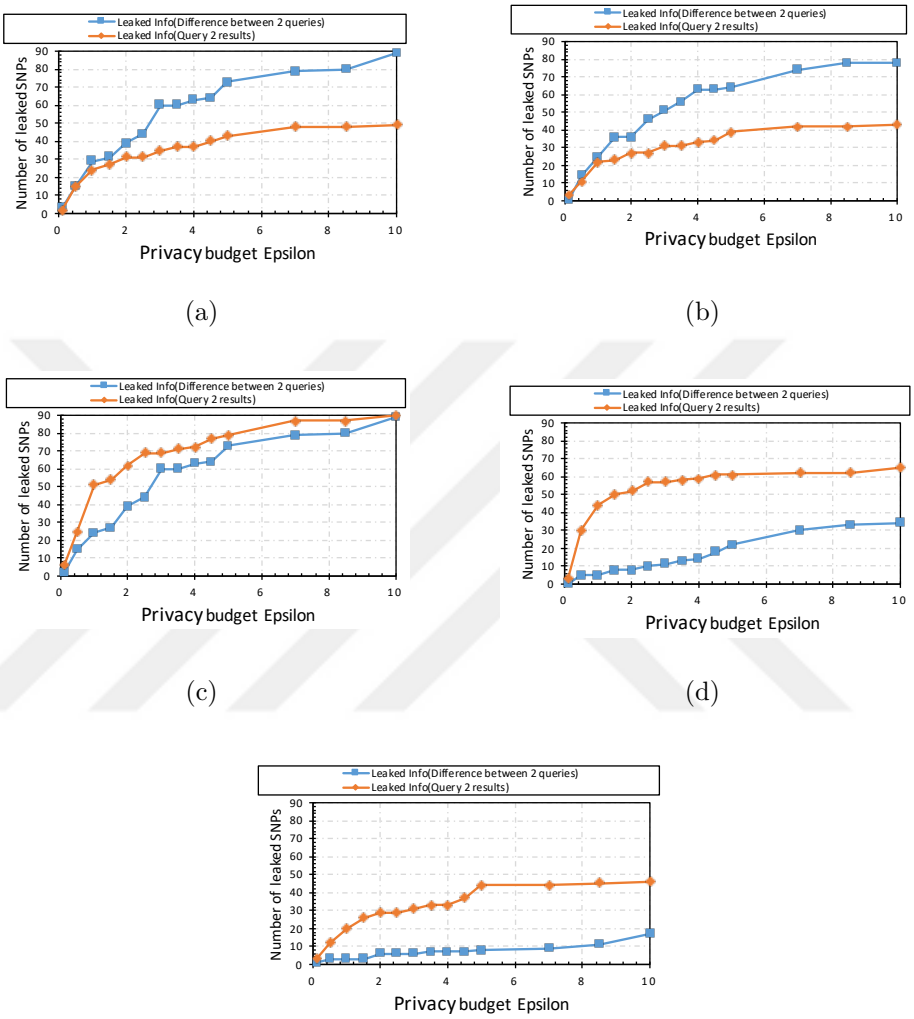


Figure 6.5: The effect of including (a) ten unrelated individuals in the first query, target j, two family members, and ten unrelated individuals in the second query (b) ten unrelated individuals in the first query, target j, twelve unrelated individuals and two family members in the second query,(c) five family members in the first query, target j and seven family members in the second query, (d) five family members in the first query, target j, two unrelated individuals and five family members in the second query, (e) two family members and seven unrelated individuals in the first query, target j, twelve unrelated individuals, and two family members in the second query, on the leaked information (i.e., number of leaked SNPs of target j) using the noisy results of MAF statistics.

Second, considering the third case, we assume that the target j join at time t with unrelated individuals, or family members or both of them. We study the following cases for our experimental results:

1. Target j and ten unrelated individuals join at time t, two family members join at time $t + \varphi$.
2. Target j and ten unrelated individuals join at time t, two unrelated individuals and two family members join at time $t + \varphi$.
3. Target j and five family members join at time t, two family members join at time $t + \varphi$.
4. Target j, two family members and five unrelated individuals join at time t, five unrelated individuals join at time $t + \varphi$

Since the target j join the dataset at time t, using the difference between the two queries will exclude the MAF statistics for the target j. Hence, in the following Figures 6.6 and 6.7 we compare the genomic information the adversary can infer for the target j using the results of the first query with the genomic information the adversary can infer for the target j using the results of the second query. Figures 6.6, we show the estimation error of the adversary in inferring target j's $m = 100$ SNPs. We make the following key observations: (i) in Figure 6.6, when all the family members of target j participating in the dataset join after time t(e.g at time $t + \varphi$), the most accurate inference of the adversary is when he exploits the second noisy MAF query result at time $t + \varphi$ as shown in Figure 6.6(a) and (b). Hence, the estimation error of the adversary to infer the actual values of the targeted SNPs decreases (i.e., its correctness increases) considerably (by 50%) as the budget privacy, ϵ, increases from 0.1 to 10, as shown in Figure 6.6(a). In a similar trend, in Figure 6.6(b), the probability of inferring the correct values for the targeted SNPs considerably increases (by 40%) as the value of the privacy budget, ϵ, increases from 0.1 to 10 when the adversaty uses second query results to exploit the two family members' MAF statistics released in the second query results as shown in Figure 6.6(b).

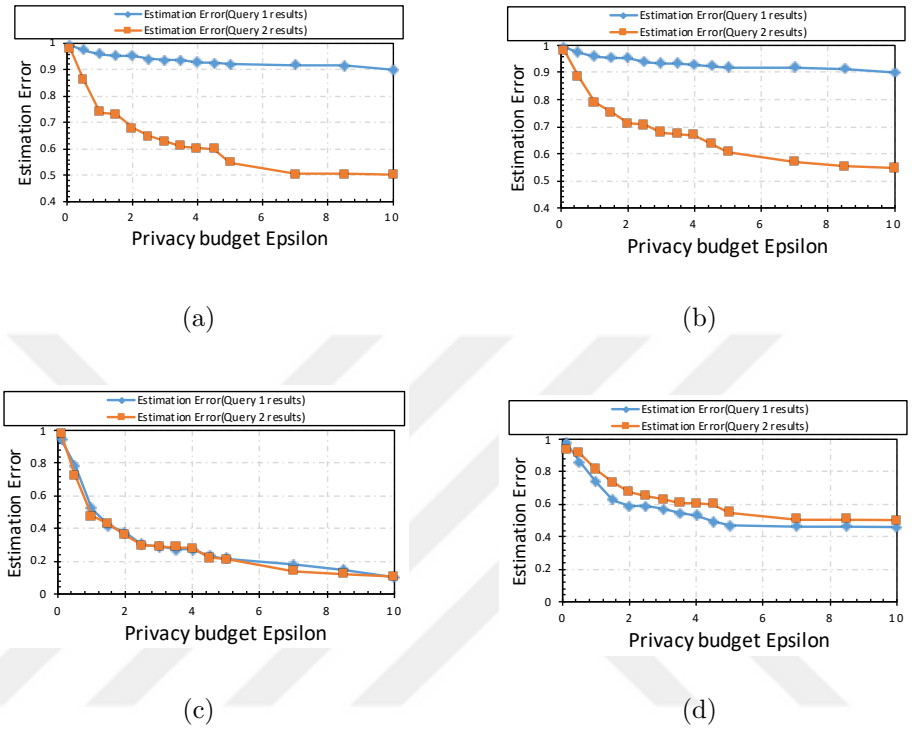


Figure 6.6: The effect of including (a) target j and ten unrelated individuals in the first query, target j, two family members and ten unrelated individuals in the second query (b) target j and ten unrelated individuals in the first query, target j, twelve unrelated individuals and two family members in the second query,(c) target j and five family members in the first query, target j and seven family members in the second query, (d) target j, two family members and five unrelated individuals in the first query, target j, two family members and ten unrelated individuals in the second query, on the probability of the adversary's correctness (1 - estimation error) in inferring the targeted SNPs.

