Saarland University

Faculty of Mathematics and Computer Science

Department of Computer Science

Health Privacy: Methods for privacy-preserving data sharing of methylation, microbiome and eye tracking data

Dissertation
zur Erlangung des Grades
des Doktors der Ingenieurwissenschaften
der Fakultät für Mathematik und Informatik
der Universität des Saarlandes

von
Inken Hagestedt

Saarbrücken,
2021

Tag des Kolloquiums:	8. Dezember 2021
Dekan:	Prof. Thomas Schuster

Prüfungsausschuss:

Vorsitzender:	Prof. Christian Rossow
Berichterstattende:	Prof. Michael Backes
	Prof. Vincent Bindschaedler
	Dr. Yang Zhang
Akademischer Mitarbeiter:	Dr. Zhikun Zhang

Zusammenfassung

Diese Dissertation beleuchtet Risiken für die Privatsphäre von biomedizinischen Daten und entwickelt Mechanismen für privatsphäre-erthaltendes Teilen von Daten. Dies zerfällt in zwei Teile: Zunächst zeigen wir die Risiken für die Privatsphäre auf, die von biomedizinischen Daten wie DNA Methylierung, Mikrobiomdaten und bei der Aufnahme von Augenbewegungen vorkommen. Obwohl diese Daten weniger stabil sind als Genomdaten, deren Risiken der Forschung gut bekannt sind, und sich mehr unter Umwelteinflüssen ändern, können bekannte Angriffe angepasst werden und bedrohen die Privatsphäre der Datenspender. Dennoch ist das Teilen von Daten essentiell um biomedizinische Forschung voranzutreiben, denn Daten von einer ausreichend großen Studienpopulation zu sammeln ist aufwändig und teuer. Deshalb entwickeln wir als zweiten Schritt privatsphäre-erhaltende Techniken, die es Wissenschaftlern erlauben, solche biomedizinischen Daten zu teilen. Diese Techniken basieren im Wesentlichen auf differentieller Privatsphäre und feindlichen Beispielen und sind sorgfältig auf den konkreten Einsatzzweck angepasst um den Nutzen der Daten zu erhalten und gleichzeitig die Privatsphäre zu schützen.

Abstract

This thesis studies the privacy risks of biomedical data and develops mechanisms for privacy-preserving data sharing. The contribution of this work is two-fold: First, we demonstrate privacy risks of a variety of biomedical data types such as DNA methylation data, microbiome data and eye tracking data. Despite being less stable than well-studied genome data and more prone to environmental changes, well-known privacy attacks can be adopted and threaten the privacy of data donors. Nevertheless, data sharing is crucial to advance biomedical research given that collection the data of a sufficiently large population is complex and costly. Therefore, we develop as a second step privacy-preserving tools that enable researchers to share such biomedical data. and second, we equip researchers with tools to enable privacy-preserving data sharing. These tools are mostly based on differential privacy, machine learning techniques and adversarial examples and carefully tuned to the concrete use case to maintain data utility while preserving privacy.

Background of this Dissertation

This dissertation is based on the papers mentioned below. For all but one papers, I was the first author, thus designed, implemented and evaluated the content by myself with valuable feedback from my co-authors. I contributed as second author to the paper [P1] and was involved only in the design, implementation and evaluation of the differentially private eye tracking part, therefore only that part of the paper is included in Chapter 7.

The first project [P2] was given to me by Mathias Humbert and Pascal Berrang. I designed, implemented and evaluated the attacks and defenses with feedback from Pascal Berrang, Mathias Humbert, Yang Zhang and Michael Backes. All authors reviewed the paper.

The idea to extend the Beacon project for methylation data was discussed on a scientific conference between Yang Zhang, Pascal Berrang, Mathias Humbert and XiaoFeng Wang, and later given to me to carry out the experiments resulting in paper [P3]. I designed, implemented and evaluated the attacks with feedback from Pascal Berrang, Mathias Humbert, Yang Zhang and Michael Backes. Together with Pascal Berrang, I designed the defense mechanism which I then implemented and evaluated, again with feedback from Pascal Berrang, Mathias Humbert, Yang Zhang and Michael Backes. All authors reviewed the paper, Dr. Ninghui Li later pointed out a technical inconsistency of the algorithm that was corrected after publication, Chapter 5 contains the correct version.

While searching for new projects, I got interested in the human microbiome and since data was publicly available, decided with Yang Zhang to study microbiome privacy [P4]. I designed, implemented and evaluated both attacks and defenses, with feedback from Yang Zhang and Michael Backes. Yuzhen Ye helped with preprocessing an additional data set. Yuzhen Ye and Haixu Tang gave additional valuable feedback in the writing phase of the paper, all authors reviewed the paper.

After discussions on ETRA 2018 and a follow-up Dagstuhl seminar, Andreas Bulling got interested in privacy for eye tracking. He proposed the project (that resulted in paper [P1]) to his student Julian Steil who needed a privacy expert and approached me. The collection of eye tracking data as well as the online survey was done by Julian Steil under supervision of Andreas Bulling and Michael Xuelin Huang. Together with Julian Steil, I designed the privacy mechanism and helped with the implementation and evaluation, Andreas Bulling and Michael Xuelin Huang gave valuable feedback. All authors reviewed the paper. Notice that this dissertation does only contain the privacy mechanism whose design and evaluation I contributed to, as well as necessary background and data set descriptions.

On ETRA 2019, Andreas Bulling, Julian Steil, Philipp Müller and me sat together and brainstormed on future eye tracking projects with focus on understanding privacy risks and designing suitable privacy-preserving mechanisms. One of the ideas, namely, to use adversarial examples as targeted defense against privacy-intrusive classifications, was later picked by Yang Zhang as my last project [P5]. I designed, implemented and evaluated the experiments with valuable feedback from Andreas Bulling. All authors reviewed the paper.

[P1] Steil, J., Hagestedt, I., Huang, M. X., and Bulling, A. Privacy-aware eye tracking using differential privacy. In: *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications, ETRA 19*. ACM. 2019, 27.

[P2] Hagestedt, I., Humbert, M., Berrang, P., Lehmann, I., Eils, R., Backes, M., and Zhang, Y. Membership inference against dna methylation databases. In: *IEEE European Symposium on Security and Privacy (EuroS&P)*. 2020.

[P3] Hagestedt, I., Zhang, Y., Humbert, M., Berrang, P., Tang, H., Wang, X., and Backes, M. Mbeacon: privacy-preserving beacons for dna methylation data. In: *Proceedings of the 26th Network and Distributed System Security Symposium (NDSS)*. 2019.

[P4] Hagestedt, I., Zhang, Y., Ye, Y., Tang, H., Wang, X., and Backes, M. *Quantifying Microbiome Privacy*.

[P5] Hagestedt, I., Backes, M., and Bulling, A. *Adversarial Examples for Privacy Protection of EyeTracking Data*.

Acknowledgments

I would like to thank my advisor Michael Backes for giving me the opportunity to persue my Ph.D. studies. I am greatful Michael Backes, Yang Zhang, Mathias Humbert and Andreas Bulling guided me through the research process of the various projects and taught me how reseach works, from experimental design to paper writing and presentations of research. Also I would like to thank my collaborators for their valuable input to the research projects. Especially, I am thankful Andreas Bulling and Julian Steil had the courage to reach out to me and pioneer the field of privacy-preserving eye tracking.

I am greatful for my colleagues that added ideas to the research process, helped with proof reading and dealing with paper rejections. Moreover, their curious questions that challenged my knowledge helped me broaden my knowledge of various topics.

Finally, I would like to thank my friends and family for their support, without them, this work would not be possible.

Contents

List of Figures

List of Tables

List of Algorithms

1

Introduction

Privacy risks of the genome are well studied[62, 159, 125], however, leakage of other epigenetic and biomedical data types has received less attention of researchers. Nevertheless, leakage of such data may lead to severe privacy risks as well. In this thesis, we move away from the genome data, representing the stable "building plan" of the cell, to study data about larger biological units and their interactions with the human host. Since these data types are influenced more by the environment [70, 127, 86, 63] we first have to answer the question of how much person-identifying information is detectable in the data in first place. On the other hand, if we can identify the person based on such biomedical data, additional inferences about the environment and the current health status of the individual can be made, as opposed to disease risk factors that can be inferred from the genome. The contribution of this work is two-fold: First, we demonstrate privacy risks of a variety of biomedical data types by adopting well-known attacks to the data, and second, we equip researchers with tools to enable privacy-preserving data sharing. Such data sharing is crucial to advance biomedical research given that measuring the data of a sufficiently large population is complex and costly. Our tools are based mostly on differential privacy [38], but also on machine learning techniques and adversarial examples [55].

First, we focus on DNA methylation data, which describes whether a small molecule, a so-called methyl group, is added to the genome at specific positions.

Our first work [P2] we study membership inference when sharing mean methylation values. Leveraging likelihood ratio tests and machine learning models, we demonstrate it is possible to distinguish members from non-members of a data set where only mean methylation values are released. This is even possible if genome values are available of the victim instead of methylation values due to the close link between the genome and methylation data. We apply differential privacy to mitigate these privacy risks, however, find a good trade-off between privacy and utility only in restricted settings with many individuals' contributions or few mean values released.

Knowledge of such population mean methylation values is the foundation of a search engine for methylation data sets. Recall that we want to facilitate biomedical research by enabling researchers to share data, so the first step is to find data sets of interest. The Global Alliance for Genomics and Health established the Beacon system, a search engine designed to help researchers find genomic data sets of interest. We extend [P3] the Beacon system to work with continuous methylation data and show that a naive implementation would again leak membership information. Thus, we propose to adopt the sparse vector technique to enable differentially private querying and show in extensive experiments that a trade-off between data donors' privacy and the utility of legitimate researchers can be found.

While for the genomic Beacon system the privacy risks were already studied by related work, the privacy of the microbiome was to the best of our knowledge just studied by one group of researchers in a single paper. The microbiome refers to the microorganisms (bacteria, viruses, and fungi) that live in and on the human body. There are at least as many microorganisms as body cells living in close symbiosis with us and influence our health and well-being. Thus, we explore [P4] whether the composition of the human microbiome uniquely identifies a person and are the first to systematically study the linkability of microbiome samples across time, across body

sites and across closely related humans. Despite complex interactions between the microbiome and the environment through direct contact or lifestyle choices such as diet, extensive experiments with microbiome data of various body sites show that linking can be performed effectively with an AUC between 0.75 and 0.9, demonstrating severe privacy risks. Simple defense mechanisms such as rounding, hiding rare features or using features only at a coarser level do not offer sufficient privacy protection.

Finally, we study another data type that is much easier accessible than methylation or the microbiome. The microbiome might be stealthily sampled from the environment of the target, but requires the attacker to directly access the person, and methylation data can only be measured from a probe of the respective cells (such as a blood sample). However, in order to access eye tracking data at scale, it suffices to compromise either eye tracking headsets or companies that offer classification services upon receiving the data stream. Given the rise in augmented and virtual reality applications and the advancements in hardware [75, 141], large amounts of data will get available soon. There is a large body of research on user modeling based on eye movements, and also higher-level user attributes such as gender and identity can be inferred.

In a first work [P1], we propose to add differentially private noise before such eye tracking data is released, which facilitates current research but can also be employed in the future as a protection against untrusted augmented and reality application providers. However, differential privacy is not targeted to the internal structure of eye tracking data itself, so in a second work [P5], we study whether better privacy protection can be reached when taking this structure into account. At the user side, this structure can be exploited by crafting adversarial examples against classifiers inferring privacy-sensitive attributes. We show that in a white-box setting, these adversarial examples do not decrease data utility, on the contrary, the data utility increases in case a re-identification classifier is targeted by the adversarial example. Additionally, we show that the service provider can take privacy into account during the feature selection phase and reach a set of features that preserves classification utility while leaking less private user attributes.

2
Background

This section introduces the different data types and explains the biological foundations. In the end, we also introduce differential privacy, a technique to ensure privacy which is used in multiple chapters. Further methods for privacy-preserving computations as well as attack methods are explained in the respecitve chapters.

2.1 Data Types

2.1.1 DNA and DNA Methylation

The DNA is a general "building plan" that is usually similar in every cell of the organism. Humans differ only in a very small amount (around 0.5% [99]) of their genes. The most common type of difference is a single position being different. These positions are referred to as single nucleotide polymorphisms (SNPs). A SNP is determined by a pair of nucleotides (among {A,C,G,T}): one that is called major allele as it is most frequent in the population and the other that is called the minor allele. Therefore, a given SNP can take three values: two major alleles, typically encoded as 0, one major allele and one minor allele, encoded as 1, and two minor alleles, encoded as 2.

DNA methylation is one of the most important epigenetic modifications, affecting both the structure and activity of the DNA molecule [70, 127]. The methylation process consists in the addition of a molecule, namely, a methyl group, to the C (cytosine) nucleotide, visualized by Figure 2.1. Because some regions of our methylation profiles are highly correlated with the genome, leakage of such data can indirectly expose family members' private data.

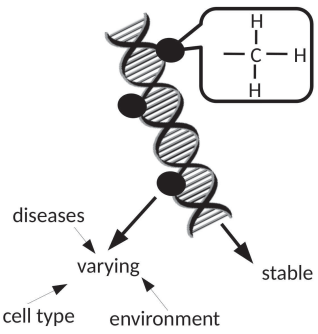


Figure 2.1: Schema of DNA and DNA methylation and their differences

Since DNA methylation may vary between copies of the DNA and across different cells, its value is quantified as the fraction of methylated nucleotides at a given genome position. Therefore, any DNA methylation position takes value in $\mathbb{R}_{[0,1]}$. With the current DNA methylation profiling technology, we can easily get access to several hundreds of thousands of DNA methylation positions in the human genome (e.g., the Illumina array provides 450k positions). We can get even more positions (up to tens of millions) by relying on more advanced technology such as whole-genome sequencing. Recent studies show that environmental factors such as exposure to stress or cigarette smoke, as well as the individual's age, correlate with changes in methylation values [12, 144, 147, 93, 92]. Moreover, aberrant DNA methylation patterns are often correlated with cancer stemming from the activation of genes such as oncogenes or the silencing of tumor suppressor genes [43]. Besides environmental factors, methylation regions can also be influenced by genomic variants at some specific positions [139, 96, 52]. We summarize the main differences between DNA methylation and SNPs in Table 2.1.

	# positions	value range	time evolution
DNA methylation	$\sim 10^7$	$\mathbb{R}_{[0,1]}$	varying
SNPs	$\sim 10^8$	$\{0, 1, 2\}$	stable

Table 2.1: Key differences between DNA methylation and genomic variants (SNPs). Note that the number (#) of positions shows the total number of currently known positions but that this number can be orders of magnitude smaller in popular profiling technology such as the Illumina array (450k positions).

2.1.2 Microbiome

There are at least as many microbiome cells as human body cells [128]. The microbiome is composed of a variety of species, they encode 150 times more unique genes than our own genome [116]. These microorganisms are bacteria, viruses and fungi, living in close symbiosis with their host and are mostly helpful rather than harmful. For example, the gut microbiome can influence diseases such as diabetes, obesity and Rheumatoid arthritis [94]. Moreover, the human host and the microbiome communicate with each other [124].

Which species of microbiome inhabit which body region depends on the body region itself, since some species need oxygen, for instance. Additionally, environmental factors such as diet [31] and personal cues such as age [29], sex and body mass index [34] influence which species thrive in the respective body region. We expect many more of these correlations to be found in the future [45].

On the one hand, this is the foundation for personalized medicine as well as early detection and better treatment of diseases in the future. On the other hand, it can also turn into a privacy threat if the data falls into the wrong hands and a diagnosis or inference of the aforementioned personal traits based on stolen microbiome data is used to discriminate against a person. Such a scenario is not far-fetched given the raise of direct-to-consumer testing by companies such as BaseClear [11] and Atlasbiomed [6]. Moreover, it is not easy to protect our privacy by not leaving any microbial traces because we all continuously shed parts of our microbiome [97]. Additionally, we expect the microbiome to be sequenced more frequently by health care providers given the increasing amount of research [45] showing the importance of the microbiome for various diseases on the one hand and the decreasing costs for sequencing on the other hand.

The microbial DNA is extracted from a probe either from the complete DNA strings (referred to as “shotgun” sequencing) or only from a part of the DNA, for example 16S rRNA genes (known as “16S rRNA” sequencing). Since 16S rRNA sequencing is cheaper and more data is available in that format, we study data from that sequencing technique. The 16S rRNA sequencing reads are first clustered into groups by similarity. These groups are called OTUs (“operational taxonomic unit”) and are likely to represent a species. Next, OTUs are assigned to a bacterial species, or a higher taxonomic rank such as genus or family if the exact species is not yet in the reference database. The more sequencing reads are found from the same species, the more abundant the species in the probe. The results are summarized in a OTU abundance table, which contains, for each OTU, its relative abundances in the probe. Due to its simplicity for non-biologists, we

will focus on this data format in our study.

2.1.3 Eye Tracking Data

The measurement and processing of eye tracking data can be used for computational user modeling [23, 126, 27], psychology research [76], human-computer interaction [20] or virtual [48] and augmented reality [60]. However, eye movements contain private information that could be highly valuable for an attacker, such as personality traits [63], mental health issues [150] or recent drug consumption [2]. Additionally, the way we move our eyes reveals person-specific information that allows for individuals to be identified from eye movements alone [66, 79]. This is particularly concerning given that people rarely think about, let alone control, their eye movements consciously in daily life. Combined with the fact that they users not (yet) aware of the rich information content available in eye movements [P1], this urgently calls for research on privacy-aware eye tracking.

The eye tracking pipeline typically consists of multiple steps: recording, event detection, feature extraction and finally classification. We explain these steps in detail in this section. First, a video of the eye movement is recorded with stationary or head-mounted cameras. From this video data, the position of the eye and pupil diameter are extracted often with specialized software provided by the manufacturer. The result is a series of "data points" containing an x and y position, the pupil diameter, and a value indicating the confidence of the extracted eye position. If the eye tracker was calibrated at the beginning of the recording, pupil positions can be mapped to gaze positions, i.e. where the person is looking, typically on a computer screen. We refer to this series of eye positions, pupil diameter and confidence information as "raw data" in our work (see Figure 2.3).

Typical preprocessing steps used by most eye tracking researchers are event detection followed by feature extraction. Event detection groups several data points into physiologically meaningful events, i.e., a fixation, a saccade, a blink or a smooth pursuit (see Figure 2.3 for sample events). A fixation occurs if the eye focuses on one point and is almost stationary. A jump from one fixation point to another is called a saccade, which is a short and fast eye movement. Smooth pursuit occurs if the eye follows a moving object, so the eye is not stationary, but moves slower than during saccades. We do not detect smooth pursuits in this work given that the sample task of reading rarely involves smooth pursuit events. It is important to note, however, that smooth pursuits could provide additional important information for other tasks. Finally, if the eye is closed, we detect a blink. The eye tracker can not detect any eye position and often returns x=y=0. We must notice that failure to detect the eye position does not necessarily indicate a blink, especially if it occurs only for a very short time. These failures happen several times in practice, as Figure 2.3 shows, e.g. due to motion blur

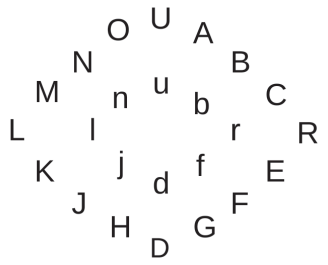


Figure 2.2: Encoding schema of saccade directions into letters

during saccades or shortly after blinks.

For example, reading text will usually result in a sequence of short saccades over the line followed by a long saccade to jump to the beginning of the next line. We can observe this in Figure 2.3 as well. Moreover, comparing the eye movements from reading this paper to reading a novel would probably result in more saccades to the right for the novel due to longer lines and more saccades and fixations in the same time (assuming a novel is "easier" and therefore faster to read than a scientific paper).

This example also demonstrates that high-level features about the sequence and number of events within a certain time window can be meaningful. Therefore, most preprocessing includes a feature extraction step that generates statistical features about the number, mean and variance of various events [23]. As the saccade directions contain crucial information especially for reading, we categorize the directions into a set of letters as shown in Figure 2.2 and distinguish shorter and longer saccades. Statistical features about the saccades can be derived by considering variance, maximum and minimum of n-grams of such saccade encoding which are referred to as wordbooks. The resulting set of features is usually classified by traditional machine learning models such as random forests [84] or Support Vector Machines (SVM), we follow previous work [P1, 23] and train SVM with radial basis function (RBF) kernel.

2.2 Differential Privacy

Differential privacy [38] is one framework to achieve privacy and is used in several of the works presented here. Differential privacy (DP) guarantees that the answer of the privacy-preserving mechanism does not depend on whether a single user contributed her data or not; hence, there is no way to infer further information about this user, say, Alice. We denote a differentially private mechanism by $\mathcal{M}$ and refer to Alice's data as a single data element in the database D. Typically, $\mathcal{M}$ adds random noise to "hide" each data element, which we will formalize in the following.

Definition 1 (ϵ-Differential Privacy [37])**.** *A mechanism $\mathcal{M}$ provides ϵ-differential privacy if for all databases D, D' that differ in at most one element and for every $S \subseteq Range(\mathcal{M})$, we have*

$$Pr[\mathcal{M}(D) \in S] \leq e^{\epsilon} \cdot Pr[\mathcal{M}(D') \in S]. \tag{2.1}$$

Differential privacy allows computing an arbitrary function g over the database, i.e. $g : \mathcal{R}^* \mapsto \mathcal{R}^d$, where d denotes the dimensionality of the output of g.
How much noise we have to add depends on the variance of the data between two arbitrary elements. Formally:

Definition 2 (L_1 Sensitivity [37])**.** *For all functions $g : \mathcal{R}^* \mapsto \mathcal{R}^d$, the L_1 sensitivity is the smallest number Δ_g s.th. for all databases D, D' differing in one element, we have*

$$||g(D) - g(D')||_{L_1} \leq \Delta_g. \tag{2.2}$$

Intuitively, the sensitivity captures the maximal influence Alice's data could have on the answer to our query. In the worst case, for her privacy, Alice's data is an outlier,

e.g. Alice's value is very high or very low. Even in this case, the difference of g applied to a database with or a database without Alice's data must be smaller than or equal to the sensitivity.

The noise to "hide" Alice's contribution is scaled to this worst case, ensuring Alice's privacy. Thus, the amount of noise depends on the sensitivity of g, which in turn depends on g itself. Intuitively, if g has small fluctuations if one database entry is changed, the sensitivity is small and only a small amount of noise has to be added, hence the result of $\mathcal{M}$ has high accuracy. Therefore, applying differential privacy requires careful choice of g and the noise mechanism to maintain utility while preserving privacy.

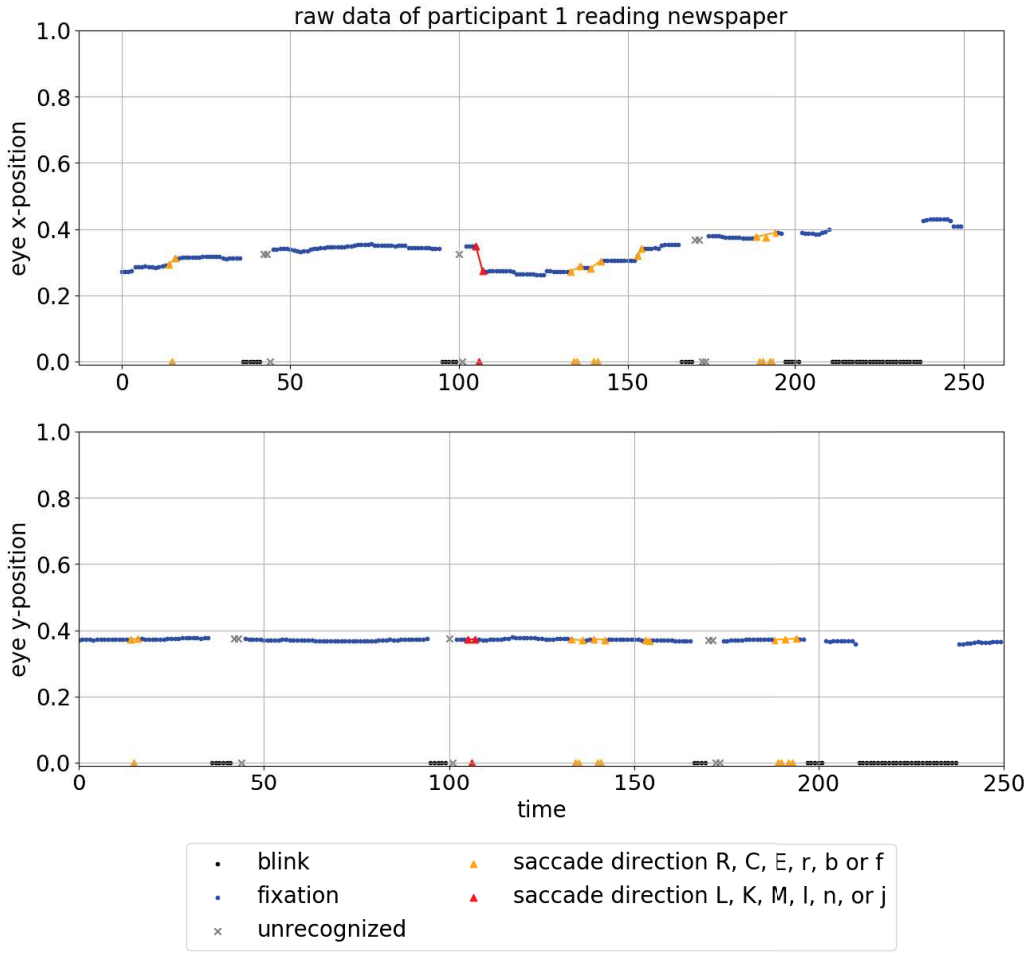


Figure 2.3: Example of x- and y-positions of the recorded eye. Each dot is a data point. We color dots differently depending on their detected events. The sample mostly contains fixations (blue), but also some reading-typical saccades (short, yellow saccades to the right) and one longer saccade to the left (red). Since the saccade direction is computed from first and last point in the saccade, we connect these points by lines in the respective color. The participant also blinks (x- and y-positions zero at least three consecutive frames, shown in black) and some points can not be recognized (shown in gray) because they are too short for fixations or blinks.

3
Related Work

In this chapter, we present related works for privacy attacks and defenses in general and also for the topics studied in this thesis.

3.1 Privacy Attacks in General

Membership inference attacks threaten the privacy of data donors to a database containing their biomedical data. In a nutshell, the goal of membership inference attacks is to infer whether a target, whose data is known to the attacker, contributed the data to the database. If so, the attacker can infer further information about the target, e.g., that the target took part in a cancer study and thus suffered from this disease. Notice that this inference goes beyond diagnosis of the target's data.

Homer et al. were the first to present a membership inference attack by relying on summary statistics over genomic data and the L_1 distance between those and the target's data [62]. An extension to this attack was proposed by Wang et al. [153] using the intra-genome correlations which allowed to rely on only a few hundreds genomic positions. The theoretical complexity was further studied by Zhou et al. as well as recovery attacks based on summary statistics [159]. Moreover, Sankararaman et al. derived an upper bound on the power of membership inference with genomic data, and showed empirically that the likelihood-ratio (LR) test was more powerful than the L_1 distance attack [125]. An overview on the privacy risks for genome data can be found in the paper by Ehrlich and Narayanan [42].

Backes et al. [9] were the first to propose a membership inference attack against another type of biomedical data, namely transcriptomic data (microRNA expression). Despite the smaller dimensionality of the microRNA profiles (a few thousands points instead of millions with the genome), the attack based on L_1 distance and the likelihood-ratio test proved to be successful against disease-related databases.

Shokri et al. studied membership inference attacks against the training data sets of machine-learning models such as neural networks [131], while Hayes et al. studied the same attacks against generative models [58]. The authors showed that their attacks can be successfully performed against medical image data sets, further demonstrating the extent of the privacy threat. Moreover, Pyrgelis et al. [115] carried out a membership inference attack against location data. They used statistical features and fit several machine learning classifiers to infer membership. Additionally, they use various differential privacy mechanisms to protect the location data.

3.2 Defenses against Privacy Attacks

As membership inference attacks and other privacy attacks pose significant risks for data donors, the databases need to be protected. Differential privacy is one of the standard tools which we explain in more detail below. Then, we present related works for another, more targeted perturbation at the end of this section.

3.2.1 Differential Privacy

Differential privacy has been studied in privacy research for more than a decade in terms of its theoretical foundations and its practical applications to different data types, such as location [115], biomedical data [122], or continuous time series data [44]. We refer the reader to [160] for a survey. A key challenge in differential privacy is to find the right trade-off between privacy and utility, that is, the right amount of random noise to "hide" an individual without hampering data utility. Fredrikson et al. demonstrated how important it is to balance privacy and utility [50]. They observed that either privacy was not preserved or that utility suffered, leading to increased health risks for the patients from unsuitable drug dosage. A good privacy-utility trade-off is possible if privacy mechanisms are tailored towards a specific use case [115, 44].

How to apply differential privacy to genomic databases has been extensively studied. Johnson and Shmatikov have proposed algorithms that protect the output of data exploration (p-values and correlations, number and location of SNPs most likely associated with a disease) with differential privacy [69]. Uhler et al. have also proposed to release differentially-private summary statistics (allele frequencies, p-values, and χ^2 statistics) [146]. This was extended by Yu et al. to allow for arbitrary numbers of case and control samples [155].

Given that the amount of noise required to achieve differential privacy is very often too high to keep enough utility, Tramèr et al. [143] proposed a relaxation of differential privacy that assumes a weaker adversary in order to reach a better privacy-utility trade-off.

Finally, Backes et al. have applied differential privacy to microRNA expression's summary statistics for preventing membership inference attacks with such data [9]. Their results confirmed the difficulty of finding an optimal privacy-utility trade-off, especially when the number of participants in the database is small.

Differential privacy was only recently applied to eye tracking data at feature level by Steil et al. [P1] and aggregated gaze heat maps by Liu et al. [89]. More closely related to our work on privacy-preserving eye tracking is the unpublished paper by Fuhl [51] which also removes certain information from eye tracking data while leaving utility intact. However, their approach is different and uses reinforcement learning on a feature representation learned by an autoencoder.

3.2.2 Adversarial Examples for Attacks on Classifiers and Beyond

Small perturbations that are imperceptible to the human observer are in some cases able to completely throw off a classifier and can often be efficiently found. These perturbations are called adversarial examples and are studied for more than a decade [15] both for traditional machine learning classifiers and for modern neural network architectures. We refer the reader to the SoK by Papernot et al [110] for an overview of the recent state of the art and focus in our summary here on those attacks that we need and the usage of adversarial examples.

Our white-box attack uses the fast gradient sign method (FSGM) developed by Goodfellow et al. [55]. In a nutshell, FGSM linearizes the gradient and perturbs the sample in the direction of the gradient. We have chosen FGSM due to its indepen-

dence of neural network architectures because we need to apply it to support vector machines (SVM) with radial basis function (RBF) kernels. FGSM just relies on the gradient, which is well defined for SVM [14]. For our black-box attack we use the recent HopSkipJump attack developed by Chen et al. [28] that only needs access to the classification labels. Given the benign sample and an initially misclassified point, HopSkipJump iteratively estimates the gradient and the decision boundary to move the misclassified sample towards the boundary. Both FGSM and HopSkipJump are implemented in the adversarial robustness toolbox ART [104] which we rely on in our experiments.

We are not the first to produce adversarial examples not only to show the vulnerability of the targeted classifier. Dong et al. [35] used adversarial examples against deep neural networks for image classification in order to understand its predictions and uncover what the model learned and failed to learn. Also, Ilyas et al. [65] use adversarial learning to uncover robust and non-robust features, this distinction can be a first step towards robust models. More closely related to our work is the proposal by Jia et al. [67] to generate "adversarial noise" in order to protect against attribute inference attacks. This noise is included in form of an additional loss function during the training process, leads to a privacy-preserving model and does not require knowing exact knowledge of private attributes. Nasr et al. [101] follow a similar approach to protect against membership inference attacks. While this is rather adversarial training than adversarial examples, Overdorf et al. [106] produce adversarial examples to poison a machine learning model in order to shift the harm from misclassification on a subset of underrepresented data to the ML provider in form of a generally increased false-positive rate.

While there is a large body of research that sees adversarial examples as a problem that needs to be fixed with robust machine learning models, other researchers, including us, see adversarial examples as a tool to understand the models and the data more thoroughly.

3.3 Privacy Risks in Specific Use Cases

In this section, we dive deeper into the detailed risks of different data types and their unique privacy risks due to their biological function.

3.3.1 Privacy Risks of Methylation Data

Other than membership attacks, Philibert et al. showed that methylation data could be relied upon to infer part of the genotype and behavioral attributes such as alcohol consumption and smoking [113]. Besides also identifying methylation points correlated with genomic variants, Dyke et al. proposed high-level guidelines for methylation data disclosure that preserves privacy [40]. Backes et al. used the correlations between certain positions of the genome and methylation data in order to re-identify DNA methylation profiles by matching them to their corresponding genome [7].

3.3.2 Privacy Risks of Genomic Beacons and Protection Mechanisms

Shringarpure and Bustamante [132] showed that even only given binary responses, it is possible to infer whether a patient is in a Beacon with the LR test. Moreover, their attack's probability estimation is not dependent on the allele frequencies, but the more stable allele distribution. While they studied the influence of several factors (population structure, Beacon size and others) on the attack's effectiveness, they did not propose any feasible solutions to establish a privacy-preserving genomic Beacon.

Raisaro et al. [117] extended the attack in [132] by adopting a sophisticated selection strategy. The attacker in this setting has direct access to allele frequencies and selects the most informative positions to query first. This setup serves as a blueprint for our attack against MBeacons.

The authors of [140] proposed an attack using the correlations between different single nucleotide polymorphisms (SNPs) to infer alleles that are missing or systematically hidden. This attack drops the number of queries necessary to infer membership with strong confidence, and renders privacy-preserving mechanisms based on hiding low-frequency SNPs useless. However, for DNA methylation, such correlations are not (yet) well studied. Therefore, we decide to postpone an in-depth study about the influence of correlations between methylation positions on the privacy risks to future work.

Privacy Protection for Beacons Besides the attack, Raisaro et al. [117] proposed three protection mechanisms and experimentally showed their effectiveness even in their stronger attacker setting. However, they do not provide any formal guarantees on their protection mechanisms.

Wan et al. [152] further analyzed the protection mechanisms presented in [117], and additionally proposed a new one. They empirically evaluated utility, privacy and effectiveness of the protection methods under several settings with respect to the hyperparameters. Here, the corresponding utility, privacy and effectiveness measures were proposed in the iDASH challenge for genomic data.

Two additional privacy protection mechanisms are proposed by Al Aziz et al. [3], one of which, the biased randomized response, is proven to be differentially private. Apart from that, they analyzed both mathematically and experimentally how the decision boundary for membership relates to the number of queries and the number of patients in the Beacon.

3.3.3 Privacy Risks of Microbiome Data

The privacy risks of microbiome data were only studied by Franzosa et al. [49]. It is unclear how their findings carry over to more complex scenarios, in which no training sets are available to generate the hitting sets. Therefore, we systematically study a variety of linkability attacks considering different attack models and methods. Additionally, we focus on the more widespread OTUs (("operational taxonomic unit") format that is also easier to access for attackers with shallow biomedical background.

Song et al. [133] view their data set on cohabiting participants from the forensics perspective rather than the privacy perspective, and concluded that family members are more similar to each other and their dogs than to unrelated people.

Other forensics researchers such as Lax et al. [86] conducted studies about the surfaces we touch or step on, or Meadow et al. [98] about (phone) surfaces touched. While showing impressive results on linkability, these studies are small-scale, Lax' phone samples are from two participants only, and Meadow et al. had 17 participants. The same is true for another work by Meadow et al. [97] who studied the particles shed in the air and observed in their small-scale study that it is possible to re-identify people both with particles in the air and fallen down to surfaces even hours after they left the sterile experimental chamber.

The impending privacy risks from large-scale measurements of microbiome has been summarized by Shamarina et al. [129]. Wagner et al. [151] describe the first step towards a solution in the case where two parties can not share their data, but want to compute a function such as differences in the abundance of microbiome species across different populations. They propose to use garbled circuits, a cryptographic method, which hides the microbiome data, but comes at the cost of increased computation time.

3.3.4 Privacy Risks of Eye Tracking Data

An ever-increasing body of work is demonstrating the rich source of information available in eye movements for computational user modeling. That is, prediction of user attributes and context from eye movements alone or in combination with other sensing modalities. Bulling et al [23, 24] pioneered the field of activity recognition and demonstrated that several activities that naturally occur in office environments, including reading, can be robustly inferred from stationary and mobile eye trackers. Kunze et al. [84] focused on reading alone. They proposed methods for distinguishing different documents by the eye movement during reading. Not only the user's current task, but also interests, cognitive load and personality traits can be inferred, as Hess et al. [59], Matthews et al. [95] and Hoppe et al. [63] showed. Moreover, Sammaknejad et al. [123] demonstrated that eye movements of men and women are different when looking at faces. Additionally, diseases such as schizophrenia [61], Parkinson's [80] and mental health issues [150] can be inferred as well as sensitive personal information such as recent drug consumption [2]. Despite the eye movement leaking highly sensitive personal and medical information, the privacy research in the field is still in its infancy.

Eye movement biometrics has emerged as a promising approach to user authentication [73]. While first works required a point stimulus that users were instructed to follow with their eyes [72, 74], later ones explored static points [13] or images [91]. Kinnunen et al. presented the first method for "task-independent" person authentication using eye movements [77]. Eberz et al. presented a biometric based on eye movement patterns. They used 20 features that allowed them to reliably distinguish and authenticate users across a variety of real-world tasks, including reading, writing, web browsing, and watching videos on a desktop screen [41]. Zhang et al. used eye movements to continuously authenticate the wearer of a VR headset by showing different visual stimuli [158].

3.4 Summary

Related works shows that different biomedical data exposes different details about the target, however, privacy attacks are often similar, such as membership inference. Well-studied tools such as differential privacy can be used to reduce privacy risks in many cases. However, such solutions need to be tailored to the use case, which is one of the contributions of this thesis.

4

Membership Inference Against DNA Methylation Databases

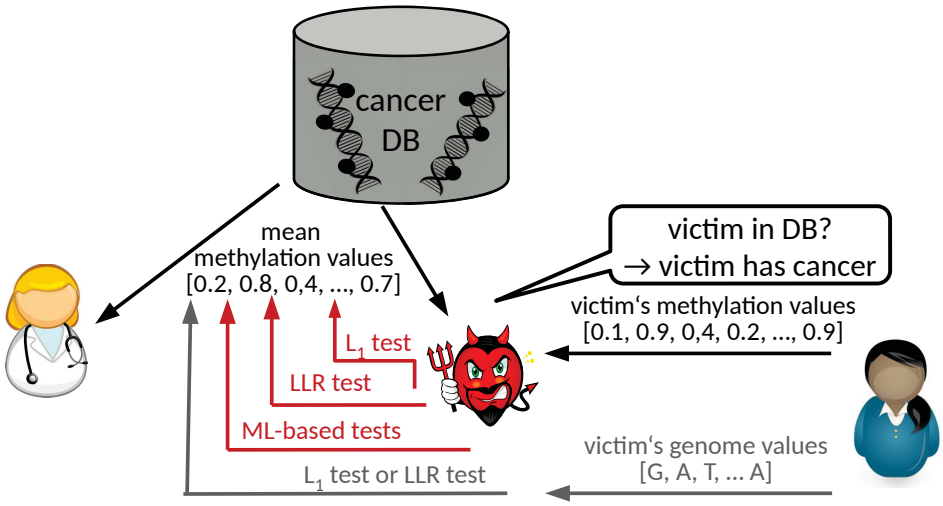


Figure 4.1: Overview of our attack scenario on methylation data: Given the victim's methylation values and mean methylation values from a database, the attacker tries to infer whether the victim is part of the database. The attacker uses statistical tests (L_1 or LLR tests) or ML-based tests. If only the genome values of the victim are available as shown in gray, the attacker first infers the respective methylation values and then carries out the statistical tests.

In this chapter, we study whether and to which extent DNA methylation data, one of the most important epigenetic elements regulating human health, is prone to membership inference attacks, a critical type of attack that reveals an individual's participation in a given database. Figure 4.1 gives a graphical overview of our different attacks. Our attacks exploit published summary statistics, among which one is based on machine learning and another exploits the dependencies between genome and methylation data, see Table 4.1 for a summary. Our extensive evaluation on six data sets containing a diverse set of tissues and diseases collected from more than 1,300 individuals in total shows that such membership inference attacks are effective, even when the target's methylation profile is not accessible. While the best performing statistical test, the LLR test, exceeds 0.9 AUC (area under the ROC curve) for one data set only, the machine-learning attack almost always reaches AUC of at least 0.9. Our empirical results also show that data is transferable between different diseases and tissues: The model trained on a different type of data set from the target data set achieves similar performance to the model trained on the same type of data set. Finally, the genome-based attack provides excellent performance with around 0.9 AUC, even though it performs slightly worse than the methylation-based attack.

The paper [P2] contains the aforementioned results. Additionally, we developed a defense mechanism based on differential privacy that we present in the end of the Chapter in Sections 4.6 and 4.7.

Organization We introduce our attacker models in Section 4.1 and the theoretical

Attacker's knowledge	Statistics about methylation database	Raw target data		External/auxiliary data	
		methy-lation	genome	methy-lation	genome + methylation
Statistical attack	✓	✓	-	-	-
Machine-learning attack	✓	✓	-	✓	-
Genome-based attack	✓	-	✓	-	✓

Table 4.1: Overview of our different attack settings. ✓means the attack needs this information, and - means it does not. When relying on methylation data, we also use training/test data from different tissues or with different diseases.

foundations of our attacks in Section 4.2. In Section 4.3, we detail the diverse datasets used for the evaluation of our attacks in Section 4.4 and conclude in Section 4.5. The additional theory and empirical evaluation of the defense mechanism with differential privacy can be found in Sections 4.6 and 4.7 respectively.

4.1 Threat Model

The adversary's objective is to determine whether an individual (referred to as a *target*) is a member of a group of study that we will refer to as a *pool*. By leveraging such an attack, the adversary can infer sensitive information about her target, as pools in medical studies can be associated with severe diseases.

To run her attack, the adversary gets access to aggregated methylation data that describe the statistical properties of the considered methylation data pool. While these aggregate data are usually published alongside biomedical case-control studies (see for example [102, 78, 138]), such aggregate data can nowadays also be queried from federated systems such as i2b2 [100], SHRINE [154] or MedCo [118]. In this work, we assume that the *mean* statistics about the pool are available to the adversary as it is the most common statistics currently available. Additionally, we assume the adversary has access to general methylation statistics of the reference population. Currently, these statistics have to be estimated by the adversary using a subset of the underlying population. However, we expect that population-wide statistics for DNA methylation will become publicly available, as for genomic data. We will refer to the subset of the reference population as the *reference group* here.

In order to perform the attack, the adversary also needs access to some raw data of the target. For our methylation-based attack, we assume the adversary knows the target's DNA methylation at m positions encoded as $\overrightarrow{x} \in \mathbb{R}^m_{[0,1]}$, from similar or different tissue/disease type as the targeted database. Full individual DNA methylation profiles are increasingly available in public databases such as the Gene Expression Omnibus (GEO) [54] or ArrayExpress [5]. Moreover, with the increasing adoption in medical practice, DNA methylation data will also certainly be stored on hospital servers, potentially putting such profiles at risk. For instance, cyber-attacks against healthcare companies have increased by 72% from 2013 to 2014 [16].

As genomic data is currently more accessible than methylation data, we also propose

and investigate a genome-based attack. We assume the adversary knows (part of) the genotype of the target instead of his methylation data. By now, more than 10 million individual genotypes have been sequenced through direct-to-consumer genetic testing [30], such as 23andMe [1] or AncestryDNA [4]. Those individuals can also share their sequenced genotypes online, on open platforms such as GEDmatch [53], OpenSNP [105], or the Personal Genome Project (PGP) [112], sometimes with their real identifiers. Therefore, even without considering the genomic databases at clinical premises, millions of genomic profiles are already freely available online.

4.2 Attacks

In this section, we present the analytical details of our membership inference attacks. We start with the methylation-based attack upon which we will build the genome-based membership inference attack.

4.2.1 Methylation-based Attack

Assuming the adversary has access to summary statistics of the pool, we analyze whether it is possible to infer whether the target is part of it by relying on statistical or machine-learning methods.

4.2.1.1 Difference of L_1 Distances

Homer et al. [62] have shown for genomic statistics that one can rely on the L_1 distance to infer membership in databases based on mean values only. We first evaluate how this method performs when applied to DNA methylation. The attack compares, for a methylation position j, the differences between the target's methylation value x^j and the mean statistics of the pool and reference group, and it determines which mean statistics is closest to x^j. Defining the mean values as μ_p^j for the pool and μ_r^j for the reference group, we have the following L_1 distances' difference:

$$D(x^j) = |x^j - \mu_r^j| - |x^j - \mu_p^j| \tag{4.1}$$

for the methylation position j. A value greater than 0 indicates that x^j is more likely to belong to the pool, while a value smaller than 0 indicates x^j is more likely to belong to the reference group. Intuitively, the L_1 test exploits the fact that the target's methylation value x^j influences the mean of its group. Therefore, the target's value x^j is expected to be closer to the mean value of the target's group than to the mean value of the other group.

Finally, we rely on the one-sided Student's t-test on the outcome of $D(x^j)$ for all methylation points j to test whether the target is part of the pool or reference group.

4.2.1.2 Log-Likelihood Ratio (LLR) Test

Additionally, we exploit the likelihood-ratio (LR) test, which has the notable advantage of reaching the maximum achievable power (true-positive rate) for a given false-positive

level. This is explained theoretically by the Neyman-Pearson lemma, and its higher power compared to the L_1 test has been demonstrated empirically with genomic data [125].

However, the LR test poses assumptions on the data distribution. We rely on the normal distribution to model the distribution of methylation values, which is the continuous probability distribution that best fits the observed methylation data.[1] We evaluate in the next section whether this model is good enough to keep LR test's power high with actual methylation data.
The general formula for the LR test at position j is:

$$LR_j(x^j) = \frac{\sigma_r^j}{\sigma_p^j} e^{\frac{(x^j-\mu_r^j)^2}{2(\sigma_r^j)^2} - \frac{(x^j-\mu_p^j)^2}{2(\sigma_p^j)^2}} \tag{4.2}$$

where σ_r^j is the standard deviation of the reference group and σ_p^j the standard deviation of the pool at methylation position j. By taking the logarithm and summing over the m known methylation positions, we get the following log-likelihood ratio (LLR) formula:

$$LLR(\overrightarrow{x}) = \sum_{j=1}^{m} \frac{(x^j-\mu_r^j)^2}{2(\sigma_r^j)^2} - \frac{(x^j-\mu_p^j)^2}{2(\sigma_p^j)^2} + \log\frac{\sigma_r^j}{\sigma_p^j} \tag{4.3}$$

In this work, we assume the adversary gets access to the mean values of the pool but not to its standard deviations. A reasonable approximation of the standard deviation can be computed from the reference population under the assumption that the standard deviation is approximately the same for the pool.

Hence, we have $\sigma_p^j \approx \sigma_r^j := \sigma^j$, and the above expression simplifies to:

$$LLR(\overrightarrow{x}) = \sum_{j=1}^{m} \frac{(x^j-\mu_r^j)^2 - (x^j-\mu_p^j)^2}{2(\sigma^j)^2} \tag{4.4}$$

Note that, following an assumption made in previous works on membership privacy [62, 125, 9], we do not consider dependencies that may exist between different methylation points.

4.2.1.3 Machine-Learning Approach

The two previous statistical tests assume implicitly that the distance between mean and methylation value is equally informative for membership inference no matter the methylation position j. This assumption may not be true: There might be methylation positions that are sensitive to environmental or genetic variants, leading to a higher variance and thereby easier membership detection in the data set.

To model a realistic attacker, we assume her to use the data itself to detect informative methylation regions and increase the success probability. We expect that an exceptionally

[1] We tested for equality to the normal distribution using the Kolmogorov-Shmirnov test and a p-value of 0.1 and observed $\frac{1}{3}$ to $\frac{2}{3}$ of the methylation regions being normally distributed, the value varying between data sets. We also tested other distributions, such as the beta distribution, but did not find anything fitting the methylation data better.

high or low distance of the target to the pool means is more informative for membership inference. Similar to the statistical approaches, we rely on the L_1 and L_2 distances, both to pool and reference means. A division by the standard deviation additionally takes the data variability of the position into account, which simplifies comparison across multiple positions.

All the aforementioned metrics have to be explored systematically, which we do by using machine learning. We fit a logistic regression classifier[2] that learns how to weight features obtained from different methylation regions. We explore the metrics using the following types of features:

1. L_1 distance to pool mean, formally: $|x^j - \mu_p^j|$ (referred to as L_1 distance feature)
2. Squared L_2 distance to pool mean, formally: $(x^j - \mu_p^j)^2$ (referred to as L_2 distance feature)
3. L_1 distance divided by the standard deviation, formally: $\frac{|x^j - \mu_p^j|}{\sigma^j}$ (referred to as scaled L_1 feature)
4. Squared L_2 distance divided by the variance, formally: $\frac{(x^j - \mu_p^j)^2}{(\sigma^j)^2}$ (referred to as scaled L_2 feature)
5. L_1 distance as used in the L_1 test, formally: $|x^j - \mu_r^j| - |x^j - \mu_p^j|$ (referred to as L_1 feature)
6. Log-likelihood ratio as used in the LLR test, formally: $\frac{(x^j - \mu_r^j)^2 - (x^j - \mu_p^j)^2}{2(\sigma^j)^2}$ (referred to as LLR feature)

To compute these features, we first obtain pool and reference means and approximate standard deviations as before for the LLR test. For each training value tr_j from a training patient, we compute the feature with the mean and standard deviation of the respective methylation position j. Features from different positions are combined into a feature vector. We then sort the features of each vector by increasing order of magnitude. This breaks the link between the learned weight and the methylation position j from which tr_j originated, but recall that our training objective is not which position j is more informative, but rather which distance is more informative for membership inference.
Subsampling: To increase the number of samples for learning while keeping the total amount of patients' data the attacker needs to know low, we generate more than one feature vector from each patient by randomly sampling s disjoint subsets of l methylation positions each. These multiple feature vectors are treated separately during training, but at test time they are combined with majority voting to eventually classify each patient into a single group. Details on the number of feature vectors per patient and length of the feature vectors are empirically evaluated in Section 4.4.

We also apply subsampling to the L_1 and LLR tests to compare these directly with the ML approach, i.e, to tell apart the effect of the different settings for machine learning and the benefit of machine learning itself.

[2]We opted for logistic regression due to its simplicity and the interpretability of the learned model.

4.2.2 Genome-based Attack

In the following, we assume the attacker does not know the target's methylation values, but the target's genome instead, while the pool still contains methylation data only. Genomic data is currently more available and easier to find online or via direct-to-consumer genetic testing services. The adversary can rely on correlations between the genome and methylation in specific regions. After inferring the methylation values, the attacker can mount the same attack as previously described, i.e., against a pool of methylation data. As some SNPs influence the methylation in specific regions, the adversary can rely on them to carry out her membership inference attack without having direct access to the target's methylation data. In the experimental evaluation, we will investigate if and to which extent the performance drops when genomic data is used instead of methylation data.

We still assume the attacker knows the mean methylation values of pool and reference group and estimates of the standard deviation from the reference group. Additionally, we assume the attacker has a set of paired methylation and genome data to identify the pairs of correlated methylation and genomic positions and to learn the conditional distribution of methylation values given the genomic values. This section shows how to extend our statistical tests and how to implement the necessary estimates to handle this attack scenario.

As demonstrated by Backes et al. [7], the conditional distribution of a methylation value x^j given a specific SNP g^i can be modeled with a normal distribution. Dropping the position index i of the SNP for simplicity, we define the probability distribution over the methylation values for a specific SNP value $g \in \{0, 1, 2\}$ as

$$f_g(x^j) = p(X_j = x^j \mid G = g) = \frac{1}{\sqrt{2\pi}\sigma_{j,g}} e^{-\frac{(x^j - \mu_{j,g})^2}{(\sigma_{j,g})^2}} \tag{4.5}$$

where $\mu_{j,g}$ and $\sigma_{j,g}$ denote the mean and standard deviation of $f_g(x^j)$, respectively.

Given this probability distribution, the following theorem shows that the expected log-likelihood ratio test for an individual carrying a given genotype boils down to using $\mu_{j,g}$ in place of the target's methylation value.

Theorem 1. *Assuming $\sigma_p^j \approx \sigma_r^j := \sigma^j$ for all methylation positions correlated with the genome, the LLR test based on the individual's genome is:*

$$LLR(g) = \sum_{j=1}^{m_c} \frac{(\mu_{j,g} - \mu_r^j)^2 - (\mu_{j,g} - \mu_p^j)^2}{2(\sigma^j)^2}, \tag{4.6}$$

where m_c represents the number of methylation positions correlated with the genome.

Proof. We derive hereafter the formula for the general case with different σ_p^j and σ_r^j. For a given methylation point j, we need to integrate x^j over all its possible values given g:

$$\begin{aligned} LLR^j(g) = &\frac{1}{2(\sigma_r^j)^2} \int_0^1 (x^j - \mu_r^j)^2 f_g(x^j) dx^j \\ &- \frac{1}{2(\sigma_p^j)^2} \int_0^1 (x^j - \mu_p^j)^2 f_g(x^j) dx^j + \log \frac{\sigma_r^j}{\sigma_p^j} \int_0^1 f_g(x^j) dx^j \end{aligned}$$

By setting $\Delta_j^c = \mu_{j,g} - \mu_p^j$ and $\Delta_j^r = \mu_{j,g} - \mu_r^j$:

$$
\begin{aligned}
LLR^j(g) &= \frac{1}{2(\sigma_r^j)^2}\int_0^1 (x^j - \mu_{j,g} + \Delta_j^r)^2 f_g(x^j)dx^j \\
&- \frac{1}{2(\sigma_p^j)^2}\int_0^1 (x^j - \mu_{j,g} + \Delta_j^c)^2 f_g(x^j)dx^j + \log\frac{\sigma_r^j}{\sigma_p^j} \\
&= \frac{1}{2(\sigma_r^j)^2}\int_0^1 (x^j - \mu_{j,g})^2 f_g(x^j)dx^j + \frac{\Delta_j^r}{(\sigma_r^j)^2}\int_0^1 (x^j - \mu_{j,g}) f_g(x^j)dx^j \\
&+ \frac{(\Delta_j^r)^2}{2(\sigma_r^j)^2}\int_0^1 f_g(x^j)dx^j - \frac{1}{2(\sigma_p^j)^2}\int_0^1 (x^j - \mu_{j,g})^2 f_g(x^j)dx^j \\
&- \frac{\Delta_j^c}{(\sigma_p^j)^2}\int_0^1 (x^j - \mu_{j,g}) f_g(x^j)dx^j - \frac{(\Delta_j^c)^2}{2(\sigma_p^j)^2}\int_0^1 f_g(x^j)dx^j + \log\frac{\sigma_r^j}{\sigma_p^j}
\end{aligned}
$$

By using the central moments of the normal distribution, we eventually get:

$$
LLR^j(g) = \frac{\sigma_{j,g}^2}{2(\sigma_r^j)^2} + \frac{(\Delta_j^r)^2}{2(\sigma_r^j)^2} - \frac{\sigma_{j,g}^2}{2(\sigma_p^j)^2} - \frac{(\Delta_j^c)^2}{2(\sigma_p^j)^2} + \log\frac{\sigma_r^j}{\sigma_p^j}
$$

If $\sigma_p^j \approx \sigma_r^j := \sigma^j$, the above formula simplifies to

$$
\begin{aligned}
LLR^j(g) &= \frac{\sigma_{j,g}^2}{2(\sigma_r^j)^2} + \frac{(\Delta_j^r)^2}{2(\sigma_r^j)^2} - \frac{\sigma_{j,g}^2}{2(\sigma_p^j)^2} - \frac{(\Delta_j^c)^2}{2(\sigma_p^j)^2} \\
&= \frac{(\Delta_j^r)^2 - (\Delta_j^c)^2}{2(\sigma^j)^2} = \frac{(\mu_{j,g} - \mu_r^j)^2 - (\mu_{j,g} - \mu_p^j)^2}{2(\sigma^j)^2}
\end{aligned}
$$

We obtain the final formula by summing over all methylation points m_c correlated with the genome. □

Similarly, for the L_1 test, we use the expected methylation value $\mu_{j,g}$ given the genotype g as the target's methylation value.

4.3 Datasets

For our evaluation, we rely on six data sets containing methylation profiles from diverse tissues of patients carrying different diseases. In total, we use the methylation profiles of 1,320 patients. Table 4.2 summarizes our data sets.

All but the last data set were generated with the Illumina 450k array that determines the DNA methylation at 450,000 fixed positions. We refer to these data sets by the disease the respective patients carry. Our last data set, the WGBS data set, contains both the genome and the methylation of 75 patients, where the DNA methylation profiles have been generated by whole-genome bisulfite sequencing (WGBS). This results in a full view of DNA methylation patterns in the whole genome of blood cells.

Preprocessing Most of the data sets have missing methylation sites (positions) for specific patients or even missing methylation sites for all the patients sharing the same

Abbreviation	Description	Tissue Type	Number of Patients	GSE identifier	by
GBM	glioblastoma	brain cancer	136	GSE36278	[138]
PA	pilocytic astrocytoma	brain cancer	61	GSE44684	[85]
IBD CD	Crohn's disease	blood	77	GSE87640	[148]
IBD UC	ulcerative colitits	blood	79	GSE87640	[148]
BC	breast cancer	breast cancer	892	not public	-
WGBS	genome and methylation data	blood	75	not public	-

Table 4.2: Datasets used in our experiments. THE GSE identifier refers to the accession number in the Gene Expression Omnibus (GEO) database. The BC data set was available on `https://portal.gdc.cancer.gov` in April 2017 but is not anymore.

disease. We remove all methylation positions with missing data, which provides us with 299,998 different methylation positions for the combination of brain cancers and IBD, and about 360,000 different methylation positions for the breast cancer data set.

For our WGBS data set, we focus on highly correlated pairs of DNA methylation positions and SNPs. We follow the approach of Backes et al. [7] and only keep the pairs with a Spearman rank correlation coefficient larger than 0.49. This way, we obtain about 300 methylation positions and the single most correlated SNP position each.

Human Subjects and Ethical Considerations The study on WGBS has received an approval from the responsible institutional ethics review board. All other data sets were publicly available in their anonymized form. All data sets have been stored and analyzed in anonymized form without access to non-anonymized data. Moreover, since we only randomly split the patients into pool and reference sets, the membership inference attacks do not reveal any more information than previously known by us. This way, we ensure that all participants were treated equally and with respect.

4.4 Attack Evaluation

We started by evaluating the statistical and machine-learning methylation-based attacks. Then, we present the results of our genome-based attack.

4.4.1 Methylation-based Attack Evaluation

Our evaluation studies the following research questions:

RQ 1 Does the LLR test outperform the L_1 test in the statistics attack setting?

RQ 2 What is the effect of our subsampling approach on the performance of the L_1 and LLR tests?

RQ 3 Which feature is best in the ML attack? Does the performance increase compared to the L_1 and LLR tests with subsampling?

RQ 4 Is it possible to train an attack model on a data set of a different tissue or disease than the target data set for the machine learning attack?

RQ 5 What is the influence of the data set size on the performance of the membership inference attacks?

While RQ1 studies the statistical approach and verifies that the Neyman-Pearson Lemma applies to our data, RQ2 and RQ3 study the foundations of the ML approach. With RQ4 and RQ5 we explore how the ML case works in non-ideal situations, namely, different training and test data and larger dataset sizes.

RQ 1: Comparing the statistical L_1 and LLR test, does the LLR test outperform the L_1 test? To apply the L_1 and LLR tests, we first define pool and reference group. We present a realistic attacker that cannot exploit any disease-specific differences between the databases. For each of our five first data sets, we first randomly sampled 60 patients,[3] which are then randomly split into a pool of 30 patients and a reference group of 30 patients. We assume the attacker has means of these 30 patients available as μ_p^j and μ_r^j respectively. Further, we sampled 15 patients from the pool and 15 from the reference group at random. The remaining 30 patients were not used in this setup, they serve as training set of the machine learning attack later in this section. We repeated the random splitting five times and present averaged results.

As discussed previously, we assume the attacker has access to the mean of the pool (μ_p^j) and reference group (μ_r^j) for each methylation position j. Moreover, we estimate σ^j by computing the standard deviation over the whole considered data set.

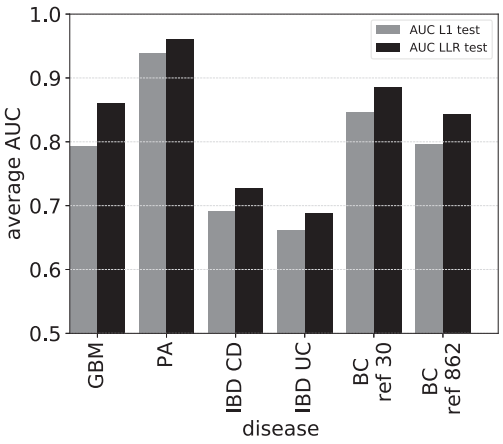


Figure 4.2: RQ1 (statistical attack setting): AUC of the L_1 and LLR tests applied to all methylation positions, averaged over five random splits of the data simulating attacks against each patient in both pool and reference groups.

We simulate membership inference attacks against each patient individually, i.e., all patients from the respective pool and reference group are attacked by applying the L_1 and LLR tests to each methylation position j and summarize across all methylation positions for the given patient as defined by the tests. Using multiple thresholds in the tests, we get a receiver operating characteristic (ROC) curve displaying the false-positive rate ($\frac{FP}{FP+TN}$) on the x-axis and the true-positive rate ($\frac{TP}{TP+FN}$) on the y-axis. The AUC is the area under this curve. An AUC of 0.5 indicates a performance similar to a random guess, whereas an AUC of 0.9 or above indicates an excellent performance. Finally, we

[3]Note that this is the maximum number we can consider if we want to compare the results across all data sets as PA contains 61 patients.

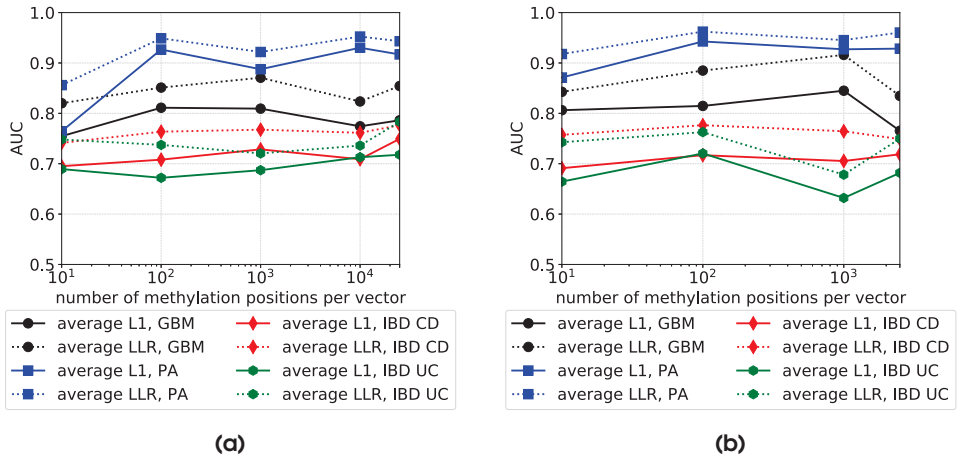


Figure 4.3: RQ2 (subsampling): Influence of the length l of the feature vectors on L_1 and LLR tests performance for four disease datasets when using (a) $s = 10$ vectors or (b) $s = 100$ vectors.

average the results over the five random splits. The results are shown in the first five groups of bars in Figure 4.2.

We observe that the LLR test outperformed the L_1 test, complying with the Neyman-Pearson Lemma. The performance reached > 0.7 AUC for all diseases when the LLR test was used, and even > 0.95 AUC for PA. Interestingly, the tissue type seems to have an influence on the attack performance, both IBD data sets were sampled from blood and were harder to attack compared to samples from brain cancer tissue for the diseases GBM and PA or breast cancer tissue for BC.

Finally, in order to evaluate a more realistic setting where the reference population is very large, we used our largest data set on breast cancer (BC) patients. Instead of sampling 30 patients as the reference group, we used all remaining patients, i.e., 862 patients to compute μ_r^j. We observed that the AUC drops by only 0.1 compared to the case with a much smaller reference group and conclude that the privacy risks remain valid with a very large reference group.

Take-home message: the LLR test outperformed the L_1 test with DNA methylation data.

RQ 2: What is the effect of subsampling on the performance on L_1 and LLR test? Which values for the hyperparameters s and l are the best? We subsampled each data vector before computing the L_1 test and the LLR test. For each patient, we randomly sampled l methylation positions s times without replacement for various settings for s and l. At the end, we combined the inference labels of the s vectors of the same patient with majority voting to get a single outcome for each patient. As before, we first randomly sampled 60 patients for each disease, which were then randomly split into 15 pool and 15 reference patients. Again, the remaining 30 patients were not used.

Figure 4.3a shows the performance of the L_1 (solid lines) and LLR tests (dotted lines) for four of our disease sets, with 10 repetitions of the sampling process. Observing

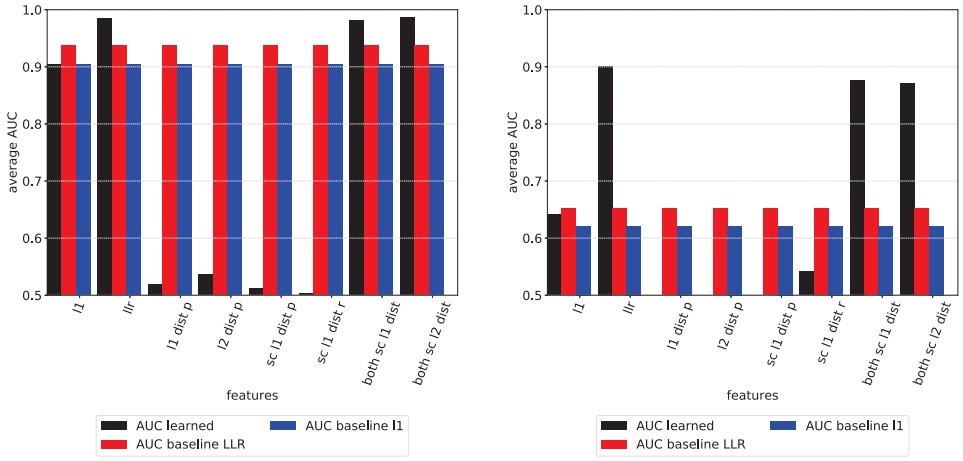


Figure 4.5: RQ3 (features): Performance of different features evaluated on disease datasets (a) PA and (b) IBD UC.

no general trend of l increasing from 10^3 to 10^4, we dropped $l = 10^4$ from the parameters which allowed increasing s to 100, see Figure 4.3b.

Comparing the AUC with the standard setup before (Figure 4.2) shows that for most diseases, the performance of the L_1 test was similar or increased slightly, and the performance of the LLR test increased slightly in almost all cases, independent of how the parameters s and l are set. For example, in GBM the traditional LLR test performance was below 0.85 AUC and with s=100 and l = 10^3 raised to more than 0.9 AUC. The difference between the previous and the current setup is how the membership information from different methylation positions j were combined. Simply taking all of them into account performed worse than first combining a few of them into a binary answer and then taking the majority vote. In the latter case, some methylation positions will therefore not contribute to the answer. This experiment shows that not all methylation positions are informative.

Finally, we observed that a reasonable trade-off between l and s satisfying the constraint $l \cdot s \leq m$ was bounding l to 10^3 and setting $s = 100$.

Take-home message: Subsampling slightly increased performance, and the hyper-parameters $l = 10^7$ and $s = 100$ represented a reasonable trade-off.

RQ 3: Which feature is best in the machine learning model, and can the performance be increased compared to the L_1 and LLR test with subsampling? For the machine-learning attack, we used the subsampling trade-off as found before and set $l = 10^3$ and $s = 100$. The remaining 15 pool and 15 reference patients were used as training set. After transforming each value in the training vectors into a feature using the formulas in Section 4.2.1.3, we sorted the vectors' values in ascending order. Then, the vectors were fed into a logistic regression classifier: We relied on the Python library *sklearn* [111] and left the regularization parameter C at its default 1.0. The classifier learned l coefficients that indicate importance of small, intermediate and large distances (as most of our features were distance-based).

Figure 4.4 shows the absolute value of the learned coefficients for IBD UC, plots for other diseases look similar. The higher the absolute value of the coefficient, the more informative the distance is for the classifier. The symmetric pattern arised due to the use of two features: the scaled L_1 distances to both pool and reference means. There was a tendency towards higher values on the right, indicating that higher distance values were more important for the attack. Nevertheless, the lower values do not get zero coefficients, which suggests they also contribute to the model. Additionally, we applied sklearn's recursive feature elimination [56], but the resulting classifiers performed worse in terms of AUC, supporting again the hypothesis that all distances are necessary.

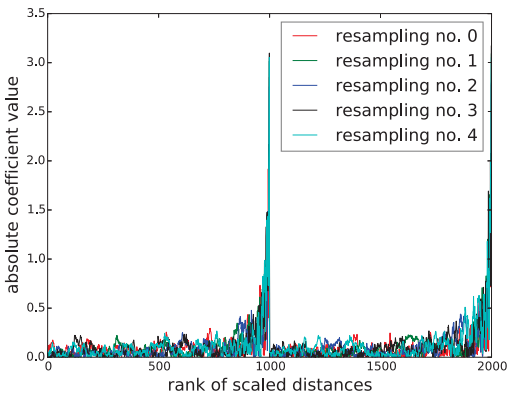


Figure 4.4: RQ3 (features): Absolute value of learned coefficients on IBD UC with both scaled L_1 features. We show the five repetitions (with different data sampling) of the experiment in different colors to check whether the coefficient values are consistent and not due to randomness. X-axis between 0 and 999 represent the coefficient values for scaled L_1 to the *pool* mean and x-axis between 1,000 and 1,999 represent the coefficient values for scaled L_1 to the *reference* mean.

We compared the performance of different features using the AUC of the learned model when applied to the test data. Figure 4.5 shows exemplarily the performances for PA and IBD UC, the "easiest" and one of the "hardest" disease data sets to attack. We tested all the feature types introduced in Section 4.2.1.3. The distance-based features exist in two versions as distance to the *pool* and to the *reference* mean, respectively, indicated by "p" and "r" in the plot. We omitted the "r" version for some features which performed similarly to their "p" versions. Additionally, we trained on *both* versions of the distance features by concatenating the respective feature vectors. As a baseline, we relied on the L_1 and LLR tests with subsampling. We observe that some features worked well, e.g., the LLR feature and using the L_1 or L_2 to both pool and reference group in their scaled form. Other features performed poorly and resulted in an AUC of around 0.5, e.g., the L_1 and L_2 both with and without scaling. This is why for those features, the black bar is barely visible in Figure 4.5. Nevertheless, the statistical tests L_1 and LLR were clearly outperformed, especially for the IBD UC and IBD CD data sets.

Take-home message: the performance was increased by using a machine-learning approach with the LLR features or L_1 or L_2 to both pool and reference groups in their scaled form.

RQ 4: Is it possible to train an attack model on a data set of a different tissue or disease than the target data set for the machine learning attack? We study now another, more challenging attack scenario where the attacker trains her machine learning model to pool and reference groups extracted from one data set and applies this model to pool and reference groups of another data set. We kept the

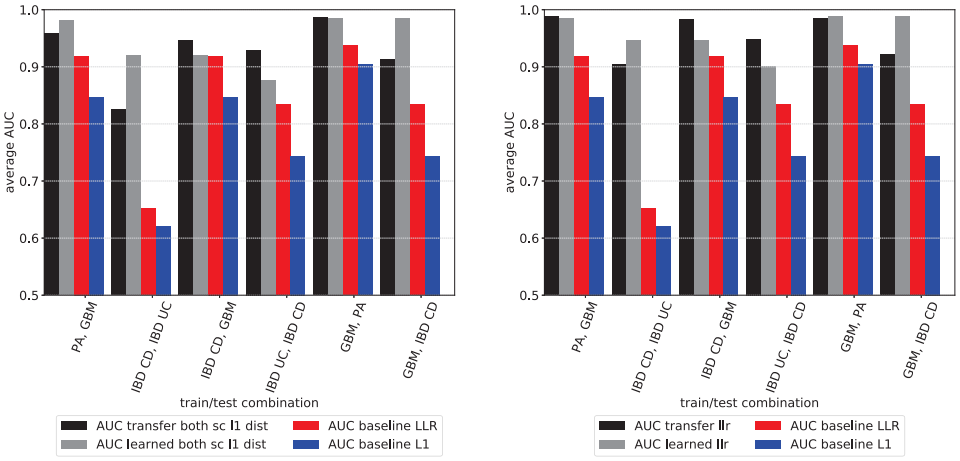


Figure 4.6: RQ4 (transferability): Transferability of learned models based on (a) both scaled L_1 and (b) LLR features.

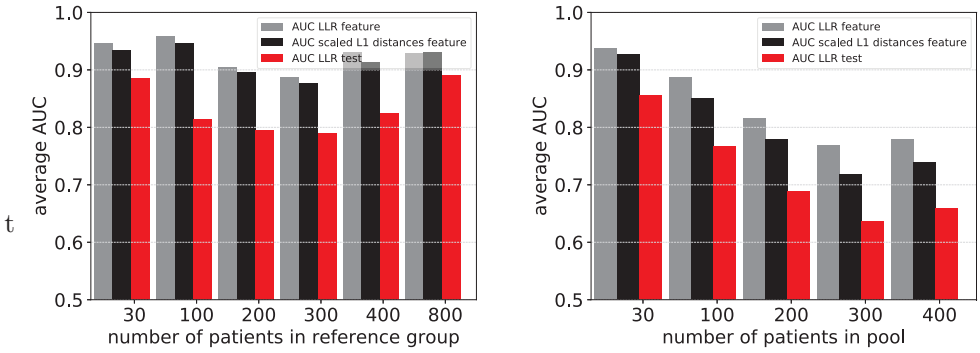
previous experimental setup, but tested the learned model on data set with different tissue or disease. This setup allows us to evaluate if our membership inference attack is prone to data transferability.

Figure 4.6 displays the resulting AUCs when learning on the first mentioned (in the x-axis labels) data set and testing on the second one. Since the performance of the scaled L_1 and scaled L_2 distances were similar (see Figure 4.5), we show in Figure 4.6 only the scaled L_1 feature and the LLR feature. Comparing the black and gray bars, we can observe that most cases show a small loss of performance when the attacker learned on patients from a different disease or tissue compared to learning from the same one. However, the learned models still performed well on the different disease set and clearly outperformed the statistical L_1 and LLR tests. Recall that the data sets GBM and PA were sampled from brain tumors, while both IBD data sets were from blood samples. According to biomedical research, part of the methylation patterns are tissue specific. However, our results show that our attack based on relative distance instead of methylation positions was prone to transferability even across different tissues.[4]

Take-home message: training and target data sets did not need to be the same for a successful attack.

RQ 5: What is the influence of a larger data set on the performance of the machine learning model? Our larger data set on breast cancer allows to study the impact of larger reference group and pool on the attack performance. First, we focused on the reference group size, increasing it from 30 to 800 patients, and kept the number of patients in the pool at 30. This allows us to evaluate whether a more realistic (i.e., larger) reference population has an impact on the attack performance. We evaluated the impact of increasing reference group size on machine learning classifiers trained on the LLR feature and both scaled L_1 features, and on the statistical LLR

[4]Note that it is very unlikely that these results are due to the same patients being in the data sets because we obtained data from different studies and different diseases.



t

(a) RQ5 (larger datasets): Performance with respect to an increasing number of patients in the reference group only.

(b) RQ5 (larger datasets): Performance with respect to an increasing number of patients in the pool and reference groups.

Figure 4.7: RQ5(larger datasets): Performance under increasing number of patients in two different configurations.

test using 30 patients for training and testing respectively. We observe in Figure 4.7a that both statistical and ML-based tests performed similarly under increased reference group sizes and that reference group size did not have any clear influence on the attack performance. This demonstrates that privacy risks remain true with a large reference population, and allows us to extrapolate that membership inference would be possible in non-closed-world settings.

Second, we increased the data set size from 30 patients in both pool and reference group to 100, 200, 300, and 400 patients. In all cases, we used disjoint training and test sets of the same size which contain the same number of pool and reference patients.

Figure 4.7b shows that the more patients there were in the pool, the worse the performance of the membership inference attack. As we see in Figure 4.7a, reference group size did not influence the attack success. This confirms previous empirical results with genomic [125] and transcriptomic [9] data, as well as theoretical findings [39]. We further observe that the attack success decreased similarly for both the statistical attack and the ML attacks. We hypothesize that the performance decrease was due to the fact that the more patients were included in the pool, the less each patient contributes to its statistics, in our case, the means, which made membership inference harder.

On the upside, we can foresee that with declining costs of molecular profiling, the size of epigenomic databases will rapidly grow. Nevertheless, we notice that the ML attack is quite robust to this increase, with still relatively good performance (AUC > 0.8) with 200 patients in the pool.

Take-home message: The attack performance was especially robust with respect to an increase in only the reference data set size. However, when increasing both pool and reference groups, the attack performance decreased. We conclude that the privacy threat remained even with larger reference population, but also with pool sizes up to 200 individuals.

4.4.2 Genome-based Attack Evaluation

Next, we evaluate the scenario in which the attacker has access to the target's genomic data instead of methylation data. We used the WGBS data set, containing methylation and genome data of 75 patients. Notice that the data was generated with a different technique (WGBS), which targets different regions of the genome than the Illumina 450k array used for the previous data sets.

We randomly sampled half of the patients as a training set (37 patients) which we used to estimate the relationship between genome and methylation data. The second half (38 patients) was used as a test: Half of the patients were chosen at random to be in the pool and the remaining half were in the reference group. The standard deviation was estimated from the training set, which we assumed the attacker has full access to. Notice that we had only $m = 300$ methylation positions correlated with the genome, which is tremendously less than $m = 299,998$ used for the previous attacks. Therefore, we did not subsample from the patients. We repeated the experiment five times with different random splits into training and test sets. Moreover, for each of these five splits, we also repeated the splitting into pool and reference group five times, effectively yielding 25 randomly generated runs. As a baseline, we computed the L_1 and LLR tests under the previous assumption that the attacker knows the target's methylation values.

Figure 4.8 compares the performance of the attacks based on methylation and genomic data. The LLR test exploited the underlying normal distribution of methylation values given a specific genome value. The same technique was used for the L_1 test. We observe first that both L_1 and LLR tests performed worse with access to genome instead of methylation values, as expected. However, they still achieve high performance, which shows that an attacker with only access to the genome of the target was still able to successfully infer her membership in methylation databases.

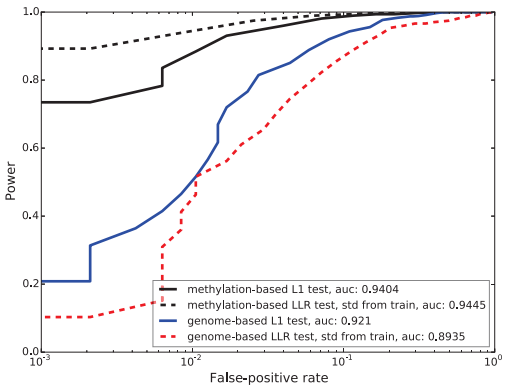


Figure 4.8: ROC curves of methylation-based L_1 and LLR tests and genome-based L_1 and LLR tests.

Surprisingly, the performance decrease was higher in the case of the LLR test, where AUC dropped from 0.94 to 0.89. For the L_1 test, the AUC decreased from 0.94 to 0.92 when we relied on the genomic data instead of the methylation data. One possible explanation for the LLR performance being lower than the L_1 performance in the genome attack is as follows. The estimation of the methylation values for the target given the genome was approximated and induced small errors. Such noisy values had a larger negative influence on the LLR test compared to the L_1 test.

We also applied our ML techniques to the WGBS set, but learning was not possible due to the relatively low number of methylation values. Nevertheless, we conclude that, despite the small drop in overall performance, membership inference was still possible with genomic data that is currently easier to obtain than methylation data.

Take-home message: We conclude that privacy is at risk even if the attacker has not access to the target's methylation data and must estimate them from their genome.

4.5 Conclusion and Future Work

In this chapter, we have thoroughly analyzed whether and to what extent DNA methylation databases are prone to membership inference attacks. In particular, we have considered two attacker models: one assuming the adversary to know her victim's methylation profile, and the second assuming the adversary to know only her victim's genotype. For both settings, we have studied traditional statistical attacks based on the L_1 distance and on the likelihood-ratio test. Additionally, we have proposed a new machine-learning attack that is able to exploit the fact that not all methylation data are equally informative for membership inference. In this setting, we have further studied data transferability, i.e., to which extent learning features from a data set different from the targeted data set influences the attack results. For the genome-based inference of membership, we have specifically designed the LLR attack to capture the probabilistic dependencies between the two types of data, and have identified a sufficient statistic for this attack.

We have evaluated our attacks on six different data sets, overall containing the DNA methylation profiles of 1,320 patients. Our empirical results consistently demonstrate the success of membership inference attacks over different tissues and diseases. Even though we were limited by the small number of patients in most of the data sets, the experiments with the larger breast cancer data set suggested that our findings may scale. We concluded that the membership privacy of contributors to DNA methylation databases is put at risk even if the adversary does not directly get access to their methylation data but only their genomes.

Performing the membership inference attacks with DNA methylation data at different points in time is a future direction that is worth investigating. Moreover, designing attacks that exploit dependencies between methylation points is another interesting direction for future work.

Given the severe privacy risks that we uncover with our attacks, future work should study protection mechanisms. One direction would be to perturb the means with randomly sampled noise to achieve differential privacy [36]. The challenge will be to tailor a mechanism to this specific case with high data dimensionality and few individuals (currently) contributing their data. In line with other applications such as MBeacon [P3], we believe that there is a clear benefit from sharing population-wide mean methylation values. Mean methylation values could become as relevant and well-studied as minor allele frequencies are today for the genome.

4.6 Defense

The evaluation section demonstrated various attacks threatening patients' privacy. Therefore, we study in the following whether researchers can share summary data in a privacy-preserving way using differential privacy mechanisms [38].

We first introduce the differential privacy definitions and how to extend them to finite ranges, and then present our utility measure.

4.6.1 Differential Privacy Mechanisms

The standard method for achieving ϵ-differential privacy or (ϵ, δ)-differential privacy consists of replacing the original statistical values $f_{\text{mean}} : \mathcal{X} \to \mathbb{R}^m_{[0,1]}$ by the sanitized statistics:

$$\tilde{f}_{\text{mean}} = f_{\text{mean}} + (N_1, \cdots, N_m) \tag{4.7}$$

which adds a given amount of noise N_j to the mean value of each released methylation point j. We test two different probability distributions to sample appropriate noise values N_j: the Laplace and Gaussian distributions.

4.6.1.1 Laplace Noise

Our first sanitization mechanism achieves ϵ-differential privacy by drawing each N_j from a Laplace distribution with scale $\frac{\Delta_1 f_{\text{mean}}}{\epsilon}$, where $\Delta_1 f_{\text{mean}}$ represents the global L_1 sensitivity of the mean function. It is based on the L_1 distance and is known to be equal to $\sum_{i=j}^{m} \frac{r_j}{n}$ [9], where r_j represents the range of values at methylation point j. In our case, $r_j = 1\ \forall j$. Notice that this is the worst-case technical range and not dependent on the actual methylation data. Therefore, $\forall j \in \{1, \ldots, m\}$, we draw N_j as

$$N_j \sim Lap\left(\frac{\frac{m}{n}}{\epsilon}\right) \tag{4.8}$$

4.6.1.2 Gaussian Noise

Our second sanitization mechanism achieves (ϵ, δ)-differential privacy and samples noise variables from the normal distribution with mean zero and standard deviation scaled to the privacy parameters and sensitivity. Formally, we set $\sigma = \sqrt{\frac{2ln(\frac{2}{\delta})}{\epsilon}} \cdot \Delta_2 f_{\text{mean}}$ where Δ_2 is the sensitivity of f_{mean} with respect to the L_2 norm [36]. In the following, we derive the Δ_2 sensitivity. Let $\mathcal{T}, \mathcal{T}'$ denote two databases differing in one element. Then the sensitivity is defined as:

$$\begin{aligned}
\Delta_2 f_{\text{mean}} &= \max_{\mathcal{T},\mathcal{T}'} ||f_{\text{mean}}(\mathcal{T}) - f_{\text{mean}}(\mathcal{T}')||_2 \\
&= \max_{\mathcal{T},\mathcal{T}'} ||(\mu_1^{\mathcal{T}}, ..., \mu_m^{\mathcal{T}}) - (\mu_1^{\mathcal{T}'}, ..., \mu_m^{\mathcal{T}'})||_2 \\
&= \max_{\mathcal{T},\mathcal{T}'} \sqrt{\sum_{j=1}^{m} (\mu_j^{\mathcal{T}} - \mu_j^{\mathcal{T}'})^2} \\
&= \max_{\mathcal{T},\mathcal{T}'} \sqrt{\sum_{j=1}^{m} (\frac{\sum_{i=1}^{n} x_{i,\mathcal{T}}^j}{n} - \frac{\sum_{i=1}^{n} x_{i,\mathcal{T}'}^j}{n})^2}.
\end{aligned}$$

As the two databases $\mathcal{T}, \mathcal{T}'$ are neighbors, for each j there is exactly one index k such that $x^j_{k,\mathcal{T}} \neq x^j_{k,\mathcal{T}'}$ and for all $i \neq k$ holds $x^j_{i,\mathcal{T}} = x^j_{i,\mathcal{T}'}$. This allows us to re-order the sums:

$$= \max_{\mathcal{T},\mathcal{T}'} \sqrt{\sum_{j=1}^{m} (\frac{1}{n} (\sum_{i=1, i\neq k}^{n} (x^j_{i,\mathcal{T}} - x^j_{i,\mathcal{T}'}) + x^j_{k,\mathcal{T}} - x^j_{k,\mathcal{T}'}))^2}.$$

Since $\forall i \neq k : x^j_{i,\mathcal{T}} = x^j_{i,\mathcal{T}'}$ and $|x^j_{k,\mathcal{T}} - x^j_{k,\mathcal{T}'}| \leq 1$, we get:

$$\Delta_2 f_{\text{mean}} = \sqrt{\sum_{j=1}^{m} (\frac{1}{n}(0+1))^2} = \sqrt{\sum_{j=1}^{m} (\frac{1}{n^2})} = \frac{\sqrt{m}}{n}.$$

4.6.1.3 Finite Ranges

We want to avoid returning values outside the natural range $[0, 1]$ of methylation values. Therefore, we clip the noised values back to the range $[0, 1]$, which does not influence the privacy guarantees due to the post-processing property [38]. Concretely, we define a function $clip(x) = \min(1, \max(0, x))$ that we apply to any value x after noising.

4.6.2 Utility Measure

In order to assess the impact of our differential privacy mechanisms, we rely on the mean relative error (MRE). Applied to our setting, we get the following formula for a set of m released methylation means:

$$\frac{1}{m} \sum_{i=1}^{m} \frac{clip(f_{\text{mean}} + N_i) - f_{\text{mean}}}{f_{\text{mean}}} \tag{4.9}$$

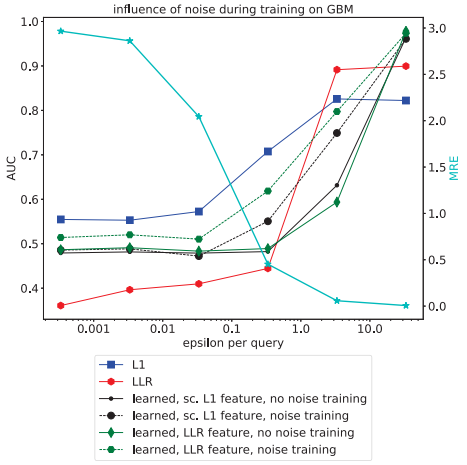
4.7 Defense Evaluation

In this section, we evaluate the effectiveness of our defense mechanisms.

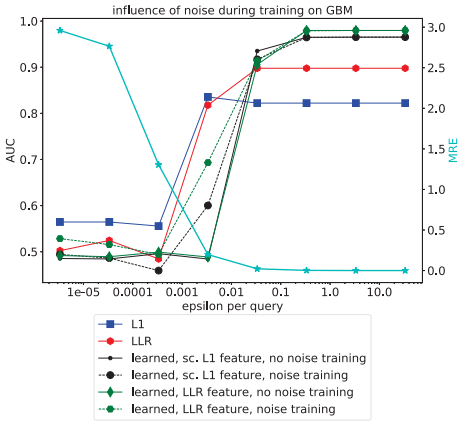
4.7.1 Methylation-based Defense

We kept the random sampling methods as for the machine learning model in the attack evaluation (see Section 4.4), added Laplace resp. Gaussian noise to the means and clipped them back to the interval [0, 1] as described in Section 4.6 for various levels of privacy parameter ϵ. For the Gaussian noise, we set the additional privacy parameter δ to 0.1 throughout all experiments. Moreover, in order to assess whether training the model with a noisy data set had an influence on the results, we trained the machine learning model (i) without noise and (ii) with the same level of noise in training and test data.