(ii) In Figure 6.6(b), using the second query results, the estimation error of the adversary is about 10% more compared to the results of Figure 6.6(a) when the adversary uses the second query results, since in Figure 6.6(b), two unrelated individuals join the dataset at time $t+\varphi$ with the two family members. (iii) If the first query results includes the target j with some of his family members, the more relatives join in the second query results, the slight decrease in the adversary's estimation error in inferring the target j genomic record, as in Figure 6.6(c). (iii)

Including more non-relatives in the second query results in a slight increase in the estimation error of the adversary, if the first query results include the target with some family members and other unrelated individuals as shown Figure 6.6(d).

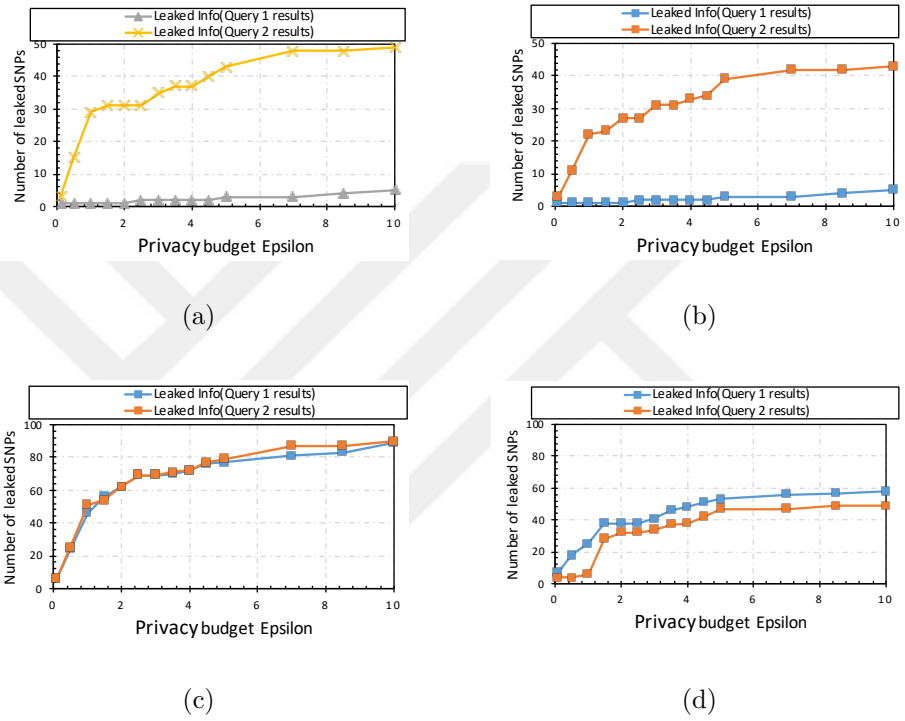


Figure 6.7: The effect of including (a) target j and ten unrelated individuals in the first query, target j, two family members and ten unrelated individuals in the second query (b) target j and ten unrelated individuals in the first query, target j, twelve unrelated individuals and two family members in the second query,(c) target j and five family members in the first query, target j and seven family members in the second query, (d) target j, two family members and five unrelated individuals in the first query, target j, two family members and ten unrelated individuals in the second query, on the leaked information (i.e., number of leaked SNPs of target j) using the noisy results of MAF statistics.

Next, in Figure 6.7, we evaluate the effect of different values of the privacy budget, ϵ, considering the first noisy MAF query results or the second noisy MAF query results, on the number of the target's leaked SNPs with different numbers of relatives (in set $\mathbf{F}$) and non-relatives (in set $\mathbf{U}$) included in the two query results.

The results we obtain are in accordance with the results of the correctness in Figure 6.6. We make the following key observations: (i) The adversary can use the second query results to infer up to 50% more SNPs of the target (when the family members join after time t) more than using the first query results when only the target and other unrelated individuals are included (Figure 6.7(a) and (b)). (ii) Increasing the number of family members in $\mathbf{F}$ from five in the first query results to seven in the second query results has an effect on increasing the number of leaked SNPs, as expected (Figure 6.7(c)). While in contrast, increasing the number of unrelated individuals in the second query results has an effect on decreasing the number of leaked SNPs, as in (Figure 6.7(d)).

6.6 Summary

We introduce the third attribute inference attack using the DP-noisy results of MAF statistics query over an external dynamic statistical genomic datasets, where the dataset's participants can join or leave at any time. We explain the threat model, describe the genomic datasets, develop the inference algorithm, show the evaluation and the experimental results. The results show that using the results of differentially-private two MAF queries at different times t and $t+\varphi$ and utilizing the dependency between tuples, an adversary can reveal up more sensitive information about the genome of a target compared to original privacy guarantees of standard DP-based mechanisms.

Chapter 7

Countermeasures

7.1 Overview

In this chapter, we formalize a DP-mechanism against the aforementioned attribute inference attacks in Chapter 3 and Chapter 4 for privacy-preserving sharing of statistics from genomic datasets to attain privacy guarantees while taking into consideration the dependence between tuples.

7.2 Detailed Discussion of Experimental Results

The experimental results of our inference attacks in Chapter 3 and Chapter 4 prove our claim that an adversary considering the relationship between dependent tuples in a genomic dataset can violate the privacy guaranteed by DP on a query result. Hence, applying DP-based mechanisms on a genomic dataset with dependent tuples will reveal more sensitive information than guaranteed.