Privacy guarantees: We comment on the privacy level ϵ that both influences the noise added to the data and the theoretic privacy guarantees. In the literature, privacy



(a) Influence of Laplace noise.



(b) Influence of Gaussian noise, the privacy parameter δ is 0.1.

Figure 4.9: Influence of two different noise mechanisms on the performance of membership inference attacks against the GBM dataset by relying on training data with and without noise.

levels of $0 < \epsilon \leq 1$ are used with the notice that ϵ should be as close to zero as possible. These values are given for one answer to one query. Answering multiple queries leads to a degraded privacy due to the composition theorem (see Chapter 3.5 in [38]). In order to compare our results with textbook privacy levels, we report the privacy level per query, i.e., $\frac{\epsilon}{m}$.

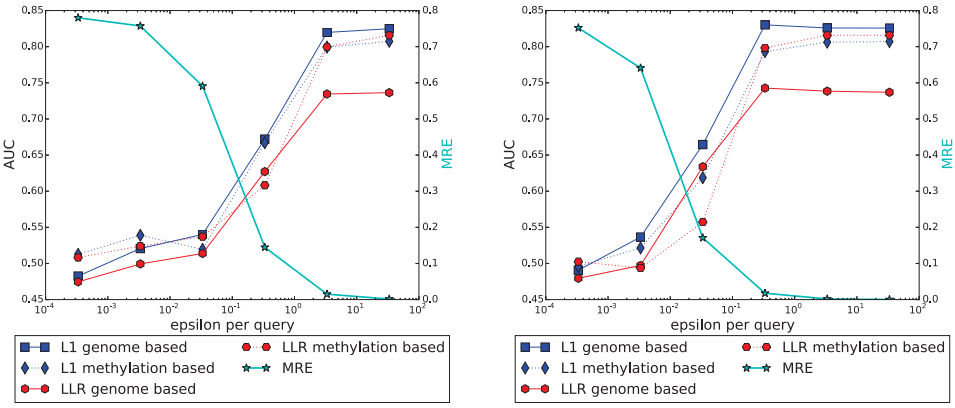
Results: We provide here only plots for the GBM datasets, the trend with other datasets being similar. In Figure 4.9a, we observe that noise dropped the performance quickly. However, in general, the statistical tests were more robust to noise than the learned models, especially the L_1 test. Additionally, we observe that using noisy statistics for training made ML approaches more resistant to noise. Moreover, the LLR feature trained on noise was harder to defend against compared to the scaled L_1 features. Nevertheless, using both the LLR feature and training on noise still leads to an attack that was easier to defend against compared to the statistical L_1 test.

A possible explanation for these observations is that both the LLR and the machine learning models relied on the distance of the target value to the mean. Due to noise, this distance changed and mislead the attacker. On the other hand, the L_1 test only considers which mean the target was closer to, but not how close. Therefore, small changes due to noise rarely mislead an attacker using the L_1 test.

Figure 4.9b shows that using Gaussian instead of Laplace noise did not significantly change the results. The only difference was that we make these observations for a different privacy level ϵ due to the different sensitivity functions.

Utility: The MRE provides information about the difference of the means induced by noising. To reach an acceptable level of privacy, i.e., an attacker AUC close to 0.5, a high MRE of 1 or more is required. That means, the level of noise is equally large or even larger than the mean value itself.

t



(a) Influence of Laplace noise.

(b) Influence of Gaussian noise with δ=0.1.

Figure 4.11: Influence of the two different noise mechanisms on the genome-based membership inference attack, where the attacker has access to genome data of the target instead of methylation data.

Influence of Data Set Size: We use our largest data set on breast cancer to study what happens if more patients were in the pool. We have already seen in the attack evaluation (Section 4.4) that the attacker's performance decreased even if no noise is involved simply due to the smaller contribution each patient had on the mean. However, increasing the number of patients in the pool by a factor of 10 (from 30 to 300) had no influence on the privacy level ϵ needed to drop the performance to 0.5 AUC, as shown in Figure 4.10. Only the MRE shows that a little less noise was added by being smaller than one for reasonable privacy protection at $\frac{\epsilon}{m} = 0.03$. We also observe that, if we aim at an AUC of around 0.6 for all attacks, we found a much better MRE of around 0.1.

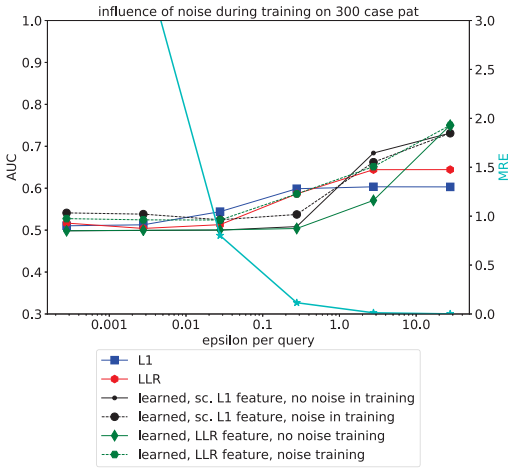


Figure 4.10: Influence of Laplace noise on the performance of membership inference attacks against 300 pool/reference patients (from the breast cancer dataset) instead of 30 patients.

4.7.2 Genome-based Defense

We also added Laplace noise, resp. Gaussian noise, to the mean methylation data of the WGBS data set where we assume the attacker had access to the target's genome data instead of methylation data.

Figure 4.11a shows that the L_1 test was harder to defend against than the LLR

test. Interestingly, the difference in performances was now smaller than in the datasets studied before (dotted lines). Due to the estimation, small changes were likely to be introduced and served as an extra layer of noise, which strengthens the hypothesis that the LLR test's smaller noise stability was due to it taking distances into account. We reach good privacy protection already for an MRE of only 0.6 using Laplace noise. This acceptable level of privacy is reached for $\frac{\epsilon}{m} = 0.03$ for Laplace noise. This was the same noise level per query as before but, since $m = 300$ instead of 300,000, the theoretical privacy protection was much stronger after composition. The evaluation on Gaussian noise can be found in Figure 4.11b and shows similar results.

4.7.3 Discussion

From our experiments with the brain cancer datasets, we conclude that few patients ($n = 30$) and many methylation points ($m = 300,000$) should not be protected with differential privacy since privacy levels degrade too much due to the composition theorem. Increasing the number of patients will not only increase the statistical power of the findings derived from the dataset, it will also decrease the privacy risks. Adding noise in that case seemed to lead to smaller perturbations given the same level of privacy parameter ϵ. The other parameter, the number of released methylation points m, changed when we studied the WGBS data set ($m \approx 300$). We showed that privacy protection with differential privacy provided a successful attack mitigation for smaller levels of ϵ and less perturbation as shown by the MRE.

To summarize, we advise to not publish summary statistics without rigorous risk analysis. In some cases, having many patients or restricting the number of methylation positions published may be possible to successfully mitigate membership inference attacks with differential privacy.

5

MBeacon: Privacy-Preserving Beacons for DNA Methylation Data

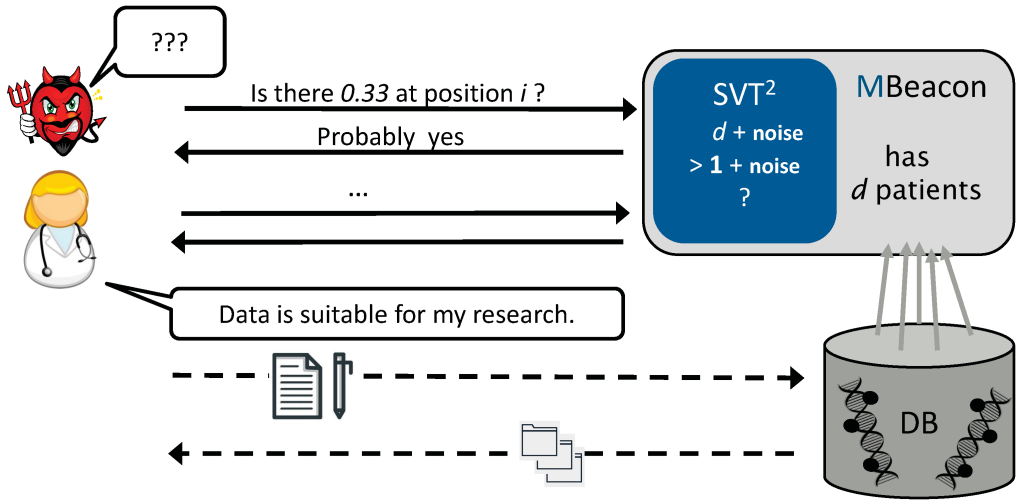


Figure 5.1: Overview of our MBeacon system: While the researcher is enabled to find datasets of interest, the attacker should learn nothing about the patients contributing their data. Our SVT² method outputs a noised answer whether data for the requested value is available in the underlying database.

The advancement of molecular profiling techniques fuels biomedical research with a deluge of data. To facilitate data sharing, the Global Alliance for Genomics and Health (GA4GH)[1] established the Beacon system[2] in 2014, a search engine designed to help researchers find data sets of interest. The Beacon system is essentially a search engine indexed over multiple *Beacons*. Each single Beacon is constructed by a partner institution of the Beacon system with its own database. Only one type of query is supported by a Beacon: whether its database contains any record with the specified nucleotide at a given position and chromosome, and the corresponding response is a binary "Yes" or "No". Upon a query from a researcher, the search engine, i.e., the Beacon system, will return the names of the partner institutions that answer "Yes", and the researcher can directly contact these institutions to obtain access to the data.

While the current Beacon system only supports genomic data, other types of biomedical data, such as DNA methylation, are also essential for advancing our understanding in the field. We propose the first Beacon system for DNA methylation data sharing: MBeacon. As the current genomic Beacon is vulnerable to privacy attacks, such as membership inference [132, 117, 3, 152], and DNA methylation data is highly sensitive, we take a privacy-by-design approach to construct MBeacon.

First, we demonstrate the privacy threat, by proposing a membership inference attack tailored specifically to unprotected methylation Beacons. Implications of such membership inference attacks are beyond membership status: For instance, if the data set is collected from individuals carrying a certain disease, then the adversary can immediately infer this sensitive information about her target(s). Our membership

[1] https://www.ga4gh.org/
[2] https://beacon-network.org/

inference attack relies on the likelihood-ratio test and uses as probability estimate a normal distribution calibrated to the mean and standard deviation of the general population's methylation values. Our experimental results show that 100 queries are sufficient to achieve a successful attack with AUC (area under the ROC curve) above 0.9. To remedy this situation, we propose a novel differential privacy mechanism, namely SVT$^{\mathbf{2}}$, which is the core component of MBeacon, see Figure 5.1 for an overview. Extensive experiments over multiple data sets show that SVT$^{\mathbf{2}}$ can successfully mitigate membership privacy risks without significantly harming utility. We consider a MBeacon's query response to be highly privacy-sensitive if it differs from the expected response over the general population data. In fact, these differences are also the major reason why our membership inference attack is effective. A MBeacon is usually constructed over a database collected from people with a certain disease, and biomedical studies show that, for data of this kind, only a few methylation regions differ from the general population. As a consequence, only a few queries are highly privacy-sensitive. Therefore, we aim for a solution that scales noise to the sensitive responses in order to reduce the overall noise level of MBeacon, thus maintaining utility. One possible solution for the problem is the sparse vector technique, a differential privacy mechanism that is designed to scale noise to a subset of highly privacy-sensitive responses. The sparse vector technique determines whether a response is sensitive by comparing it to a fixed threshold. However, it cannot be applied to MBeacon, as we need to check whether the MBeacon response and the expected response agree with each other. The novelty of our proposed SVT$^{\mathbf{2}}$ lies in checking this agreement through two comparisons to a fixed threshold: one for the MBeacon response, the other for the expected response. We prove that SVT$^{\mathbf{2}}$ guarantees differential privacy.

The goal of the MBeacon system is to facilitate DNA methylation data sharing. Therefore, the main users of the system are researchers who want to discover institutions that possess data of interest. In order to quantify the impact of SVT$^{\mathbf{2}}$ on the real-world utility of our MBeacon system, we introduce a new utility metric by simulating a legitimate researcher who tries to find other institutions that possess methylation data similar to her own data.

We evaluate the performance of our privacy-preserving MBeacon through extensive experiments (simulating 2,100 researchers). The results show that the privacy loss on membership inference attacks can be minimized while the researcher utility still remains high. For carefully chosen privacy parameters, it is possible to decrease the attacker's performance to random guessing (AUC < 0.6) while preserving a high utility for the researcher (AUC > 0.8). Furthermore, we conduct a large-scale evaluation of privacy parameters for SVT$^{\mathbf{2}}$ and provide the necessary tools for an institution to tune these parameters to their needs.

Organization The rest of the chapter is organized as follows. We briefly introduce the current Beacon system in Section 5.1. MBeacon is formally defined in Section 5.2. Section 5.3 and 5.4 present our membership inference attack and its evaluation, respectively. In Section 5.5, we describe our defense mechanism SVT$^{\mathbf{2}}$. Section 5.6 introduces the utility metric. The effectiveness of our defense is evaluated in Section 5.7. Finally, we conclude in Section 5.8.

5.1 Background

In this section, we provide the necessary background on the current Beacon system as well as on DNA methylation.

5.1.1 Beacon System

Current biomedical data sharing has limited success due to its inherent privacy risks. To tackle this problem, GA4GH has established the Beacon system, also referred to as the Beacon network.

The Beacon system is a search engine that allows researchers to query whether any of the institutions taking part in the system possesses data of their interests. Each partner institution implements its own Beacon with its onsite data. These Beacons only support one simple type of query, i.e., the presence of a specified nucleotide (A, C, G, T) at a given position within a certain chromosome. The response is a binary "Yes" or "No". To give a concrete example, query "13 : 32936732 G > C" stands for "Are there any patients that have allele C at position 32936732 (with reference allele G) on chromosome 13?". When the Beacon system receives such a query, it forwards the query to each of its partner institutions' Beacons. If an institution's data set contains at least one record matching the query, then the Beacon answers "Yes". The names of all Beacons with "Yes" answers are sent back to the questioner. In the end, the questioner can contact the corresponding institutions for data access offline.

In this chapter, we propose the first Beacon system for sharing DNA methylation data, namely the MBeacon system. Since an individual's methylation data may carry information about her current disease status and environmental factors influencing her health, methylation data is considered highly privacy-sensitive. Also, a recent study has shown that methylation data can be re-identified by inferring the corresponding genomes [7] given an individual's methylation profile. Therefore, our MBeacon system is built following a privacy-by-design approach.

5.2 MBeacon Design

The MBeacon system is a search engine that indexes over multiple MBeacons. Each MBeacon is established by an institution with its own database, and this institution is referred to as a partner of the MBeacon system. We denote an institution by $\mathbb{I}$ and its MBeacon by $B_{\mathbb{I}}$. Without ambiguity, we also use $\mathbb{I}$ to represent the institution's database itself, which consists of multiple patients' methylation profiles. Moreover, we denote a patient by v, and her methylation profile, i.e., the sequenced methylation values, by a vector $m(v) \in \mathbb{R}^M_{[0,1]}$. The vector length M is equal to the total number of methylation positions considered, e.g., $M = 450,000$.

Similar to the genomic Beacon, our MBeacon supports one type of query, that is "*Are there any patients with this methylation value at a specific methylation position?*". Formally, we define a query q as a tuple (pos, val) where pos represents the queried position and val represents the queried value. A Beacon $B_{\mathbb{I}}$ is essentially a function,

$$B_{\mathbb{I}} : q \to \{0, 1\}, \tag{5.1}$$

Notation	Description
v	A victim
$m(v)$	Methylation profile of v
$\mathbb{I}$	An institution's database
$B_{\mathbb{I}}$	A MBeacon built on $\mathbb{I}$
q	A query to a MBeacon
$\overrightarrow{Q}$	A vector of queries
K	An adversary's background knowledge
b	No. of bins for methylation values
$\mathbb{A}$	Membership inference attack
δ	Measurement error
$\text{SVT}^{\mathbf{2}}$	The defense mechanism for MBeacon
α_i	No. of patients for q_i in MBeacon
β_i	Estimated No. of patients for q_i
P	Methylation of interest for researcher
D	Methylation of no-interest for researcher
$B_{P,D}$	MBeacon built with P and D
B_D	MBeacon built with D
T	MBeacon responses "Yes" if there are $p \geq T$ patients s with the requested value

Table 5.1: Notations.

where 0 represents "No" and 1 represents "Yes". It is worth noting that this general query format also allows researchers to infer answers to more complex queries, such as "*Are there any patients with methylation value above some threshold for a specific position?*". When a researcher issues a query to the MBeacon system, the system forwards this query to all the MBeacons, and returns the names of those MBeacons with "Yes" answers to the researcher.

For presentation purposes, we summarize the notations introduced here and in the following sections in Table 5.1.

5.3 Membership Inference Attack

To demonstrate the privacy risks of unprotected methylation Beacons, we propose a membership inference attack against them. In this section, we first present the considered adversarial model, then the methodology of our attack.

5.3.1 Threat Model

In general, the goal of membership inference attacks is to predict whether the victim is a member of the database given certain knowledge about the victim. For instance, an attacker with access to the sequenced methylation values of her victim aims to infer whether the victim is in the database containing methylation data collected from some

HIV carriers. By knowing who is member of the study, the attacker is able to infer the HIV status of her victim, even though (to the best of our knowledge) the HIV status is not directly detectable from the methylation values. This example demonstrates the severe consequence of membership inference. Moreover, all the existing attacks against genomic Beacons are membership inference attacks [132, 117, 3, 152].

We assume that the adversary has access to the victim's methylation data $m(v)$ and additional background knowledge K that we instantiate later. The adversary's goal is to perform an attack $\mathbb{A}$, to decide whether v is in the database of institution $\mathbb{I}$ by querying the MBeacon $B_{\mathbb{I}}$. Formally, the membership inference attack is defined as follows:

$$\mathbb{A} : (m(v), B_{\mathbb{I}}, K) \rightarrow \{0, 1\}, \tag{5.2}$$

where 1 means that the victim is in the MBeacon database and 0 that she is not. If v's methylation values are indeed part of the MBeacon's database ($m(v) \in \mathbb{I}$) and the attack output is 1, then the attack achieves a true positive for v. If the output is 0, then it is a false negative. However, if v's methylation values are not part of $B_{\mathbb{I}}$ (i.e., $m(v) \notin \mathbb{I}$) and the attack output is 0, this is a true negative, otherwise, if the output is 1, it is a false positive.

5.3.2 Attacking Methylation Beacons

We rely on the likelihood-ratio (LR) test to realize our membership inference attack for two main reasons. First, by the Neyman-Pearson Lemma [87, 134], the LR test achieves the highest power (true-positive rate) for a given false-positive rate in binary hypothesis testing if the theoretical preconditions are met. Second, the LR test has been successfully used by Shringarpure and Bustamante [132] and Raisaro et al. [117] for attacking genomic Beacons.

In general, the LR test formulates a null hypothesis H_0 and an alternative hypothesis H_1, and compares the quotient of the two hypotheses' likelihoods to a threshold. Our null hypothesis H_0 is defined as the queried victim v not being in the MBeacon ($m(v) \notin \mathbb{I}$), and the alternative hypothesis H_1 as the queried victim being in the MBeacon ($m(v) \in \mathbb{I}$).

The adversary submits a series $\overrightarrow{Q} = \langle q_1, \ldots, q_n \rangle$ ($n \leq M$) of queries to $B_{\mathbb{I}}$ with her victim's methylation values, i.e., $m(v)$, and get a list of responses, denoted by $B_{\mathbb{I}}(\overrightarrow{Q}) = \langle B_{\mathbb{I}}(q_1), \ldots, B_{\mathbb{I}}(q_n) \rangle$. Assuming that the different responses are independent,[3] the log-likelihood of the responses is

$$\begin{aligned} L(B_{\mathbb{I}}(\overrightarrow{Q})) = & \sum_{i=1}^{n} B_{\mathbb{I}}(q_i) \log(\Pr(B_{\mathbb{I}}(q_i) = 1)) + \\ & (1 - B_{\mathbb{I}}(q_i)) \log(\Pr(B_{\mathbb{I}}(q_i) = 0)). \end{aligned} \tag{5.3}$$

To implement the two hypotheses H_0 and H_1, we need to model $\Pr(B_{\mathbb{I}}(q) = 1)$ and $\Pr(B_{\mathbb{I}}(q) = 0)$. The approach in [132] cannot be directly applied as it is designed for

[3] We assume the adversary does not submit a single query for multiple times, and we assume correlations between different methylation positions are not exploited, because they are not (yet) well studied. Note that the same independence assumption has been used in previous works on genomic Beacons [132, 117].

Abbreviation	Description	number of patients	GSE identifier	by
Ependymoma	Ependymoma	48	GSE45353	[120]
GBM	glioblastoma	136	GSE36278	[138]
PA	pilocytic astrocytoma	61	GSE44684	[85]
ETMR-PNET	embryonal brain tumor and primitive neuroectodermal tumor	38	GSE52556	[78]
mHGA	4 different subtypes of pediatric glioblastomas	96	GSE55712	[47]
DIPG	diffuse intrinsic pontine glioma	28	GSE50022	[18]
IBD CD	Crohn's disease	77	GSE87640	[148]
IBD UC	ulcerative colitits	79	GSE87640	[148]

Table 5.2: Datasets used for our experiments.

genomic data, which is discrete. In contrast to that, methylation data is represented as a continuous value between 0 and 1. We propose to bin the methylation values into b equal-width bins that represent the range of values the questioner might be interested in.[4] Here, b is a parameter of the MBeacon system, and we empirically study the influence of different values for b on the attack performance in Section 5.4.

Thus, we represent a methylation Beacon as $B_{\mathbb{I}}^b$. The probability $\Pr(B_{\mathbb{I}}^b(q) = 0)$ to get a "No" answer, respectively $\Pr(B_{\mathbb{I}}^b(q) = 1)$ to get a "Yes" answer can be described in our case as:

$$\Pr(B_{\mathbb{I}}^b(q) = 0) = (1 - \tau^b(q))^N \tag{5.4}$$

$$\Pr(B_{\mathbb{I}}^b(q) = 1) = 1 - (1 - \tau^b(q))^N \tag{5.5}$$

Here, N is the number of patients in the Beacon. Following previous works on genomic Beacons [132, 117], we assume N to be publicly known and therefore being part of the attacker's background knowledge K. Meanwhile, τ^b is the probability of a patient having a methylation value in the interval determined by the respective bin. We can assume that the adversary has the exact probability as part of her background knowledge K. However, if the exact probability is not available and the adversary only knows the mean and standard deviation of people's methylation values at a certain position, she can approximate the probability with normal (Gaussian) distribution using μ_{pos} as the mean and σ_{pos} as the standard deviation of the queried position.[5] Concretely, $\tau^b(q)$ is

[4] There are two reasons why we only study equal-width bins: First, without further knowledge about the data distribution underlying the Beacon, it is hard to define a suitable bin width. Second, all Beacons should share the same interface to combine the answers in a well-defined way. This would not be possible if the bins vary across different Beacons based on the data set they are composed of.

[5] We experimentally found that the normal distribution fits methylation data best, using the Kolmogorov-Smirnov test and a p-value of 0.1. Other ways to approximate the probability are left for future work.

estimated as:

$$\begin{aligned}&\widetilde{\tau^b}(q) = \widetilde{\tau^b}((pos, val)) = \\ &cdf(\mu_{pos}, \sigma_{pos}, b_r) - cdf(\mu_{pos}, \sigma_{pos}, b_l)\end{aligned} \tag{5.6}$$

where *cdf* is the cumulative distribution function of the normal distribution, and b_r (b_l) denotes the value of the corresponding bin's right (left) edge. Notice that, like in the genomic setting, the general probability of having a specific allele is required as well, and it is realized by assuming the population's allele frequencies are part of the attacker's background knowledge K.

By inserting the probabilities from Equations 5.4 and 5.5 into Equation 5.3, we get

$$\begin{aligned}L_{H_0}(B^b_{\mathbb{I}}(\overrightarrow{Q})) = &\sum_{i=1}^{n} B^b_{\mathbb{I}}(q_i)\log(1-(1-\tau^b(q_i))^N)+ \\ &(1-B^b_{\mathbb{I}}(q_i))\log((1-\tau^b(q_i))^N)\end{aligned} \tag{5.7}$$

$$\begin{aligned}L_{H_1}(B^b_{\mathbb{I}}(\overrightarrow{Q})) = &\sum_{i=1}^{n} B^b_{\mathbb{I}}(q_i)\log(1-\delta(1-\tau^b(q_i))^{N-1})+ \\ &(1-B^b_{\mathbb{I}}(q_i))\log(\delta(1-\tau^b(q_i))^{N-1}).\end{aligned} \tag{5.8}$$

Notice that for the H_0 hypothesis, we consider all N patients in the database. However, for the H_1 hypothesis where we assume the target being part of the database, we consider only $N-1$ other patients that contribute to the answer in addition to the target. It might occur that two measurements of methylation data from the same patient and tissue type differ, either due to measurement errors or changes over time. Thus, the target may be part of the Beacon, but the attacker's data differs from the data entry in the Beacon. Similar to previous works, we denote this probability, i.e., measurement error, by δ and empirically evaluate its influence on our attack. We assume δ to be part of the attacker's background knowledge.

In the end, the log of the likelihood-ratio is given by:

$$\begin{aligned}\Lambda =&L_{H_0}(B^b_{\mathbb{I}}(\overrightarrow{Q})) - L_{H_1}(B^b_{\mathbb{I}}(\overrightarrow{Q})) \\ =&\sum_{i=1}^{n}(1-B^b_{\mathbb{I}}(q_i))\log\left(\frac{(1-\tau^b(q_i))^N}{\delta(1-\tau^b(q_i))^{N-1}}\right)+ \\ &B^b_{\mathbb{I}}(q_i)\log\left(\frac{1-(1-\tau^b(q_i))^N}{1-\delta(1-\tau^b(q_i))^{N-1}}\right).\end{aligned} \tag{5.9}$$

If Λ is lower than some threshold t, we reject the null hypothesis and predict that the victim is in the MBeacon database. Otherwise, we conclude that the victim is not.

Finally, the choice of the set of queries $\langle q_1, \ldots, q_n\rangle$ influences the attack performance as well. We follow the same approach as Raisaro et al. [117] to rank all possible queries with their expected information gain: For each methylation position *pos*, the attacker computes the difference between the victim's methylation value $m(v)_{pos}$ and the general population's value $\widetilde{\tau^b}(pos, m(v)_{pos})$. The larger this difference, the higher the probability of getting a "Yes" answer if the target is part of the Beacon, and simultaneously, the

higher the probability of getting a "No" answer if the target is not part of the Beacon. Therefore, we assume the attacker decides on the set of queries $\langle q_1, \ldots, q_n \rangle$ using this difference and querying the n most informative queries.

5.4 Attack Evaluation

In this section, we evaluate the performance of our membership inference attack against simulated methylation Beacons to demonstrate the privacy threat.

5.4.1 Data Sets

For our experiments, we relied on eight diverse data sets containing methylation profiles of patients carrying specific diseases. In total, we used methylation profiles of 563 individuals. The data sets are available online in the Gene Expression Omnibus database (GEO),[6] and we summarize them in Table 5.2. We used six brain tumor data sets, where the methylation data was sequenced from the respective brain tumor. Moreover, we also made use of an additional data set with two types of inflammatory bowel disease, where the methylation data was sequenced from blood samples, reported in the last two lines of Table 5.2. All of these data were generated with the Illumina 450k array, effectively determining the DNA methylation at 450,000 fixed positions.

Preprocessing. Most of the data sets had missing methylation sites for specific patients or even for all the patients sharing the same disease. We removed all methylation positions with missing data, which left us with 299,998 different methylation sites for the combination of all our eight data sets.

Human Subjects and Ethical Considerations. All data sets were publicly available in their anonymized form. Moreover, they have been stored and analyzed in anonymized form without having access to non-anonymized data. The membership inference we carried out did not reveal any more information than previously known by us.

5.4.2 Evaluation Results

We used our three largest[7] data sets, i.e., GBM, and both IBD data sets (referred to as IBD CD and IBD UC), to simulate three methylation Beacons, respectively. For each methylation Beacon, we randomly sampled 60 patients to construct its Beacon database. We followed the approach of previous works on Beacons testing with uniform sets of patients [132, 117, 3, 152]. This ensured the attacker can only exploit individual variances and not disease-induced systematic differences, i.e., variances that are unavoidably in the data. Later in Section 5.7, we explored another attack scenario on heterogeneous methylation sets.

We assume the adversary had access either to a randomly chosen sample from the methylation Beacon ("in" patient), or from the patients with the same disease who are not included in the methylation Beacon ("out" patient). For the "out" patients, we used the remainder of the patients that we did not sample into the methylation Beacon.

[6] `https://www.ncbi.nlm.nih.gov/geo/`

[7] We exclude the mHGA dataset, since it was not uniform but a combination of 4 subtypes.

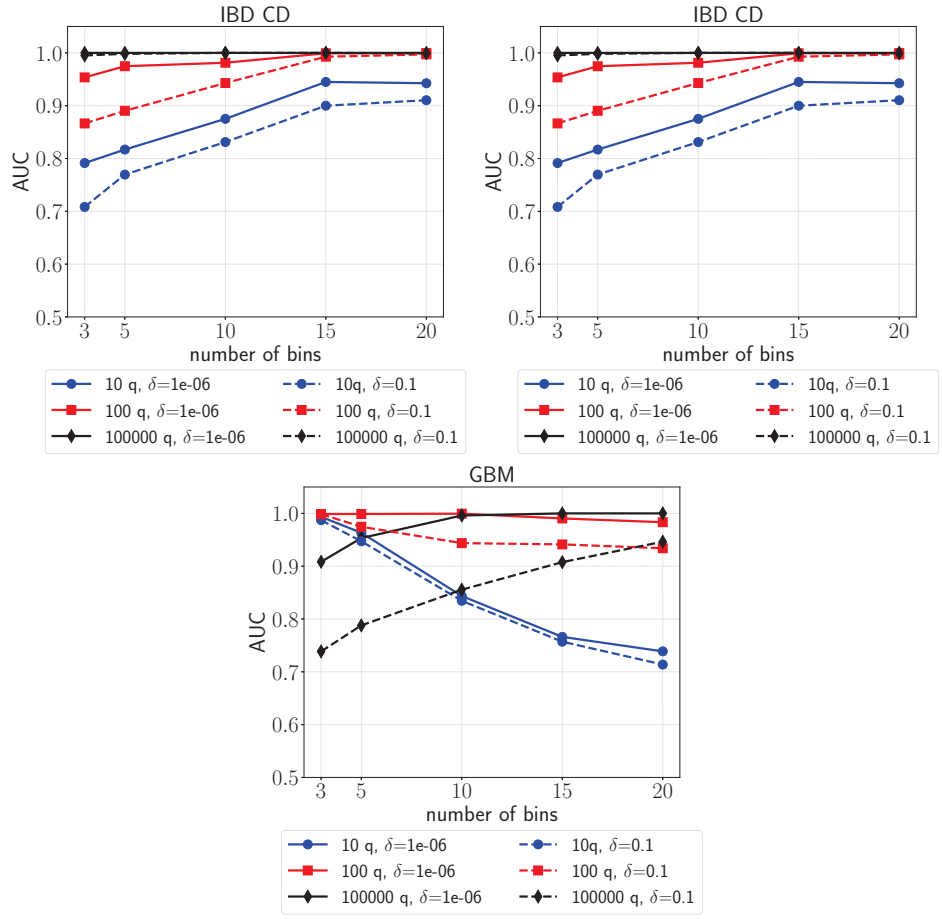


Figure 5.2: Influence of number of bins used and number of queries submitted on attacker's performance of the membership inference attack (a) on IBD CD, (b) IBD UC and (c) GBM.

For the "in" patients, we sampled the same number of patients from the methylation Beacon to not introduce a bias between "in" and "out" test patients. To reduce the size bias between GBM and the two IBD sets, we sampled at most 25 test patients. We repeated the random split of patients into methylation Beacon and testing set 10 times, which corresponds to a simulation of 500 attackers for GBM, 340 for IBD CD and 300 for IBD UC.

The attackers carried out the LR test as described previously in Section 5.3. We simulated attackers without access to the exact probability $\tau^b(q)$, because it is an unrealistic assumption that these are available. In fact, if such knowledge would be available, a lot of privacy would already be lost. Instead, we modeled attackers estimating the probabilities from a general background population. We combined the main data sets GBM, IBD UC, IBD CD with the other data sets (Ependymoma, mHGA, ETMR-PNET, PA and DIPG) as an estimate for the general population.[8] From this combined background data, we computed the attacker's background knowledge K as mean and standard deviation for each methylation position. Apart from being used in the LR test to estimate frequencies, the means were used to rank possible queries by their expected information gain, as discussed in Section 5.3.

We adopted the AUC, i.e., area under the ROC curve, as our evaluation metric since it does not involve picking a specific threshold for the LR test. The ROC curve is a 2D plot which reflects the relation between true positive rate and false positive rate over a series of thresholds for the LR test. The AUC summarizes the ROC curve as a single value. A ROC curve closer to top-left border of the plot, thus a larger AUC value, indicates a better prediction performance. Moreover, there exists a conventional standard[9] to interpret AUC values: AUC = 0.5 is equivalent to random guessing, whereas an AUC greater than 0.9 shows the prediction is excellent. It is worth noting that AUC has been adopted by many recent works for assessing privacy attacks [9, 107, 50, 103, 10, 108, 114].

To get an overview of the attack and the influence of various parameters, we varied the number of bins b from 3 to 20, and let the attacker submit 10, 100, and 100,000 unique queries to the respective methylation Beacon. We varied δ between 0.1 and 10^{-6}.

Figure 5.2 shows the attacker's performance as a function of b. Different numbers of queries submitted are displayed in different colors, and line styles indicate two choices for δ. As expected, the number of bins influenced the attacker's performance. The more bins, the fewer patients' values were expected in each of them, which made the membership inference easier.

The attacker's performance was high as soon as the number of bins is reasonably large (larger than 3), no matter whether 100,000 or just 10 queries are submitted. This demonstrates the privacy risk of unprotected methylation Beacons. Nevertheless, the GBM curve for only 10 queries demonstrates that asking too few queries may just not be enough for a successful attack. The choice of δ had only little influence on the attack performance in case more than 100 queries are submitted.

[8] Since general population statistics did not exist yet for methylation values, we had to estimate them. If the estimate was not accurate and a realistic attacker could get better estimates, the attack performance could increase.

[9] `http://gim.unmc.edu/dxtests/roc3.htm`

We observe a different attack performance depending on the data set, which was expected because we were testing different populations, diseases and tissues here. We note that both IBD data sets provided similar high AUCs, which can be explained by the fact that they were taken from the same tissue, namely blood cells.

As the increase in the attacker's performance was only slight for more than 10 bins, we fixed the number of bins to 10 in the remainder of the experiments to reduce the number of parameters and simplify the presentation. Additionally, we fixed δ to 10^{-6} to model the worst-case for privacy, even though the privacy risk differed not much for other choices of δ.

5.5 Defense

The results in Section 5.4 demonstrate the privacy risks stemming from unprotected methylation Beacons. To mitigate this threat, we present our defense mechanism, the double sparse vector technique (SVT$^{\mathbf{2}}$). We first explain the intuition behind it and then the defense mechanism in detail. In the end, we prove that our defense mechanism is differentially private.

5.5.1 Intuition

Recall that we assume the background knowledge K contains the means and standard deviations of the general population at the methylation positions of interest. That means, if one judges by the background knowledge that there should (or should not) be an individual with some value in a MBeacon and the MBeacon output confirms this, then not much privacy is lost. Yet, if MBeacon's answer deviates from the background knowledge, one learns an additional piece of information about the real distribution in the MBeacon for the queried position. In consequence, the privacy of patients in the MBeacon is at risk. More formally, we consider a MBeacon response as highly privacy-sensitive if it deviates from the answer we expect from the general population.

A MBeacon is usually built with data collected from people with certain disease. According to biomedical research [139, 145, 148], for data of this kind, only a few methylation regions differ from the general population. This indicates that just a few query responses are expected to be privacy-sensitive. Therefore, we aim for a solution that calibrates the noise specifically to those few responses in order to reduce the overall noise level of MBeacon, thus maintaining utility.

5.5.2 Background on SVT

t One possible solution in such a scenario is the sparse vector technique (SVT), a differential privacy mechanism which is designed to scale noise to a subset of sensitive responses.

In SVT, whether a response is sensitive or not is determined by a threshold T defined by the data owner: A response $\alpha \geq T$ is considered as privacy-sensitive, and one assumes most responses will yield $\alpha < T$. SVT guarantees differential privacy while scaling noise only to the privacy-sensitive answers. To this end, SVT has an additional privacy parameter c which refers to as the maximal amount of answers $\alpha \geq T$ the

mechanism can give over its whole lifetime. SVT adds noise to all queries (no matter whether they are privacy sensitive or not) before comparing to the threshold to ensure differential privacy. However, this noise is scaled to c instead of the much larger number of queries in total. For a detailed and formal description of SVT, we refer the reader to [38].

Algorithm 1: $\mathcal{A}$ outputs whether the database and prior agree on the number of patients in the queried position being above the threshold in a differentially private manner.

Input: base threshold T, privacy parameters ϵ_1, ϵ_2 and c, query sensitivity Δ, query vector $\overrightarrow{Q}$, database $\mathbb{I}$ and prior frequency $\mathtt{P}$

Result: sanitized responses R such that $r_i \in \{\perp, \top\}$ for each i

$z_1 = \text{LAP}(\frac{\Delta}{\epsilon_1})$; $\quad z_2 = \text{LAP}(\frac{\Delta}{\epsilon_1})$;
count = 0;
for *each query* q_i *in* $\overrightarrow{Q}$ **do**
  $y_i = \text{LAP}(\frac{2c\Delta}{\epsilon_2})$; $\quad y_i' = \text{LAP}(\frac{2c\Delta}{\epsilon_2})$;
  get α_i from $\mathbb{I}$ and β_i from $\mathtt{P}$;
  if *(*$\alpha_i + y_i < T + z_1$ *and* $\beta_i + y_i < T + z_1$*) or (*$\alpha_i + y_i' \geq T + z_2$ *and* $\beta_i + y_i' \geq T + z_2$*)* **then**
    $r_i = \perp$;
  else
    $r_i = \top$;
    count = count + 1 ;
  end
  if *count* $\geq c$ **then**
    Halt
  end
end

5.5.3 SVT²

However, we cannot directly apply SVT to protect our methylation Beacon, as our privacy-sensitive responses depend on whether we expect a "No" or a "Yes" answer, thus cannot be judged by a simple, fixed threshold. Concretely, suppose that we expect β patients in the queried bin, then the true number of patients in the bin, i.e., α, is privacy-sensitive if β and α lie on opposite sides of a predefined threshold T and the Beacon gives another answer than the one we expected. This means we need two comparisons to determine whether the answer is privacy-sensitive. Therefore, we propose double sparse vector technique (SVT$^{\mathbf{2}}$) to protect MBeacon. Since SVT is not applicable, we cannot compare our new technique SVT$^{\mathbf{2}}$ to SVT.

Formally, the ith query is not privacy-sensitive if the following expectation is met:

$$\begin{aligned}&((\alpha_i + y_i < T + z_1) \wedge (\beta_i < T + z_1))\\ \vee&((\alpha_i + y_i' \geq T + z_2) \wedge (\beta_i \geq T + z_2))\end{aligned} \tag{5.10}$$

Algorithm 2: $\mathcal{B}$ transforms the output of Algorithm 1 to the MBeacon output format.

Input: base threshold T, privacy parameters ϵ_1, ϵ_2 and c, query sensitivity Δ, query vector $\overrightarrow{Q}$, database $\mathbb{I}$ and prior frequency $\mathtt{P}$
Result: sanitized MBeacon responses $B_{\mathbb{I}}(\overrightarrow{Q})$

```
1 R⃗ = A(T, ε1, ε2, c, Δ, Q⃗, I, P) ;
2 for each query q_i in Q⃗ do
3 |   get r_i from R⃗;   get β_i from P;
4 |   if r_i = ⊥ then
5 |   |   B_I(q_i) = β_i ≥ T;
6 |   else
7 |   |   B_I(q_i) = ¬(β_i ≥ T);
8 |   end
9 end
```

where α_i is the number of patients in the MBeacon that corresponds to the query q_i, β_i is the estimated number of patients given by the general population,[10] and T is the threshold determining whether the α_i and β_i agree with each other. This (dis-)agreement is used to check whether the current query is privacy-sensitive or not: Only Condition 5.10 being false implies the query is privacy-sensitive. Moreover, z_1, z_2 and y_i, y_i' are noise variables sampled independently from the Laplace distribution. The sampling procedure is explained in detail later in this section.

Similar to the sparse vector technique, SVT$^{\mathbf{2}}$ bounds the total number of highly privacy-sensitive queries by maintaining a counter. Each privacy-sensitive query increases the counter. If a predefined maximal budget c is exceeded, the algorithm stops answering. In practice, that would mean that the corresponding MBeacon goes offline. We study when this is the case and whether this negatively influences the MBeacon utility in Section 5.7.

We disassemble our method SVT$^{\mathbf{2}}$ into Algorithms 1 and 2, also referred to as $\mathcal{A}$ and $\mathcal{B}$, for technical reasons of the differential privacy proof. Algorithm 1 answers whether the Beacon returns the requested answer in a differentially private way, Algorithm 2 then transforms this into the desired MBeacon answer format. Moreover, we formulate the expected answer as a query to a database to allow practitioners to instantiate it with the most suitable estimation for their purpose. In our evaluation, we use the normal distribution fitted to population-wide means and standard deviations, since the LR test also relies on their knowledge.

Algorithm 1 determines whether the prior and the MBeacon database agree on the answer. Condition 5.10 can be found in its generalized form in line 6 of Algorithm 1, where noise is added to the prior as well. This removes the assumption that β is publicly known from Algorithm 1. In the less privacy relevant case, answer can be directly given (line 7); in the more privacy relevant case, the privacy budget has to be decreased

[10] We assume the number of patients in the MBeacon database to be publicly known, so we can set $\beta_i = \tau^b(q_i)^N$.

in addition to returning the answer. If the current privacy budget *count* exceeds the maximal budget c, the algorithm has to stop answering (lines 12 and 13).

Algorithm 2 takes the output of Algorithm 1 and provides the differentially-private MBeacon answer by flipping the expected answer if necessary (line 7).

Notice that genomic Beacons usually set $T = 1$, but we generalize that setting by allowing other threshold values in a k-anonymity like fashion. For low values of T, the regions where the MBeacon answer differs from the expected answer grow, while for higher values they shrink. Furthermore, a user might not ask all queries at once, but adaptively, This is taken into consideration by SVT and consequently by SVT$^{\mathbf{2}}$, another important aspect in the on-line setting of MBeacon.

Repeated Queries. All differential privacy mechanisms, including our proposed mechanism, assume all queries are unique. Otherwise, the noise might eventually cancel out. A single person has no (legitimate) interest in asking the same query multiple times, but in an online Beacon setting, multiple users might ask the same question. However, the assumption is not a limitation: we maintain a database of responses and, if a question has been asked before, we answer the same way we did before. Initially, such a database can be empty and it gets filled with responses over time. Its size is in O(number of methylation regions×number of bins), but the total MBeacon database is O(number of methylation regions×number of patients) and we expect much more patients than bins, so the space overhead is acceptable.

5.5.4 Differential Privacy Proof

We first prove that Algorithm 1, i.e., $\mathcal{A}$, is differentially private. Then, we show that the transformation of its output to our desired MBeacon output using Algorithm 2, i.e., $\mathcal{B}$, is also differentially private. The combination of these arguments proves that SVT$^{\mathbf{2}}$ is differentially private.

Theorem 2. *Algorithm 1 is* $2(\epsilon_1 + \epsilon_2)$*-differentially private.*

We present a proof sketch of Theorem 2 in the following, the full proof is presented in the end of the chapter in Section 5.9

Proof sketch. Consider any output of $\mathcal{A}$ as a vector $\overrightarrow{R} \in \{\top, \bot\}^l$, we refer to its elements as $\overrightarrow{R} = \langle r_1, ..., r_l \rangle$. We define two sets $I_\top = \{i : r_i = \top\}$ and $I_\bot = \{i : r_i = \bot\}$ of indices for the different answers. For the analysis, let the noise values y_i, y_i' for all $i \in I_\top \cup I_\bot$ be arbitrary but fixed [38]. We concentrate on the probabilities over the randomness of z_1, z_2, i.e., the noise added to the threshold T. Moreover, let the two databases $\mathbb{I}$ and $\mathbb{I}'$ be arbitrary but fixed, such that $\mathbb{I}$ and $\mathbb{I}'$ are neighboring databases.

We begin by disassembling the probability of Algorithm 1 getting a specific answer $\overrightarrow{R}$ from $\mathbb{I}$ as follows.[11]

$$\Pr[\mathcal{A}(\mathbb{I}) = \overrightarrow{R}] = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \Pr[\rho_1 = z_1 \wedge \rho_2 = z_2] \\ f_{\mathbb{I}}(z_1, z_2) g_{\mathbb{I}}(z_1, z_2) dz_1 dz_2 \tag{5.11}$$

[11] As the other inputs are fixed, we use $\mathcal{A}(\mathbb{I})$ to represent Algorithm 1 in the proof, omitting the other input parameters for better readability.

where

$$f_{\mathbb{I}}(z_1, z_2) = \Pr[\wedge_{i \in I_\perp} r_i = \perp | \rho_1 = z_1 \wedge \rho_2 = z_2] \tag{5.12}$$

$$g_{\mathbb{I}}(z_1, z_2) = \Pr[\wedge_{i \in I_\top} r_i = \top | \rho_1 = z_1 \wedge \rho_2 = z_2] \tag{5.13}$$

To prove the theorem, it is sufficient to show that, for sensitivity Δ, the following inequalities hold:

$$f_{\mathbb{I}}(z_1, z_2) \leq f_{\mathbb{I}'}(z_1 + \Delta, z_2 - \Delta) \tag{5.14}$$

$$g_{\mathbb{I}}(z_1, z_2) \leq e^{2\epsilon_2} g_{\mathbb{I}'}(z_1 + \Delta, z_2 - \Delta) \tag{5.15}$$

$$\Pr[\rho_1 = z_1 \wedge \rho_2 = z_2] \leq e^{2\epsilon_1} \Pr[\rho_1 = z_1 + \Delta \wedge \rho_2 = z_2 - \Delta] \tag{5.16}$$

which gives us the required connection between the two neighboring databases $\mathbb{I}$ and $\mathbb{I}'$.

To prove Inequality 5.14, we utilize only the sensitivity Δ, i.e., $|\alpha_i - \alpha_i'| \leq \Delta$ and $|\beta_i - \beta_i'| \leq \Delta$. For Inequality 5.15, as g argues about the negation of the query formulation, if we simply follow the proof for Inequality 5.14, we would get $g_{\mathbb{I}'}(z_1 - \Delta, z_2 + \Delta)$. Therefore, we rely on the fact that noise values y_i are Laplace distributed (formally, $\textsc{Lap}(\frac{2c\Delta}{\epsilon_2})$) and use Inequalities 5.17 and 5.18 to prove it.

$$\Pr[\rho = y_i] \leq e^{\frac{\epsilon_2}{c}} \Pr[\rho = v_i + 2\Delta] \tag{5.17}$$

$$\Pr[\rho = y_i] \leq e^{\frac{\epsilon_2}{c}} \Pr[\rho = v_i - 2\Delta] \tag{5.18}$$

To prove Inequality 5.16, we use the fact that z_1 and z_2 are sampled from $\textsc{Lap}(\frac{\Delta}{\epsilon_1})$.

In the end, by combining Inequalities 5.14, 5.15 and 5.16, we prove Theorem 2 as follows:

$$\begin{aligned}
&\Pr[\mathcal{A}(\mathbb{I}) = \overrightarrow{R}] \\
=&\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \Pr[\rho_1 = z_1 \wedge \rho_2 = z_2] \\
&f_{\mathbb{I}}(z_1, z_2) g_{\mathbb{I}}(z_1, z_2) dz_1 dz_2 \\
\leq&\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{2\epsilon_1} \Pr[\rho_1 = z_1 + \Delta \wedge \rho_2 = z_2 - \Delta] \\
&f_{\mathbb{I}'}(z_1 + \Delta, z_2 - \Delta) e^{2\epsilon_2} g_{\mathbb{I}'}(z_1 + \Delta, z_2 - \Delta) dz_1 dz_2 \\
=&e^{2\epsilon_1 + 2\epsilon_2} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \Pr[\rho_1 = z_1' \wedge \rho_2 = z_2'] \\
&f_{\mathbb{I}'}(z_1', z_2') g_{\mathbb{I}'}(z_1', z_2') dz_1' dz_2' \\
=&e^{2(\epsilon_1 + \epsilon_2)} \Pr[\mathcal{A}(\mathbb{I}') = \overrightarrow{R}]
\end{aligned}$$

■

The purpose of Algorithm 1 is to answer whether the database is approximated well by the background knowledge in a differentially private way. To output a Beacon answer of the format "Yes, such data is available" resp. "No, such data is not available", we need to remove the background knowledge from Algorithm 1's answer. This is performed by Algorithm 2, which preserves the differential privacy of the answer. Intuitively, the transformation maintains differential privacy due to the composition and post-processing theorems. However, these theorems are not directly applicable due to our database format. Therefore, we prove the following theorem.

Theorem 3. *Algorithm 2 is* $2(\epsilon_1 + \epsilon_2)$*-differentially private.*

Proof. Once the prior frequency P is fixed, the output of Algorithm 2 only depends on the output of Algorithm 1, namely, whether the prior is correct or has to be flipped. Formally, we describe this as follows.

First, fixing any output $\overrightarrow{\mathcal{R}} \in \{$ "Yes", "No" $\}^l$ of Algorithm 2 on $Q = \langle q_1, ..., q_l \rangle$, we have:

$$\frac{\Pr[\mathcal{B}(T, \epsilon_1, \epsilon_2, c, Q, \mathbb{I}, \mathtt{P}) = \overrightarrow{\mathcal{R}}]}{\Pr[\mathcal{B}(T, \epsilon_1, \epsilon_2, c, Q, \mathbb{I}', \mathtt{P}) = \overrightarrow{\mathcal{R}}]} = *$$

As Algorithm 2 is deterministic, we have:

$$* = \frac{\Pr[\mathcal{A}(T, \epsilon_1, \epsilon_2, c, Q, \mathbb{I}, \mathtt{P}) = \overrightarrow{R}]}{\Pr[\mathcal{A}(T, \epsilon_1, \epsilon_2, c, Q, \mathbb{I}', \mathtt{P}) = \overrightarrow{R}]} = *$$

Algorithm 1 is $2(\epsilon_1 + \epsilon_2)$-differentially private, thus:

$$* \leq e^{2(\epsilon_1 + \epsilon_2)}$$

■

Notice that, for technical reasons, we disassemble our proposed method into two stages. However, one can of course perform both stages at once and directly output the MBeacon response. Since we assume the prior frequency is publicly known, we do not have to add noise to its result, which yields Condition 5.10 above.

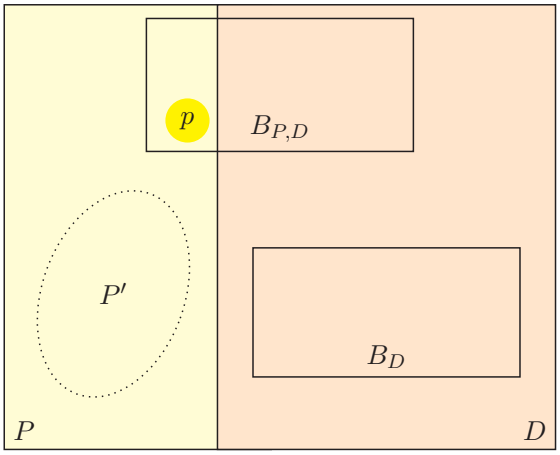
Setting the Parameters. We have shown that $\mathcal{A}$ is $2(\epsilon_1 + \epsilon_2)$-differentially private to make the connection between privacy-sensitive and less privacy-sensitive queries as well as the connection to the sparse vector technique visible. However, for tuning parameters, it is desirable to have only a single privacy parameter ϵ in addition to the budget c. Lyu et al. [90] showed that the ratio $\epsilon_1 : \epsilon_2 = 1 : (2c)^{\frac{2}{3}}$ maximizes utility, while preserving $\epsilon = \epsilon_1 + \epsilon_2$. We adopt Lyu's ratio between ϵ_1 and ϵ_2. The sensitivity Δ is 1 in our case, since removing a participant's entry from the database or changing it can affect the bin count by at most one. For a given privacy parameter and using $\Delta = 1$, we set:

$$\epsilon_1 = \frac{\frac{\epsilon}{2}}{(2c)^{\frac{2}{3}} + 1} \qquad \epsilon_2 = (2c)^{\frac{2}{3}} \epsilon_1$$

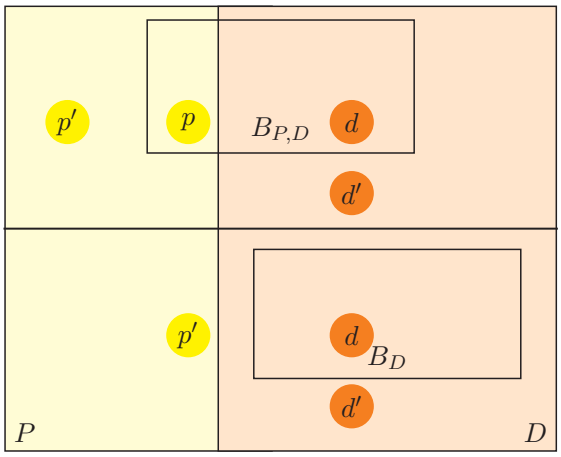
Application to other Domains. We emphasize that SVT$^{\mathbf{2}}$ is a general differential privacy mechanism, and can be applied in other cases beyond MBeacon: SVT$^{\mathbf{2}}$ is useful for comparing a database to a general belief in a differentially-private way. Moreover, comparing two databases is possible using Algorithm 1 since it applies noise to both databases α and β. In the future, we plan to apply SVT$^{\mathbf{2}}$ to other data domains, such as location data [109, 156], social network data [103, 157], and other types of biomedical data [8].

5.6 Researcher Utility

The goal of the MBeacon system is to facilitate biomedical data sharing among the research community. Therefore, we quantify the utility of MBeacon as the ability of a legitimate researcher to find methylation data of interest.



(a) The researcher knows patient(s) from P' and is interested in patients from P in $B_{P,D}$, shown exemplified by patient p. The researcher's task is to find that B_D is not interesting for research, while $B_{P,D}$ is interesting. We focus on the worst-case of the researcher by assuming P being a minority in $B_{P,D}$ to give a lower bound on utility.



(b) The attacker either queries $B_{P,D}$ or B_D (without knowing which one), and might have a target p' from P outside the MBeacon, a target p from P in the MBeacon or a target d resp. d' from D in resp. outside of the MBeacon. To compare side-by-side with the researcher, we again assume P to be a minority in $B_{P,D}$.

Figure 5.3: A graphical overview on the general utility setup for researcher (a) and the general utility setup for the attacker (b).

Concretely, a researcher is interested in methylation profiles of people with a certain phenotype or disease. We use the set P to represent all these methylation profiles. Moreover, the researcher already has multiple profiles in P on her site, denoted by P' with $P' \subset P$. Then, her goal is to find those MBeacons with methylation profiles from $P \backslash P'$. A central assumption here is that methylation profiles in P are similar to each other.

As the MBeacon system only supports queries on single methylation positions, the researcher also relies on the LR test to find MBeacons that contain patients in P. Moreover, there often exist measurement errors when collecting methylation values. To increase the reliability of her LR test, the researcher further averages all the methylation profiles in P'.

Ideally, the researcher queries a MBeacon B_P only containing patients of interest. To simulate a more realistic case, we assume the existence of another population D the researcher is not interested in. Notice that D might also be a mixture of populations. The researcher tries to distinguish a MBeacon B_D containing no patients of interest from a MBeacon $B_{P,D}$ containing some patients of interest. In the worst case, there are only a few patients from P in $B_{P,D}$. In that case, the contribution of patients from P is small and may be hidden due to the SVT$^{\mathbf{2}}$ protection.

To get the lower bound of the MBeacon utility, we concentrate on this worst case scenario. Figure 5.3a depicts a graphical summary of the researcher setup. The researcher achieves a true positive if the MBeacon she selects contains some profiles in P. A false positive indicates that the MBeacon she finds does not have the data of her interest. True negative and false negative are defined accordingly. Given these numbers, in particular the true-positive and false-positive rates, we can derive the AUC as our core utility metric.

Attack Scenarios. In order to find a good trade-off between utility and privacy, we have to evaluate the attacker's success under the same scenario as the researcher. The attacker's goal is to detect with high probability whether a target is part of the MBeacon database or not. Of course, the attacker does not know whether she is querying a MBeacon of the form $B_{P,D}$ or B_D, similar to the researcher not knowing the distinction a priori. Moreover, the attacker's target might be a patient in D or in P. We refer to such an attacker as "full" attacker; a graphical overview is displayed in Figure 5.3b.

The evaluation of the "full" attacker is comparable to the researcher evaluation, but not to existing works [132, 117, 3, 152], where the MBeacon and the victim are from one uniform data set. Therefore, we additionally model an attacker querying only B_D and trying to infer whether a victim in D is part of the MBeacon. We refer to this second attacker as the "standard" attacker, since it is the same as the one considered in Section 5.4.

5.7 Defense Evaluation

We evaluate our defense mechanism SVT$^{\mathbf{2}}$ in this section with respect to the attack performance and utility as defined in Section 5.6.

5.7.1 Experimental Setup

For the set of researcher's interest, P, we used Ependymoma, which contained data from 48 patients. For the set D the researcher was not interested in, we used either GBM, IBD CD or IBD UC as before, forming three different types of MBeacons.

Each of these MBeacons consisted of a certain number of patients in P, we tested 7 different choices for this number including 1, 3, 5, 10, 13, 15 and 20. The remaining patients were randomly sampled from the respective D such that a total size of 60 was reached. Moreover, we sampled randomly 60 patients from the respective D to construct B_D. We simulated 5 researchers querying each pair of corresponding MBeacons $B_{P,D}$ and B_D. The researcher possessed P' containing 5 randomly sampled patients in P that were not used in the MBeacon.[12] As mentioned in Section 5.6, the researcher averaged these patients' profiles to reduce measurement errors. The whole sampling process was repeated 10 times to ensure the observations were not due to randomness.

For the attacker simulations, we re-used the MBeacons we constructed before for the researcher, but sampled test patients differently. The "full" attacker has access to only a single patient. We randomly sampled 12 patients from each of $B_{P,D}$ and B_D as the ones in the MBeacon. Accordingly, we sample 24 patients from $P \cup D$ as the patients that were outside the MBeacon. Since we assumed throughout the experiments that patients in P were the minority, we used only up to a third of patients in P and the remainder in D. As before, we repeated random sampling 10 times. The "best" attacker did not have access to $B_{P,D}$ and, consequently, did not get test patients in P. Instead, we sampled 24 test patients from B_D and 24 test patients from $D \setminus B_D$ for each of the B_D MBeacons.

We assumed both researchers and attackers had access to the mean and standard deviation of the general population, that we estimated by a union of all our available data sets as before. These means and standard deviations were used to carry out LR tests and rank queries, up to 250,000 queries were allowed per researcher or attacker, respectively. Moreover, both researchers and attackers sorted their queries based on expected information gain as explained in Section 5.3 and used in the previous experiments in Section 5.4.

To sum up, we tested three different choices for D, and 7 different numbers of patients from P in $B_{P,D}$, simulated 5 researchers querying each of the MBeacons and re-sampled the experiments 10 times, so simulate in total 2,100 researchers. Due to the attackers not averaging over multiple patients, we could simulate more membership inference attacks: 10,500 carried out by the "full" and the "standard" attacker each.

5.7.2 Evaluation of SVT²

First, we evaluated the influence of the number of patients in P in the MBeacons of type $B_{P,D}$. We observe that if there were 5 or more patients of interest, the researcher's performance was maximized. The "full" attacker, however, suffered from more patients in P, probably due to the higher variance in the MBeacon.

[12] If the researcher averaged fewer patients, the performance could decrease slightly since individual, non-disease related changes in the patients' methylation values become more pronounced in the search.

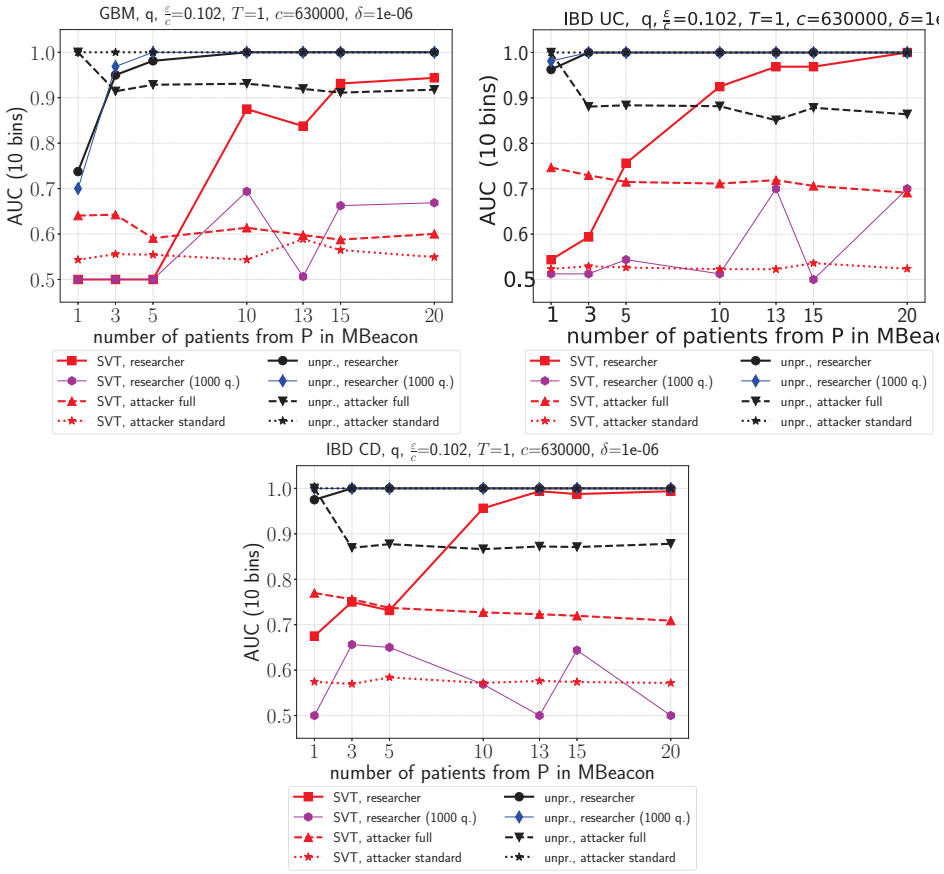


Figure 5.4: Comparison of researchers' and attackers' performances in unprotected MBeacon (black, abbreviated as "unpr.") and protected MBeacon (red) using GBM (left), IBD UC (middle) and IBD CD (right) as D using up to 100,000 queries. Additionally, we plot the researchers' performances for 1,000 queries in blue (unprotected) and magenta (protected). AUCs with values smaller than 0.5 are displayed as 0.5.

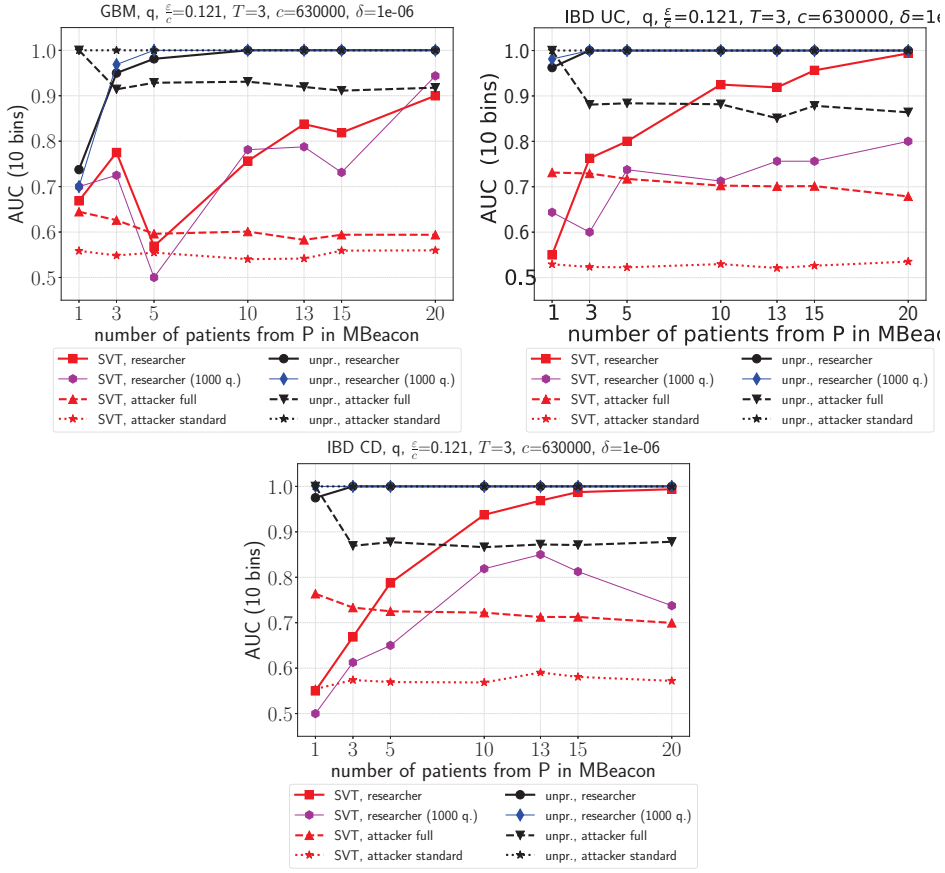


Figure 5.5: Comparison of researchers' and attackers' performances when setting $T = 3$ in unprotected MBeacon (black, abbreviated as "unpr.") and protected MBeacon (red) using GBM (left), IBD UC (middle) and IBD CD (right) as D using up to 100,000 queries. Additionally, we plot the researchers' performances for 1,000 queries in blue (unprotected) and magenta (protected). AUCs with values smaller than 0.5 are displayed as 0.5.

Second, we focused on SVT$^{\mathbf{2}}$. Our protection mechanism has three parameters: a threshold T determining how many patients have to be in the respective bin to answer "Yes", as well as the privacy parameter ϵ and the query budget c, which both calibrate the noise.

The Privacy Budget. We aimed for parameters that drop the "standard" attackers' performance to about 0.5 AUC, equivalent to random guessing, while minimizing the noise. Moreover, exceeding the query budget is something MBeacon providers would want to avoid, because the MBeacon has to stop answering in that case. Therefore, we chose a budget that was never exceeded in our simulations. The researchers and the two different types of attackers ("standard" and "full") were all simulated separately, so our budget had to be sufficient for 50 attackers submitting 12,500,000 (50×250,000) queries in total. Notice that not all of those queries were expected to be unique and not all of them fell into the category of privacy-sensitive queries for which the budget must be reduced.

Threshold $T = 1$. We started with the default threshold $T = 1$, i.e., the MBeacon answers "Yes" if there was at least one patient's methylation value in the queried bin. A budget of $c = 630,000$ was sufficient for our simulations. This might seem large at first glance, but notice that, having 10 bins, there are 300,000×10 different queries that can be asked, so our c corresponds to about 21% of them. We report the privacy level that we found as a suitable trade-off between privacy and utility at $\frac{\epsilon}{c} = 0.102$. We report the privacy levels as in [130].

As shown in Figure 5.4, the privacy level was sufficient to drop the "standard" attackers' performance to less than 0.6 AUC which shows that the privacy threat can be mitigated successfully. In the more realistic "full" attacker scenario, however, the attacker's performance was higher, which is explained by the fact that membership attacks with patients from P against the B_D MBeacon were most successful. Nevertheless, we see a significant drop in performance due to the application of SVT$^{\mathbf{2}}$.

The researcher's performance was still good with 0.8 AUC or more, depending on the number of patients from P in the MBeacon.

The impact of noise got even more pronounced if we assumed the researcher to submit only 1,000 queries. On the unprotected methylation Beacon, the AUC was about the same, however, the researcher could not get good answers from an SVT$^{\mathbf{2}}$-protected MBeacon. This shows the price of the SVT$^{\mathbf{2}}$ protection: more queries have to be submitted.

Threshold $T = 3$. Next, we increase the threshold. We kept the same budget c since we just wanted to study the influence of the increased threshold, for which we also have to increase the privacy budget to $\frac{\epsilon}{c} = 0.121$. Figure 5.5 shows the result, we see a similar performance. This was the same for $T = 2$, which we therefore do not show here. A threshold $T > 3$ would probably not be accepted by researchers given this MBeacon sizes, therefore, we did not experiment with higher thresholds.

Setting the Parameters. The above results demonstrate that the threshold and other privacy parameters have to be chosen dependent on the use case to maximize utility and minimize the privacy loss. We believe that our general method of parameter tuning, namely, setting a budget c that was not exceeded, then changing values of ϵ based on attacker's and researcher's performance and increasing c if needed by a higher noise

level (or reducing it if the consumed budget is much smaller), yields a good trade-off between utility and privacy for any dataset and MBeacon size.

5.8 Conclusion

In this chapter, we propose the first Beacon system for sharing DNA methylation data, namely, the MBeacon system. Due to the severe privacy risks stemming from DNA methylation data, our construction of MBeacon follows a privacy-by-design approach.

We first illustrate the severe privacy risks by conducting a membership inference attack based on the LR test. Experimental results on multiple data sets show that with 100 queries, the adversary is able to achieve a superior performance. Then, we propose a defense mechanism, SVT$^{\mathbf{2}}$, to implement our privacy-preserving MBeacon. Our SVT$^{\mathbf{2}}$ is an advancement of the sparse vector technique, one type of differential privacy algorithms. We theoretically prove that SVT$^{\mathbf{2}}$ is differentially private. Since the goal of MBeacon is to facilitate biomedical data sharing, we propose a new metric for measuring researchers' utility considering a realistic scenario. Extensive experiments demonstrate that, using carefully chosen parameters, MBeacon can degrade the performance of the membership inference attack significantly without substantially hurting the researchers' utility.

There are two directions we want to explore in the future. First, we plan to extend the Beacon-style system to other types of biomedical data, such as gene expression, microRNA or laboratory tests. In particular, this requires to adapt the estimate of the general population accordingly. Second, the current Beacon systems only support queries on a single position. We plan to extend the Beacon system to support multiple-position queries. On one hand, this new system should improve the utility for the researchers. On the other hand, it will raise new privacy challenges.

5.9 Full Proof of Theorem 2

We begin by disassembling the probability of getting a specific answer $\overrightarrow{R}$ from a database $\mathbb{I}$ as in Equation 5.19.

$$\Pr[\mathcal{A}(\mathbb{I}) = \overrightarrow{R}] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \Pr[\rho_1 = z_1 \wedge \rho_2 = z_2] \\ f_{\mathbb{I}}(z_1, z_2) g_{\mathbb{I}}(z_1, z_2) dz_1 dz_2 \tag{5.19}$$

where

$$f_{\mathbb{I}}(z_1, z_2) = \Pr[\wedge_{i \in I_\perp} r_i = \perp | \rho_1 = z_1 \wedge \rho_2 = z_2] \tag{5.20}$$

$$g_{\mathbb{I}}(z_1, z_2) = \Pr[\wedge_{i \in I_\top} r_i = \top | \rho_1 = z_1 \wedge \rho_2 = z_2] \tag{5.21}$$

Intuitively, g deals with the positive answers indicating highly privacy-sensitive results and f deals with the negative answers. We will show that, for sensitivity Δ,

$$f_{\mathbb{I}}(z_1, z_2) \leq f_{\mathbb{I}'}(z_1 + \Delta, z_2 - \Delta) \tag{5.22}$$

$$g_{\mathbb{I}}(z_1, z_2) \leq e^{2\epsilon_2} g_{\mathbb{I}'}(z_1 + \Delta, z_2 - \Delta) \tag{5.23}$$

$$\Pr[\rho_1{=}z_1 \wedge \rho_2{=}z_2] \leq e^{2\epsilon_1} \Pr[\rho_1 = z_1 + \Delta \wedge \rho_2 = z_2 - \Delta] \tag{5.24}$$

which gives us the required connection between the two neighboring databases $\mathbb{I}$ and $\mathbb{I}'$.
Proof of Inequality 5.22. Due to the independence of the database entries Equation 5.20 is equivalent to

$$f_{\mathbb{I}}(z_1, z_2) = \prod_{i \in I_\perp} \Pr[r_i = \perp | \rho_1 = z_1 \wedge \rho_2 = z_2] = *$$

By plugging in our query formula, we have:

$$\begin{aligned} * &= \prod_{i \in I_\perp} \Pr[(\alpha_i + y_i < T + z_1 \wedge \beta_i + y_i < T + z_1) \\ &\qquad \vee (\alpha_i + y_i' \geq T + z_2 \wedge \beta_i + y_i' \geq T + z_2)] \\ &= \prod_{i \in I_\perp} \Pr[(y_i < T + z_1 - \alpha_i \wedge y_i < T + z_1 - \beta_i) \\ &\qquad \vee (y_i' \geq T + z_2 - \alpha_i \wedge y_i' \geq T + z_2 - \beta_i)] = * \end{aligned}$$

Next, we want to exploit the sensitivity to change to the other database. We know that $|\alpha_i - \alpha_i'| \leq \Delta$ leads to

$$\alpha_i \leq \alpha_i' + \Delta \quad \text{and} \quad \alpha_i \geq \alpha_i' - \Delta. \tag{a}$$

Similarly, $|\beta_i - \beta_i'| \leq \Delta$ indicates

$$\beta_i \leq \beta_i' + \Delta \quad \text{and} \quad \beta_i \geq \beta_i' - \Delta. \tag{b}$$

By using Equation (a) and (b), we have the following relation.

$$\begin{aligned} * &\leq \prod_{i \in I_\perp} \Pr[(y_i < T + z_1 - (\alpha_i' - \Delta) \wedge y_i < T + z_1 - (\beta_i' - \Delta)) \\ &\qquad \vee (y_i' \geq T + z_2 - (\alpha_i' + \Delta) \wedge y_i' \geq T + z_2 - (\beta_i' + \Delta))] \\ &= \prod_{i \in I_\perp} \Pr[(\alpha_i' + y_i < T + (z_1 + \Delta) \wedge \beta_i' + y_i < T + (z_1 + \Delta)) \\ &\qquad \vee (\alpha_i' + y_i' \geq T + (z_2 - \Delta) \wedge \beta_i' + y_i' \geq T + (z_2 - \Delta))] \\ &= f_{\mathbb{I}'}(z_1 + \Delta, z_2 - \Delta) \end{aligned}$$

Therefore, Inequality 5.22 if proven. Notice that the last step of simplification would not be possible if we had just used one noise variable $z = z_1 = z_2$. □
Proof of Inequality 5.23. Again, by independence of the database entries and the negation of our query formulation, we have:

$$\begin{aligned} g_{\mathbb{I}}(z_1, z_2) = \prod_{i \in I_\top} \Pr[\neg((\alpha_i + y_i < T + z_1 \wedge \beta_i + y_i < T + z_1) \\ \vee (\alpha_i + y_i' \geq T + z_2 \wedge \beta_i + y_i' \geq T + z_2))] = * \end{aligned}$$

We push the negation inwards:

$$\begin{aligned} * = \prod_{i \in I_\top} \Pr[&(\alpha_i + y_i \geq T + z_1 \vee \beta_i + y_i \geq T + z_1) \\ &\wedge (\alpha_i + y_i' < T + z_2 \vee \beta_i + y_i' < T + z_2)] = * \end{aligned}$$

The sensitivities $|\alpha_i - \alpha_i'| \leq \Delta$ and $|\beta_i - \beta_i'| \leq \Delta$ allow us to introduce the other database $\mathbb{I}'$ similar to before:

$$\begin{aligned} * \leq \prod_{i \in I_\top} \Pr[&(y_i \geq T + z_1 - \alpha_i' - \Delta \vee y_i \geq T + z_1 - \beta_i' - \Delta) \\ &\wedge (y_i' < T + z_2 - \alpha_i' + \Delta \vee y_i' < T + z_2 - \beta_i' + \Delta)] = * \end{aligned}$$

We could go on as before with f, but it would not provide the desired bounds, as the signs of Δ would be flipped. Instead, we exploit that the noise values y_i are $\textsc{Lap}(\frac{2c\Delta}{\epsilon_2})$ distributed:

$$\Pr[\rho = y_i] \leq e^{\frac{\epsilon_2}{c}} \Pr[\rho = y_i + 2\Delta] \tag{c}$$

$$\Pr[\rho = y_i] \leq e^{\frac{\epsilon_2}{c}} \Pr[\rho = y_i - 2\Delta] \tag{d}$$

We cannot use that directly, as we have a logical formula in the probabilities. The outer conjunction can be rewritten to a multiplication due to independence of the noise variables v_i, v_i'. The inner disjunction is not problematic, as we show below. We prove it generally for any $x, x', Y_1, Y_2, Y_3, Y_4$ to increase readability. Later, we just need the following instantiations:

$$\begin{aligned} &x = y_i \qquad x' = y_i' \\ &Y_1 = T + z_1 - \alpha_i - \Delta \qquad Y_2 = T + z_1 - \beta_i - \Delta \\ &Y_3 = T + z_2 - \alpha_i + \Delta \qquad Y_4 = T + z_2 - \beta_i + \Delta \end{aligned}$$

We want to re-formulate $\Pr[x \geq Y_1 \vee x \geq Y_2]$ for some arbitrary, but fixed x, Y_1, Y_2. For probabilities, the following holds:

$$\Pr[x \geq Y_1 \vee x \geq Y_2] = \Pr[x \geq \min(Y_1, Y_2)]$$

Then, we apply (c):

$$\begin{aligned} &\Pr[x \geq \min(Y_1, Y_2)] = \Pr[x \geq M] = \int_M^\infty \Pr[x = m] dm \\ &\leq e^{\frac{\epsilon_2}{c}} \int_M^\infty \Pr[x = m + 2\Delta] dm \text{ (substitute } t = \phi(m) = m + 2\Delta) \\ &= e^{\frac{\epsilon_2}{c}} \int_{\phi(M)}^{\phi(\infty)} \Pr[x = t] dt = e^{\frac{\epsilon_2}{c}} \Pr[x \geq \phi(M)] \\ &= e^{\frac{\epsilon_2}{c}} \Pr[x \geq \min(Y_1, Y_2) + 2\Delta] \\ &= e^{\frac{\epsilon_2}{c}} \Pr[x - 2\Delta \geq \min(Y_1, Y_2)] \\ &= e^{\frac{\epsilon_2}{c}} \Pr[x - 2\Delta \geq Y_1 \vee x - 2\Delta \geq Y_2] \end{aligned} \tag{5.25}$$

Similarly, we re-formulate $\Pr[x' < Y_3 \vee x' < Y_4]$ for some arbitrary, but fixed x', Y_3, Y_4.

$$\Pr[x' < Y_3 \vee x' < Y_4] = \Pr[x' < \max(Y_3, Y_4)]$$

Now, we apply (d) as above:

$$\begin{aligned} &\Pr[x'<\max(Y_3, Y_4)] \leq e^{\frac{\epsilon_2}{c}} \Pr[x'<\max(Y_3, Y_4) - 2\Delta] \\ =&e^{\frac{\epsilon_2}{c}}\Pr[x'+2\Delta<\max(Y_3,Y_4)]=e^{\frac{\epsilon_2}{c}}\Pr[x'+2\Delta<Y_3\vee x'+2\Delta<Y_4] \end{aligned} \tag{5.26}$$

Now, we come back to the proof for Inequality 5.23. Since v_i and v_i' are independent, we have the following.

$$\begin{aligned} *=&\prod_{i\in I_\top}\Pr[y_i\geq T+z_1-\alpha_i'-\Delta\vee y_i\geq T+z_1-\beta_i'-\Delta] \\ &\Pr[y_i'<T+z_2-\alpha_i'+\Delta\vee y_i'<T+z_2-\beta_i'+\Delta] = * \end{aligned}$$

Next, by utilizing Inequalities 5.25 and 5.26, we have:

$$\begin{aligned} *\leq&\prod_{i\in I_\top} e^{\frac{\epsilon_2}{c}} \Pr[y_i\geq T + z_1 - \alpha_i' + \Delta\vee y_i\geq T + z_1 - \beta_i' + \Delta] \\ &e^{\frac{\epsilon_2}{c}} \Pr[y_i'<T + z_2 - \alpha_i' - \Delta\vee y_i'<T + z_2 - \beta_i' - \Delta] \\ =&\prod_{i\in I_\top} e^{2\frac{\epsilon_2}{c}}\Pr[y_i\geq T+z_1-\alpha_i'+\Delta\vee y_i\geq T + z_1 - \beta_i' + \Delta] \\ &\Pr[y_i' < T + z_2 - \alpha_i' - \Delta \vee y_i' < T + z_2 - \beta_i' - \Delta] \\ =&e^{\frac{2\epsilon_2|I_\top|}{c}}\prod_{i\in I_\top}\Pr[y_i\geq T+z_1-\alpha_i'+\Delta\vee y_i\geq T+z_1-\beta_i'+\Delta] \\ &\Pr[y_i' < T + z_2 - \alpha_i' - \Delta \vee y_i' < T + z_2 - \beta_i' - \Delta] = * \end{aligned}$$

As we have at most c answers for privacy-sensitive queries, i.e., $|I_\top| \leq c$, thus we have:

$$\begin{aligned} * \leq& e^{2\epsilon_2}\prod_{i\in I_\top}\Pr[((y_i\geq T+z_1-\alpha_i'+\Delta)\vee(y_i\geq T+z_1-\beta_i'+\Delta)) \\ &\wedge((y_i'<T+z_2-\alpha_i'-\Delta)\vee(y_i'<T+z_2-\beta_i'-\Delta))] \\ =& e^{2\epsilon_2} g_{\mathbb{I}'}(z_1+\Delta, z_2-\Delta) \qquad \square \end{aligned}$$

Proof of Inequality 5.24. As ρ_1 and ρ_2 are sampled independently, $\Pr[\rho_1 = z_1 \wedge \rho_2 = z_2]$ equals to:

$$\Pr[\rho_1 = z_1 \wedge \rho_2 = z_2] = \Pr[\rho_1 = z_1]\Pr[\rho_2 = z_2] = *$$

Moreover, as ρ_1 and ρ_2 are sampled from $\textsc{Lap}(\frac{\Delta}{\epsilon_1})$, we have

$$\begin{aligned} *\leq& e^{\epsilon_1}\Pr[\rho_1 = z_1 + \Delta] * e^{\epsilon_1}\Pr[\rho_2 = z_2 - \Delta] \\ =&e^{2\epsilon_1}\Pr[\rho_1 = z_1 + \Delta \wedge \rho_2 = z_2 - \Delta] \qquad \square \end{aligned}$$

Let us wrap up using the above proofs on Inequalities 5.22 to 5.24 on 5.19.

$$
\begin{aligned}
&\Pr[\mathcal{A}(\mathbb{I}) = \overrightarrow{R}] \\
=&\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \Pr[\rho_1 = z_1 \wedge \rho_2 = z_2] f_{\mathbb{I}}(z_1, z_2) g_{\mathbb{I}}(z_1, z_2) dz_1 dz_2 \\
\leq&\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} e^{2\epsilon_1} \Pr[\rho_1 = z_1 + \Delta \wedge \rho_2 = z_2 - \Delta] \\
&f_{\mathbb{I}'}(z_1 + \Delta, z_2 - \Delta) e^{2\epsilon_2} g_{\mathbb{I}'}(z_1 + \Delta, z_2 - \Delta) dz_1 dz_2 \\
=&e^{2\epsilon_1 + 2\epsilon_2} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \Pr[\rho_1 = z_1' \wedge \rho_2 = z_2'] \\
&f_{\mathbb{I}'}(z_1', z_2') g_{\mathbb{I}'}(z_1', z_2') dz_1' dz_2' \\
=&e^{2(\epsilon_1 + \epsilon_2)} \Pr[\mathcal{A}(\mathbb{I}') = \overrightarrow{R}]
\end{aligned}
$$

■

6

Quantifying Microbiome Privacy

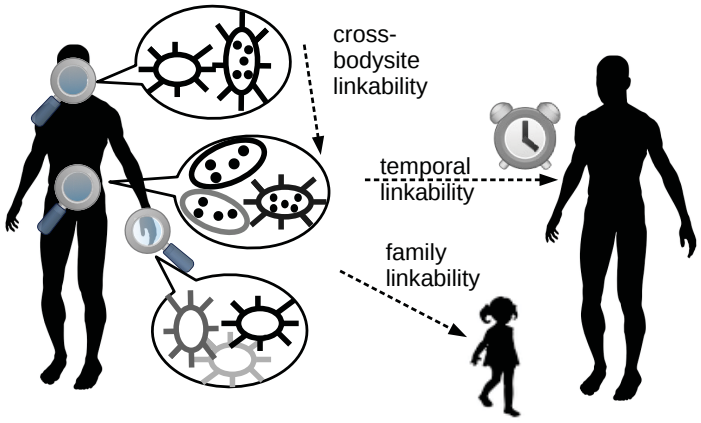


Figure 6.1: Overview of our research questions: Can the microbiome be linked across bodysites, across time and to a cohabiting person?

The human microbiome refers to the genetic material of microorganisms such as bacteria, viruses and fungi, that live on the body surface or within the body, e.g., in the gut or on the skin. There are at least as many microbiome cells as human body cells [128]. The microbiome is composed of a variety of species, they encode 150 times more unique genes than our own genome [116].

Recent studies demonstrate the close relationship between human health and the microbiome [94, 124, 31, 29, 34]

Research question This raises the question whether the composition of different microbiome species can uniquely identify a person. Franzosa et al. [49] started answering this question by studying temporal linkability, i.e., is a person re-identifiable after time. Their conclusion is alarming - yes, we can be re-identified by our microbiome.

Results Given that we all continuously shed parts of our microbiome [97], we look at this privacy issue in more depth. First, we confirm the results of Franzosa et al. [49] with a simpler attack that does require minimal knowledge. The attack achieves an AUC (area under the Receiver Operating Characteristic curve) of 0.75 or higher.

Second, we extend our attack to cross-body linkability. We find that it is possible to link samples from different body sites, e.g., skin and stool, even across time. That means a skin sample that was sampled stealthily from a surface the target touched can be linked to a stool sample, which contains the gut microbiome. Thirdly, we extend our study to linkability across cohabiting family members using an additional data set. We find their common environment, diet and ancestry can be exploited by various linkability attacks that reach an AUC of 0.7 or higher depending on the body site and attack mode. We give a graphical overview of our research questions in Figure 6.1.

Privacy Implications The privacy risks we demonstrate are twofold:

- Stealing a sample and linking it to another sample obtained from clinical context *reveals the metadata* attached to the latter, e.g., a diagnosis. The attacker obtains access to the clinical context either by being member of the medical staff or by launching an attack on the hospital infrastructure.

- Linking samples from sparsely studied body sites (e.g., the skin) to the microbiome that is better understood (e.g., in the gut) allows to *carry out standard tests* on the second site. This supports inference of further private information about the target, for example about diet [94] or diseases and discrimination against the target based on this information.

Methodology All our attacks require minimal biological knowledge and could therefore be carried out by any attacker with a decent level of knowledge about data science and machine learning. Concretely, we first explore a similarity-based approach that uses statistical distances to express microbiome similarity. For two feature vectors encoding the microbiome, we compute the pairwise distance (euclidean and cosine distance). Note that this approach is unsupervised and does not require any training data. Second, we model an attacker with basic knowledge about machine learning that trains random forests. To model an attacker with more advanced knowledge, we design a custom neural network layout inspired by work of Huang et al. [64] and Siamese Networks [17]. The network learns a low-dimensional representation of the microbiome features and then outputs whether these representations are related.

Our findings demonstrate severe privacy risks, therefore, we make a first step towards mitigation of these threats. We explore tree simple defense mechanisms that reduce the granularity of microbiome measurements, but find them not sufficient to protect microbiome privacy. We hope our analysis inspires researchers from biology, medicine and data science to work jointly on new methods for privacy-preserving microbiome data processing.

Organization We introduce the attacks for temporal and cross-bodysite linkability in Section 6.1 together with the empirical evaluation. In Section 6.2 we adopt the attacks for cross-person linkability and show how these perform empirically. Finally, we evaluate simple defense mechanisms in Section 6.3 and conclude in Section 6.4.