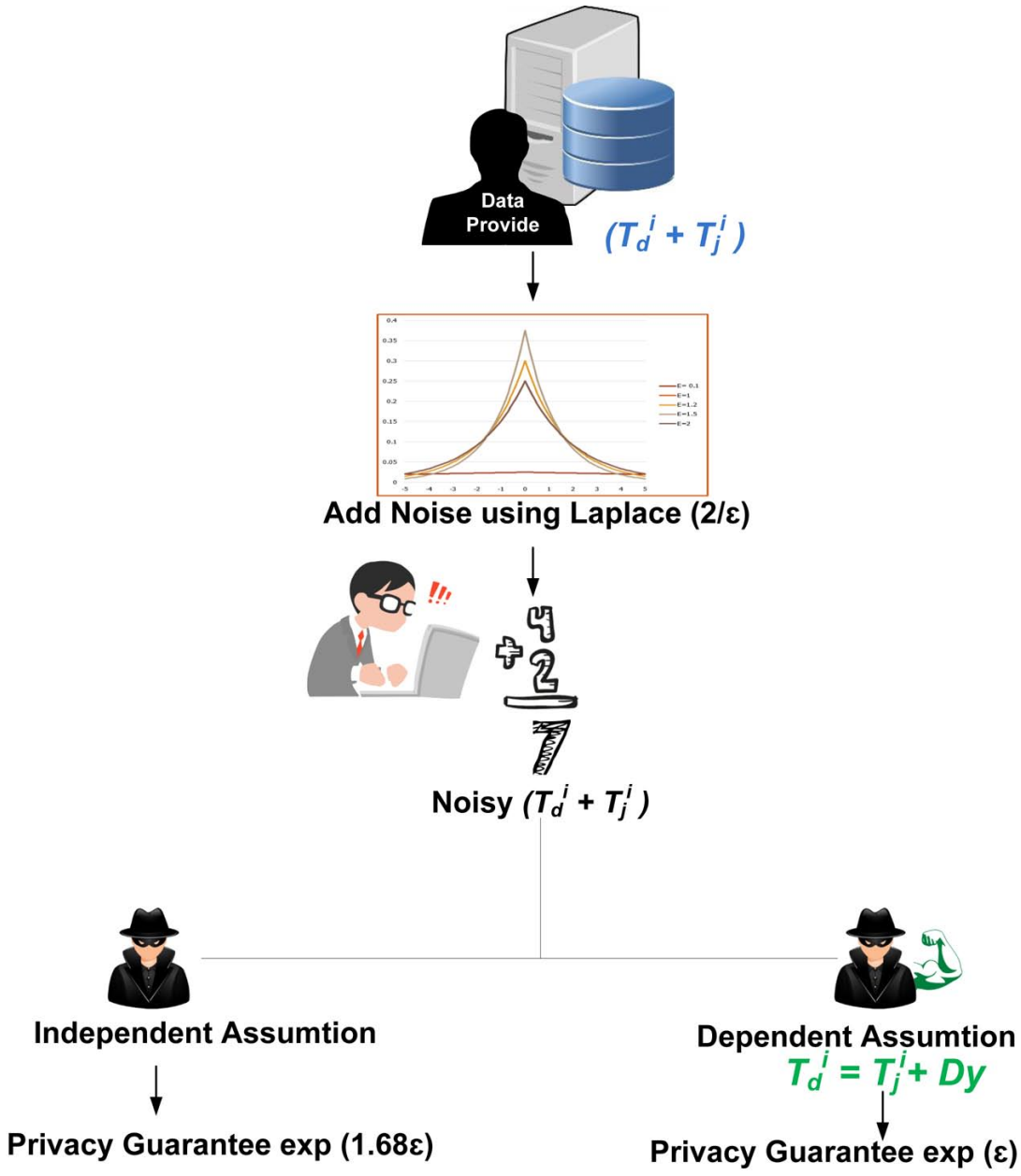


Figure 7.1: Dependence between tuples violate the privacy guarantees of DP. For instance, including up to 9 family members of the target j in the query results, degrades the privacy guarantees from ϵ to 1.68ϵ.

Following the adversary model proposed in [19], we also show that the dependencies between tuples may weaken the privacy in the following example. We represent the genomic dataset as T, where T_p^i denotes the sum of SNP values

for SNP i from p-users, in which they all can be correlated (p=d) or only part of the p participants are correlated (p¿d). We have probabilistic dependence as $T_p^i = T_j^i + Dy$. By considering a simple inference attack using the result of a sum query, where $Q(T) = (T_p^i + T_j^i)$, the adversary can infer the value of T_j^i.

If T_p^i and T_j^i are assumed to be independent, as stated in [48, 49], adding Laplace noise with $2/\epsilon$ parameter achieves ϵ-DP. However, if the adversary knows the familial relationship between the tuples T_p^i and T_j^i, using the average of different 8 cases which include up to 9 family members of the target j in the query result, using 13 different ϵ values in LPM, we show that this noise can achieve 1.68ϵ-DP as shown in Figure 7.1.

7.3 Countermeasures

As shown in Chapter 3 and Chapter 4, a genomic dataset with dependent tuples requires a stronger privacy notion than the existing DP mechanisms to get the same level of privacy guarantees. According to the evaluation results, using the adversary model described in Section 3.2, in this section, we formalize the notion of ϵ-differential privacy for genomic datasets with dependent tuples to avoid inferring more sensitive information by an adversary with prior knowledge about the dependency between tuples. For any dataset T we denote the number of dependent tuples in T as d.[1] We run the attack on a victim among these d dependent tuples. We define the dependence relationship between two tuples j and h as $R_{j,h}$, where R represents the familial relationships in real-world genomic datasets. In Chapter 3, we show an instance of R, where the dependence R can be known through the online information about the participants of the genomic studies and it can be formulated using the probabilistic dependence $T_p^i = T_j^i + Dy$. Like DP, ϵ-DP for genomic datasets with dependent tuples uses the notion of neighboring datasets, which can be defined as follows:

Definition 7.3.1. The datasets T and T' with d dependent tuples, which is the

[1]There may be different sets of dependent tuples in the dataset, we just focus on the largest set of dependent tuples with size d.

largest number of dependent tuples having probabilistic genomic relationship R, are neighboring dependent datasets if the change of one tuple value in T causes change of at most *d-1* tuple values in T'.

Accordingly, we define the ϵ-DP for genomic datasets with dependent tuples as follows:

Definition 7.3.2. A randomized algorithm A satisfies ϵ-differential privacy if for any pair of neighboring datasets T and T' with d dependent tuples, and for any $O \subseteq Range\,(A)$,

$$Pr\,[A\,(T) \in O] \leq e^{\epsilon} Pr\,[A\,(T') \in O]$$

.

Note that when R represents no dependency between tuples (R=0), our privacy model is equivalent to DP mechanism. In order to restrict an adversary from inferring more sensitive information of an individual, we compute the value of the privacy budget (ϵ) for datasets that include dependent tuples so that the privacy guarantee will be the same as the datasets with independent tuples.

Analyzing Laplace Perturbation Mechanism (LPM): Recall the results we got for the threat model discussed in Section 3.2. We have a tuple T_j^i that has a probabilistic dependence relationship R with T_p^i as $T_p^i = T_j^i + Dy$, considering the result of a sum query where $Q(T) = (T_p^i + T_j^i)$. To achieve DP, we add Laplace noise with parameter $2/\epsilon$ for the sum query. We analyze the LPM-based differential privacy mechanism while considering two assumptions:

- Independent tuples
- Dependent tuples

From the results we have the following observations:

1. For any dataset with independent tuples, the noisy query output guarantees achieving DP with the same budget of ϵ value.

2. We need a smaller ϵ value to achieve DP for any dataset with dependent tuples. In other words, reducing the ϵ value used to achieve DP causes the Laplace noise to be augmented.

3. There may be different sets of dependent tuples in the dataset, according to the size of the largest dependent tuples, the added noise will be determined.

From our observations, we analyze the results of different queries for different number of dependent tuples. We compare the leaked information in a dataset assuming dependent and independent tuples in order to compute the DP sensitivity for different dependency size in a genomic dataset. The sensitivity can be defined as follows:

Definition 7.3.3. The dependent sensitivity for publishing the results of any query Q over a genomic dataset with correlated tuples is

$$\varsigma = \sigma \Delta Q, \tag{7.1}$$

where σ is the variable used to obtain the new value of ϵ. we describe computation of σ later. Also, ΔQ is the query Q's global sensitivity, which is the maximum difference in the query's result on any two neighboring datasets. Therefore, to achieve privacy guarantees in a genomic dataset, we formalize the mechanism to get ϵ-differential privacy for genomic datasets with dependent tuples as follows:

Theorem 7.3.1. Let A be a randomized algorithm. Then, for a dataset T with d genomic dependent tuples, $A(T)$ provides ϵ-differential privacy for a query Q with global sensitivity ς, if $A(T) = Q(T) + LAP(\varsigma/\epsilon)$, where ς is computed as in Equation 7.1.

Proof. Given a query function Q over a dataset T. ς evaluates the sensitivity of the query function Q for a dataset T with dependent tuples d. Hence, we can prove that applying Laplace mechanism with scale ς/ϵ for two neighbouring dependent datasets provides ϵ-differential privacy as:

$\frac{Pr(Q(T)+Lap(\varsigma/\epsilon)=O)}{Pr(Q(T')+Lap(\varsigma/\epsilon)=O)} \leqslant e^{\epsilon}$

$= \frac{exp-(\frac{|O-Q(T)|\epsilon}{\sigma\Delta Q})}{exp-(\frac{|O-Q(T')|\epsilon}{\sigma\Delta Q})}$

$= exp(\frac{\epsilon}{\sigma\Delta Q}(|O - Q(T')| - |O - Q(T)|))$

$\leqslant exp(\frac{\epsilon}{\sigma\Delta Q}(|Q(T) - Q(T')|)$

$\leqslant e^{(\frac{\epsilon}{\sigma})} \leqslant e^{\epsilon}$, where the last follows from the definition of sensitivity ς, and the fact that $\sigma \geq 1$. □

Consider the leaked information an adversary can get without dependency assumption to be L_0, the leaked information any adversary can get with dependency assumption to be L_1, and the set of different ϵ values used in the query results over the dataset T to be v. The following equation gives the value of σ:

$$\sigma = v/(\sum_{\epsilon \in v}^{|v|} L_0/L_1)$$

(7.2)

From our results, we calculate σ which allows accurate computation of the sensitivity for dependent data using 12 different ϵ values ($|v| = 12$).

7.4 Methodology for Countermeasures

The data provider get the query and identify the largest set of dependent tuples in the query results. There may be more than one dependent tuples set (i.e.,

different sets of families) that are included in the calculation of the query results. We provide two practical strategies for computing the sensitivity. Based on the size of the dependent tuples the data provider can compute the value of σ. The data provider can select a proper model according to the query of the querier.

1. Using the query results over only the dependent tuples in the dataset: The data provider receives a query and observes the size of the largest dependent tuples set on it. He assumes that the querier has a complete knowledge about the correlation between tuples and the query results will only contain information from these dependent tuples.

   For example, in our evaluation scenario in which the maximum number of first-degree relatives that can be included in the same dataset together is 9. Hence, the data provider can compute the value of σ directly from the size of correlated tuples d as:

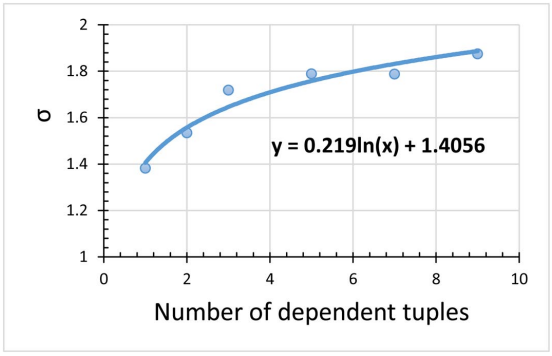
   $$\sigma = 0.219 \ln(d) + 1.4056 \tag{7.3}$$

   We show how Equation 7.3 is derived in Figure 7.2(a)

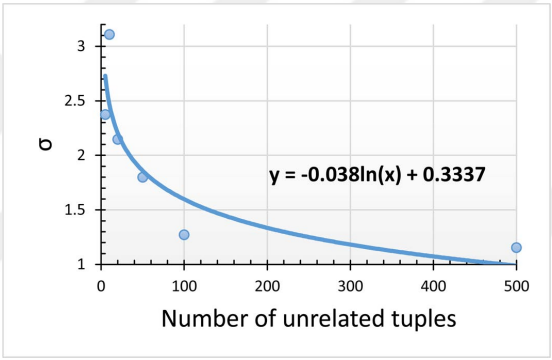
2. Using the query results over the dependent tuples and unrelated tuples in the dataset: The data provider assumes that the querier has a complete knowledge about the correlation between tuples, but the query results will contain information from these dependent tuples and other unrelated tuples. Here, we compute the σ value for 6 different values of unrelated members included in the results of query over the dataset T. The number of unrelated other members starts from 5 and gradually increases to 500. The data provider can compute the value of σ directly from the size of unrelated tuples u as:

   $$\sigma = -0.038 \ln(u) + 0.3337 \tag{7.4}$$

   Figure 7.2(b) shows how Equation 7.4 is derived for this scenario.



(a)



(b)

Figure 7.2: (a) σ values for computing the sensitivity when the query results contain the dependent tuples only (b) σ values for computing the sensitivity when the query results contain the dependent tuples and other unrelated tuples.

7.5 Evaluation of Countermeasures

In this section, we evaluate the performance of the proposed countermeasures to release the query results of dependent data over two real genomic datasets. We apply our algorithms over two datasets containing genomic data from (i) 1000 Genome Phase 3 data and CEPH/Utah Pedigree 1463, and (ii) 1000 Genomes Phase 3 and Manuel Corpas Family Pedigree. We use 100 SNPs from chromosome 1 and chromosome 22 to analyze the resistance of our privacy mechanism to the threat model presented in Section 3.2.

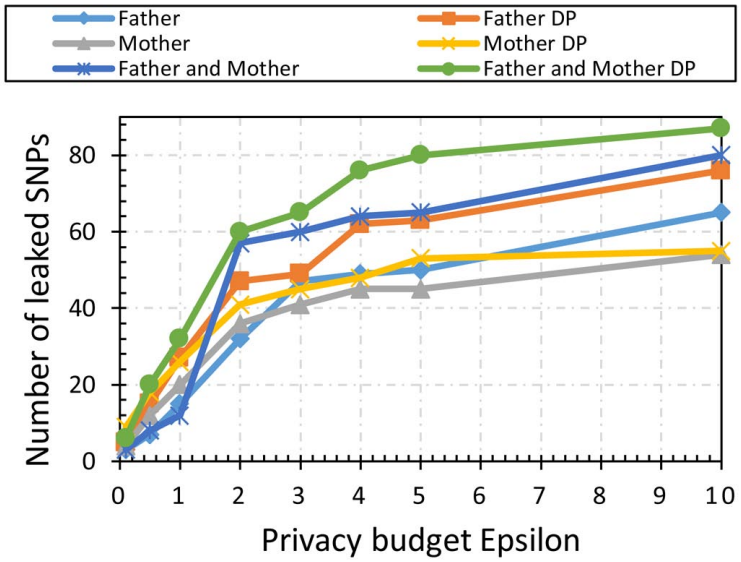
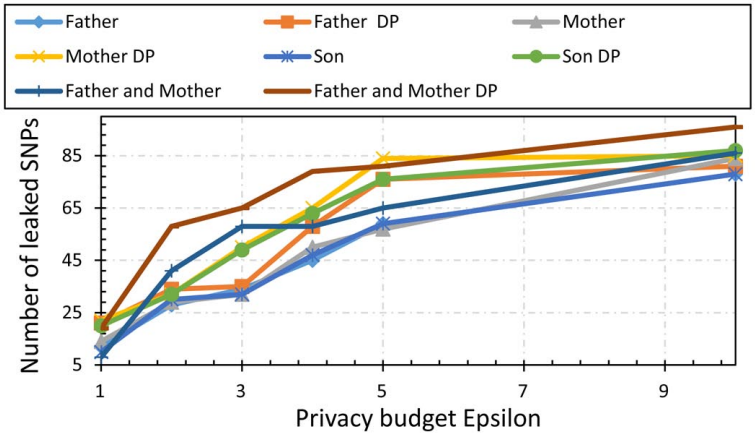


Figure 7.3: The effect of applying our proposed countermeasure for different values of the privacy budget, ϵ. "DP" lines stand for applying differential privacy mechanism (over 3 different sets of family members in the dataset) and the other three lines show the leaked SNPs when our proposed mechanism is applied.