6.1 Temporal and Cross-Bodysite Linkability

In this section, we introduce our attack techniques, then explain the application to the temporal and cross-bodysite case and show the resulting performance on an exemplary data set. A summary of the notation can be found in Table 6.1.

6.1.1 Overview

Our attacks are based on three main types of attack models that use simple similarities, random forest classifiers or neural network classifiers. We test these models on several linkability tasks in this and the following sections.

All our models get two different samples, and output a classification label. Formally, let s_i and s_j be two samples that should be linked. Notice that s_i and s_j are vectors. By S_i and S_j we denote sets (databases) of such samples to be linked independently. We implement various functions

$$\mathbb{A} : S_i \times S_j \mapsto \{\top, \bot\} \tag{6.1}$$

where $\top, \bot$ are class labels. The label $\top$ indicates the samples s_i and s_j are classified as being related, while $\bot$ indicates they are classified as not related. If not mentioned otherwise, we do binary classifications. We explain how matching and non-matching samples are drawn below for the individual attack types, because this setup depends on the type of attack.

Table 6.1: Notations

Symbol	explanation
m	the number of microbiome features in one sample
s_i	microbiome sample, represented as a vector
S_i	a database of microbiome samples
T	a threshold (used for distance metrics)
L_1	L_1 distance between two vectors, or respective combination method
L_2	squared L_2 distance between two vectors, or the resp. combination method
avg	combination by elementwise average
$\times$	combination by elementwise multiplication
$c((,\bullet),\bullet)$	combination function, using one of the four methods above
nnS	neural networks first combining the input vectors
nnC	neural networks learning a representation
RF	random forest classifiers
b	a body site to sample from
t	the point in time a sample was taken
f_i	unique family identifier
r_i	unique relationship identifier
p_b^t	a sample taken from an individual at b and t
P_b^t	a set of samples, all taken at b and t
$p_b^{f_i,r_j}$	a sample taken from individual at b, individual belongs to family f_i and has relationship type r_j
$P_b^{\cup f,\cup r}$	set of samples all taken from b, combining all different families and relationship types

6.1.1.1 Cross-Bodysite Linkability Attacks

We test whether it is possible to link samples (from the same point in time) of different body regions as belonging to the same person. This is a privacy risk since some samples are easier to access in a stealthy way than others. Additionally, the microbiome in some body regions is studied more frequently. As an example, the gut microbiome is well studied and linked with diet and immune responses [31, 94], therefore, an attacker might be interested in linking to a stool sample to infer this information about the target. However, it may be much easier to get a skin sample from swiping a surface the target touched, thus linkability of skin to stool samples makes the attack more stealthy. On

the other hand, cross-bodysite linkability may be challenging due to the vastly different conditions like the availability of oxygen in the different body regions which dictate the species that can survive in these regions.

Formally, let p_b^t be a sample from body site b and time t, we denote the set of all samples from different individuals taken at body site b and time t as P_b^t. Our first attack database S_i contains of samples from body site b_1 sampled at time t, i.e., $S_i = P_{b_1}^t$. The second attack database S_j contains samples from a different body site b_2 sampled at the same time t: $S_j = P_{b_2}^t$. We carry out different types of attacks

$$\mathbb{A} : P_{b_1}^t \times P_{b_2}^t \mapsto \{\top, \bot\} \tag{6.2}$$

6.1.1.2 Temporal Linkability Attacks

Changes in the environment, diseases and age influence the microbiome [29]. Additionally, part of the microbiome is most probably shared by related people (be it in a family, friendship, co-workers or randomly meeting strangers) due to continuous shedding of particles [97].

Given these influences, the entire microbiome is most certainly not stable over time. We study whether at least some parts remain stable for re-identification. If we find such linkability over time, the attack surface grows, for example, a recently stolen skin sample could be linked to a stool sample from a year ago when the target visited a hospital, thus the medical condition from a year ago can be inferred. This example also illustrates temporal linkability has to be studied both for the same body site and across different body sites.

For temporal linkability, let $p_b^{t_1}$ be an individual from which a sample is taken at body site b and time t, we denote the set of samples from different individuals taken at body site b and time t as P_b^t. Our first attack database S_i contains samples from a body site b sampled at time t_1: $S_i = P_b^{t_1}$. The second attack database S_j contains samples from the same body site b sampled at a later point in time t_2: $S_j = P_b^{t_2}$. We then carry out different types of attacks

$$\mathbb{A} : P_b^{t_1} \times P_b^{t_2} \mapsto \{\top, \bot\} \tag{6.3}$$

For temporal and cross-bodysite linkability, we also change the body sites: for two different body sites b_1, b_2 and different time points t_1, t_2 we test

$$\mathbb{A} : P_{b_1}^{t_1} \times P_{b_2}^{t_2} \mapsto \{\top, \bot\} \tag{6.4}$$

Finally, we investigate whether having access to multiple body sites increases attack performance. We fit temporal linkability attack functions for two body sites b_1, b_2 separately, i.e., $\mathbb{A} : P_{b_1}^{t_1} \times P_{b_1}^{t_2} \mapsto \{\top, \bot\}$ and $\mathbb{A} : P_{b_2}^{t_1} \times P_{b_2}^{t_2} \mapsto \{\top, \bot\}$. Their probability output is combined by averaging and then converted into the binary label as usual.

6.1.2 Attack Models

We explain three fundamentally different ways to instantiate the attack function $\mathbb{A}$ in this subsection.

6.1.2.1 Similarity Attacks

Our simplest attack model does not involve learning, only a distance metric. The resulting distance is then compared to a threshold T based on which the label is decided. We use the euclidean distance and the cosine distance, other distance metrics have similar or lower performance and are therefore not included. For the cosine similarity, we instantiate the general attack function with

$$\mathbb{A}_{cos}(s_i, s_j) = 1 - \frac{s_i s_j^T}{||s_i||||s_j||} < T_{cos} \tag{6.5}$$

for a suitable threshold T_{cos}. Note that $||\cdot||$ denotes the L_2 norm of the vector and s_j^T the transposed vector.

For the euclidean distance, we instantiate $\mathbb{A}$ with

$$\mathbb{A}_e(s_i, s_j) = \sqrt{(s_i * s_i) - 2(s_i * s_j) + (s_j * s_j)} < T_e \tag{6.6}$$

for a suitable threshold T_e, where $x * y$ denotes the dot product between the vectors x and y.

Testing multiple samples with different thresholds allows to compute the ROC (Receiver Operating Characteristic[1]) curve and the area under this curve, referred to as AUC, expresses the performance as one value, where 0.5 is as bad as random guessing and 1.0 indicates perfect performance. We rely on the AUC to measure performance across different experiments. This method implicitly assumes the attacker to know the best threshold T_{cos} or T_e respectively and is therefore an upper bound for the privacy threat resulting from these methods.

6.1.2.2 Neural Network Attack Classifiers

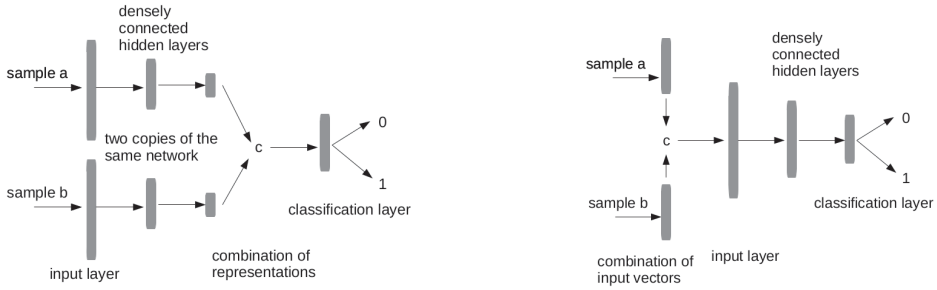
We hypothesize neural networks are better at identifying structure in the microbiome data that simplifies linkability. Since the format of our training and testing data are pairs, we design a custom network layout that first extracts structure from the data and then carries out the target classification. Figure 6.2a shows a schematic overview. The structure extraction was inspired by Huang et al. [64] and Siamese Networks [17]. It consists of two densely connected layers with the second one having half the number of neurons as the first one. Both parts of the pair are fed through the *same* network. Afterwards, we combine the representations using one out of four different methods, finally, the result is classified in one densely connected layer. The four combination methods are average, product, L_1 distance and L_2 distance between each feature in the respective feature vector. For the latter, we remove the square root, i.e., compute the squared euclidean distance. Formally, let $< F_1, F_2, ..., F_m >$, $< G_1, G_2, ...G_m >$ be the two feature vectors of the pair, and let $c(\bullet, \bullet)$ denote our combination methods, then we compute $< c(F_1, G_1), c(F_2, G_2), ..., c(F_m, G_m) >$. For $c(\bullet, \bullet)$, we use:

- L_1 distance: $c(F_i, G_i) = |F_i - G_i|$

[1]The Receiver Operating Characteristic (ROC) curve is plotted using the false-positive rate on the x-axis and the true-positive rate on the y-axis for different thresholds.

- squared L_2 distance: $c(F_i, G_i) = (F_i - G_i)^2$
- average: $c(F_i, G_i) = \frac{(F_i+G_i)}{2}$
- product: $c(F_i, G_i) = F_i \times G_i$

This results in four different attack functions $\mathbb{A}_{nnC,L_1}, \mathbb{A}_{nnC,L_2}, \mathbb{A}_{nnC,avg}, \mathbb{A}_{nnC,\times}$.



(a) Our proposed network structure learning a representation

(b) Standard network structure first combining the inputs

Figure 6.2: Visualization of the two different neural network structures used in this work.

We compare the custom network layout with a standard layout that first applies one of the four different combination methods and then trains two densely connected layers and a classification layer (see Figure 6.2b). Again, the second densely connected layer has half the number of neurons as the first layer. We refer to the resulting attack functions from these neural networks as $\mathbb{A}_{nnS,L_1}, \mathbb{A}_{nnS,L_2}, \mathbb{A}_{nnS,avg}, \mathbb{A}_{nnS,\times}$. In order to decrease the training time, we carry out zero-variance removal[2] for all types of neural networks.

Hyperparameters: We have two types of neural networks:

- The custom neural network has two densely connected layers, feeds the two parts of the samples through the network separately and combines them right before the classification layer. This is implemented by re-using layers for both parts of the samples. The final classification layer uses the sigmoid activation function. Due to the four combination methods there are four variants.
- The standard neural network that first combines the pairs of samples and then fits a two-layer densely connected neural network, where the number of nodes reduces by the factor 2 in each step. After the two densely connected layers, there is a classification layer with sigmoid activation. Again, we have four variants due to the four combination methods.

Each neural network is trained either with ReLU (rectified linear unit) or with ELU (exponential linear unit) as activation function for the inner layers. SGD (stochastic

[2]Zero variance removal means, a feature f_i is excluded if it shows zero variance in the training data, effectively reducing the number of features m. In the corresponding testing data set, we remove this feature as well, independent on whether it has non-zero variance in the test data set or not.

gradient descent) is used for training, and we keep the learning rate at its default of 0.01, binary cross entropy is used as loss function. Thus, we have 16 different neural network classifiers. Initial experiments revealed more layers or fewer nodes in inner layers do not increase the performance consistently and significantly, we present this setting.

6.1.2.3 Random Forest Attack Classifiers

To complete our attack methods, we add a traditional machine learning approach, namely, random forests. We first apply one of the four combination methods $c(\bullet, \bullet)$ mentioned above and then fit random forest classifiers with recursive feature elimination, resulting in the attack functions $\mathbb{A}_{RF,L_1}, \mathbb{A}_{RF,L_2}, \mathbb{A}_{RF,avg}, \mathbb{A}_{RF,\times}$. We carry out recursive feature elimination in order to do automatic feature engineering, which is also done by neural networks internally. As before for the neural network classifiers, we use zero variance removal after applying the combination method but before fitting the classifiers to reduce the overall run time.
Hyperparameters: We use the following grid:

- 10, 20 or 25 samples must be left in each leaf of the individual decision trees to reduce the risk of overfitting
- 50, 80, 90, 95% of the features remain after recursive feature elimination, or no features are removed at all
- 5 features are removed at each step during recursive feature elimination
- the pairs of samples are combined using either L_1 distance or L_2 distance or the average or the product.

We test all 60 different hyperparameter combinations and always fit 100 decision trees as a random forest. Since the combinations at the "edges" of the grid do not show significantly better performance, we expect other hyperparameters outside of our grid not to perform much better.

6.1.2.4 Training/Test Data Split

For all attacks, we use 5-fold cross-validation and report the performance in terms of AUC averaged over all five folds. There is no training involved in the statistical test as it is an unsupervised attack. To make a fair comparison, we sample the same quantity of testing data as other methods' evaluation and measure the average AUC.

Our method requires sample pairs (s_i, s_j) from different points in time or different body sites. Setting up samples that match, i.e., have label $\top$ is straight-forward. For the non-matching samples with label $\bot$, we simply randomly draw a sample that from the remaining sample set $S_j \setminus \{s_j\}$. This method requires no additional metadata about the samples and is therefore suitable for all published data sets.

6.1.3 Data Sets

We use the OTUs tables generated from 16S rRNA measurements using the tool Mothur published by Franzosa et al. [49]. The features are strings indicating the identified species (or unknown) as well as the species' taxonomic hierarchy. This is an example of such a hierarchy: `Bacteria`→`Actinobacteria` →`Actinobacteria`→`Actinomycetales` →`Actinomycetaceae`→`Actinobaculum`→`39`. The last number is an identifier, "Actinobaculum" is the genus of the identified species. It belongs to the family of "Actinomycetaceae", which in turn belongs to the family of "Actinomycetales" and so on. At the coarsest level of hierarchy, this is a bacterium. This representation allows studying coarser grained hierarchical levels as well by summarizing finer-grained levels. For each site, we have two databases from two different visits. The participants have been sampled twice at different points of time (up to 300 days apart) and are linked by their identifier. Each body site was sampled from 30 to 105 participants.

6.1.4 Evaluation

We apply the attack models introduced in subsection 6.1.2 on the data and evaluate cross-bodysite linkability and temporal linkability.

6.1.4.1 Cross-Bodysite Linkability

We needed to preprocess the data because the species reported were (partially) different. This preprocessing was a projection into the union of species, and we filled up the "missing" entries with zeros, the neutral element for the OTUs features. We focus only on the machine learning based attacks, since the similarity metrics did not work well with largely non-overlapping OTUs features.

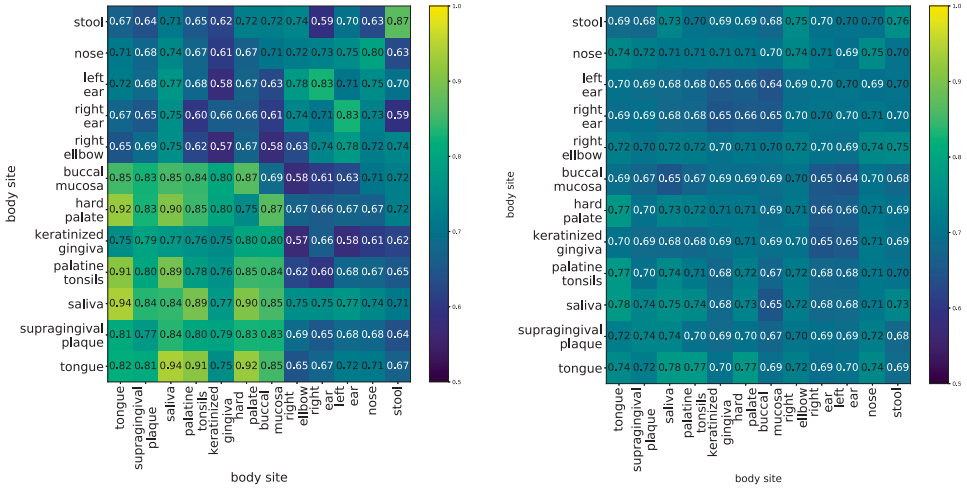
Figure 6.3a and Figure 6.3b visualize the AUC of the best performing classifiers as heat maps, lighter colors imply higher performance.

The diagonal is filled with performances of temporal linkability attacks to allow for a comparison of how much performance was lost when weakening the attacker by the access to different body sites[3].

We observe that the results of the neural networks were all quite similar, independent of body site, and were around 0.7 AUC. Interestingly, cross-bodysite linkability could be easier than temporal linkability of the same body site, see the tongue and saliva samples. On the other hand, stool samples behaved as one might have expected, here, cross-bodysite linkability was harder than temporal linkability.

The picture was different for random forests, where the performance varied between 0.5 AUC up to more than 0.9 AUC for some combinations. We observe a cluster of high performance at the saliva, tongue, hard palate, supragingvial plaque and buccal mucosa sites. All of these sites lie in the mouth, so the performance might be due to the biological link between the microbiome of these sites. Notice that a similar cluster of higher performance can also be seen in the neural networks, however, to a much

[3]Notice that the plots are symmetric, the results from body site A on the x-axis and B on the y-axis are the same as site A on the y-axis and B on the x-axis. This is due to the projection in the shared space that lead to similar shared spaces, thus our combinations of body sites are sets, not ordered pairs.



(a) Best perfomances of cross body site attacks with neural networks.

(b) Best performances of cross body site attacks with random forests.

Figure 6.3: Performances of temporal cross body attacks on all body sites. The diagonal are filled with attacks on the same body site.

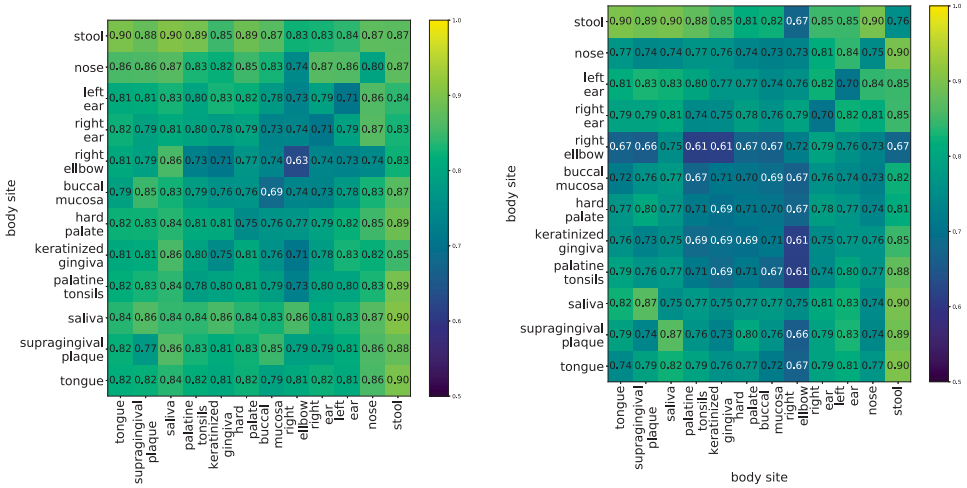
lower degree. Nevertheless, the skin sites right and left ear as well as elbow and nose were hard to match. However, they were easier to match to other skin samples than to samples collected in the mouth.

A possible explanation for the larger performance gaps in random forests compared to neural networks could be that the mouth sites share a larger portion of the microbiome species. Thus, when first combining the features of the two samples as the random forest does, there were more feature dimensions where the combination method did not have a zero for one of the sites. These signals could be directly used in the classification process. However, our custom neural networks got the two samples separately, they seem to be relying on other features so it made almost no difference whether the feature was present in both sites, or just in one.

6.1.4.2 Temporal Linkability

For temporal linkability, we used pairs of samples from the same body site that were taken at different points in time. We tested our statistical and machine-learning based attacks and plotted the resulting AUCs in Figure 6.4 for the different body sites. For the neural networks and random forest classifiers, we show the best performance only.

We observe that some body sites, such as stool or saliva, were easier to link with more than 0.8 AUC, while others were harder to link, and we achieved an AUC of 0.7. That means, privacy is at risk for all body sites. Additionally, we see that simplistic methods such as our distance-based attacks could work well in most cases, even though higher performance could be achieved when using random forest classifiers, or machine learning in general.



(a) random forest performance **(b)** neural network performance

Figure 6.5: Performance of the temporal attacks when using two body sites instead of one, the diagonals show the performance of just one body site.

Using Multiple Body Sites for Temporal Linkability Next, we studied whether using multiple data sources increased the performance of the temporal attack. We tested all combinations of two body sites, Figure 6.5a and Figure 6.5b visualize the results for neural networks and random forests, respectively. On the diagonals, we show the performance of the single body site for comparison. We restricted training and test data set to the individuals who had samples for both body sites. Since some people did not submit samples for all body sites, these data sets were smaller and got too small if we added even more body sites.

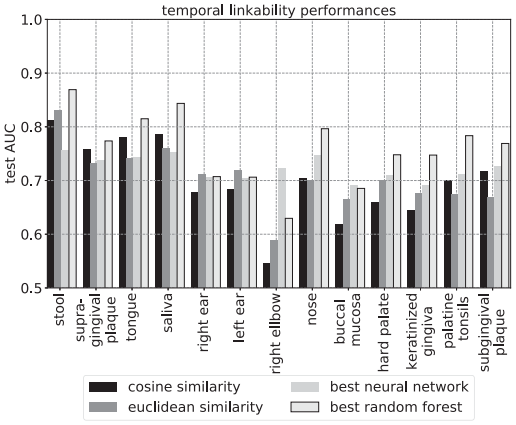
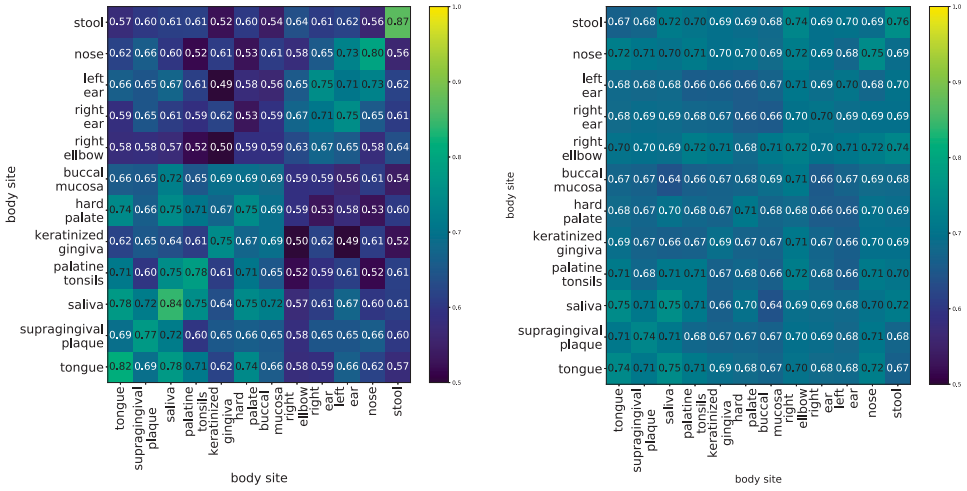


Figure 6.4: Temporal linkability performances of the four different attack methods

We observe for both neural networks and random forests that in general, performance increased slightly. For some combinations, i.e., stool and oral cavity based samples, the performance increased to about 0.9 AUC. This may be related to the common influencing factor of the host diet. In general, random forest classifiers profited more in terms of performance gain than neural networks. We conclude that the privacy risks increased when more data was available.

Cross-Bodysite and Cross-Time Linkability Figure 6.6a and Figure 6.6b show the neural network and random forest classifiers' performances at different points in time. For

the random forest classifiers (see Figure 6.6b) we observe a drastic drop in performance in the mouth-related samples that were easy to link across different body sites at the same time point, but not much better than the other body sites when measured at different time points. For the other areas, we also observe a drop in performance, however, not so pronounced. It seems like those features that make cross-bodysite linkability easy were not stable over time.



(a) random forest performance **(b)** neural network performance

Figure 6.6: Performance of the cross-bodysite attacks at different points in time.

Interestingly, this pattern did not show up in the neural network results. The performance dropped only slightly when using samples from another point in time. It seems like random forest classifiers and neural network classifiers picked up different features from the data, with those used by neural networks being more stable, but may lead to a lower performance.

6.2 Family Linkability

In this section, we extend our attacks to link family members in various ways: either to link two samples as being in a specific relationship, i.e., parent-child relationship, or, more generally, two samples being related, independent of their relationship. Additionally, we also experiment a non-binary classifier that outputs the type of relationship, or none.

6.2.1 Methods

We refer to the general attack model that classifies two people as related irrespective of their relationship type as "same family" attacks. If we take the relationship type into account, we call it "family member" attack. The difference between these two attack models simulates different background knowledge of the attacker, knowing age and gender for example, unlikely pairs can be excluded. Additionally, this fine-grained study

may give us further inside which kind of relationship is easier to detect and therefore poses higher privacy risks. Technically, the difference is only in how the training and testing data is composed, which we now explain formally.

Let f_i be an unique index for a family, r_j a unique index for the relationship type, and b a body site. A sample pair $(p_b^{f_i,r_j}, p_b^{f_i,r_j})$ should be correctly classified as related ($\top$) and a sample pair $(p_b^{f_i,r_j}, p_b^{f_k,r_j}), k \neq i$ should be correctly classified as non-cohabiting ($\bot$).

The difference between the attacks lies in the sets used for training and testing: for family-member linkability, we fix one relationship type r_j, i.e.,

$$\mathbb{A} : P_b^{\cup f,r_j} \times P_b^{\cup f,r_k} \mapsto \{\top, \bot\} \tag{6.7}$$

and test over all families, denoted as $\cup f$. For same-family linkability, we additionally test all relationship types $\cup r$:

$$\mathbb{A} : P_b^{\cup f,\cup r} \times P_b^{\cup f,\cup r} \mapsto \{\top, \bot\} \tag{6.8}$$

We also test cross-bodysite: instead of always using the same body site b in both sets, we use two different body sites b_1, b_2.

Finally, for the classification of relationships, we modify the output format of the attack function, it is not binary anymore, but the set of different relationships $\{r_j\}$ in our data set or no relation at all (denoted as $\bot$):

$$\mathbb{A} : P_b^{\cup f,\cup r} \times P_b^{\cup f,\cup r} \mapsto r \cup \{\bot\} \tag{6.9}$$

For our data set, we have $r \in \{$partner, child-parent, sibling, pet-owner$\}$, but that can easily be adopted if different relationship metadata is available.

We use the same data sets for the first and last approach, i.e., the classification into family relationships has imbalanced classes, half of the samples are not related. We use such an imbalanced training and testing set since it is more realistic compared to a balanced one.

6.2.2 Data Sets

To study family linkability, we use the data published by Song et al. [133]. The data set contains sequenced samples of several families, i.e., couples and their children as well as couples without children, living together in one household. Additionally, if they own a dog, samples from the dog are included as well. Other pets are not studied. Four different body sites are sampled: face, hand, tongue and feces. For each individual, we are usually given additionally metadata about the person's gender, age and diet, which we use in our experiments. Further information, e.g., about weight and drug usage, is included as well, but not used in our experiments.

We download the data set from the QIIME database with the study ID given in the paper [133]. Neither the amount of data (2GB) nor access itself (only a valid mail address) are an obstacle for an attacker. There are about 16000 features for between 16 and 79 people for each body site and relationship combination.

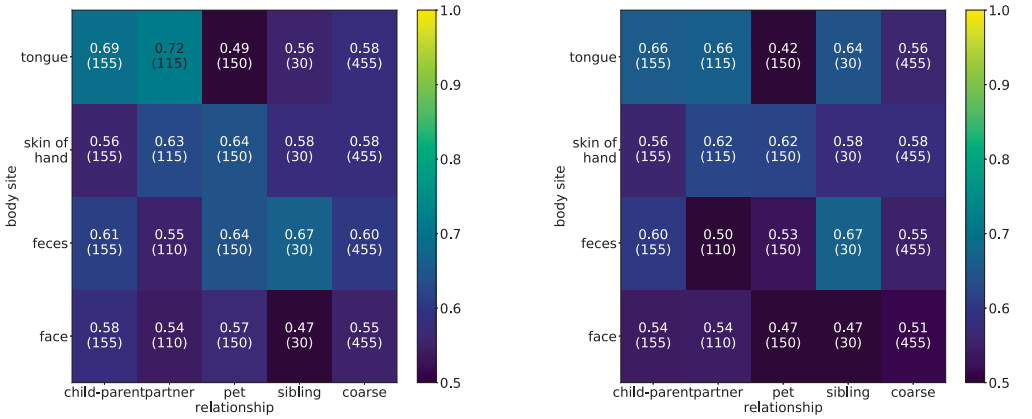
We use the same three techniques introduced before in subsection 6.1.2 to do the family linkability as well. The only difference is in how we set up training and test data. Recall that we need paired samples (s_i, s_j). For the non-matching samples with label $\perp$, we choose the most similar sample $s'_j \neq s_j$ to form a non-matching pair (s_i, s'_j) based on the matching pair. To this end, we define the most similar sample as the one with the same gender and smallest age difference. We believe this setup is most suitable for the family member attack, where the attacker already knows the type of relationship, e.g., a mother and a child, and aims to know whether the adult and the child are related or not. Since the age difference between children and adults is known to influence the gut microbiome [29], we would overestimate the privacy risk if we add easily detectable non-related adult-adult pairs or child-child pairs to the training set. We suspect similar influences of age and gender on other body sites as well, therefore, we make use of available metadata and sample "reasonable" pairs as non-matching members.

6.2.3 Evaluation

In the following, we evaluate our different attack scenarios.

6.2.3.1 Same-Body-Site Inference

First, we inferred whether two samples, taken from the same body site, belong to related people using only simple cosine and euclidean distance metrics between the samples.



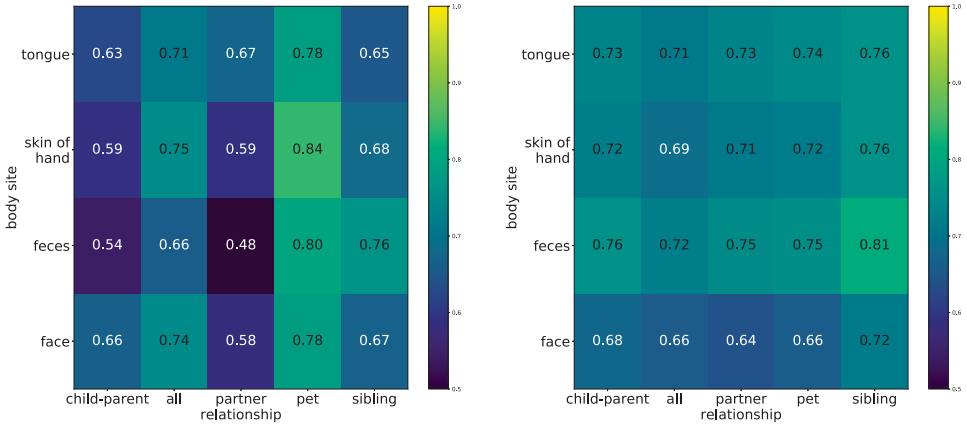
(a) Attack based on cosine distance

(b) Attack based on euclidean distance

Figure 6.7: Performance of family linkability attack with distance metrics.

Figure 6.7a shows the results in terms of AUC of the family inference attacks for cosine distance, the plot for euclidean distance Figure 6.7b is similar. In some cases, cosine distances were better than euclidean distances, in other cases, the euclidean distances. In general, family inference was often possible with an AUC between 0.6 and 0.7, with 0.715 at most. Thus, there is a privacy threat, but it is not immense (yet). Interestingly, it was also possible to infer whether a person is the pet's owner based on the skin of the hand (and the dog's paw, respectively), even though the environment

for the microbiome on skin and in fur should be vastly different. The relatedness may be due to exchange of microbiome over the air, over the floor or due to direct contact such as stroking the pet. Among human family members, the tongue seemed to be the body site with the best chances of success. If the family relationship was not taken into account ("coarse" column in Figure 6.7a), the attack had similar performance.



(a) Best random forest classifier. **(b)** Best neural network classifier.

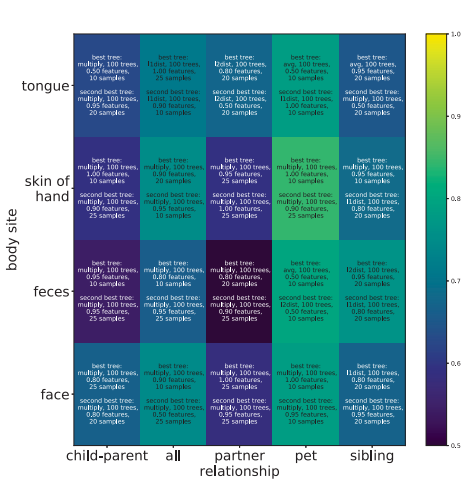
Figure 6.8: Performances of linking family members using the same body sites, the color represents the AUC of the best classifier.

Figure 6.8a shows the performance of the best random forest classifiers. We observe mixed performances, from 0.43 AUC up to 0.84 AUC. For all but two body-site relationship combinations, the AUC was above 0.55. In general, it was easiest to infer whether dogs and their owners belong together and hardest to identify partners. The performance of the coarse-grained relationship inference (displayed as "all" relationship in Figure 6.8a) were well within the bounds of the fine-grained inference results. That means, it was in general not easier or harder when training and testing on data of various relationship forms.

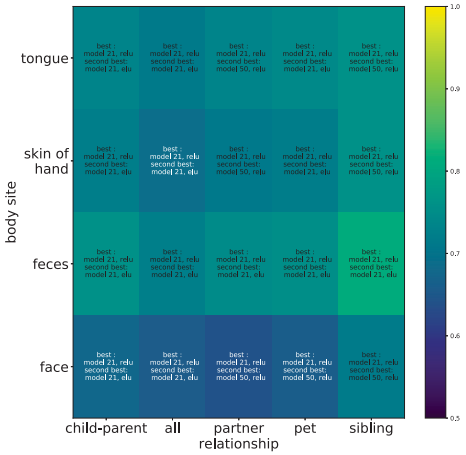
In most cases, the multiplication operation for combining the two parts of the samples and little to no feature elimination showed the best performance. We show the best performing hyperparameters in Figure 6.9.

The performances of the best neural network classifiers for each body site combination ranged from 0.64 to 0.81 as Figure 6.8b shows. The model $\mathbb{A}_{nnC,L_1}$ performed best, which first learned an internal representation with $\frac{1}{4}$th of the original size and then computed the absolute difference between the representations. Activating the neurons with the ReLU function was best in all cases. In general, neural networks were best in classifying the sibling relationship and worst with the partner relationship. Moreover, face samples were hardest to link while stool samples are easiest. Compared to random forest classifiers, there was less variance in the performances.

We hypothesize that neural networks and random forests rely on different features. In general, neural networks show a better performance, however, there were noticeable exceptions like linking pets and their owners based on skin samples, where random



(a) Hyperparameters of the best and second best performing random forest classifier.



(b) Hyperparameters of the best and second best performing neural network classifier.

Figure 6.9: Hyperparameters of the best performing classifiers for family linkability with the same body sites. The colors represent the AUC.

forests clearly outperform neural networks.

6.2.3.2 Multiclass Classification Based on Same Body Sites

We generated the training and test data as before and combined all relationships. The only difference was the generation of labels. For neural networks, we used a one-hot encoding[4]. Random forest classifiers support arbitrary labels, therefore, we used the string representation of all classes directly as label.

We also had to change the neural network classifiers setup slightly. We replaced the last classification layer: It had 5 instead of one node (one for each class) and used softmax instead of sigmoid activation.

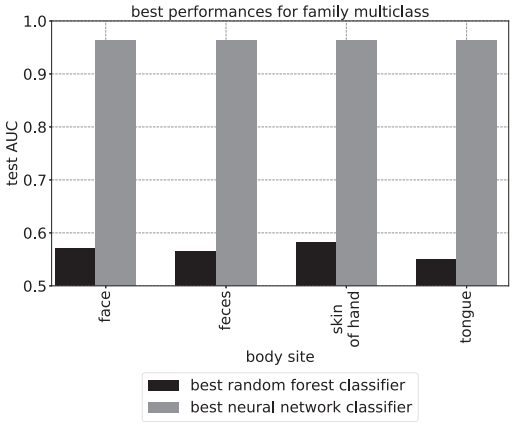


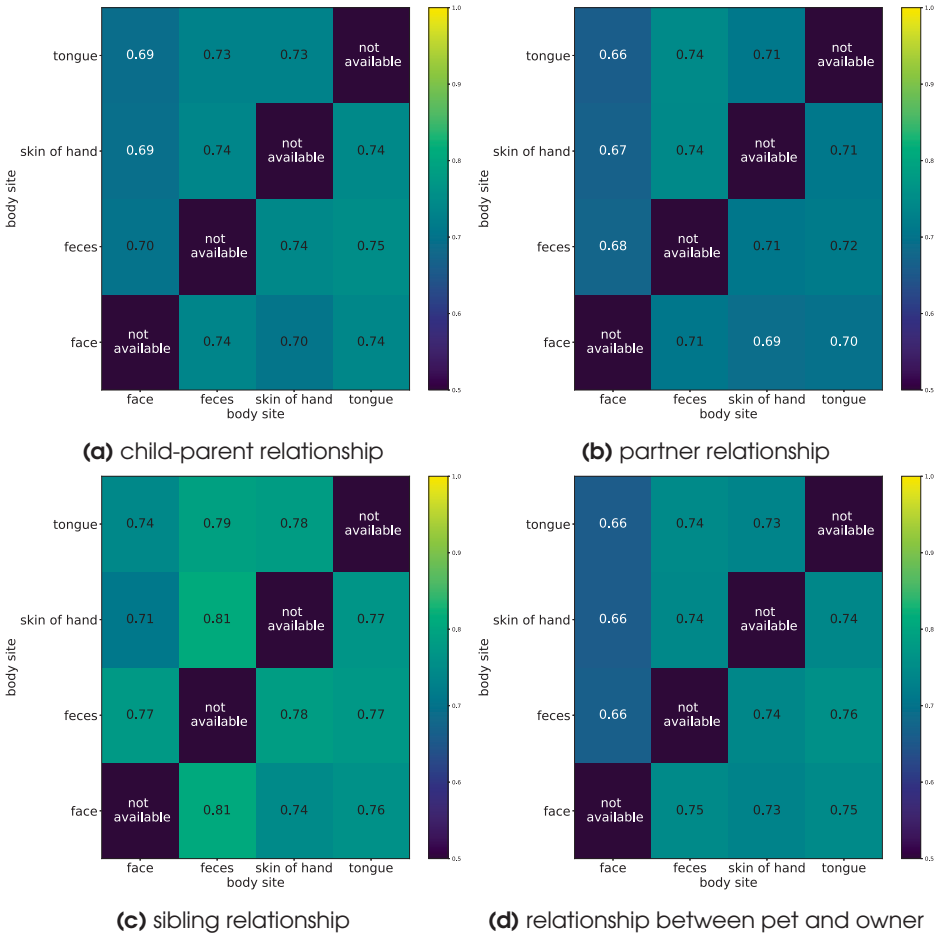
Figure 6.10: Performance of neural networks classifiers and random forest classifiers in multi class mode.

Figure 6.10 shows the best performances of both neural networks and random forest classifiers. In contrast to previous results, the neural networks performed very good with >0.9 accuracy and the random forest classifiers failed with less than

[4]We generated a vector of length five, since we have four relationship classes and one for non-related samples. The vector had an entry of one for the respective class, and zeros elsewhere.

0.6 accuracy. Notice that half of the data was not related, so the trivial classifier that outputs "not related" all the time has an accuracy of 0.5.

6.2.3.3 Cross-Bodysite Inference



(a) child-parent relationship

(b) partner relationship

(c) sibling relationship

(d) relationship between pet and owner

Figure 6.11: Performance of neural networks classifiers on cross-bodysite family linkability for different relationships.

Figure 6.11 visualizes the AUCs of the neural network classifiers, each relationship is shown in one subplot, and we use the x- and y-dimensions for the body site combination. As we have observed previously, linkability was easier for the sibling relationship. In general, face samples were a bit harder, and feces samples were a bit easier to link.

The performance did not differ much from the previous experiments. The results suggest it was not harder when having only samples from different body sites available. In terms of privacy, that is alarming, since some samples are easier to obtain (e.g., skin samples from surfaces touched) and other samples are linked with serious privacy violations (e.g., feces samples that leak information about the person's diet).

As before for the temporal attacks, we did not carry out similarity-based attacks across different body sites.

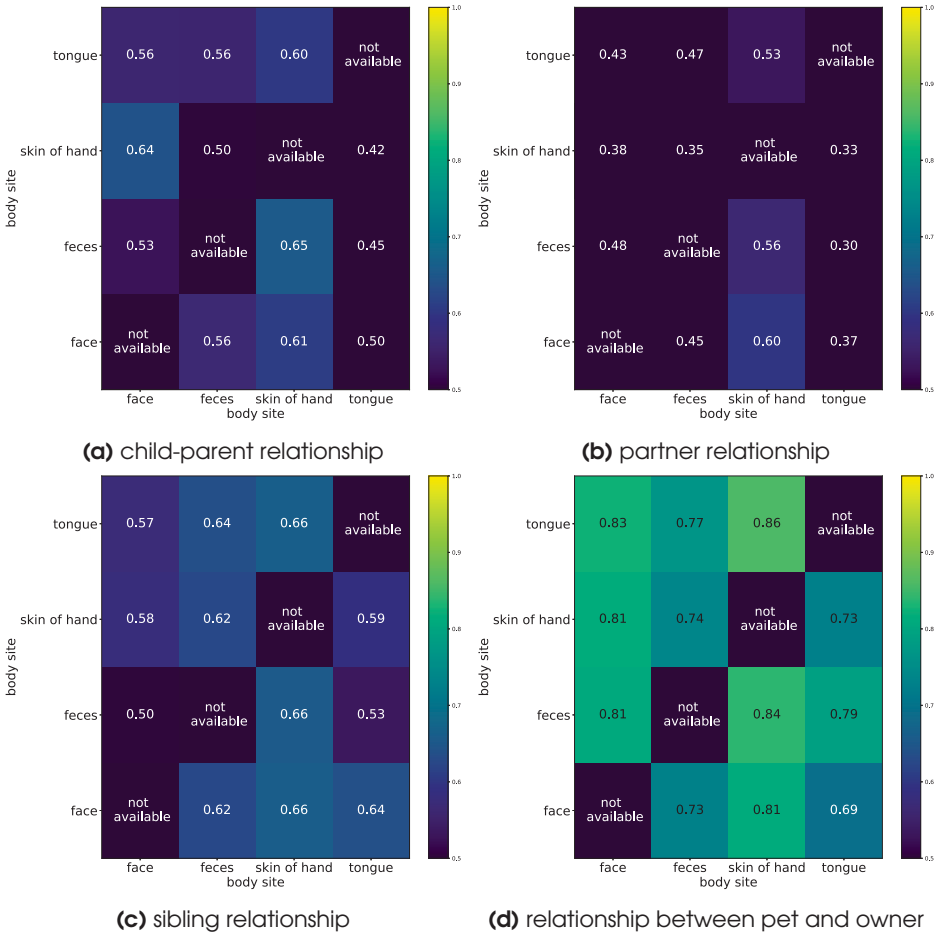


Figure 6.12: Performance of random forest classifiers on cross-bodysite family linkability, for different relationships.

Figure 6.12 visualizes the AUCs of the random forest classifiers similarly as before. Our previous experiments revealed that random forest classifiers were less stable. This was visible for the cross-bodysite case in an even more pronounced way: While it was possible to carry out family linkability with the pet with good performance, often higher than with a neural network, linkability failed in most of the cases for partners and performed poorly for the child-parent relationship and the sibling relationship.

Finally, Figure 6.13 visualizes the AUCs of the classifiers when not differentiating between different types of relationships. Again, we observe the greater stability of neural network classifiers at the cost of peak performance, e.g., face and skin samples could be linked better with random forests than with neural networks.

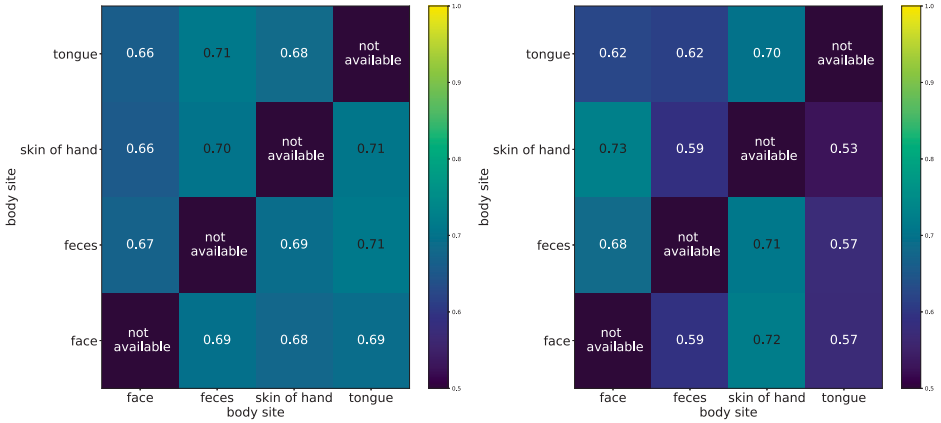


Figure 6.13: Performance of neural network and random forest classifiers on cross-body site family linkability when not differentiating the relationships (coarse grained mode).