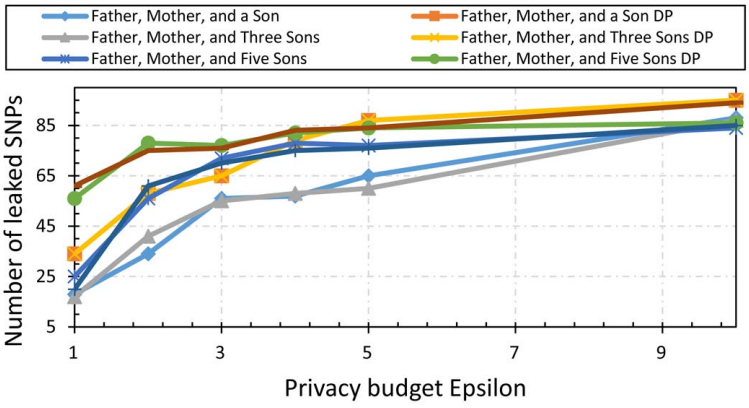
Consider the first scenario of our privacy model in which the data provider publishes the perturbed genomic data of $T_p^i + T_j^i$ where the query results only contain information from these dependent tuples. We use the empirically determined values in Equation 7.3 to compute σ for different number of dependent tuples d. Then, according to Equation 7.1, we compute the dependent sensitivity ς. Figure 7.3 analyzes the amount of leaked information for individual j, an adversary can reconstruct from the perturbed query results assuming two cases for the dependent tuple size d. As before, we assume the adversary target to be the son of Manuel Corpas family. The first query results include the data of the target and his father. We can see under the same privacy budget, ϵ, our privacy model has much lower leaked information than the DP approach except for ϵ=3 where we get almost the same number of leaked SNPs. The second query results include the data of the target and his mother. Similarly, our privacy model achieves better privacy for various privacy budgets. In the third query results, the dependent tuples size in the dataset increased to three; we have the target,

his father, and mother included in the query results. As illustrated in Figure 7.3, we can confirm that our privacy model provides better privacy performance in terms of leaked information metric.

In all different values of ϵ, using leaked information and correctness metrics our privacy model provides better privacy and it can reach up to 63% less leaked information in the case of $d = 3$ and ϵ=1. Therefore our privacy model achieves better privacy guarantees than the existing approaches of DP for genomic studies, and this advantage increases for smaller ϵ values.



(a)



(b)

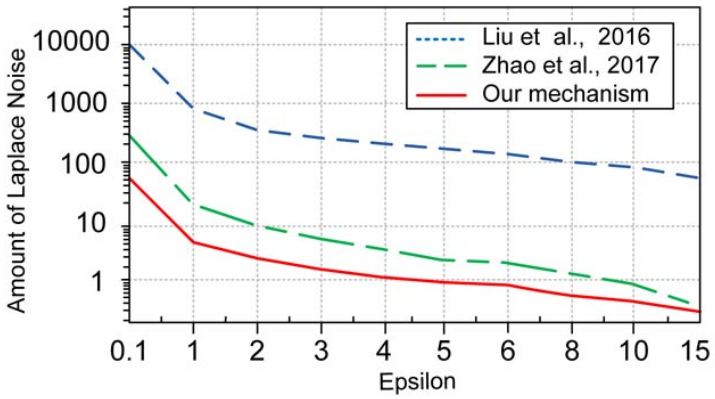
Figure 7.4: (a)(b) The effect of applying our privacy model using different values of the privacy budget ϵ, and different number of family relatives starting from 1 until 7 first degree family members over UTAH family dataset.

Next, in Figure S7.4 we apply our mechanism to the second dataset, in which we target the par 1 in CEPH/Utah dataset and try to protect him against any inference attack aiming to detect his genomic data exploiting that his family members are included in the same genomic dataset. We assume 8 cases for different numbers of correlated tuples d starting from 2 dependent tuples (the target j and 1 first degree family member) and gradually increase until 8 dependent tuples d (the target j and 7 first degree family member) in the dataset. Our model decreases the leaked information better than DP in all the eight cases. Hence, we increase the correctness of the adversary and decrease the leaked information about the genomic data of the target j as elaborated in Figure S7.4, in which we can reduce the leaked information up to 69% as in the case of d=9 and ϵ=1.

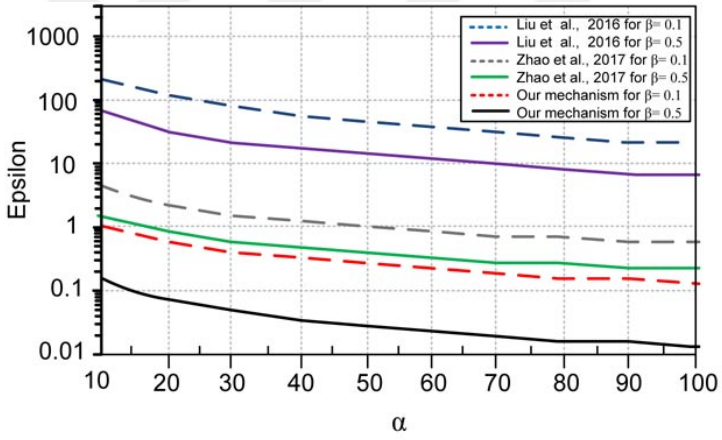
7.6 Comparison with Existing Work

In the following, we compare our mechanism with the most similar existing work [18, 19] using a sum query over a dataset with n=1000 tuples. Since [18] and [19] consider Markov chain-based correlations, in their models, all 1000 tuples are correlated. Thus, for this comparison, we also report the results of our scheme for 1000 dependent tuples.

Figure 7.5 compares our mechanism with [18] and [19] in terms of privacy (Figure 7.5(a)) and utility (Figure 7.5(b)). Figure 7.5(a) shows the amount of noise added to achieve ϵ-DP by considering the dependence between tuples. Here, we can see that for all ϵ values, our proposed scheme adds significantly smaller amount of noise, and hence provides better utility. For example, when $\epsilon = 0.1$, the amount of noise added in our scheme is 0.58% of the noise added by [19] and 17.32% of the noise added by [18].



(a)



(b)

Figure 7.5: (a) The amount of Laplace noise added for different values of privacy budget ϵ. (b) The privacy performance of different mechanisms which guarantee the (α, β)-usefulness. Here, the noisy output of the query should deviate by at most α from the real value (in terms of L1-norm) with probability $(1 - \beta)$.

Figure 7.5(b) shows the $(\alpha, \beta)-$usefulness defined by [90] which is commonly used for evaluating the utility guarantees for privacy mechanisms. It means the noisy output of the query should deviate by at most α from the real value (in terms of L1-norm) with probability $(1 - \beta)$. Figure 7.5(b) shows the smallest privacy parameter (ϵ) for different α values. For instance, to have $\alpha = 10$ and $\beta = 0.1$ (i.e., deviate by at most 10 from the original query result with a probability of 0.9), our proposed scheme requires a privacy budget of $\epsilon = 1.34$. To achieve the

same $(\alpha, \beta)-$usefulness, $\epsilon = 230$ shall be used for required of [19] and $\epsilon = 3.8$ shall be used for required of [18]. Thus, compared to existing work, for all α values, our mechanism requires a significantly smaller ϵ, and hence better privacy guarantees.

7.7 Summary

We introduce the notion of ϵ-differential privacy for genomic datasets with dependent tuples to avoid inference of sensitive information by any adversary with prior knowledge about the dependency between tuples.By evaluating our mechanism on different genomic datasets, our experimental results empirically demonstrate the following observations:

- Our model better minimizes the leaked information for genomic datasets compared to the state-of-the-art approaches [16, 19, 30, 67, 66, 91, 68]. Thus, we can select an appropriate privacy budget to achieve the optimal desired privacy while maintaining utility of the data for different genomic applications.

- Our model can achieve up to 50% on average better privacy guarantees based on the estimated error and the leaked information metrics than DP approaches, based on the leaked information metric, for publishing the average number of SNP values for a group of members participating on any genomic studies.

- Our model is resistant to state-of-the-art inference attacks [19, 65]. It reduces the leaked information even with a larger number of dependent tuples for various values of privacy budgets ϵ.

Chapter 8

Conclusion and Future Work

DP is considered as a concept that provides rigorous privacy guarantees. However, it suffers from weak privacy performance due to some limitations such as ignoring the dependence between the tuples in the dataset. In this thesis, we have demonstrated the limitations of the state-of-the-art DP-based mechanisms for genomic datasets with dependent tuples. We have utilized an inference attack to assess the vulnerability of these approaches and we have shown the effect of data dependence on the genomic privacy. For this, first, we have identified attribute inference attacks using three queries over real-life genomic datasets. We have shown how kinship relationships between individuals in a genomic dataset cause a significant reduction in the privacy guarantees of traditional DP-based mechanisms. An adversary, knowing the familial relationship between some individuals in a genomic dataset, may infer more information about the dataset participants than what is guaranteed by traditional DP. Furthermore, we have shown that these inferred genomic records (as a result of the attribute inference attack) can be utilized by the attacker to perform successful membership inference attacks to other statistical datasets. Our results demonstrate that the adversary can successfully determine that the target participated in the genomic study using its inferred SNPs. Our results also show the shortcomings of traditional DP-based mechanisms for privacy-preserving data sharing from static and dynamic statistical genomic datasets that include dependent tuples. To mitigate such privacy

risks, we have introduced ϵ-differential privacy for genomic datasets with dependent tuples that takes into consideration the probabilistic dependence relationship between data tuples and provides rigorous privacy guarantees. Furthermore, we have proposed a perturbation mechanism and have evaluated our perturbation mechanism over different genomic datasets to prove that it can achieve the required privacy guarantees. Our results show that our privacy model for genomic datasets performs significantly better than the existing DP-based mechanisms.

8.1 Future Research Directions

As future work, we plan to consider the followings:

1. Develop novel techniques to ensure privacy and utility preserving data sharing from dynamic statistical genomic datasets.
2. Theoretically compute the dependent sensitivity for applying LPM-DP for genomic statistical datasets with dependent tuples.
3. Find a new mechanism to mitigate the risks of membership privacy.
4. Applying other techniques like selective hiding to distort the dependent tuples.
5. Consider more specific inheritance patterns and disease-related genomic features to study the genomic correlation probabilities between tuples.

Bibliography

[1] N. Almadhoun, E. Ayday, and Ö. Ulusoy, "Differential privacy under dependent tuples—the case of genomic privacy," *Bioinformatics*, vol. 36, no. 6, pp. 1696–1703, 2020.

[2] D. G. Hert, C. P. Fredlake, and A. E. Barron, "Advantages and limitations of next-generation sequencing technologies: a comparison of electrophoresis and non-electrophoresis methods," *Electrophoresis*, vol. 29, no. 23, pp. 4618–4626, 2008.

[3] S. A. Kraft, M. K. Cho, K. Gillespie, M. Halley, N. Varsava, K. E. Ormond, H. S. Luft, B. S. Wilfond, and S. Soo-Jin Lee, "Beyond consent: Building trusting relationships with diverse populations in precision medicine research," *The American Journal of Bioethics*, vol. 18, no. 4, pp. 3–20, 2018.

[4] D. J. Carey, S. N. Fetterolf, F. D. Davis, W. A. Faucett, H. L. Kirchner, U. Mirshahi, M. F. Murray, D. T. Smelser, G. S. Gerhard, and D. H. Ledbetter, "The Geisinger MyCode community health initiative: an electronic health record–linked biobank for precision medicine research," *Genetics in Medicine*, vol. 18, no. 9, p. 906, 2016.

[5] G. Nanibaa'A, N. A. Sathe, A. H. M. Antommaria, I. A. Holm, S. C. Sanderson, M. E. Smith, M. L. McPheeters, and E. W. Clayton, "A systematic literature review of individuals' perspectives on broad consent and data sharing in the United States," *Genetics in Medicine*, vol. 18, no. 7, p. 663, 2016.

[6] E. Kobayashi and N. Satoh, "Public involvement in pharmacogenomics research: a national survey on public attitudes towards pharmacogenomics

research and the willingness to donate DNA samples to a DNA bank in Japan," *Cell and tissue banking*, vol. 10, no. 4, p. 281, 2009.

[7] J. M. Pulley, M. M. Brace, G. R. Bernard, and D. R. Masys, "Attitudes and perceptions of patients towards methods of establishing a DNA biobank," *Cell and tissue banking*, vol. 9, no. 1, pp. 55–65, 2008.

[8] C. L. Storr, F. Or, W. W. Eaton, and N. Ialongo, "Genetic research participation in a young adult community sample," *Journal of Community Genetics*, vol. 5, no. 4, pp. 363–375, 2014.

[9] I. Ishiyama, A. Nagai, K. Muto, A. Tamakoshi, M. Kokado, K. Mimura, T. Tanzawa, and Z. Yamagata, "Relationship between public attitudes toward genomic studies related to medicine and their level of genomic literacy in Japan," *American Journal of Medical Genetics Part A*, vol. 146, no. 13, pp. 1696–1706, 2008.

[10] A. K. Rahm, M. Wrenn, N. M. Carroll, and H. S. Feigelson, "Biobanking for research: a survey of patient population attitudes and understanding," *Journal of Community Genetics*, vol. 4, no. 4, pp. 445–450, 2013.

[11] M. Alser, N. Almadhoun, A. Nouri, C. Alkan, and E. Ayday, "Can you really anonymize the donors of genomic data in today's digital world?," in *Data Privacy Management, and Security Assurance*, pp. 237–244, Springer, 2015.

[12] A. L. R. Commission *et al.*, "Essentially yours–the protection of human genetic information in australia," vol. 1 and 2, no. 96, 2003.

[13] A. DeWan, M. Liu, S. Hartman, S. S.-M. Zhang, D. T. Liu, C. Zhao, P. O. Tam, W. M. Chan, D. S. Lam, M. Snyder, *et al.*, "HTRA1 promoter polymorphism in wet age-related macular degeneration," *Science*, vol. 314, no. 5801, pp. 989–992, 2006.

[14] N. Homer, S. Szelinger, M. Redman, D. Duggan, W. Tembe, J. Muehling, J. V. Pearson, D. A. Stephan, S. F. Nelson, and D. W. Craig, "Resolving individuals contributing trace amounts of DNA to highly complex mixtures using high-density SNP genotyping microarrays," *PLoS Genetics*, vol. 4, no. 8, p. e1000167, 2008.

[15] M. Y. Galperin, D. J. Rigden, and X. M. Fernández-Suárez, "The 2015 nucleic acids research database issue and molecular biology database collection," *Nucleic Acids Research*, vol. 43, no. D1, pp. D1–D5, 2015.

[16] C. Dwork, "Differential privacy: A survey of results," in *International Conference on Theory and Applications of Models of Computation*, pp. 1–19, Springer, 2008.

[17] D. Lv and S. Zhu, "Achieving correlated differential privacy of big data publication," *Computers & Security*, vol. 82, pp. 184–195, 2019.

[18] J. Zhao, J. Zhang, and H. V. Poor, "Dependent differential privacy for correlated data," in *2017 IEEE Globecom Workshops (GC Wkshps)*, pp. 1–7, IEEE, 2017.

[19] C. Liu, S. Chakraborty, and P. Mittal, "Dependence makes you vulnerable: Differential privacy under dependent tuples," in *NDSS*, vol. 16, pp. 21–24, 2016.