6.2.3.4 Inference with Two Body Sites in Parallel

We set up the training and test data as for the cross-bodysite experiments. But instead of learning on one body site and testing on the other, we fitted classifiers for both body sites and combined their predictions by adding the predicted vectors and dividing by two to scale them back to the original range.

Figure 6.14 visualizes the random forest classifiers' performance. Combining two classifiers did not change the performance much, in some cases, it slightly increased, e.g., when using only the skin samples for pet linkability, an AUC of 0.84 was reached, while it increased to 0.85 when using additional face samples. But there were cases where the overall performance drops, e.g., linking siblings reached 0.76 AUC when using feces samples, but feces and face samples together could only be classified with 0.72 AUC.

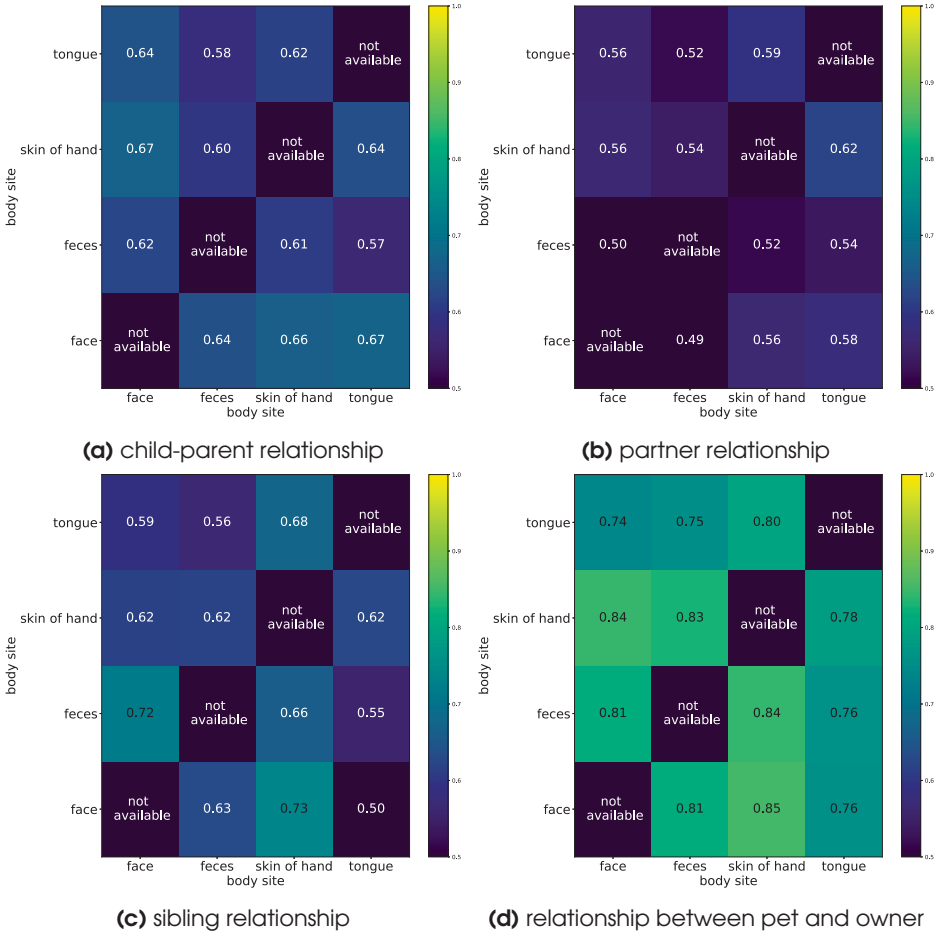
The results for the neural network classifiers, see Figure 6.15 were disappointing. The performance decreased to less than 0.7 AUC, often less than 0.6 AUC, with one exception: sibling linkability with face and skin samples had 0.8 AUC, while using only face samples leads to 0.716 AUC and only skin samples to 0.759 AUC.

6.3 Defense Discussion

In this section, we discuss possible defense mechanisms. We have tried three types of protective mechanisms: generalization, rounding and hiding rare features.

6.3.1 Generalization

The hierarchical structure within the OTUs species has natural "family tree", e.g., `Bacteria→Actinobacteria→Actinobacteria→Actinomycetales →Actinomycetaceae→Actinobaculum→39` where the last number is a unique id, and we can deduce that this OTU denotes a bacteria species of type "Actinobacteria", subtype "Actinobacteria" and so on. We generate coarser-grained features by summing



(a) child-parent relationship

(b) partner relationship

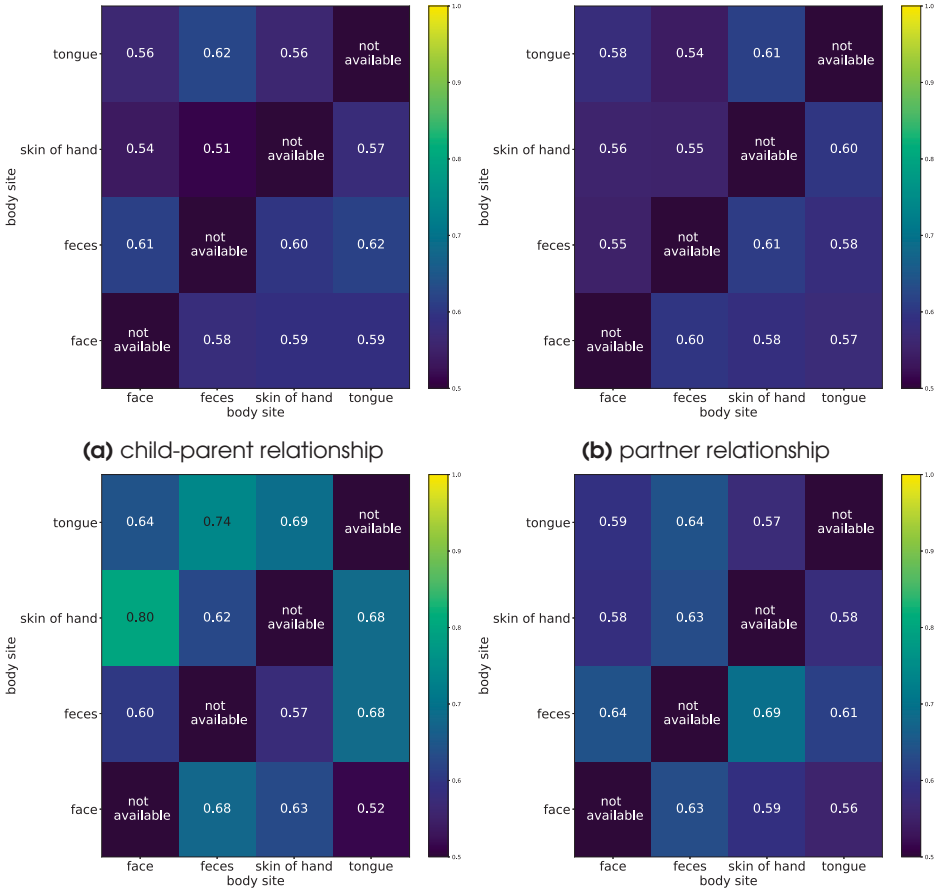
(c) sibling relationship

(d) relationship between pet and owner

Figure 6.14: Performance of random forest classifiers on ensemble family linkability, different plots show different kinds of relationships.

over all OTU with the same prefix, i.e., continuing with the above example, we summed over all OTUs of the form `Bacteria→Actinobacteria→Actinobacteria →Actinomycetales→Actinomycetaceae` to generate one new feature. This feature has now depth 5 in the family tree as opposed to depth 7 in the original family tree. As the first depth only distinguishes between bacteria, viruses and fungi, we stopped generalization at depth two, i.e., the coarsest family of each type of microbiome species.

We repeated the two types of temporal linkability experiments with neural networks and random forests, Figure 6.16 shows the resulting performances. We do not show our results for same body site family linkability here. In most body sites, we observe a drop in performance. However, the resulting performance was not close to 0.5 AUC which would be sufficient for defense, especially for random forests.



(a) child-parent relationship

(b) partner relationship

(c) sibling relationship

(d) relationship between pet and owner

Figure 6.15: Performance of neural network classifiers on ensemble family linkability, different plots show different kinds of relationships.

6.3.2 Rounding

Our data is in range [0, 1] and each number has at most 7 decimals. We round the features rounded to 6,5,4 or 3 decimals. We repeated the same-body site temporal linkability experiment and the family member linking both with neural networks and random forest classifiers.

Figure 6.17 visualizes the family member linking attack performance after rounding to 3 decimals, the strongest "protection" possible. In the labels, we do not only report the AUCs of the respective best classifier, but also the performance decrease (negative number) or increase (positive number) compared to the respective best performance without any rounding. We observe only small drops in performance, in some cases, the performance even increased slightly. The observations on the temporal linkability experiments were similar and are therefore not shown. Therefore, rounding is not a suitable defense either.

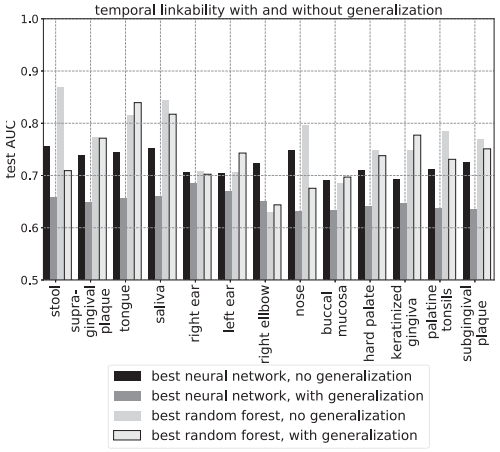


Figure 6.16: Performances of temporal linkability attacks on the same body site. We generalized to depth 2, the strongest generalization possible for the data set.

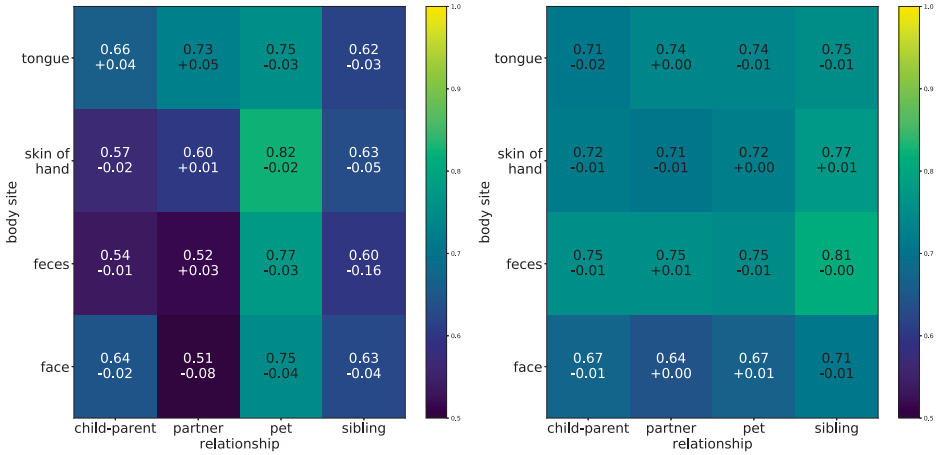
6.3.3 Hiding

Finally, hiding sets rare features that occur in less than 10% or less than 25% of the samples to zero. In case rare features were used by the machine learning models, this method would solve the root cause. Again, we repeated the same-body site temporal linkability experiment and the family member linking with both neural networks and random forest classifiers.

Figure 6.18 visualizes the results of the same-body site temporal linkability experiments, with various protection levels as well as the baseline without hiding on the y-axis. The results for the family member linking were similar and therefore not shown here. We observe only slight drops in performance, and especially for the random forests, the performance sometimes increased slightly. Therefore, also hiding is not a suitable defense mechanism.

6.3.4 Discussion

We tested all three methods with a subset of our attack methods, but as the results are consistent, we expect them to carry over to all different attacks. In general, we could not observe a large drop in performance, on the contrary, in some cases the performance even increased. These first experiments confirmed that it is not easy to defend against microbiome linking attacks in general. We leave more sophisticated methods, such as adding differentially private noise or working with encrypted data only, to future work and to the biomedical researchers that are more familiar with the exact use cases these methods have to be designed for.



(a) Random forest performance after rounding

(b) Neural network performance after rounding

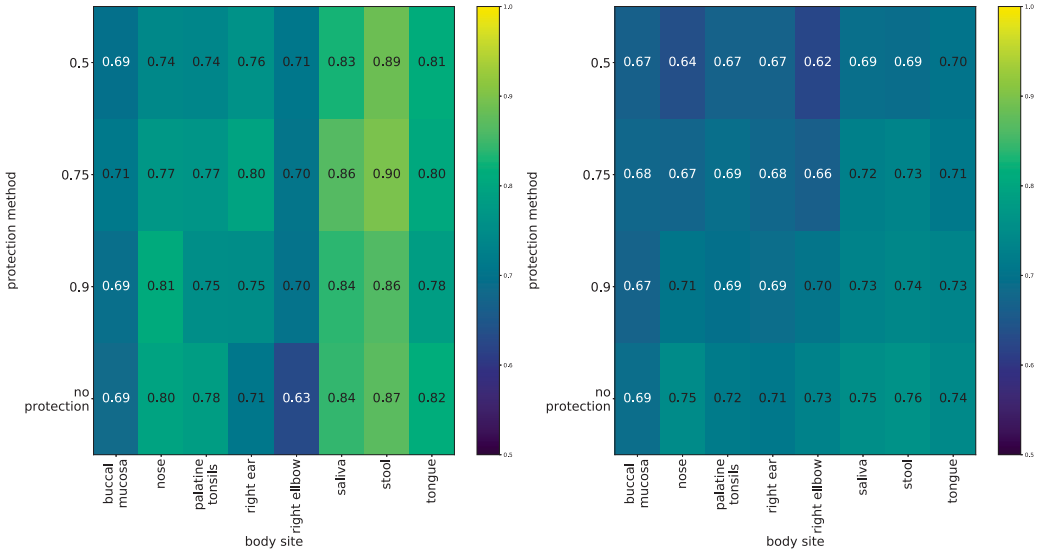
Figure 6.17: Performances of family member linking same body site attacks. We rounded to 3 decimals, the strongest rounding possible for the data set. The second line in each label is the performance decrease (increase) w.r.t. the non-rounded performance.

6.4 Conclusion

We have studied a variety of attack scenarios and attack methods. Table 6.2 summarizes the minimal and maximal AUC over all tested body sites and/or relationships respectively. We conclude that already simplistic methods based on euclidean distance or cosine similarity can work surprisingly well. Using more sophisticated machine-learning models, an AUC of 0.8 is often reached or exceeded, which demonstrates that privacy is at risk. This is also true if the attacker possesses samples from different points in time, or from different body sites. During our attacks, we have not incorporated specific biological knowledge, for example, the "family tree" of the OTUs features and the induced distances. Biomedical studies taking these into account show higher performance in their respective classification task [46, 119]. Therefore, we hypothesize that our analysis rather underestimates the privacy risk, and plan to explore this hypothesis in future work.

A limitation of our study is the amount of data per body site that we had available, which is only up to 100 samples from different people. While this is significantly more than previous small-scale studies with less than 20 people, our results should be re-evaluated as soon as data sets of several hundreds of people become available.

Additionally, we studied three simple defense mechanisms, but none of these methods showed a convincing drop in performance. These findings call for more sophisticated methods, such as adding differentially private noise, working with well-chosen subsets of the data or encrypted data only. We leave it to future work to explore these options.



(a) Random forest performance after hiding, various thresholds

(b) Neural network performance after hiding, various thresholds

Figure 6.18: Performances of same-body site temporal linkability attacks. The y-axis depicts the different protection levels.

attack type	body site(s)		
	same	cross	two
cross body			
same person	-		
neural network	-	0.50 - 0.94	
random forest	-	0.64 - 0.78	
temporal			
statistic	0.55 - 0.81	-	-
neural network	0.70 - 0.75	0.65 - 0.76	
random forest	0.63 - 0.83	0.50 - 0.87	
same family			
statistic	0.52 - 0.60	-	-
neural network	0.66 - 0.69	0.66 - 0.71	-
random forest	0.66 - 0.75	0.50 - 0.73	-
family member			
statistic	0.50 - 0.72	-	-
neural network	0.64 - 0.81	0.66 - 0.81	0.54 - 0.80
random forest	0.50 - 0.84	0.50 - 0.86	0.50 - 0.84
family multiclass			
neural network	0.96 - 0.96	-	-
random forest	0.55 - 0.57	-	-

Table 6.2: AUC ranges of the various attack scenarios and methods, reporting the lowest and highest AUC of the different body sites in the data set.
The "same family" attack returns whether two people are in the same family, regardless of their relationship, the "family member" attack returns whether two people are in a given relationship, and the "family multi class" attack returns the relationship the two people have, or none, if they are unrelated.

7

Privacy-Aware Eye Tracking Using Differential Privacy

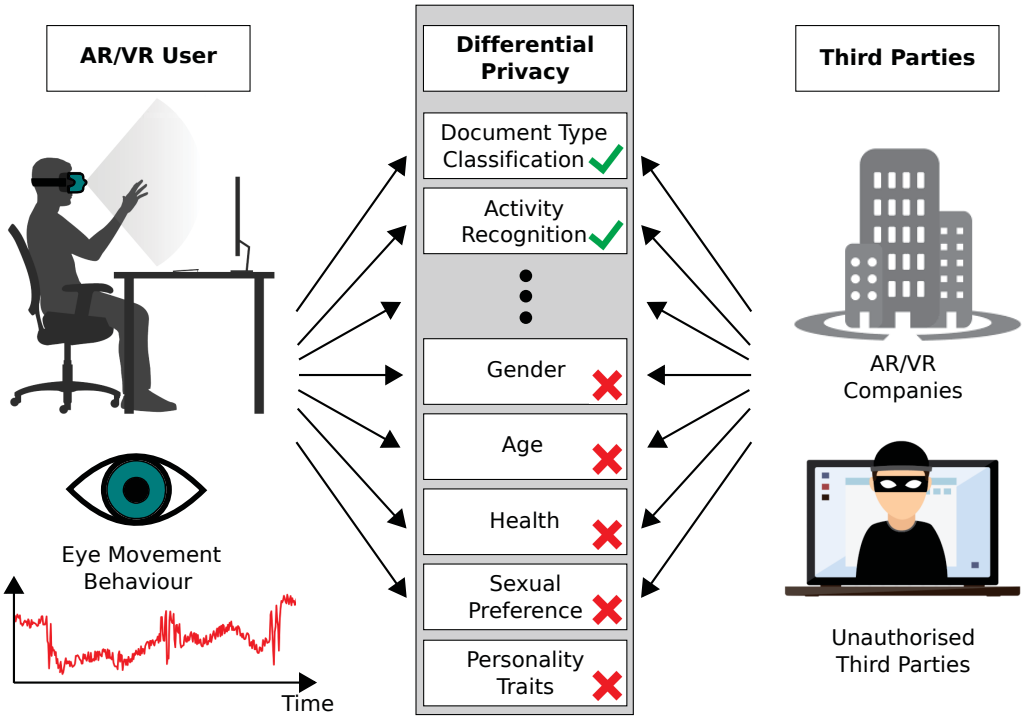


Figure 7.1: Using differential privacy prevents third parties, like companies or hackers, from deriving private attributes from a user's eye movement behaviour while maintaining the data utility for non-private information.

With eye tracking becoming pervasive [21, 142], preserving users' privacy has emerged as an important topic in the eye tracking, eye movement analysis, and gaze interaction research communities. Privacy is particularly important in this context given the rich information content available in human eye movements [22], on one hand, and the rapidly increasing capabilities of interactive systems to sense, analyze, and exploit this information in every day life [57, 137, 149] on the other. The eyes are more privacy-sensitive than other input modalities: They are typically not consciously controlled; they can reveal unique private information, such as personal preferences, goals, or intentions. Moreover, eye movements are difficult to remember, let alone reconstruct in detail, in retrospect, and hence do not easily allow users to "learn from their mistakes", i.e. to reflect on their past and change their future privacy-related behavior.

These unique properties and rapid technological advances call for new research on next-generation eye tracking systems that are *privacy-aware*, i.e. that preserve users' privacy in all interactions they perform with other humans or computing systems in everyday life. However, *privacy-aware eye tracking* remains under-investigated as of yet [88].

There is a lack of eye tracking methods to preserve users' privacy, corresponding systems, and user interfaces that implement (and hence permit the evaluation of) these methods with end users. Our work aims to address these limitations and, as such, make

the first crucial step towards a new generation of eye tracking systems that respect and actively protect private information that can be inferred from the eyes.

We contribute the first method to protect users' privacy in eye tracking based on differential privacy (DP), a well-studied framework in the privacy research community. In a nutshell, DP adds noise to the data to minimize chances to infer privacy-sensitive information or to (re-)identify a user while, at the same time, still allow use of the data for desired applications (the so-called utility task), such as activity recognition or document type classification (see Figure 7.1). We illustrate the use of differential privacy for a sample virtual reality (VR) gaze interface. We opted for a VR interface given that eye tracking will be readily integrated into upcoming VR head-mounted displays, and hence, given the significant and imminent threat potential: Eye movement data may soon be collected at scale on these devices, recorded in the background without the user noticing, or even transferred to hardware manufacturers.

The paper [P1] contains the above mentioned approach for privacy-preserving eye tracking. Additionally, it also contains a large-scale online user study to understand users' privacy concerns. The author of this thesis was not involved in the design and evaluation of the user study and therefore excluded that part of the research paper.

Organization We first introduce the theoretical foundations of privacy-preserving eye tracking in Section 7.1. Section 7.2 describes how the first author of the paper [P1] collected the data. We evaluate the proposed solution in Section 7.3, discuss the findings in Section 7.4 and conclude in Section 7.5.

7.1 Privacy-preserving Eye Tracking

The findings from our survey underline the urgent need to develop *privacy-aware eye tracking systems* – systems that provide a formal guarantee to protect the privacy of their users. Additionally, it is important not to forget that eye movement data typically also serves a desired task – a so-called *utility*. For example, eye movement data may be used in a reading assistant to detect the documents a user is reading [84] or to automatically estimate how many words a user reads per day [82, 83]. Therefore, it is important to ensure that any privacy-preserving method does not render the utility dysfunctional, i.e. that the performance on the utility task will not drop too far. The key challenge can thus be described as *ensuring privacy without impeding utility*.

We assume in the following that multiple users share their eye tracking data in the form of aggregated features. This database can be downloaded both for legitimate use cases as well as for infringing on users' privacy, for example, to train classifiers for various tasks. Therefore, our proposed privacy mechanism is applied prior to the release by a trusted curator.

7.1.1 Threat Models

We have identified two attack vectors on users' privacy in the context of eye tracking that we formalize in two threat models. They differ in their assumption about the attackers' prior knowledge about their target.

Without Prior Knowledge In the first threat model, we assume that an attacker has no prior knowledge about the target and wants to infer a private attribute; we focus on gender in our example study. The attacker can only rely on a training data set from multiple participants different from the target. This data can be gathered by companies or game developers we share our data with in exchange for a specific service. Some users might opt in to share their data with a third party to receive personalized advertisements, or they might create a user account to remove advertisements. These companies with eye tracking data can misuse the data, forward it to third parties or get hacked by external attackers. Another source for attackers to get eye tracking data sets is publicly available data sets generated for research purposes. Concretely, VR glasses are offered in gaming centers and used by multiple visitors, which we refer to as the one-device-multiple-users scenario. An attacker with access to the eye tracking data might be interested in inferring the gender of the current user to show gender-specific advertisements.

With Prior Knowledge The second threat model assumes that the attacker has already gathered prior knowledge about the target. Observing further eye tracking data, the attacker wants to re-identify the target to inspect the target's habits. Concretely, the target might be using different user accounts or even different devices for work and leisure time (a one-user-multiple-devices scenario). We assume the attacker is able to link the target's work data to the target's identity and now wants to identify the target's data from his/her leisure activities. Again, the attacker could be a VR/AR company exploiting their data to check whether a device is only used by one person, or re-identify a user automatically to adapt device settings. Moreover, data could be released intentionally to a third party for money or unintentionally through a hack.

7.1.2 Differential Privacy for Eye Tracking

We propose to mitigate the privacy threats emerging from our two threat models using *differential privacy*,

Next, we formalize the exponential mechanism that is one way to generate differentially private data:

Definition 3 (Exponential Mechanism [38])**.** *The exponential mechanism selects and outputs an element $r \in \mathcal{R}$ in the range of permissible output elements with probability equal to (written: $r \sim$)*

$$r \sim exp(\frac{\epsilon \cdot u(x, r)}{2\Delta_u}) \tag{7.1}$$

where u is a utility function judging the quality of r with respect to the original data element x.

In order to apply the exponential mechanism to our example database of fixation durations, we would first need to define a utility function u and the set of permissible outputs. Valid answers to the query "What are the average fixation rates when reading a text, sampled at 30 second windows?" are vectors of length d containing real-numbered entries; thus, $\mathcal{R} = \mathbb{R}^d_{\geq 0}$. The utility function u is a measure of quality for the output

r with respect to the original data entry x. The exponential mechanism ensures that high-quality outputs r are generated exponentially more often than low-quality r.

Finally, we state one theorem that allows combining several differentially private mechanisms into one.

7.1.3 Implementing Differential Privacy

Our dataset contains data from n participants, which we refer to as $p_1, ..., p_n$. For each participant, we measure m features, $f_1, ..., f_m$ at different points in time. In summary, p_{1,f_7,t_5} denotes the value of the 7th feature at time point 5 of participant 1, and the vector $(p_{1,f_7,t_0}, ..., p_{1,f_7,t_{max,1}})$ contains all measurements of feature 7 for participant 1. Notice that the data entries available may have different lengths, i.e. $t_{max,1}$, the last time point of participant 1, may be different from another participant's last time point, e.g. $t_{max,2}$.

The sensitivity for our mechanism then depends on the range of the features, which is different across our m features. For example, feature f_{15} is the fixation duration in our data set, and it has an estimated range of [0.11, 2.75] seconds, while f_{22}, which describes the pupil diameter size, has an estimated range of [21.9, 133.9] pixels. Therefore, we derive one privacy mechanism $\mathcal{M}_{f_i}$ for each feature separately and use the composition theorem to combine the m mechanisms into our final mechanism. The exponential mechanism requires a utility function u. We choose the L_1 distance for simplicity of the derivation:

$$u(p_{f_i}, r) = \sum_{j=1}^{t_{max,p}} |p_{f_i,j} - r_j| \tag{7.2}$$

According to Definition 2, the sensitivity Δ_{u,f_i} is

$$\Delta_{u,f_i} = \max_{p_{f_i}, q_{f_i}} ||(p_{f_i,t_0}, ..., p_{f_i,t_{max,p}}) - (q_{f_i,t_0}, ..., q_{f_i,t_{max,q}})||_{L_1} \tag{7.3}$$

i.e. the maximal difference between the data vectors of two arbitrary participants p and q for the i-th feature. Next, we unify the length by padding the data vector with the shorter length. Let $tmax$ be the maximal length: $tmax = max(t_{max,p}, t_{max,q})$. Using this and the definition of the L_1 norm:

$$\Delta_{u,f_i} \leq \max_{p_{f_i}, q_{f_i}} \sum_{j=1}^{tmax} |p_{f_i,t_j} - q_{f_i,t_j}| = tmax \cdot \delta_i \tag{7.4}$$

In the last step, we used the fact that we can derive the range δ_i of feature f_i, either estimated from the data or by theoretic constraints.

We rely on the exponential mechanism (see Definition 3) to obtain a vector r that is differentially private for each participant p and feature f_i:

$$r \sim exp(\frac{\epsilon_i u(p_{f_i}, r)}{2\Delta_{u,f_i}}) \overset{\text{Eq. 7.2}}{=} exp(\frac{\epsilon_i \sum_{j=1}^{t_{max,p}} |p_{f_i,j} - r_j|}{2 \cdot tmax \cdot \delta_i}) \tag{7.5}$$

To increase readability, we define $\lambda_i = \frac{\epsilon_i}{2 \cdot tmax \cdot \delta_i}$, which is constant once i and ϵ_i are fixed. We generate such a vector r from the exponential distribution by first sampling a

random scalar y from the exponential distribution with location 0 and scale parameter $\frac{1}{\lambda_i}$. Our differentially private vector r is derived from y as follows:

$$y = exp(\lambda_i \cdot \sum_{j=1}^{t_{max,p}} |p_{f_i,j} - r_j|) \Leftrightarrow \frac{log_e(y)}{\lambda_i} = \sum_{j=1}^{t_{max,p}} |p_{f_i,j} - r_j| \tag{7.6}$$

Selecting $r_j = \pm\frac{log_e(y)}{\lambda_i \times tmax} + p_{f_i,j}$ fulfils the above constraint with randomly sampled sign.

The privacy guarantee of the combined mechanism $\mathcal{M}$ is, by the composition theorem [37], $\sum_{i=1}^{m} \epsilon_i$.

Subsampling In order to achieve a higher privacy guarantee, we propose to subsample the data. Given a window size w, we draw one sample from $(p_{k,i,n\cdot w}, ..., p_{k,i,(n+1)\cdot w})$ for each participant k and feature i independently where $n \in \mathbb{N}$, such that the sampling windows are non-overlapping. Notice that this subsampling approach and the corresponding window size are independent of the feature generation process. This method decreases the sensitivity further by a factor of w: $\Delta_{u,f_i,w} \leq \frac{tmax}{w} \cdot \delta_i$.

7.2 Data Collection

Given the lack of a suitable data set for evaluating privacy-preserving eye tracking using differential privacy, we recorded our own data set. As a utility task, we opted to detect different document types the users read, similar to a reading assistant [84]. Instead of printed documents, participants read in VR, wearing a corresponding headset. The recording of a single participant consists of three separate recording sessions, in which a participant reads one out of three different documents: a comic, online newspaper, or textbook (see Figure 7.2). All documents include a varying proportion of text and images. Each of these documents was about a 10-minute read, depending on a user's reading skill (about 30 minutes in total).

Participants We recruited 20 participants (10 male, 10 female) aged 21 to 45 years through university mailing lists and adverts in different university buildings on campus. Most participants were BSc and MSc students from a large range of subjects (e.g. language science, psychology, business administration, computer science) and different countries (e.g. India, Pakistan, Germany, Italy). All participants had little or no experience, with eye tracking studies and had normal or corrected-to-normal vision (contact lenses).

Apparatus The recording system consisted of a desktop computer running Windows 10, a 24" computer screen, and an Oculus DK2 virtual reality headset connected to the computer via USB. We fitted the headset with a Pupil eye tracking add-on [75] that provides state-of-the-art eye tracking capabilities. To have more flexibility in the applications used by the participants in the study, we opted for the Oculus "Virtual Desktop" that shows arbitrary application windows in the virtual environment. To record a user's eye movement data, we used the capture software provided by Pupil.

(a) Comic **(b)** Newspaper **(c)** Textbook

Figure 7.2: Each participant read three different documents: (a) comic, (b) online newspaper, and (c) textbook.

We recorded a separate video from each eye and each document. Participants used the mouse to start and stop the document interaction. We encouraged participants to read at their usual speed and did not tell them what exactly we were measuring.

Recording Procedure After arriving at the lab, participants were given time to familiarize themselves with the VR system. We showed each participant how to behave in the VR environment, given that most of them had never worn a VR headset before. We did not calibrate the eye tracker but only analyzed users' eye movements from the eye videos post hoc. This was so as not to make participants feel observed, and to be able to record natural eye movement behavior. Before starting the actual recording, we asked participants to sign a consent form. Participants then started to interact with the VR interface, in which they were asked to read three documents floating in front of them (see Figure 7.2). After finishing reading a document, the experimental assistant stopped and saved the recording and asked participants questions on their current level of fatigue, whether they liked and understood the document, and whether they found the document difficult using a 5-point Likert scale (1: Strongly disagree to 5: Strongly agree). Participants were further asked five questions about each document to measure their text understanding. The VR headset was kept on throughout the recording.

After the recording, we asked participants to complete a questionnaire on demographics and any vision impairments. We also assessed their Big Five personality traits [68] using established questionnaires from psychology. In this work we only use the given ground truth information of a user's gender from all collected (private) information, the document type, and IDs we assigned to each participant, respectively.

Eye Movement Feature Extraction We extracted a total of 52 eye movement features, covering fixations, saccades, blinks, and pupil diameter (see Table 1 in the supplementary material). Similar to [25], we also computed wordbook features that encode sequences of n saccades. We extracted these features using a sliding window of 30 seconds (step size of 0.5 seconds).

7.3 Evaluation

The overall goal of our evaluations was to study the effectiveness of the proposed differential privacy method and its potential as a building block for privacy-aware eye tracking. In these evaluations, gaze-based document type classification served as the utility task, while gender prediction exemplified an attacker without prior knowledge about the target, and user re-identification an attacker with prior knowledge.

7.3.1 Classifier Training

For each task, we trained a support vector machine (SVM) classifier with radial basis function (RBF) kernel and bias parameter $C = 1$ on the extracted eye movement features. We opted for an SVM due to the good performance demonstrated in a large body of work for eye-based activity recognition [25, 135]. As the first work of its kind, one goal was to enable readers to compare our results to the state of the art. We standardized the training data (zero mean, unit variance) before training the classifiers; the test data was standardized with the same parameters. Majority voting was used to summarize all classifications from different time points for the respective participant. We randomly sampled training and test sets with an equal distribution of samples for each of the respective classes, i.e. for the three document classes, two gender classes and 20 classes for user identification.

Document Type Classification We trained a multi-class SVM for document type classification and used leave-one-person-out cross-validation, i.e. we trained on the data of 19 participants and tested on the remaining one – iteratively over all combinations – and averaged the performance results in the end. We envision that in the future, only differentially private data will be available; therefore, we applied our privacy-preserving mechanism to the training and test sets. However, currently there is non-noised data available as well: thus, we set up an additional experiment using clean data for training and noised data for testing.

Gender Prediction We trained a binary SVM for gender prediction, using reported demographics as ground truth, and applied it again with a person-independent (leave-one-person-out) cross-validation. Since we are in the *without prior knowledge* threat model, we trained on differentially private and non-noised data to model both the future and current situation, as for document type classification.

User (Re-)Identification We trained a multi-class SVM for user (re-)identification but without a leave-one-person-out evaluation scheme. Instead, we used the first half of the extracted aggregated feature vectors from each document and each participant for training. We tested on the remaining half, since here we are in the *with prior knowledge* threat model. In this scenario, we assumed a powerful attacker that was able to obtain training data from multiple people without noise and was able to map their samples to their identities. The attacker's goal was to re-identify these people when given noised samples without identity labels.

Implementing the Differential Privacy Mechanism We applied the exponential mechanism for each of our $n = 20$ participants and for each of the $m = 52$ features, using a subsampling window size $w = 10$ to reduce sensitivity. In preliminary evaluations, we observed that subsampling alone had no negative effect on the performance of the SVM. The sensitivity for our differentially private mechanism was generated by data-driven constraints: For each feature i, we estimated δ_i by calculating the global minimum min_i and maximum max_i over all participants and time points and set $\delta_i = max_i - min_i$. This way, the sensitivity ensures privacy protection even of outliers. The noise we added in our study can be understood as reading-task-specific noise. For all f_i, we used the same ϵ_i so that the released data of the whole dataset is $\sum_{i=1}^{52} \epsilon_i$-private.

We repeated our experiments five times each and report averaged results to account for random subsampling and noise generation effects. As a performance metric, we report $Accuracy = \frac{TP+TN}{TP+FP+TN+FN}$, where TP, FP, TN, and FN represent sample-based true positive, false positive, true negative, and false negative counts.

7.3.2 Without Prior Knowledge

In Figure 7.3, we first evaluated the gender prediction task, our example for the attacker *without prior knowledge*, trained on differentially private (noised) data (Gender DP) for decreasing ϵ values. As one might expect, decreasing ϵ, and thereby increasing the noise, negatively influences the testing performance when trained on differentially private data with $\epsilon < 30$. For $\epsilon = 15$, the performance almost drops to the chance level of 54% (random guessing in a slightly imbalanced case due to the leave-one-person-out cross-validation). We conclude that on our dataset, privacy of the participants' gender information is preserved for $\epsilon \leq 15$.

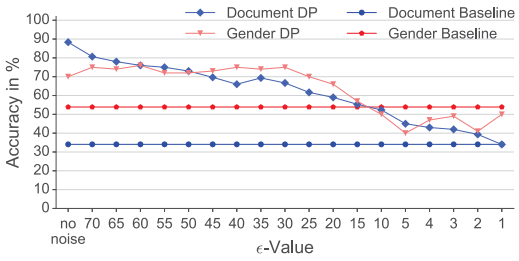
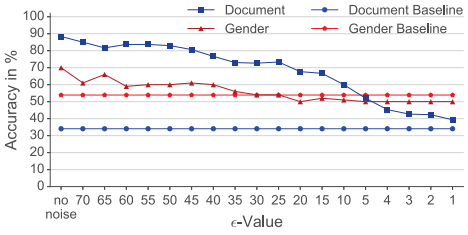


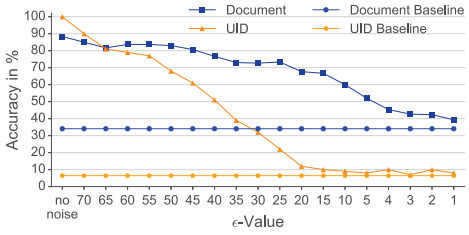
Figure 7.3: Performance for the threat model without prior knowledge trained on differentially private data.

We then evaluated the impact of the noise level for this ϵ-value on utility (see Figure 7.3) using the SVMs trained for document type classification on noised data. As expected, noise negatively influences document type classification as well, but to a lesser extent compared to gender prediction. For privacy preservation, it is sufficient to set $\epsilon = 15$, resulting in an accuracy of about 55% for docu- ment type classification, which is still about 22% over chance level.

So far, we have assumed the SVMs were trained on noised data (Document DP). At present, to the best of our knowledge, all available eye movement data sets are not noised. To study this current situation, we trained both the gender prediction SVM and the document type classification SVM without noise and tested at various noise levels. Figure 7.4a shows the results of this evaluation. As can be seen, also in this scenario, privacy can be preserved: For $\epsilon = 20$, the accuracy of the gender prediction has dropped below chance level, while document type classification is still around 70%. We observed



(a) Performance for the threat model without prior knowledge



(b) Performance for the threat model with prior knowledge

Figure 7.4: Performances when training on clean data with different threat models.

that even $\epsilon = 30$ would already preserve privacy, since training with noise seems to balance out some negative noise effects. Thus, we conclude that for both current and future situations, privacy preservation is possible while preserving most of the utility.

7.3.3 With Prior Knowledge

Finally, we evaluated the *with prior knowledge* threat model, in which we assumed the attacker trained a SVM on the data of multiple users without noise and wanted to re-identify which person a set of noised samples belongs to. Figure 7.4b shows the results of this evaluation for varying ϵ-values. We again added the document type classification performance to be able to judge the effects on utility. As expected, the noise on the test data disturbed the attacker's classification ability: for $\epsilon = 40$, the attacker's accuracy dropped to 50%. For $\epsilon = 15$, it dropped down almost to chance level (6.4%) while the utility preserved an accuracy of about 70%. We conclude that, in this scenario as well, it is possible to preserve a user's privacy with acceptable costs on utility.

7.4 Discussion

7.4.1 Privacy Concerns in Eye Tracking

The ever-increasing availability of eye tracking to end users, e.g. in recent VR/AR headsets, in combination with the rich and sensitive information available in the eyes (e.g. on personality [63]), creates significant challenges for protecting users' privacy. Our large-scale online survey on privacy implications of pervasive eye tracking, the first of its kind, yielded a number of interesting insights on this important, yet so far largely unexplored, topic (see the supplementary material for the full results).

To prevent inference of users' private attributes from eye tracking data, not every data representation is suitable. We recommend using statistical or aggregated feature representations that summarize temporal and appearance statistics of a variety of eye movements, such as fixation, saccades, and blinks. We are the first to propose a practical solution to this challenge by using differential privacy that effectively protects private information, while at the same time maintaining data utility.

7.4.2 Privacy-Preserving Eye Tracking

Informed by our survey results, we presented a privacy-aware eye tracking method in a VR setting. This is the first of its kind to quantitatively evaluate the practicability and effectiveness of privacy-aware eye tracking. For that purpose, we study 1) two realistic threat models (*with* and *without prior knowledge* about the target user), and 2) different scenarios in training with and without clean/non-noised data. We conducted an extensive evaluation on a novel 20-participant data set and 3) demonstrated the effectiveness of the trained threat models on two example privacy-infringing tasks, namely gender inference and user identification.

Applying differential privacy mitigates these privacy threats. The fundamental principle of differential privacy is to apply appropriate noise on the data to deteriorate the accuracy of a privacy-infringing task while maintaining that of a utility task. As such, the level of noise should be smaller than the inter-class difference in the utility task but larger than that of the privacy-infringing task.

We showed in our practical evaluations that users' privacy can be preserved with acceptable accuracy of the utility task by applying differential privacy. This conclusion was consistent across different evaluation paradigms in our example study, which aimed to perform gaze-based document type classification while preserving the privacy of users' gender and identity.

Our mechanism can be used to sanitize data not only before releasing it to the public, but also in VR/AR devices themselves, since it sanitizes one user at a time. Although our example study focuses only on reading, we expect our method to generalize to any other activity involving eye tracking. Due to our data-driven approach, sensitivity can be adapted so that a similar trade-off can be found. Depending on sensitivity and data vector length, the privacy level ϵ of this trade-off may differ from the presented results. Similarly, our study was evaluated on a typical HCI data set size, and we expect our approach to generalize to larger data sets that will be available in the future, given the rapid emergence of VR and eye tracking technology.

To conclude, the proposed method is an effective and low-cost solution to preserve users' privacy while maintaining the utility task performance.

7.5 Conclusion

We presented the first privacy-aware gaze interface that uses differential privacy. We opted for a virtual reality gaze interface, given the significant and imminent threat potential created by upcoming eye tracking technology equipped VR headsets. Our experimental evaluations on a new 20-participant data set demonstrated the effectiveness of the proposed approach to preserve private information while maintaining performance on a utility task – hence, implementing the principle *ensure privacy without impeding utility*.

8

Adversarial Examples for Privacy Protection of Eye Tracking Data

Recent advances in the miniaturization of eye tracking hardware have paved the way for the development of lightweight and fully-integrated head-mounted systems [75, 141]. These allow continuous recording and analysis of eye movements in daily life, potentially over long periods of time. The potential applications for analyzing eye movements pervasively in daily life are far-reaching and include, for example, computational user modeling [23, 126, 27], psychology research [76], human-computer interaction [20] or virtual [48] and augmented reality [60].

However, in addition to these positive use cases, eye movements also contain private information that could be highly valuable for an attacker, such as personality traits [63], mental health issues [150] or recent drug consumption [2]. Additionally, the way we move our eyes reveals person-specific information that allows for individuals to be identified from eye movements alone [66, 79]. This is particularly concerning given that people rarely think about, let alone control, their eye movements consciously in daily life. Combined with the fact that they users not (yet) aware of the rich information content available in eye movements [P1], this urgently calls for research on *privacy-aware eye tracking* – a topic that has only recently started to attract attention in the research community [P1, 136, 89].

We contribute to this emerging body of research by exploring the potential of adversarial examples for privacy-aware eye tracking. Adversarial examples are small perturbations to data that prevent classifier inference while still enabling legitimate usage (the so-called utility). While adversarial examples are well-studied in computer vision, machine learning and security [33, 32] only few works in the privacy community have used adversarial examples as a defense [106, 71, 81] and, to the best of our knowledge, we are the first to explore them for physiological signals and particularly eye tracking data. Recent work has explored differential privacy as a means to preserve privacy in eye tracking [89, P1] but this approach affects the whole signal. In contrast, adversarial perturbations are targeted to specific gaze data characteristics and, as such, promise higher privacy and better utility preservation.

Through evaluations on a recent data set by Steil et al. [P1] for document type recognition from gaze during reading, we demonstrate that adversarial examples can minimize leakage of gender information and re-identification while usefulness of the data is preserved. We study different scenarios ranging from modifications at the raw gaze data level with access to only classification labels to modifications at the feature level with white-box access to the targeted classifier. Classifiers for eye tracking data are often using Support Vector Machines (SVM) [P1, 23, 26, 24], thus we study SVM with radial basis function (RBF) kernels as well.

We additionally propose a privacy-preserving feature selection that enables classifier designers to protect users' privacy. Our feature selection method yields a subset of features that drastically reduces the amount of private information that can be inferred (e.g. an accuracy reduction by 30% for re-identification) while maintaining over 90% of the utility.

Organization We first introduce our threat models in Section 8.1 before explaining how to generate privacy-protective adversarial examples on raw data level in Section 8.2 and on feature level 8.3. We introduce the dataset in Section 8.4 that we use for the evaluation in Section 8.5. In Section 8.6, we dig deeper into how and why these

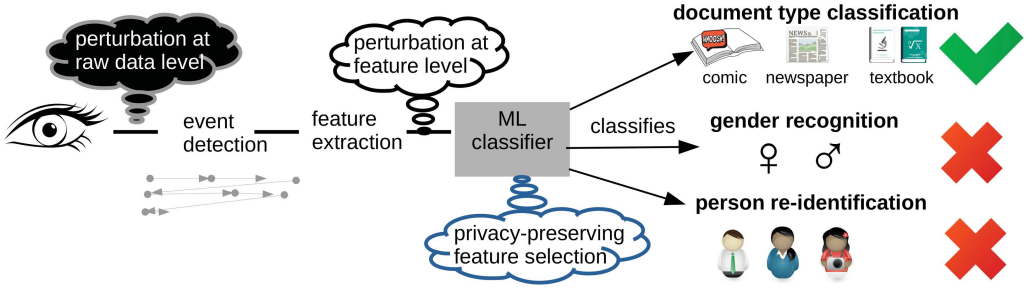


Figure 8.1: Summary of our method: To prevent from inference of private attributes, such as gender and identity, while maintaining document type recognition utility, we propose the user to generate adversarial examples at the raw or feature level without (black-box) or with (white-box) knowledge of the classifier. Additionally, the service provider can select a subset of features that leaks less private information while still performing well for the relevant utility task.

adversarial perturbations are protective and derive a privacy-preserving feature selection method in Section 8.7 before we summarize and discuss our results in Section 8.8.

8.1 Threat Models

We focus on two different scenarios that differ in terms of the user's knowledge and privacy threats. Our first threat model assumes the preprocessing and feature extraction steps are compromised, so the user perturbs the raw data, i.e., the pupil positions detected by the manufacturer's software. The user obtains only the classification label for the perturbed data. We refer to this scenario as "black-box" in the following.

In a setting where eye tracking data is streamed to a service provider for classification, the service provider probably chooses a compressed representation of the data to be transmitted. One natural way for such a compression would be to carry out the preprocessing and feature extraction steps in the user's device. Therefore, we assume in our second scenario the user is able to modify the feature representation before it is sent to the classification service. The data may be eavesdropped or the classification service itself extracts privacy-sensitive attributes. We additionally assume the user can train a classification model for these attributes, thus we refer to this scenario as "white-box" in the following. Notice that eye tracking data is often freely shared for research purposes. Since the classifiers are often SVM with RBF kernels [P1, 23, 26, 24], they are straight-forward to be trained without dedicated hardware or knowledge about machine learning beyond applying using existing libraries such as sklearn [19].

8.2 Adversarial Examples at Raw Data Level

In order to relax the assumption on the attacker's knowledge about the preprocessing pipeline and classifier, we adopt the HopSkipJump attack by Chen et al. [28]. It is a black-box attack that queries the classifier and obtains the labels.

Figure 8.2: Snippets of the different document types shown to the participants

In our case, the user submits raw data to the black-box oracle, which internally runs the preprocessing pipeline and queries the SVM classifier. The first step in HopSkipJump is to find an initial sample that is misclassified, and as a second step, this misclassified example is changed to be closer to the benign sample by estimating the classifier's decision boundary. HopSkipJump was developed for image perturbation and samples a random image initially. In contrast, for eye tracking data we observed that such a randomly generated sample was usually not misclassified. Therefore, we replaced this random initialization step with an iterative process that perturbs individual data points in the sample. First, a set of data points is selected and both measurements of x and y eye positions are replaced with zero. We call this kind of point a "zero point", and it occurs naturally in the data if the tracker is not able to measure the eye's position correctly, e.g., because the participant is blinking or due to motion blur during saccades. We allow for up to 25% of data points that are not "zero points" to be perturbed this way, which is usually sufficient to find an initial misclassification. HopSkipJump then further optimizes this initial sample as described by the authors in [28] and implemented in ART [104]. Notice that HopSkipJump optimizes only those feature dimensions where the initial sample deviates from the benign sample, so our initialization also decreases the search space for the adversarial perturbations made, since at most 25% of the data points are changed initially.

8.3 Adversarial Examples at Feature Level

Our feature level attack is a white-box attack where we assume not only the preprocessing pipeline is known, but also the targeted SVM. This allows us to use the gradient information for a gradient-based attack. We opted for the Fast Gradient Sign Method (FGSM) [55] due to its applicability to SVM, which we explain in the following. Goodfellow et al. propose to add the following perturbation η:

$$\eta = \epsilon \text{sign}(\nabla_x J(\theta, x, y)) \tag{8.1}$$

for a model parameterized by θ with linearized cost function J for the sample x and its label y. The gradient of the cost function J with respect to the input x is denoted by ∇_x and ϵ is the parameter that determines the size of the perturbation.

In our case of SVM (support vector machine) with RBF (radial basis function) kernel, $\nabla_x J(\theta, x, y)$ equals to:

feature type	by	explanation	number of features
Fixation	[23]	rate, computed over pupil positions within one fixation: mean, max, variance of duration mean of mean, variance of variance	8
Saccades	[23]	rate, ratio of (small/large/right/left) saccades, mean, max and variance of amplitudes	12
Combined	[23]	ratio saccades to fixations	1
Wordbooks	[23]	for n-grams of length up to four (including): number of non-zero entries, maximum and minimum of entries	24
Blinks	[23]	rate, mean and variance of blink duration	3
Pupil Diameter	[23]	mean of mean and variance of variance during fixations	4
reading features	[84]	euclidean distance between 5% and 95% quantile of fixation coordinates, the slope of saccade directions using linear regression over fixations	2

Table 8.1: We extracted 54 eye movement features to describe a user's eye movement behavior.

$$\sum_i \alpha_i y_i - 2\gamma e^{(-\gamma||x-sv_i||_2}(x - sv_i) \tag{8.2}$$

according to Biggio et al. [14] where sv_i denotes the i-th support vector with label y_i and weight parameter α_i and γ is the RBF-kernel parameter that describes the locality of the kernel and $||\cdot||_2$ denotes the L_2 norm.

We use the implementation of FGSM in the Adversarial Robustness Toolbox [104], which comes in two modes: The one-step procedure described in the original paper by Goodvellow et al. [55] and the "minimal" mode that step wise increases the amount of allowed perturbation until a maximal perturbation ε_{max} is reached or the sample is misclassified. Since the "minimal" mode allows more fine-grained control over the amount of perturbation, it leads to better empirical results in our experiments.

8.4 Dataset

We used the publicly available data set by Steil et al. [P1]. It contains recordings of 20 participants (10 female and 10 male) that read three different types of documents (comic, newspaper, textbook) in virtual reality, Figure 8.2 displays snippets of these document types. Notice that the data set size is common for eye-tracking research data sets.

Eye movements were recorded by the Oculus DK2 Virtual Reality headset using the Pupil eye-tracking software [75] at a sampling rate of 30Hz. The three different

documents for reading were displayed using the program "Virtual Desktop" by Oculus for Windows. Users were encouraged to read with their usual reading speed and could read the documents in their preferred order. Notice that our eye tracking measurements only contain information about the viewing behavior, but not about what exactly the users looked at. This is because calibration, i.e., the mapping of the eye tracker's coordinate system to the coordinate system of the stimulus, was skipped to preserve users' privacy.

The software provided by Pupil was used to extract x- and y-positions of the eye, pupil diameter and confidence level as described in chapter 2. The event detection first extracts fixations with a dispersion-based algorithm, i.e., if the x- and y-positions of the eyes are within a radius of 0.02 for at least 0.1 seconds. Saccades can occur between fixations and are categorized by their direction as in Figure 2.2. Finally, blinks are recognized if x- and y-position of the eye is zero for at least 0.1 seconds.

The feature extraction processes the series of events from a sliding window into 52 high-level features as described by Bulling et al. [23] and two additional reading-specific features from Kunze et al. [84], see Table 8.1 for details.

8.5 Evaluation of Privacy Protection Using Adversarial Examples

At the example of re-identification and gender classification as privacy-sensitive tasks, we evaluate the influence of adversarial examples. Our goal was not only to protect the users' privacy by evading re-identification or gender classifiers, but also to preserve utility of the data. As a proxy for utility, we used document type classification while reading. We start by explaining how the classifiers for re-identification, gender and document classification are trained and then detail the feature level and raw data level attacks.

8.5.1 Classifier Training

We trained SVM with RBF (radial basis function) kernels with leave-one-person-out cross-validation using the implementation provided by sklearn [19]. We left the regularization parameter C at its default 1.0 and the kernel coefficient γ for the locality of the RBF kernel at its default $1/\text{number of features}$. The data was mean-centered and scaled to unit variance. First, we found the optimal window size for document recognition by training the SVMs on the window sizes 15, 20, 25, 30, 45, 60, 90, 120 seconds and testing on 200 randomly drawn samples per document type from the respective testing participant. Second, we used the same 200 samples per participant and document type to train one SVM with the optimal window size to find the best subset of features for document recognition with forward feature selection. Later, we reported test accuracy on the remaining samples from the testing participant averaged over all participants. For gender prediction, we used the same leave-one-person-out cross-validation scheme and the same samples for testing and trained SVMs for each document type separately. For re-identification, we also trained for each document type separately. However, we used one multi-class one-vs-rest SVM for the 20 participants using 200 samples from the

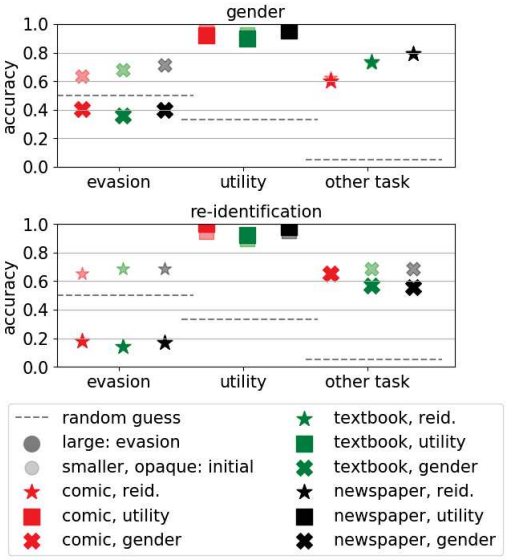


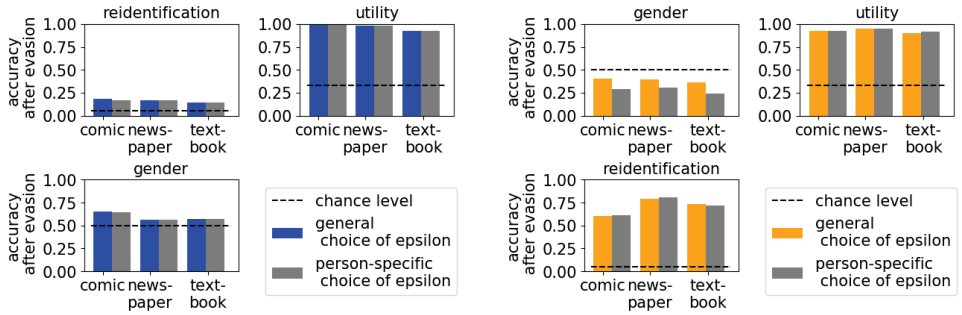
Figure 8.3: Accuracy before and after evasion, the opaque smaller markers show the initial accuracy and the larger markers after evasion, different marker shapes show different classifiers and the colors different documents. The dashed gray line symbolizes the random guess accuracy. The maximal perturbation ε_{max} was chosen independently of the target participant.

beginning of the respective recording and the remaining samples were used for testing. This corresponds to the assumption that the attacker owns data from the target and wants to re-identify the target based on a similar stimulus.

8.5.2 Adversarial Examples at Feature Level

We first explain the concrete setting of FGSM's hyperparameters, and then evaluate their impact in different attack scenarios. FGSM in minimal mode comes with three parameters: ε_{max}, ε_s and the choice between targeted and untargeted attacks. For the maximal allowed perturbation ε_{max}, we tried values between 0.1 and 2.5 with increases of 0.1. The perturbation per step ε_s was fixed to 0.1. The resulting adversarial examples were evaluated on the targeted classifier before and after the perturbation. Additionally, we also evaluated them on the utility task (document type recognition) and the other privacy-sensitive task. We focused on untargeted attacks to allow for a fair comparison between the binary gender classification problem and the multi-class re-identification problem.

We found that for different participants, different levels of perturbation are best. Therefore, we evaluated our results once assuming the user is able to infer the optimal ε_{max} for the participant (person-specific mode) and once assuming the user is not and uses the ε_{max} that performs best on average over all participants (general mode). Notice that the user chooses ε_{max} only based on the attack success, independent of the rate of utility preservation or the performance of any other task.



(a) Evasion of the re-identification classifier. **(b)** Evasion of the gender classifier.

Figure 8.4: Influence of person-specific or general choice of ε_{max} on the accuracy of the different tasks.

General Choice of ε_{max}: Figure 8.3 shows that the evasion is successful: the gender prediction accuracy drops below the chance level of 0.5 (dashed line) for all three documents. For re-identification, the chance level itself is low ($1/20$) due to the 20 different participants), and the performance drops not to chance level, but below 0.2. Thus, adversarial examples provide a considerable amount of privacy protection. Nevertheless, utility was preserved both when gender or re-identification models are targeted, as Figure 8.3 shows. Especially for the re-identification case, utility even increased from 0.946 to 0.999 (comic), 0.951 to 0.981 (newspaper) and 0.8942 to 0.9220 (textbook). We shed light on the reasons for this in subsection 8.6.2.

Additionally, we also observed the effect of adversarial examples on the orthogonal privacy-sensitive task. Adversarial examples that successfully hide the participants' gender still leak their identity. However, adversarial examples that hide the identity also offer some degree of protection against gender inference since the accuracy dropped, even though the attack was not targeted at gender.

Person-specific Choice of ε_{max}: Comparing the general choice of the maximal perturbation ε_{max} with the person-specific choice, we observed a lower accuracy after evasion for gender in Figure 8.4b, and similar accuracy for re-identification, see Figure 8.4a. The former might be explained by different degree of similarity between the participants and slightly different classification boundaries of the 20 different classifiers.

Using a Subset of Features: So far, we have used a feature set of 54 features, however, not all of them are relevant for the benign task of document type recognition. In order to model an honest service provider that optimizes the feature set for higher accuracy, we evaluated the task with a subset of features. Figure 8.5 displays the results of gender recognition evasion. The task got harder for the newspaper document, and we see that the initial accuracy is higher for all three document types. Nevertheless, utility could still be preserved. The accuracy of re-identification increased due to the feature subset, however, the adversarial perturbation decreased the accuracy to a larger extend compared to the full feature set. If re-identification itself was targeted, the task did not get harder even though the initial accuracy was higher. The decrease in accuracy for gender recognition was similar to the full feature set and the impact of the evasion on utility stayed small.

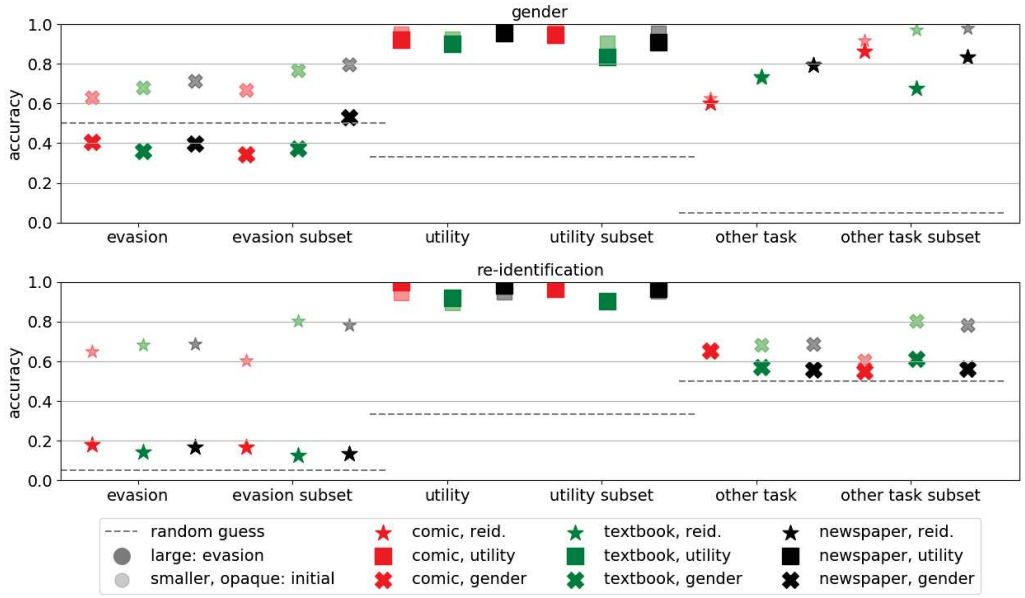


Figure 8.5: We display the results of the white-box evasion on the subset of features that is best for document recognition and compare it to the full feature set. The opaque smaller markers show the initial accuracy and the larger markers after evasion, different marker shapes show different classifiers and the colors different documents. The dashed gray line symbolizes the random guess accuracy.

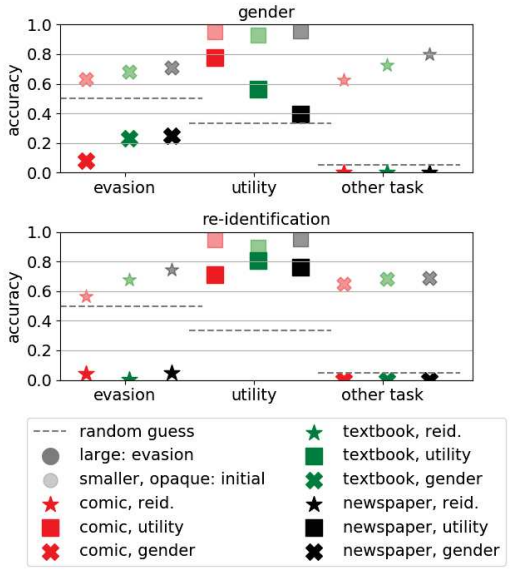


Figure 8.6: Accuracy before and after evasion in black-box model, the opaque smaller markers show the initial accuracy and the larger markers after evasion, different marker shapes show different classifiers and the colors different documents. The dashed gray line symbolizes the random guess accuracy.

8.5.3 Adversarial Examples at Raw Data Level

In Figure 8.6 we evaluate the black-box attack on gender (upper part) and re-identification (lower part) similarly to before. In case of the gender evasion, the accuracy was well below chance level and even smaller than the previously shown white-box attacks with FGSM. It is also possible to evade the re-identification classifier. However, the utility was severely impacted and reached only 0.4 for newspaper (recall that chance level is $\frac{1}{3}$ due to the three document types), and was highest for comic with almost 0.77. We investigated this further in subsection 8.6.1.

When we evaded the re-identification classifier, the utility was preserved better and stayed above 0.7. Again, the other privacy-intrusive task, here gender classification, also suffered from the perturbations and it was almost impossible to predict gender correctly even though that was not the goal of the evasion.

We observe that evading the classifiers on raw data in the black-box model is less targeted to the privacy-sensitive task compared to the white-box model, but it was still possible. Our findings suggest that such an evasion might even be beneficial for other privacy-sensitive attributes without explicitly optimizing for them.

Using a Subset of Features: The evasion on raw data still worked if just a subset of features was used, see Figure 8.7. For re-identification, the difference in accuracies between full feature set and feature subset is small. Gender evasion, however, is harder on the feature subset compared to the full set of features. It might be necessary to induce larger changes on the raw data to influence the remaining features. The effect is clearly more pronounced for gender, where the utility changed drastically, from almost 0.8

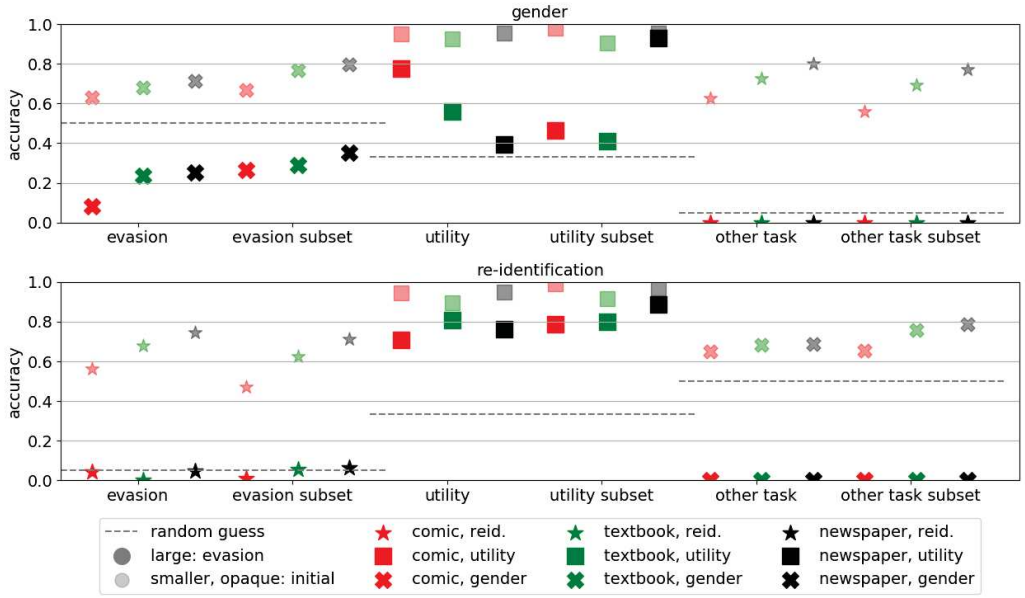


Figure 8.7: We display the results of the black-box evasion on the subset of features that is best for document recognition and compare it to the full feature set. The opaque smaller markers show the initial accuracy and the larger markers after evasion, different marker shapes show different classifiers and the colors different documents. The dashed gray line symbolizes the random guess accuracy.

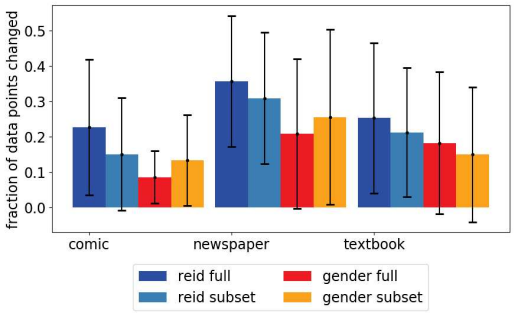


Figure 8.8: Average fraction of data points labeled differently due to adversarial perturbation on raw data level in different settings. The error bars show the standard deviation among the 20 participants.

down to less than 0.5 accuracy for comic, while the utility for newspaper increased from 0.4 to 0.9 accuracy, and for textbook from 0.4 to 0.55. Notice that the subset of features was optimized for document recognition, thus a higher initial accuracy is expected. The utility changes from full feature set to subset were small for the re-identification attack, where we again observe slight increase for newspaper, very little changes for textbook, but an increase for comic.

8.6 Understanding the Impact of Adversarial Examples

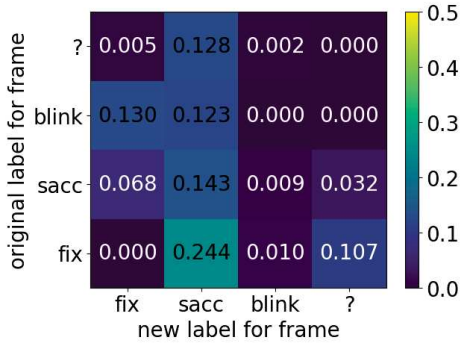
The impact of the different methods for adversarial perturbation on utility are diverse and unexpected in some cases, thus we study the effects in more depth to explain them better.

8.6.1 Impact of Adversarial Perturbations at Raw Data Level

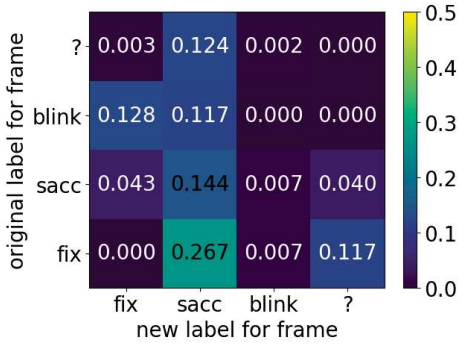
We observed the changes in the first processing step, event detection, when submitting a perturbed data sample instead of the original raw eye tracking data.

We measured the fraction of data points that were labeled differently in the adversarially crafted sample compared to the original sample by the event detection step, Figure 8.8 visualizes the results for all events. The perturbed samples changed between 8% and over 35% of the data points' labels. Recall that we set a bound of up to 25% of changed data points in the initializer, however, the event detection considered multiple data points at once which explains the additional changes. The high standard deviation indicates large differences between the participants, nevertheless, we can say that comic samples were changed less overall than newspaper and textbook samples. There tended to be fewer changes necessary in order to evade the gender classifier compared to re-identification. Also, if just a subset of features was used by the classifier, fewer changes were necessary on average, with perturbations against the gender classifier on newspaper being an exception.

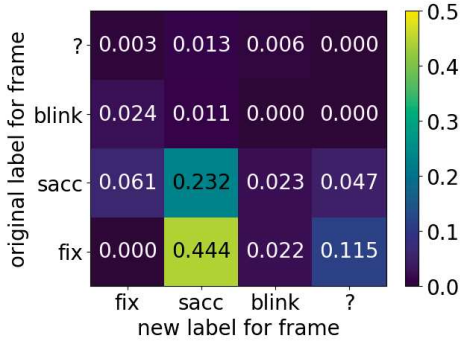
As a second step, we studied the changes in more detail and plot confusion maps in Figure 8.9. We observed that mostly fixations turn into saccades. This is not surprising



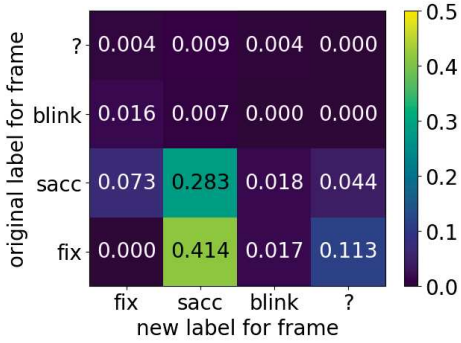
(a) Evading the gender classifier on the full feature set



(b) Evading the gender classifier on the subset of features



(c) Evading the re-identification classifier on the full feature set



(d) Evading the re-identification classifier on the subset of features

Figure 8.9: Fraction of data points that were changed due to perturbations at raw data level, the y-axis denotes the original label and the x-axis the label after perturbation. The question mark indicates no event could be recognized. The results are averaged over samples and participants. We only show the newspaper document here since the other documents show similar patterns.

given that our event detection algorithm required points to be within a small radius to be recognized as a fixation, and changes to individual points are likely to exceed this radius. For re-identification evasion, we observe that saccade directions change, i.e., the data point was still recognized as a saccade, but the saccade is mapped to a different "letter" (see Figure 2.2) that encodes the direction. The changes induced to evade the gender classifier however were wider spread and include all event types being mistaken for saccades, even though to a smaller degree than fixations. These wide spread changes may be an explanation why gender evasion had such a negative impact on utility. The patterns described here are visible for all document types, thus we visualize only newspaper here.

We conclude that black-box evasion with HopSkipJump induce a considerable amount of "fake" saccades that result in wide-spread changes in the feature extraction step, especially for gender evasion.

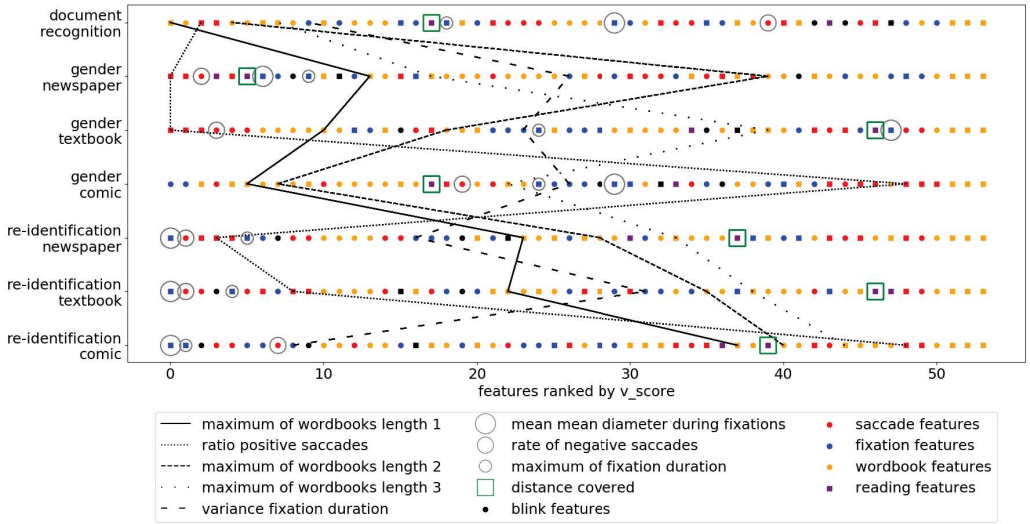


Figure 8.10: Features ranked by v_score for the different tasks, different marker colors represent different types of features. We highlight the distinguishing features with lines and further features of interest with squares and circles. We observe different features being distinguishing for the different tasks.

8.6.2 Impact of Adversarial Perturbations at Feature Level

We expected utility to drop as privacy increased, thus we were surprised by the result that white-box attacks with FGSM preserved utility when defending against gender classification and even increased utility for re-identification defense. In order to understand the reasons for this positive surprise, we first used clustering to identify a small set of features to focus on. Then, we studied the changes in data distributions due to adversarial perturbations on these features.

Clustering each feature separately allows to judge the information of each feature separately, as opposed to classifiers such as random forests or linear SVMs that output feature importance. Additionally, findings from clustering are more likely to generalize to different classifiers. Thus, we used sklearn's[19] implementation of k-means clustering on the complete data set in each feature dimension separately. We set the number of clusters to three, i.e., the number of different documents in our data set, and checked the performance of clustering with respect to the document ground truth labels using the v_score. The higher the v_score, the better the data separates in this feature dimension, thus, the feature may be more important for the classifier. Formally, the v_score is defined by Rosenberg et al. [121] as the harmonic mean between homogeneity and completeness. Homogeneity is maximized if all clusters contain only members of the same class, and completeness is maximized if all data points of the same class are in the same cluster, so the v_score is given by:

$$v\_score = \frac{(1+\beta) * homogeneity * completeness}{\beta * homogeneity + completeness} \tag{8.3}$$

where β can be used to give stronger weight to homogeneity or completeness, but we

feature name	v_score
maximum of wordbooks length 1	0.401
difference between maximum and minimum wordbooks length 1	0.401
ratio positive saccades	0.373
ratio negative saccades	0.373
maximum of wordbooks length 2	0.366
difference between maximum and minimum wordbooks length 2	0.366
variance wordbooks length 1	0.349
maximum of wordbooks length 3	0.338
difference between maximum and minimum wordbooks length 3	0.338
variance fixation duration	0.327

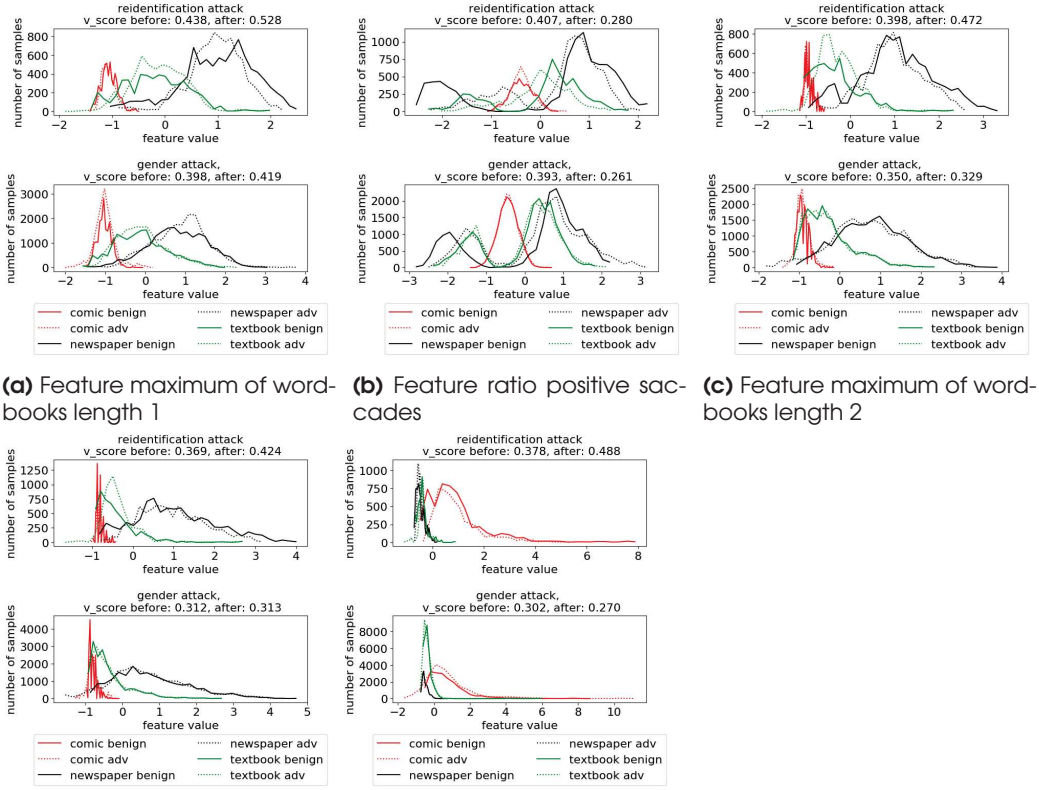
Table 8.2: The top 10 v_scores of clustering feature-wise, we used the document labels as ground truth. Gray marked features are excluded from later analysis because of similar behavior to the other features.

leave it at the default 1.0.

Distinguishing Features: The features with the ten highest v_scores are listed in Table 8.2. We notice that several features are very similar: The maximum of the wordbooks with length one is equal to the difference of maximum and minimum since the minimum is always zero on on our data, similarly for wordbooks of length 2 and length 3. The ratio of positive saccades is dual to the ratio of negative saccades. Finally, the variance of wordbooks with length one is distributed very similar to the maximum when plotting a histogram of the data. Thus, we reached a set of five features that we refer to as "distinguishing features" in the following and marked black in Table 8.2. These features lead to a testing accuracy of 0.8 for document recognition when training the same SVM with RBF kernel on the same samples, thus, these features are not only distinguishing in general, but also for our specific classifier.

We applied clustering to each document type individually and computed v_scores with respect to gender labels (and set k=2) and identity labels (and set k=20). In Figure 8.10 we show how distinguishing our five features are for the gender task and the re-identification task. We see that these distinguishing features had low rank for identity, and the features distinguishing identities had low rank for document recognition. The effect was smaller for gender recognition, which also had larger differences in feature ranks between newspaper and textbook on the one hand and comic on the other hand. We conclude that especially the re-identification task and the document type recognition task rely on different feature dimensions. Thus, if FGSM changes mostly those dimensions relevant for re-identification or gender respectively, the distinguishing features for document recognition can still be used by the respective classifier.

Effect of Perturbations: Next, we looked at the adversarial perturbations in these five distinguishing feature dimensions. Figure 8.11a to Figure 8.11e compare the histograms



(a) Feature maximum of wordbooks length 1

(b) Feature ratio positive saccades

(c) Feature maximum of wordbooks length 2

(d) Feature maximum of wordbooks length 3

(e) Feature variance fixation duration

Figure 8.11: Histograms with 30 bins of the five most distinguishing features on the original (solid lines) and perturbed (dotted lines) data for gender and re-identification evasion respectively, colors denote different ground truth document labels.

of these adversarial examples with the benign data they were generated from, in both cases, we used the best values of ε_{max} as found above in the general setting. Additionally, we also applied k-means clustering on the adversarial examples to observe the change in v_score with respect to document type labels due to the perturbations. Notice that the benign distributions are not identical across gender and re-identification because test data for re-identification was sampled temporally while the data for gender classification was sampled uniformly. The data was mean-centered and scaled to unit variance with respect to the document classification training data before adversarial examples were generated.

The most distinguishing feature is wordbooks of length 1, Figure 8.11a shows the shifts in distributions due to the perturbations. We observe for re-identification that the perturbation modified newspaper samples that were between -1 and 0, which resulted in better distinction to textbook samples. This is also visible in the increase of the v_score. The effect was smaller for gender and barely visible from the histogram. In general, the reading of a comic seems to trigger more diverse saccade patterns that lead

to smaller minima in each type of wordbook features.

Also, the second most distinguishing feature captures saccades, namely, positive saccades going upwards or to the right. Figure 8.11b shows that both textbook and newspaper distributions had two peaks, while comic had only one peak in between. The re-identification evasion shifted the lower parts of both newspaper and textbook distributions towards higher values resulting in less variance and a smaller v_score. This effect was smaller for gender.

The next two distinguishing features are both wordbook features of lengths two and three, i.e., n-grams of two and three saccades. Newspaper tends to have larger values, comic smaller values and much less variance. In both cases the re-identification attack shifted the lower part of the newspaper distribution making the distributions easier to distinguish by eye and by v_score, see Figure 8.11c, Figure 8.11d.

Finally, the variance in fixation duration was in general higher for comic, and very similar for newspaper and textbook. Again, Figure 8.11e shows that the re-identification evasion helped to distinguish the document classes by moving the comic samples of lower variance, this effect was lower for the gender evasion.

Conclusion: For the most distinguishing features, the perturbation for the re-identification classifier decreased person-specific information in the samples and simultaneously increased the difference between the documents. This showcases that privacy and utility are not necessarily competing goals. For the eye tracking user, this is a positive finding because utility increases while privacy is protected better.

8.7 Privacy-Preserving Feature Selection

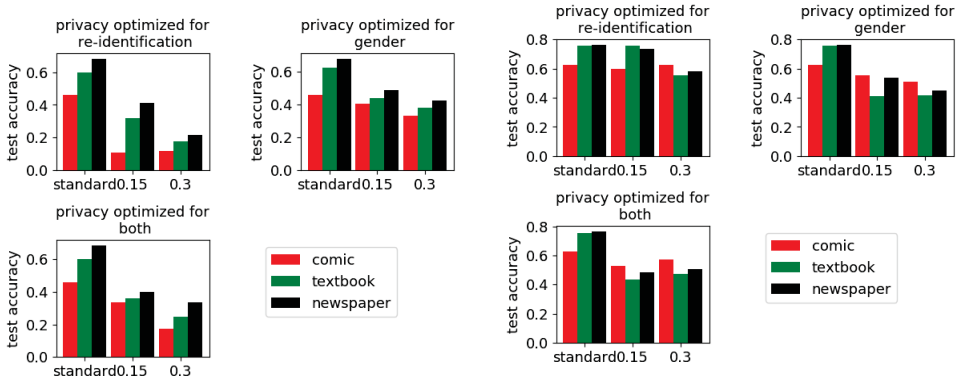
In the previous section, we observed that the features that were important for our three classification tasks differ. Therefore, we study in this section whether it is possible to select a feature set that enables the utility task while minimizing the amount of private information contained in the features. If such a privacy-preserving feature set can be found, classification providers can use it to reduce the amount of private information that can be eavesdropped while the data is streamed for classification.

Method: We extended the classical forward feature selection, but notice that our findings generalize to other forms of feature selection. Forward feature selection iteratively grows the set of selected features by adding all possible features f, one at a time, evaluating the accuracy $\alpha_{u,f}$, and growing the feature set by the feature that lead to the largest accuracy. We extended this scoring function $s(f)$ by fitting a second privacy classifier obtaining $\alpha_{p,f}$ and then using a privacy weight w to tune the trade-off between utility and privacy:

$$s(f) = \alpha_{u,f} * (1 - w) + (1 - \alpha_{p,f}) * w \tag{8.4}$$

For w, we try 0.1, 0.15, 0.2, 0.25, 0.3, 0.5 and 0.6 and estimate $\alpha_{p,f}$ based on re-identification accuracy, gender classification accuracy, or both. In the latter case, we distributed w evenly to both tasks.

Evaluation: We tested the resulting feature set and the validation accuracy shows that at 0.15 there was already some privacy improvement with almost no influence on utility. Increasing w beyond 0.3 did not improve privacy further. Thus, we picked these two trade-off values and evaluated the feature set on the test set.



(a) Impact on re-identification accuracy

(b) Impact on gender classification accuracy

Figure 8.13: Test accuracy of privacy-sensitive classifications when selecting features with different privacy notions.

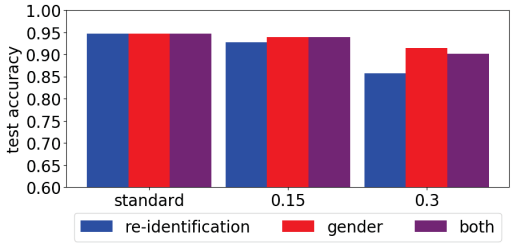


Figure 8.12: Impact of different privacy-preserving feature selection methods on document type classification accuracy.

In Figure 8.12 we show the document type classification accuracy for the two selected w in comparison to the standard feature selection for the three scenarios. The accuracy stays above 0.85 and suffered less from optimizing for gender privacy than for re-identification privacy. We observe in Figure 8.13a and Figure 8.13b that the highest privacy protection was achieved when the feature selection was targeted to the privacy task, this is expected. Nevertheless, also optimizing for the other privacy-relevant task decreased the privacy leakage to some degree. Thus, there is hope that such privacy-preserving feature selection decreases leakage even for tasks that are overlooked at the point of feature selection. When we optimized for both privacy tasks, we could not achieve both good gender and re-identification privacy at the same time.

In order to find an explanation for that, we study the selected features, shown in Table 8.3 for the higher privacy weight 0.3. Taking re-identification into account lead to a very small feature set and the increase of w from 0.15 to 0.3 yielded a proper subset of the previously selected features. Notice that we selected features for different values of w independently, but for the same training data. The feature set was larger when taking gender classification into account instead, however, still smaller than the set returned by standard feature selection. Moreover, the set for w=0.3 was not a subset. With one exception, all features selected with re-identification privacy were also part of feature set selected by standard feature selection, while we obtain more features that would be rejected when taking gender classification into account.

There are only two features that are selected in all three cases, the maximum of

wordbooks with length 1 and the distances covered during the time window. Figure 8.10 highlights that the maximum of wordbooks with length 1 was most distinguishing for document type recognition, and much less distinguishing for gender classification and re-identification. The distance covered, marked in green squares, was almost irrelevant to re-identification, more distinguishing for document type recognition and had very mixed leakage of gender information across the different document types. Thus, it was not surprising that these two features are often a good trade-off between privacy and utility. We conclude that features which increase document classification accuracy without leaking too much private attributes heavily depends on the privacy notion.

8.8 Discussion and Conclusion

Our goal was to use adversarial perturbations to reduce the leakage of private information in eye tracking data. Our experiments in section 8.5 show that users can protect their privacy with and without knowledge about the classifier inferring their private attributes as the success rates are comparable. However, knowledge about the classifier and preprocessing pipeline lead to smaller perturbations that maintain or even increase utility. Moreover, the white-box attack has low run time and could be used in an online setting where data is streamed to the service provider. Thus, our evaluations demonstrate that adversarial examples can be used as an ad hoc defense against leakage of private information from eye tracking signals.

We observed that privacy and utility were not conflicting goals, and that an increase of the utility is even possible while the private information is hidden due to the adversarial perturbation. Thus, we studied the phenomenon in depth in subsection 8.6.2 using classifier-agnostic clustering and found that features important for the utility task were positively influenced by adversarial examples against re-identification. Future work can explore the usage of adversarial examples for data augmentation and data quality improvement. Given the small number of participants in typical eye-tracking studies, removal of person-specific characteristics with adversarial perturbations might be an important preprocessing step for eye tracking research.

Clustering also demonstrated that the features important for utility and for distinguishing participants are vastly different. That finding inspired the privacy-preserving feature selection to trade-off privacy and utility already in the design state of the classifier. This method might be adopted in the following scenario: A service provider wants to hide protect the intellectual property in the classifier for eye-tracking data, thus requires data to be uploaded to the service provider's servers. The selection of a feature subset that minimizes the amount of private information in transfer drastically reduces the privacy risks and might be crucial for privacy-conscious users.