[20] A. Chaabane, G. Acs, M. A. Kaafar, *et al.*, "You are what you like! information leakage through users' interests," in *Proceedings of the 19th Annual Network & Distributed System Security Symposium (NDSS)*, 2012.

[21] A.-M. Olteanu, K. Huguenin, R. Shokri, M. Humbert, and J.-P. Hubaux, "Quantifying interdependent privacy risks with location data," *IEEE Transactions on Mobile Computing*, vol. 16, no. 3, pp. 829–842, 2016.

[22] D. Kifer and A. Machanavajjhala, "No free lunch in data privacy," in *Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data*, pp. 193–204, ACM, 2011.

[23] M. Humbert, E. Ayday, J.-P. Hubaux, and A. Telenti, "Addressing the concerns of the lacks family: quantification of kin genomic privacy," in *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*, pp. 1141–1152, ACM, 2013.

[24] S. Song, Y. Wang, and K. Chaudhuri, "Pufferfish privacy mechanisms for correlated data," in *Proceedings of the 2017 ACM International Conference on Management of Data*, pp. 1291–1306, ACM, 2017.

[25] R. Wang, Y. F. Li, X. Wang, H. Tang, and X. Zhou, "Learning your identity and disease from research papers: information leaks in genome wide association study," in *Proceedings of the 16th ACM Conference on Computer and Communications Security*, pp. 534–544, 2009.

[26] S. Sankararaman, G. Obozinski, M. I. Jordan, and E. Halperin, "Genomic privacy and limits of individual detection in a pool," *Nature Genetics*, vol. 41, no. 9, p. 965, 2009.

[27] J. Neyman and E. S. Pearson, "IX. on the problem of the most efficient tests of statistical hypotheses," *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, vol. 231, no. 694-706, pp. 289–337, 1933.

[28] N. Almadhoun, E. Ayday, and Ö. Ulusoy, "Inference attacks against differentially private query results from genomic datasets including dependent tuples," *Bioinformatics*, vol. 36, pp. i136–i145, 07 2020.

[29] Z. Wang and J. Moult, "SNPs, protein structure, and disease," *Human Mutation*, vol. 17, no. 4, pp. 263–270, 2001.

[30] D. Liben-Nowell and J. Kleinberg, "The link-prediction problem for social networks," *Journal of the American Society for Information Science and Technology*, vol. 58, no. 7, pp. 1019–1031, 2007.

[31] D. G. Wang, J.-B. Fan, C.-J. Siao, A. Berno, P. Young, R. Sapolsky, G. Ghandour, N. Perkins, E. Winchester, J. Spencer, *et al.*, "Large-scale identification, mapping, and genotyping of single-nucleotide polymorphisms in the human genome," *Science*, vol. 280, no. 5366, pp. 1077–1082, 1998.

[32] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 05, pp. 557–570, 2002.

[33] L. Sweeney, A. Abu, and J. Winn, "Identifying participants in the personal genome project by name," *Available at SSRN 2257732*, 2013.

[34] M. Gymrek, A. L. McGuire, D. Golan, E. Halperin, and Y. Erlich, "Identifying personal genomes by surname inference," *Science*, vol. 339, no. 6117, pp. 321–324, 2013.

[35] M. Kayser and P. de Knijff, "Improving human forensics through advances in genetics, genomics and molecular biology," *Nature Reviews Genetics*, vol. 12, no. 3, pp. 179–192, 2011.

[36] D. R. Nyholt, C.-E. Yu, and P. M. Visscher, "On Jim Watson's APOE status: genetic information is hard to hide," *European Journal of Human Genetics*, vol. 17, no. 2, p. 147, 2009.

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

[38] A. J. Pakstis, W. C. Speed, R. Fang, F. C. Hyland, *et al.*, "SNPs for a universal individual identification panel," *Human Genetics*, vol. 127, no. 3, pp. 315–324, 2010.

[39] E. E. Schadt, S. Woo, and K. Hao, "Bayesian method to predict individual SNP genotypes from gene expression data," *Nature Genetics*, vol. 44, no. 5, pp. 603–608, 2012.

[40] X. Zhou, B. Peng, Y. F. Li, Y. Chen, H. Tang, and X. Wang, "To release or not to release: evaluating information leaks in aggregate human-genome data," in *European Symposium on Research in Computer Security*, pp. 607–627, Springer, 2011.

[41] M. Backes, P. Berrang, M. Humbert, and P. Manoharan, "Membership privacy in MicroRNA-based studies," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, pp. 319–330, 2016.

[42] M. Naveed, E. Ayday, E. W. Clayton, J. Fellay, C. A. Gunter, J.-P. Hubaux, B. A. Malin, and X. Wang, "Privacy in the genomic era," *ACM Computing Surveys (CSUR)*, vol. 48, no. 1, pp. 1–44, 2015.

[43] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 1, no. 1, pp. 3–es, 2007.

[44] N. Li, T. Li, and S. Venkatasubramanian, "t-closeness: Privacy beyond k-anonymity and l-diversity," in *2007 IEEE 23rd International Conference on Data Engineering*, pp. 106–115, IEEE, 2007.

[45] E. Ayday, J. L. Raisaro, U. Hengartner, A. Molyneaux, and J.-P. Hubaux, "Privacy-preserving processing of raw genomic data," in *Data Privacy Management and Autonomous Spontaneous Security*, pp. 133–147, Springer, 2014.

[46] E. Ayday, J. L. Raisaro, J.-P. Hubaux, and J. Rougemont, "Protecting and evaluating genomic privacy in medical tests and personalized medicine," in *Proceedings of the 12th ACM workshop on Privacy in the Electronic Society*, pp. 95–106, ACM, 2013.

[47] P. baldi, R. Baronio, E. De Cristofaro, P. Gasti, and G. Tsudik, "Countering gattaca: efficient and secure testing of fully-sequenced human genomes," in *Proceedings of the 18th ACM Conference on Computer and Communications Security*, pp. 691–702, ACM, 2011.

[48] C. Uhlerop, A. Slavković, and S. E. Fienberg, "Privacy-preserving data sharing for genome-wide association studies," *The Journal of Privacy and Confidentiality*, vol. 5, no. 1, p. 137, 2013.

[49] F. Yu, S. E. Fienberg, A. B. Slavković, and C. Uhler, "Scalable privacy-preserving data sharing methodology for genome-wide association studies," *Journal of Biomedical Informatics*, vol. 50, pp. 133–141, 2014.

[50] 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.

[51] P. Baldi, R. Baronio, *et al.*, "Countering GATTACA: efficient and secure testing of fully-sequenced human genomes," in *Proceedings of the 18th ACM Conference on Computer and Communications Security*, pp. 691–702, ACM, 2011.

[52] E. Ayday, J. L. Raisaro, U. Hengartner, *et al.*, "Privacy-preserving processing of raw genomic data," in *Proceedings of 8th Data Privacy Management (DPM 2013) International Workshop (in conjunction with ESORICS)*, 2013.

[53] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Theory of Cryptography Conference*, pp. 265–284, Springer, 2006.

[54] K. Nissim, S. Raskhodnikova, and A. Smith, "Smooth sensitivity and sampling in private data analysis," in *Proceedings of the thirty-ninth Annual ACM Symposium on Theory of Computing*, pp. 75–84, ACM, 2007.

[55] F. McSherry and K. Talwar, "Mechanism design via differential privacy," in *48th Annual IEEE Symposium on Foundations of Computer Science*, pp. 94–103, IEEE, 2007.

[56] R. Drmanac, A. B. Sparks, M. J. Callow, A. L. Halpern, N. L. Burns, B. G. Kermani, P. Carnevali, I. Nazarenko, G. B. Nilsen, G. Yeung, *et al.*, "Human genome sequencing using unchained base reads on self-assembling DNA nanoarrays," *Science*, vol. 327, no. 5961, pp. 78–81, 2010.

[57] A. Machanavajjhala, J. Gehrke, and M. Götz, "Data publishing against realistic adversaries," *Proceedings of the VLDB Endowment*, vol. 2, no. 1, pp. 790–801, 2009.

[58] J. Lee and C. Clifton, "Differential identifiability," in *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1041–1049, ACM, 2012.

[59] N. Li, W. Qardaji, and D. Su, "On sampling, anonymization, and differential privacy or, k-anonymization meets differential privacy," in *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security*, pp. 32–33, 2012.

[60] N. Li, W. Qardaji, D. Su, Y. Wu, and W. Yang, "Membership privacy: a unifying framework for privacy definitions," in *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security*, pp. 889–900, ACM, 2013.

[61] F. Tramèr, Z. Huang, J.-P. Hubaux, and E. Ayday, "Differential privacy with bounded priors: reconciling utility and privacy in genome-wide association studies," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pp. 1286–1297, ACM, 2015.

[62] J. Gehrke, M. Hay, E. Lui, and R. Pass, "Crowd-blending privacy," in *Advances in Cryptology–CRYPTO 2012*, pp. 479–496, Springer, 2012.

[63] R. Bassily, A. Groce, J. Katz, and A. Smith, "Coupled-worlds privacy: Exploiting adversarial uncertainty in statistical data privacy," in *IEEE 54th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 439–448, IEEE, 2013.

[64] E. Lui and R. Pass, "Outlier privacy," in *Theory of Cryptography Conference*, pp. 277–305, Springer, 2015.

[65] M. Fredrikson, E. Lantz, S. Jha, S. Lin, D. Page, and T. Ristenpart, "Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing.," in *USENIX Security Symposium*, pp. 17–32, 2014.

[66] D. Kifer and A. Machanavajjhala, "A rigorous and customizable framework for privacy," in *Proceedings of the 31st ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, pp. 77–88, ACM, 2012.

[67] X. He, A. Machanavajjhala, and B. Ding, "Blowfish privacy: Tuning privacy-utility trade-offs using policies," in *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, pp. 1447–1458, ACM, 2014.

[68] R. Chen, B. C. Fung, P. S. Yu, and B. C. Desai, "Correlated network data publication via differential privacy," *The VLDB Journal-The International Journal on Very Large Data Bases*, vol. 23, no. 4, pp. 653–676, 2014.

[69] B. Yang, I. Sato, and H. Nakagawa, "Bayesian differential privacy on correlated data," in *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, pp. 747–762, ACM, 2015.

[70] Y. Cao, M. Yoshikawa, Y. Xiao, and L. Xiong, "Quantifying differential privacy under temporal correlations," in *2017 IEEE 33rd International Conference on Data Engineering (ICDE)*, pp. 821–832, IEEE, 2017.

[71] B. L. Browning, Y. Zhou, and S. R. Browning, "A one-penny imputed genome from next-generation reference panels," *The American Journal of Human Genetics*, vol. 103, no. 3, pp. 338–348, 2018.

[72] J. D'Errico, "Partitions of an integer," *MATLAB Central File Exchange*, vol. 12009, 2018.

[73] D. B. Goldstein, A. Allen, J. Keebler, E. H. Margulies, S. Petrou, S. Petrovski, and S. Sunyaev, "Sequencing studies in human genetics: design and interpretation," *Nature Reviews Genetics*, vol. 14, no. 7, p. 460, 2013.

[74] J.-W. Byun, Y. Sohn, E. Bertino, and N. Li, "Secure anonymization for incremental datasets," in *Workshop on Secure Data Management*, pp. 48–63, Springer, 2006.

[75] S. Kabou, S. mohamed Benslimane, and M. Mosteghanemi, "A survey on privacy preserving dynamic data publishing," *International Journal of Organizational and Collective Intelligence (IJOCI)*, vol. 8, no. 4, pp. 1–20, 2018.

[76] B. C. Fung, K. Wang, R. Chen, and P. S. Yu, "Privacy-preserving data publishing: A survey of recent developments," *ACM Computing Surveys (Csur)*, vol. 42, no. 4, pp. 1–53, 2010.

[77] X. Xiao and Y. Tao, "M-invariance: towards privacy preserving republication of dynamic datasets," in *Proceedings of the 2007 ACM SIGMOD international conference on Management of data*, pp. 689–700, 2007.

[78] A. Anjum, G. Raschia, M. Gelgon, A. Khan, N. Ahmad, M. Ahmed, S. Suhail, M. M. Alam, *et al.*, "τ-safety: A privacy model for sequential

publication with arbitrary updates," *Computers & Security*, vol. 66, pp. 20–39, 2017.

[79] O. Temuujin, J. Ahn, and D.-H. Im, "Efficient l-diversity algorithm for preserving privacy of dynamically published datasets," *IEEE Access*, vol. 7, pp. 122878–122888, 2019.

[80] E. Shmueli, T. Tassa, R. Wasserstein, B. Shapira, and L. Rokach, "Limiting disclosure of sensitive data in sequential releases of databases," *Information Sciences*, vol. 191, pp. 98–127, 2012.

[81] O. Gkountouna, S. Angeli, A. Zigomitros, M. Terrovitis, and Y. Vassiliou, "k m-anonymity for continuous data using dynamic hierarchies," in *International Conference on Privacy in Statistical Databases*, pp. 156–169, Springer, 2014.

[82] J. Salas and V. Torra, "A general algorithm for k-anonymity on dynamic databases," in *Data privacy Management, Cryptocurrencies and Blockchain Technology*, pp. 407–414, Springer, 2018.

[83] T. Zhu, G. Li, W. Zhou, and S. Y. Philip, "Differentially private data publishing and analysis: A survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 8, pp. 1619–1638, 2017.

[84] R. Chen, B. C. Fung, B. C. Desai, and N. M. Sossou, "Differentially private transit data publication: a case study on the Montreal transportation system," in *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 213–221, 2012.

[85] D. Su, J. Cao, N. Li, and M. Lyu, "Privpfc: differentially private data publication for classification," *The VLDB Journal*, vol. 27, no. 2, pp. 201–223, 2018.

[86] X. Cheng, P. Tang, S. Su, R. Chen, Z. Wu, and B. Zhu, "Multi-party high-dimensional data publishing under differential privacy," *IEEE Transactions on Knowledge and Data Engineering*, 2019.

[87] N. Johnson, J. P. Near, and D. Song, "Towards practical differential privacy for SQL queries," *Proceedings of the VLDB Endowment*, vol. 11, no. 5, pp. 526–539, 2018.

[88] F. Li and S. Zhou, "Challenging more updates: Towards anonymous republication of fully dynamic datasets," *arXiv preprint arXiv:0806.4703*, 2008.

[89] M. Corpas, "Crowdsourcing the corpasome," *Source Code for Biology and Medicine*, vol. 8, no. 1, p. 13, 2013.

[90] A. Blum, K. Ligett, and A. Roth, "A learning theory approach to noninteractive database privacy," *Journal of the ACM (JACM)*, vol. 60, no. 2, p. 12, 2013.

[91] T. Zhu, P. Xiong, G. Li, and W. Zhou, "Correlated differential privacy: hiding information in non-IID data set," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 2, pp. 229–242, 2015.

Appendix A

Data and Code

All the datasets and codes of our algorithms are available online on GitHub:

- Differential-privacy-genomic-inference-attack: `https://github.com/nourmadhoun/Differential-privacy-genomic-inference-attack`
- Inference-Attacks-Differential-Privacy: `https://github.com/nourmadhoun/Inference-Attacks-Differential-Privacy`