We hope this combination of adversarial examples, clustering and feature selection fuels further research towards understanding eye tracking data and its privacy risks and leads to more robust and more secure data processing pipelines.

standard feature selection	re-id. privacy	gender privacy	gender and re-id. privacy
rate of large saccades	-	-	-
rate of positive saccades	-	-	-
ratio small / saccades	-	●	●
ratio positive saccades	-	-	-
ratio negative saccades	-	-	-
mean saccade amplitude	-	-	-
var saccade amplitude	-	●	-
max fixation duration	-	●	-
var var y	-	-	-
mean mean diameter during fixations	-	-	-
var mean diameter during fixations	-	-	-
mean var diameter during fixations	-	●	-
var blink duration	-	●	●
non zero entries WB1	-	-	-
max WB1	●	●	●
min WB1	●	-	●
var WB1	-	-	-
min WB2	●	-	●
var WB2	-	-	-
max WB3	-	-	-
min WB3	●	-	●
var WB3	-	●	●
non zero entries WB4	-	-	-
max WB4	-	●	-
min WB4	●	-	●
var WB4	-	-	●
distance covered	●	●	●
slopes saccade direction	-	-	-

Table 8.3: Features selected by different feature selection techniques. • denotes the feature is selected for privacy weights w 0.3. Features that we identified before as distinguishing features are marked in italic font, and features not part of the standard feature selection set are gray.

9
Conclusion

The contributions of this thesis are two-fold: first, to uncover privacy risks of a variety of biomedical data types, and second, to develop tools for privacy-preserving data sharing.

As opposed to genome data, other biomedical data such as methylation, microbiome and eye tracking data have not been studied extensively for their privacy risks. However, because these data types are influenced more by the environment [70, 127, 86, 63], they leak personal information about the target such as the current health status or the current environmental settings. On the other hand, data sharing is crucial to advance biomedical research given that measuring the data of a sufficiently large population is complex and costly. Therefore, we propose methods for privacy-preserving data sharing that protect the data donors' privacy and maintain utility for research purposes.

In the first work presented in Chapter 4, we have thoroughly analyzed whether and to what extent DNA methylation databases are prone to membership inference attacks. In particular, we have considered two attacker models: one assuming the adversary to know her victim's methylation profile, and the second assuming the adversary to know only her victim's genotype. For both settings, we have studied traditional statistical attacks based on the L_1 distance and on the likelihood-ratio test. Additionally, we have proposed a new machine-learning attack that is able to exploit the fact that not all methylation data are equally informative for membership inference. In this setting, we have further studied data transferability, i.e., to which extent learning features from a data set different from the targeted data set influences the attack results. For the genome-based inference of membership, we have specifically designed the LLR attack to capture the probabilistic dependencies between the two types of data, and have identified a sufficient statistic for this attack.

We have evaluated our attacks on six different data sets, overall containing the DNA methylation profiles of 1,320 patients. Our empirical results consistently demonstrate the success of membership inference attacks over different tissues and diseases. Even though we were limited by the small number of patients in most of the data sets, the experiments with the larger breast cancer data set suggested that our findings may scale. We concluded that the membership privacy of contributors to DNA methylation databases is put at risk even if the adversary does not directly get access to their methylation data but only their genomes.

Performing the membership inference attacks with DNA methylation data at different points in time is a future direction that is worth investigating. Moreover, designing attacks that exploit dependencies between methylation points is another interesting direction for future work.

Given the severe privacy risks that we uncover with our attacks, protection mechanisms are necessary. We explored one direction, the perturbation with randomly sampled noise to achieve differential privacy [36]. The challenge was to tailor a mechanism to this specific case with high data dimensionality and few individuals (currently) contributing their data. In line with other applications such as MBeacon [P3], we believe that there is a clear benefit from sharing population-wide mean methylation values. Mean methylation values could become as relevant and well-studied as minor allele frequencies are today for the genome.

In Chapter5 we also used DNA methylation data, but studied a concrete data-sharing

scenario. We showcase that a protection mechanism designed specifically for the use case and data can effectively mitigate privacy threats while maintaining utility.

We extended the Beacon system, a search engine for genome data whose answers are either "Yes, we have such data" or "No, we do not have such data". Knowing whether similar data is available helps researchers to find further data sets of interest. Despite a coarse-grained input format (queries only for a single position) and output format (binary yes/no), Beacons are vulnerable to membership inference attacks. Therefore, we followed a privacy-by-design principle for our extension to methylation data that we called MBeacon.

We first illustrated the severe privacy risks by conducting a membership inference attack based on the LR test. Experimental results on multiple data sets showed that with 100 queries, the adversary was able to achieve a superior performance. Then, we proposed a defense mechanism, SVT$^{\mathbf{2}}$, to implement our privacy-preserving MBeacon. Our SVT$^{\mathbf{2}}$ is an advancement of the sparse vector technique, one type of differential privacy algorithms. We theoretically proved that SVT$^{\mathbf{2}}$ is differentially private. Since the goal of MBeacon is to facilitate biomedical data sharing, we proposed a new metric for measuring researchers' utility considering a realistic scenario. Extensive experiments demonstrated that, using carefully chosen parameters, MBeacon can degrade the performance of the membership inference attack significantly without substantially hurting the researchers' utility.

There are two directions we want to explore in the future. First, we plan to extend the Beacon-style system to other types of biomedical data, such as gene expression, microRNA or laboratory tests. In particular, this requires to adapt the estimate of the general population accordingly. Second, the current Beacon systems only support queries on a single position. We plan to extend the Beacon system to support multiple-position queries. On one hand, this new system should improve the utility for the researchers. On the other hand, it will raise new privacy challenges.

We demonstrated in Chapter5 the importance of a concrete scenario and a thorough understanding of the data and privacy risks to develop a targeted defense mechanism that is able to balance privacy and utility. For microbiome data, the underlying privacy risks need to be uncovered first, which we did in Chapter 6. We defined and evaluated a variety of attack scenarios and attack methods and found that already simplistic methods based on euclidean distance or cosine similarity can work surprisingly well. Using more sophisticated machine-learning models, an AUC of 0.8 was often reached or exceeded, which demonstrated that privacy is at risk. This was also true if the attacker possesses samples from different points in time, or from different body sites. During our attacks, we have not incorporated specific biological knowledge, for example, the "family tree" of the OTUs features and the induced distances. Biomedical studies taking these into account show higher performance in their respective classification task [46, 119]. Therefore, we hypothesize that our analysis rather underestimated the privacy risk, and plan to explore this hypothesis in future work.

A limitation of our study was the amount of data per body site that we had available, which is only up to 100 samples from different people. While this was significantly more than previous small-scale studies with less than 20 people, our results should be re-evaluated as soon as data sets of several hundreds of people become available.

Additionally, we studied three simple defense mechanisms, but none of these methods showed a convincing drop in performance. These findings call for more sophisticated methods, such as adding differentially private noise, working with well-chosen subsets of the data or encrypted data only. We leave it to future work to explore these options.

While access to methylation and microbiome data require direct contact with the target or a hack into medical databases, eye tracking data is easier to obtain. Compromising eye tracking headsets or classification providers suffices to leak eye movement data at scale. In Chapter 7 we therefore presented the first privacy-aware gaze interface that uses differential privacy. We opted for a virtual reality gaze interface, given the significant and imminent threat potential created by upcoming eye tracking technology equipped VR headsets. Our experimental evaluations on a new 20-participant data set demonstrated the effectiveness of the proposed approach to preserve private information while maintaining performance on a utility task – hence, implementing the principle *ensure privacy without impeding utility*.

We additionally explored another technique to reduce the leakage of private information in eye tracking data in Chapter 8. Instead of differential privacy, we proposed targeted noise generated with adversarial examples. Our experiments show that users can protect their privacy with and without knowledge about the classifier inferring their private attributes as the success rates are comparable. However, knowledge about the classifier and preprocessing pipeline lead to smaller perturbations that maintain or even increase utility. Moreover, the white-box attack has low run time and could be used in an online setting where data is streamed to the service provider. Thus, our evaluations demonstrate that adversarial examples can be used as an ad hoc defense against leakage of private information from eye tracking signals.

We observed that privacy and utility were not conflicting goals, and that an increase of the utility is even possible while the private information is hidden due to the adversarial perturbation. Thus, we studied the phenomenon in depth in subsection 8.6.2 using classifier-agnostic clustering and found that features important for the utility task were positively influenced by adversarial examples against re-identification. Future work can explore the usage of adversarial examples for data augmentation and data quality improvement. Given the small number of participants in typical eye-tracking studies, removal of person-specific characteristics with adversarial perturbations might be an important preprocessing step for eye tracking research.

Clustering also demonstrated that the features important for utility and for distinguishing participants are vastly different. That finding inspired the privacy-preserving feature selection to trade-off privacy and utility already in the design state of the classifier. This method might be adopted in the following scenario: A service provider wants to hide protect the intellectual property in the classifier for eye-tracking data, thus requires data to be uploaded to the service provider's servers. The selection of a feature subset that minimizes the amount of private information in transfer drastically reduces the privacy risks and might be crucial for privacy-conscious users.

While the privacy risks of various biomedical data types depend on their biological function, attack techniques are often similar. Membership inference attacks and classification of private attributes have been explored here. The biological function also dictates the use cases and in order to not decrease usefulness of the data after

the application of privacy protection measures, these must be tailored to the use case. Future work should investigate whether use cases and attacks can be formalized into a general framework that simplifies the choice and implementation of standard techniques such as differential privacy.

Bibliography

Author's Papers for this Thesis

[P1] Steil, J., Hagestedt, I., Huang, M. X., and Bulling, A. Privacy-aware eye tracking using differential privacy. In: *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications, ETRA 19*. ACM. 2019, 27.

[P2] Hagestedt, I., Humbert, M., Berrang, P., Lehmann, I., Eils, R., Backes, M., and Zhang, Y. Membership inference against dna methylation databases. In: *IEEE European Symposium on Security and Privacy (EuroS&P)*. 2020.

[P3] Hagestedt, I., Zhang, Y., Humbert, M., Berrang, P., Tang, H., Wang, X., and Backes, M. Mbeacon: privacy-preserving beacons for dna methylation data. In: *Proceedings of the 26th Network and Distributed System Security Symposium (NDSS)*. 2019.

[P4] Hagestedt, I., Zhang, Y., Ye, Y., Tang, H., Wang, X., and Backes, M. *Quantifying Microbiome Privacy*.

[P5] Hagestedt, I., Backes, M., and Bulling, A. *Adversarial Examples for Privacy Protection of EyeTracking Data*.

Other references

[1] *23andMe*. `https://www.23andme.com`. Accessed: 2019-18-11.

[2] Abel, L. A. and Hertle, R. W. Effects of psychoactive drugs on ocular motor behavior. In: *Neuropsychology of Eye Movement*. Psychology Press, 2013, 93–126.

[3] Al Aziz, M. M., Ghasemi, R., Waliullah, M., and Mohammed, N. Aftermath of bustamante attack on genomic beacon service. *BMC medical genomics* 10, 2 (2017), 43.

[4] *AncestryDNA*. `https://www.ancestry.com/dna`. Accessed: 2019-18-11.

[5] *ArrayExpress*. `http://www.ebi.ac.uk/arrayexpress`. Accessed: 2019-20-07.

[6] *Atlasbiomed*. `https://atlasbiomed.com/uk/microbiome`. Accessed: 2019-10-09.

[7] Backes, M., Berrang, P., Bieg, M., Eils, R., Herrmann, C., Humbert, M., and Lehmann, I. Identifying Personal DNA Methylation Profiles by Genotype Inference. In: *Proceedings of the 38th IEEE Symposium on Security and Privacy (S&P)*. IEEE, 2017, 957–976.

[8] Backes, M., Berrang, P., Hecksteden, A., Humbert, M., Keller, A., and Meyer, T. Privacy in Epigenetics: Temporal Linkability of MicroRNA Expression Profiles. In: *Proceedings of the 25th USENIX Security Symposium (USENIX)*. USENIX Association, 2016, 1223–1240.

[9] Backes, M., Berrang, P., Humbert, M., and Manoharan, P. Membership Privacy in MicroRNA-based Studies. In: *Proceedings of the 23rd ACM Conference on Computer and Communications Security (CCS)*. ACM, 2016, 319–330.

[10] Backes, M., Humbert, M., Pang, J., and Zhang, Y. walk2friends: Inferring Social Links from Mobility Profiles. In: *Proceedings of the 24th ACM Conference on Computer and Communications Security (CCS)*. ACM, 2017, 1943–1957.

[11] *BaseClear*. `https://www.baseclear.com/services/next-generation-sequencing/microbial-community-analysis/`. Accessed: 2019-05-27.

[12] Bauer, T., Trump, S., Ishaque, N., Thu rmann, L., Gu, L., Bauer, M., Bieg, M., Gu, Z., Weichenhan, D., et al. Environment-induced Epigenetic Reprogramming in Genomic Regulatory Elements in Smoking Mothers and Their Children. *Molecular Systems Biology* 12, 3 (2016), 861–861.

[13] Bednarik, R., Kinnunen, T., Mihaila, A., and Fränti, P. Eye-movements as a biometric. In: *Scandinavian conference on image analysis*. Springer. 2005, 780–789.

[14] Biggio, B., Corona, I., Maiorca, D., Nelson, B., Šrndić, N., Laskov, P., Giacinto, G., and Roli, F. Evasion attacks against machine learning at test time. In: *Joint European conference on machine learning and knowledge discovery in databases*. Springer. 2013, 387–402.

[15] Biggio, B. and Roli, F. Wild patterns: ten years after the rise of adversarial machine learning. *Pattern Recognition* 84 (2018), 317–331.

[16] *The Black Market For Stolen Health Care Data*. `http://www.npr.org/sections/alltechconsidered/2015/02/13/385901377/the-black-market-for-stolen-health-care-data`. Accessed: 2019-10-08.

[17] Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., and Shah, R. Signature verification using a" siamese" time delay neural network. In: *Advances in neural information processing systems*. 1994, 737–744.

[18] Buczkowicz, P., Hoeman, C., Rakopoulos, P., Pajovic, S., Letourneau, L., Dzamba, M., Morrison, A., Lewis, P., Bouffet, E., Bartels, U., et al. Genomic analysis of diffuse intrinsic pontine gliomas identifies three molecular subgroups and recurrent activating ACVR1 mutations. *Nature genetics* 46, 5 (2014), 451–456.

[19] Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., Niculae, V., Prettenhofer, P., Gramfort, A., Grobler, J., Layton, R., VanderPlas, J., Joly, A., Holt, B., and Varoquaux, G. API design for machine learning software: experiences from the scikit-learn project. In: *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*. 2013, 108–122.

[20] Bulling, A. Pervasive attentive user interfaces. *IEEE Computer* 49, 1 (2016), 94–98.

[21] Bulling, A. and Gellersen, H. Toward Mobile Eye-Based Human-Computer Interaction. *IEEE Pervasive Computing* 9, 4 (2010), 8–12.

[22] Bulling, A., Roggen, D., and Tröster, G. What's in the eyes for context-awareness? *IEEE Pervasive Computing* 10, 2 (2011), 48–57.

[23] Bulling, A., Ward, J. A., Gellersen, H., and Troster, G. Eye movement analysis for activity recognition using electrooculography. *IEEE transactions on pattern analysis and machine intelligence* 33, 4 (2010), 741–753.

[24] Bulling, A., Ward, J. A., and Gellersen, H. Multimodal Recognition of Reading Activity in Transit Using Body-Worn Sensors. *ACM Transactions on Applied Perception* 9, 1 (2012), 2:1–2:21.

[25] Bulling, A., Ward, J. A., Gellersen, H., and Tröster, G. Eye Movement Analysis for Activity Recognition Using Electrooculography. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33, 4 (2011), 741–753.

[26] Bulling, A., Weichel, C., and Gellersen, H. Eyecontext: recognition of high-level contextual cues from human visual behaviour. In: *Proceedings of the sigchi conference on human factors in computing systems*. ACM. 2013, 305–308.

[27] Bulling, A. and Zander, T. O. Cognition-aware computing. *IEEE Pervasive Computing* 13, 3 (2014), 80–83.

[28] Chen, J., Jordan, M. I., and Wainwright, M. J. Hopskipjumpattack: a query-efficient decision-based attack. *arXiv preprint arXiv:1904.02144* (2019).

[29] Cheng, J., Ringel-Kulka, T., Heikamp-de Jong, I., Ringel, Y., Carroll, I., De Vos, W. M., Salojärvi, J., and Satokari, R. Discordant temporal development of bacterial phyla and the emergence of core in the fecal microbiota of young children. *The ISME journal* 10, 4 (2016), 1002.

[30] *2017 was the year consumer DNA testing blew up*. `https://www.technologyreview.com/s/610233/2017-was-the-year-consumer-dna-testing-blew-up/`. Accessed: 2018-06-28.

[31] David, L. A., Maurice, C. F., Carmody, R. N., Gootenberg, D. B., Button, J. E., Wolfe, B. E., Ling, A. V., Devlin, A. S., Varma, Y., Fischbach, M. A., et al. Diet rapidly and reproducibly alters the human gut microbiome. *Nature* 505, 7484 (2014), 559.

[32] Demontis, A., Melis, M., Biggio, B., Maiorca, D., Arp, D., Rieck, K., Corona, I., Giacinto, G., and Roli, F. Yes, machine learning can be more secure! a case study on android malware detection. *IEEE Transactions on Dependable and Secure Computing* (2017).

[33] Demontis, A., Melis, M., Pintor, M., Jagielski, M., Biggio, B., Oprea, A., Nita-Rotaru, C., and Roli, F. Why do adversarial attacks transfer? explaining transferability of evasion and poisoning attacks. In: *28th {USENIX} Security Symposium ({USENIX} Security 19)*. 2019, 321–338.

[34] Dominianni, C., Sinha, R., Goedert, J. J., Pei, Z., Yang, L., Hayes, R. B., and Ahn, J. Sex, body mass index, and dietary fiber intake influence the human gut microbiome. *PloS one* 10, 4 (2015), e0124599.

[35] Dong, Y., Su, H., Zhu, J., and Bao, F. Towards interpretable deep neural networks by leveraging adversarial examples. *arXiv preprint arXiv:1708.05493* (2017).

[36] Dwork, C., Kenthapadi, K., McSherry, F., Mironov, I., and Naor, M. Our data, ourselves: Privacy via distributed noise generation. In: *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer. 2006, 486–503.

[37] Dwork, C., McSherry, F., Nissim, K., and Smith, A. Calibrating noise to sensitivity in private data analysis. In: *Theory of cryptography conference*. Springer. 2006, 265–284.

[38] Dwork, C., Roth, A., et al. The Algorithmic Foundations of Differential Privacy. *Foundations and Trends® in Theoretical Computer Science* 9, 3–4 (2014), 211–407.

[39] Dwork, C., Smith, A., Steinke, T., Ullman, J., and Vadhan, S. Robust Traceability from Trace Amounts. In: *Proceedings of the 56th IEEE Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2015, 650–669.

[40] Dyke, S. O., Cheung, W. A., Joly, Y., Ammerpohl, O., Lutsik, P., Rothstein, M. A., Caron, M., Busche, S., Bourque, G., Rönnblom, L., et al. Epigenome Data Release: A Participant-centered Approach to Privacy Protection. *Genome Biology* 16 (2015), 1–12.

[41] Eberz, S., Rasmussen, K. B., Lenders, V., and Martinovic, I. Looks like eve: exposing insider threats using eye movement biometrics. *ACM Transactions on Privacy and Security (TOPS)* 19, 1 (2016), 1.

[42] Erlich, Y. and Narayanan, A. Routes for Breaching and Protecting Genetic Privacy. *Nature Reviews Genetics* 15, 6 (2014), 409–421.

[43] Esteller, M. and Herman, J. G. Cancer as an Epigenetic Disease: DNA Methylation and Chromatin Alterations in Human Tumours. *The Journal of Pathology* 196, 1 (2002), 1–7.

[44] Fan, L. and Xiong, L. Adaptively sharing time-series with differential privacy. *arXiv preprint arXiv:1202.3461* (2012).

[45] Findley, K., Williams, D. R., Grice, E. A., and Bonham, V. L. Health disparities and the microbiome. *Trends in microbiology* 24, 11 (2016), 847–850.

[46] Fioravanti, D., Giarratano, Y., Maggio, V., Agostinelli, C., Chierici, M., Jurman, G., and Furlanello, C. Phylogenetic convolutional neural networks in metagenomics. *BMC bioinformatics* 19, 2 (2018), 49.

[47] Fontebasso, A. M., Papillon-Cavanagh, S., Schwartzentruber, J., Nikbakht, H., Gerges, N., et al. Recurrent somatic mutations in ACVR1 in pediatric midline high-grade astrocytoma. *Nature genetics* 46, 5 (2014), 462–466.

[48] *FOVE Eye Tracking Virtual Reality headset*. `https://www.getfove.com/`. Accessed: 2020-03-04.

[49] Franzosa, E. A., Huang, K., Meadow, J. F., Gevers, D., Lemon, K. P., Bohannan, B. J., and Huttenhower, C. Identifying personal microbiomes using metagenomic codes. *Proceedings of the National Academy of Sciences* (2015), 201423854.

[50] Fredrikson, M., Lantz, E., Jha, S., Lin, S., Page, D., and Ristenpart, T. Privacy in Pharmacogenetics: An End-to-end Case Study of Personalized Warfarin Dosing. In: *Proceedings of the 23rd USENIX Security Symposium (USENIX)*. USENIX Association, 2014, 17–32.

[51] Fuhl, W. Reinforcement learning for the manipulation of eye tracking data. *arXiv preprint arXiv:2002.06806* (2020).

[52] Gaunt, T. R., Shihab, H. A., Hemani, G., Min, J. L., Woodward, G., Lyttleton, O., Zheng, J., Duggirala, A., McArdle, W. L., Ho, K., Ring, S. M., Evans, D. M., Davey Smith, G., and Relton, C. L. Systematic Identification of Genetic Influences on Methylation Across the Human Life Course. *Genome Biology* 17, 1 (2016), 61.

[53] *GEDmatch*. `https://www.gedmatch.com`. Accessed: 2019-18-11.

[54] *Gene Expression Omnibus*. `https://www.ncbi.nlm.nih.gov/geo`. Accessed: 2019-20-07.

[55] Goodfellow, I. J., Shlens, J., and Szegedy, C. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* (2014).

[56] Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. Gene selection for cancer classification using support vector machines. *Machine learning* 46, 1-3 (2002), 389–422.

[57] Hansen, J. P., Johansen, A. S., Hansen, D. W., Itoh, K., and Mashino, S. Command without a click: dwell time typing by mouse and gaze selections. In: *Proceedings of Human-Computer Interaction–INTERACT*. 2003, 121–128.

[58] Hayes, J., Melis, L., Danezis, G., and De Cristofaro, E. Logan: membership inference attacks against generative models. *Proceedings on Privacy Enhancing Technologies (PoPETs)* (2019).

[59] Hess, E. H. and Polt, J. M. Pupil size as related to interest value of visual stimuli. *Science* 132, 3423 (1960), 349–350.

[60] *Hololens*. `https://www.microsoft.com/en-us/hololens`. Accessed:2020-03-04.

[61] Holzman, P. S., Proctor, L. R., Levy, D. L., Yasillo, N. J., Meltzer, H. Y., and Hurt, S. W. Eye-tracking dysfunctions in schizophrenic patients and their relatives. *Archives of general psychiatry* 31, 2 (1974), 143–151.

[62] Homer, N., Szelinger, S., Redman, M., Duggan, D., Tembe, W., Muehling, J., Pearson, J. V., Stephan, D. A., Nelson, S. F., and Craig, D. W. Resolving individuals contributing trace amounts of dna to highly complex mixtures using high-density snp genotyping microarrays. *PLoS Genet* 4, 8 (2008), e1000167.

[63] Hoppe, S., Loetscher, T., Morey, S. A., and Bulling, A. Eye movements during everyday behavior predict personality traits. *Frontiers in Human Neuroscience* 12 (2018), 105.

[64] Huang, P.-S., He, X., Gao, J., Deng, L., Acero, A., and Heck, L. Learning deep structured semantic models for web search using clickthrough data. In: *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*. ACM. 2013, 2333–2338.

[65] Ilyas, A., Santurkar, S., Tsipras, D., Engstrom, L., Tran, B., and Madry, A. Adversarial examples are not bugs, they are features. In: *Advances in Neural Information Processing Systems*. 2019, 125–136.

[66] Jäger, L. A., Makowski, S., Prasse, P., Liehr, S., Seidler, M., and Scheffer, T. Deep eyedentification: biometric identification using micro-movements of the eye. *arXiv preprint arXiv:1906.11889* (2019).

[67] Jia, J. and Gong, N. Z. Attriguard: a practical defense against attribute inference attacks via adversarial machine learning. In: *27th {USENIX} Security Symposium ({USENIX} Security 18)*. 2018, 513–529.

[68] John, O. P. and Srivastava, S. The big five trait taxonomy: history, measurement, and theoretical perspectives. *Handbook of personality: Theory and research* 2, 1999 (1999), 102–138.

[69] Johnson, A. and Shmatikov, V. Privacy-preserving Data Exploration in Genome-wide Association Studies. In: *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*. ACM, 2013, 1079–1087.

[70] Jones, P. A. Functions of DNA Methylation: Islands, Start Sites, Gene Bodies and Beyond. *Nature Reviews Genetics* 13, 7 (2012), 484–92.

[71] Joon Oh, S., Fritz, M., and Schiele, B. Adversarial image perturbation for privacy protection–a game theory perspective. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, 1482–1491.

[72] Kasprowski, P. Human identification using eye movements. *Praca doktorska, Politechnika Œląska* (2004).

[73] Kasprowski, P. and Ober, J. Eye movement tracking for human identification. In: *6th World Conference BIOMETRICS*. 2003.

[74] Kasprowski, P. and Ober, J. Enhancing eye-movement-based biometric identification method by using voting classifiers. In: *Biometric Technology for Human Identification II*. Vol. 5779. International Society for Optics and Photonics. 2005, 314–324.

[75] Kassner, M., Patera, W., and Bulling, A. Pupil: an open source platform for pervasive eye tracking and mobile gaze-based interaction. In: *Adj. Proc. ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*. 2014, 1151–1160.

[76] Kingstone, A., Smilek, D., Ristic, J., Kelland Friesen, C., and Eastwood, J. D. Attention, researchers! it is time to take a look at the real world. *Current Directions in Psychological Science* 12, 5 (2003), 176–180.

[77] Kinnunen, T., Sedlak, F., and Bednarik, R. Towards task-independent person authentication using eye movement signals. In: *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*. ACM. 2010, 187–190.

[78] Kleinman, C. L., Gerges, N., Papillon-Cavanagh, S., Sin-Chan, P., Pramatarova, A., Quang, D.-A. K., Adoue, V., Busche, S., Caron, M., Djambazian, H., et al. Fusion of TTYH1 with the C19MC microRNA cluster drives expression of a brain-specific DNMT3B isoform in the embryonal brain tumor ETMR. *Nature genetics* 46, 1 (2014), 39–44.

[79] Komogortsev, O. V. and Holland, C. D. Biometric authentication via complex oculomotor behavior. In: *Biometrics: Theory, Applications and Systems (BTAS), 2013 IEEE Sixth International Conference on*. IEEE. 2013, 1–8.

[80] Kuechenmeister, C. A., Linton, P. H., Mueller, T. V., and White, H. B. Eye tracking in relation to age, sex, and illness. *Archives of General Psychiatry* 34, 5 (1977), 578–579.

[81] Kulynych, B., Hayes, J., Samarin, N., and Troncoso, C. Evading classifiers in discrete domains with provable optimality guarantees. *arXiv preprint arXiv:1810.10939* (2018).

[82] Kunze, K., Kawaichi, H., Yoshimura, K., and Kise, K. The wordometer–estimating the number of words read using document image retrieval and mobile eye tracking. In: *12th International Conference on Document Analysis and Recognition (ICDAR)*. 2013, 25–29.

[83] Kunze, K., Masai, K., Inami, M., Sacakli, Ö., Liwicki, M., Dengel, A., Ishimaru, S., and Kise, K. Quantifying reading habits: counting how many words you read. In: *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM. 2015, 87–96.

[84] Kunze, K., Utsumi, Y., Shiga, Y., Kise, K., and Bulling, A. I know what you are reading: recognition of document types using mobile eye tracking. In: *Proceedings of the 2013 International Symposium on Wearable Computers*. ACM. 2013, 113–116.

[85] Lambert, S. R., Witt, H., Hovestadt, V., Zucknick, M., Kool, M., Pearson, D. M., Korshunov, A., Ryzhova, M., Ichimura, K., Jabado, N., et al. Differential expression and methylation of brain developmental genes define location-specific subsets of pilocytic astrocytoma. *Acta neuropathologica* 126, 2 (2013), 291–301.

[86] Lax, S., Hampton-Marcell, J. T., Gibbons, S. M., Colares, G. B., Smith, D., Eisen, J. A., and Gilbert, J. A. Forensic analysis of the microbiome of phones and shoes. *Microbiome* 3, 1 (2015), 21.

[87] Lehmann, E. L. and Romano, J. P. *Testing statistical hypotheses*. Springer Science & Business Media, 2006.

[88] Liebling, D. J. and Preibusch, S. Privacy considerations for a pervasive eye tracking world. In: *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*. ACM. 2014, 1169–1177.

[89] Liu, A., Xia, L., Duchowski, A., Bailey, R., Holmqvist, K., and Jain, E. Differential privacy for eye-tracking data. In: *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications, ETRA 19*. ACM. 2019.

[90] Lyu, M., Su, D., and Li, N. Understanding the Sparse Vector Technique for Differential Privacy. *Proceedings of the VLDB Endowment* 10, 6 (2017), 637–648.

[91] Maeder, A. J. and Fookes, C. B. A visual attention approach to personal identification (2003).

[92] Marioni, R. E., Shah, S., McRae, A. F., Chen, B. H., Colicino, E., Harris, S. E., Gibson, J., Henders, A. K., Redmond, P., Cox, S. R., et al. DNA methylation age of blood predicts all-cause mortality in later life. *Genome biology* 16, 1 (2015), 25.

[93] Marioni, R. E., Shah, S., McRae, A. F., Ritchie, S. J., Muniz-Terrera, G., Harris, S. E., Gibson, J., Redmond, P., Cox, S. R., Pattie, A., et al. The epigenetic clock is correlated with physical and cognitive fitness in the Lothian Birth Cohort 1936. *International journal of epidemiology* 44, 4 (2015), 1388–1396.

[94] Maslowski, K. M. and Mackay, C. R. Diet, gut microbiota and immune responses. *Nature immunology* 12, 1 (2010), 5.

[95] Matthews, G., Middleton, W., Gilmartin, B., and Bullimore, M. Pupillary diameter and cognitive load. *Journal of Psychophysiology* (1991).

[96] McClay, J. L., Shabalin, A. A., Dozmorov, M. G., Adkins, D. E., Kumar, G., Nerella, S., Clark, S. L., Bergen, S. E., Hultman, C. M., Magnusson, P. K. E., Sullivan, P. F., Aberg, K. A., and Oord, E. J. C. G. van den. High Density Methylation QTL Analysis in Human Blood via Next-generation Sequencing of the Methylated Genomic DNA Fraction. *Genome Biology* 16, 1 (2015), 291.

[97] Meadow, J. F., Altrichter, A. E., Bateman, A. C., Stenson, J., Brown, G., Green, J. L., and Bohannan, B. J. Humans differ in their personal microbial cloud. *PeerJ* 3 (2015), e1258.

[98] Meadow, J. F., Altrichter, A. E., and Green, J. L. Mobile phones carry the personal microbiome of their owners. *PeerJ* 2 (2014), e447.

[99] Mittos, A., Malin, B., and De Cristofaro, E. Systematizing genomic privacy research–a critical analysis. *arXiv preprint arXiv:1712.02193* (2017).

[100] Murphy, S. N., Weber, G., Mendis, M., Gainer, V., Chueh, H. C., Churchill, S., and Kohane, I. Serving the enterprise and beyond with informatics for integrating biology and the bedside (i2b2). *Journal of the American Medical Informatics Association* 17, 2 (2010), 124–130.

[101] Nasr, M., Shokri, R., and Houmansadr, A. Machine learning with membership privacy using adversarial regularization. In: *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*. 2018, 634–646.

[102] Naumova, A. K., Al Tuwaijri, A., Morin, A., Vaillancout, V. T., Madore, A.-M., Berlivet, S., Kohan-Ghadr, H.-R., Moussette, S., and Laprise, C. Sex-and age-dependent dna methylation at the 17q12-q21 locus associated with childhood asthma. *Human genetics* 132, 7 (2013), 811–822.

[103] Ni, M., Zhang, Y., Han, W., and Pang, J. An Empirical Study on User Access Control in Online Social Networks. In: *Proceedings of the 2016 ACM Symposium on Access Control Models and Technologies (SACMAT)*. ACM, 2016, 12–23.

[104] Nicolae, M.-I., Sinn, M., Tran, M. N., Buesser, B., Rawat, A., Wistuba, M., Zantedeschi, V., Baracaldo, N., Chen, B., Ludwig, H., Molloy, I., and Edwards, B. Adversarial robustness toolbox v1.0.1. *CoRR* 1807.01069 (2018).

[105] *OpenSNP*. `https://opensnp.org`. Accessed: 2019-18-11.

[106] Overdorf, R., Kulynych, B., Balsa, E., Troncoso, C., and Gürses, S. Pots: protective optimization technologies. *arXiv preprint arXiv:1806.02711* (2018).

[107] Pang, J. and Zhang, Y. Location Prediction: Communities Speak Louder than Friends. In: *Proceedings of the 2015 ACM Conference on Online Social Networks (COSN)*. ACM, 2015, 161–171.

[108] Pang, J. and Zhang, Y. DeepCity: A Feature Learning Framework for Mining Location Check-Ins. In: *Proceedings of the 2017 International Conference on Weblogs and Social Media (ICWSM)*. AAAI, 2017, 652–655.

[109] Pang, J. and Zhang, Y. Quantifying Location Sociality. In: *Proceedings of the 2017 ACM Conference on Hypertext and Social Media (HT)*. ACM, 2017, 145–154.

[110] Papernot, N., McDaniel, P., Sinha, A., and Wellman, M. P. Sok: security and privacy in machine learning. In: *2018 IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE. 2018, 399–414.

[111] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research (JMLR)* 12 (2011), 2825–2830.

[112] *Personal Genome Project*. `http://www.personalgenomes.org`. Accessed: 2017-20-07.

[113] Philibert, R. A., Terry, N., Erwin, C., Philibert, W. J., Beach, S. R., and Brody, G. H. Methylation Array Data Can Simultaneously Identify Individuals and Convey Protected Health Information: An Unrecognized Ethical Concern. *Clinical Epigenetics* 6 (2014), 28.

[114] Pyrgelis, A., Troncoso, C., and Cristofaro, E. D. Knock Knock, Who's There? Membership Inference on Aggregate Location Data. In: *Proceedings of the 25th Network and Distributed System Security Symposium (NDSS)*. 2018.

[115] Pyrgelis, A., Troncoso, C., and De Cristofaro, E. Knock Knock, Who's There? Membership Inference on Aggregate Location Data. *arXiv preprint arXiv:1708.06145* (2017).

[116] Qin, J., Li, R., Raes, J., Arumugam, M., Burgdorf, K. S., Manichanh, C., Nielsen, T., Pons, N., Levenez, F., Yamada, T., et al. A human gut microbial gene catalogue established by metagenomic sequencing. *nature* 464, 7285 (2010), 59.

[117] Raisaro, J. L., Tramèr, F., Ji, Z., Bu, D., Zhao, Y., Carey, K., Lloyd, D., Sofia, H., Baker, D., Flicek, P., et al. Addressing Beacon re-identification attacks: quantification and mitigation of privacy risks. *Journal of the American Medical Informatics Association* (2017), ocw167.

[118] Raisaro, J. L., Troncoso-Pastoriza, J. R., Misbach, M., Gomes de Sá, E. S., André, J., Pradervand, S., Missiaglia, E., Michielin, O., Ford, B. A., and Hubaux, J.-P. *MedCo: Enabling Privacy-Conscious Exploration of Distributed Clinical and Genomic Data*. Tech. rep. 2017.

[119] Reiman, D., Metwally, A. A., and Dai, Y. Popphy-cnn: a phylogenetic tree embedded architecture for convolution neural networks for metagenomic data. *bioRxiv* (2018), 257931.

[120] Rogers, H. A., Kilday, J.-P., Mayne, C., Ward, J., Adamowicz-Brice, M., Schwalbe, E. C., Clifford, S. C., Coyle, B., and Grundy, R. G. Supratentorial and spinal pediatric ependymomas display a hypermethylated phenotype which includes the loss of tumor suppressor genes involved in the control of cell growth and death. *Acta neuropathologica* 123, 5 (2012), 711–725.

[121] Rosenberg, A. and Hirschberg, J. V-measure: a conditional entropy-based external cluster evaluation measure. In: *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*. 2007, 410–420.

[122] Saleheen, N., Chakraborty, S., Ali, N., Rahman, M. M., Hossain, S. M., Bari, R., Buder, E., Srivastava, M., and Kumar, S. Msieve: differential behavioral privacy in time series of mobile sensor data. In: *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM. 2016, 706–717.

[123] Sammaknejad, N., Pouretemad, H., Eslahchi, C., Salahirad, A., and Alinejad, A. Gender classification based on eye movements: a processing effect during passive face viewing. *Advances in cognitive psychology* 13, 3 (2017), 232.

[124] Sandrini, S., Aldriwesh, M., Alruways, M., and Freestone, P. Microbial endocrinology: host–bacteria communication within the gut microbiome. *Journal of Endocrinology* 225, 2 (2015), R21–R34.

[125] Sankararaman, S., Obozinski, G., Jordan, M. I., and Halperin, E. Genomic Privacy and Limits of Individual Detection in a Pool. *Nature Genetics* 41, 9 (2009), 965–967.

[126] Sattar, H., Müller, S., Fritz, M., and Bulling, A. Prediction of search targets from fixations in open-world settings. In: *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, 981–990.

[127] Schübeler, D. Function and Information Content of DNA Methylation. *Nature* 517, 7534 (2015), 321–326.

[128] Sender, R., Fuchs, S., and Milo, R. Are we really vastly outnumbered? revisiting the ratio of bacterial to host cells in humans. *Cell* 164 (2016).

[129] Shamarina, D., Stoyantcheva, I., Mason, C. E., Bibby, K., and Elhaik, E. Communicating the promise, risks, and ethics of large-scale, open space microbiome and metagenome research. *Microbiome* 5, 1 (2017), 132.

[130] Shokri, R. and Shmatikov, V. Privacy-Preserving Deep Learning. In: *Proceedings of the 22nd ACM conference on computer and communications security (CCS)*. ACM. 2015, 1310–1321.

[131] Shokri, R., Stronati, M., Song, C., and Shmatikov, V. Membership Inference Attacks against Machine Learning Models. In: *Proceedings of the 38th IEEE Symposium on Security and Privacy (S&P)*. IEEE. 2017, 3–18.

[132] Shringarpure, S. S. and Bustamante, C. D. Privacy Risks from Genomic Data-Sharing Beacons. *The American Journal of Human Genetics* 97, 5 (2015), 631–646.

[133] Song, S. J., Lauber, C., Costello, E. K., Lozupone, C. A., Humphrey, G., Berg-Lyons, D., Caporaso, J. G., Knights, D., Clemente, J. C., Nakielny, S., et al. Cohabiting family members share microbiota with one another and with their dogs. *elife* 2 (2013), e00458.

[134] Statistics, T. F. M. *A Decision Theoretic Approach*. 1967.

[135] Steil, J. and Bulling, A. Discovery of everyday human activities from long-term visual behaviour using topic models. In: *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM. 2015, 75–85.

[136] Steil, J., Koelle, M., Heuten, W., Boll, S., and Bulling, A. Privaceye: privacy-preserving head-mounted eye tracking using egocentric scene image and eye movement features. In: *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications*. ACM. 2019, 26.

[137] Stellmach, S. and Dachselt, R. Look & touch: gaze-supported target acquisition. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM. 2012, 2981–2990.

[138] Sturm, D., Witt, H., Hovestadt, V., Khuong-Quang, D.-A., Jones, D. T., Konermann, C., Pfaff, E., Tönjes, M., Sill, M., Bender, S., et al. Hotspot mutations in H3F3A and IDH1 define distinct epigenetic and biological subgroups of glioblastoma. *Cancer cell* 22, 4 (2012), 425–437.

[139] Teh, A. L., Pan, H., Chen, L., Ong, M.-L., Dogra, S., Wong, J., MacIsaac, J. L., Mah, S. M., McEwen, L. M., Saw, S.-M., et al. The Effect of Genotype and in Utero Environment on Interindividual Variation in Neonate DNA Methylomes. *Genome Research* 24, 7 (2014), 1064–1074.

[140] Thenen, N. von, Ayday, E., and Cicek, A. E. Re-identification of individuals in genomic data-sharing beacons via allele inference. *bioRxiv* (2017), 200147.

[141] Tonsen, M., Steil, J., Sugano, Y., and Bulling, A. Invisibleeye: mobile eye tracking using multiple low-resolution cameras and learning-based gaze estimation. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT)* 1, 3 (2017), 106:1–106:21.

[142] Tonsen, M., Steil, J., Sugano, Y., and Bulling, A. Invisibleeye: mobile eye tracking using multiple low-resolution cameras and learning-based gaze estimation. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (2017), 106.

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

[144] Trump, S., Bieg, M., Gu, Z., Thürmann, L., Bauer, T., Bauer, M., Ishaque, N., Röder, S., Gu, L., Herberth, G., et al. Prenatal Maternal Stress and Wheeze in Children: Novel Insights into Epigenetic Regulation. *Scientific Reports* 6 (2016), 28616.

[145] Tsaprouni, L. G., Yang, T.-P., Bell, J., Dick, K. J., Kanoni, S., Nisbet, J., Viñuela, A., Grundberg, E., Nelson, C. P., Meduri, E., et al. Cigarette Smoking Reduces DNA Methylation Levels at Multiple Genomic Loci but the Effect is Partially Reversible upon Cessation. *Epigenetics* 9, 10 (2014), 1382–1396.

[146] Uhler, C., Slavković, A., and Fienberg, S. E. Privacy-preserving data sharing for genome-wide association studies. *The Journal of Privacy and Confidentiality* 5, 1 (2013), 137.

[147] Van Dongen, J., Nivard, M. G., Willemsen, G., Hottenga, J.-J., Helmer, Q., Dolan, C. V., Ehli, E. A., Davies, G. E., Van Iterson, M., Breeze, C. E., et al. Genetic and Environmental Influences Interact with Age and Sex in Shaping the Human Methylome. *Nature Communications* 7 (2016), 11115.

[148] Ventham, N., Kennedy, N., Adams, A., Kalla, R., Heath, S., O'leary, K., Drummond, H., Wilson, D., Gut, I. G., Nimmo, E., et al. Integrative epigenome-wide analysis demonstrates that DNA methylation may mediate genetic risk in inflammatory bowel disease. *Nature communications* 7 (2016), 13507.

[149] Vertegaal, R. et al. Attentive user interfaces. *Communications of the ACM* 46, 3 (2003), 30–33.

[150] Vidal, M., Turner, J., Bulling, A., and Gellersen, H. Wearable eye tracking for mental health monitoring. *Computer Communications* 35, 11 (2012), 1306–1311.

[151] Wagner, J., Paulson, J. N., Wang, X., Bhattacharjee, B., and Corrada Bravo, H. Privacy-preserving microbiome analysis using secure computation. *Bioinformatics* 32, 12 (2016), 1873–1879.

[152] Wan, Z., Vorobeychik, Y., Kantarcioglu, M., and Malin, B. Controlling the signal: Practical privacy protection of genomic data sharing through Beacon services. *BMC medical genomics* 10, 2 (2017), 39.

[153] Wang, R., Li, Y. F., Wang, X., Tang, H., and Zhou, X. 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 (CCS)*. ACM, 2009, 534–544.

[154] Weber, G. M., Murphy, S. N., McMurry, A. J., MacFadden, D., Nigrin, D. J., Churchill, S., and Kohane, I. S. The shared health research information network (shrine): a prototype federated query tool for clinical data repositories. *Journal of the American Medical Informatics Association* 16, 5 (2009), 624–630.

[155] Yu, F., Fienberg, S. E., Slavković, A. B., and Uhler, C. Scalable privacy-preserving data sharing methodology for genome-wide association studies. *Journal of biomedical informatics* 50 (2014), 133–141.

[156] Zhang, Y., Humbert, M., Rahman, T., Li, C.-T., Pang, J., and Backes, M. Tagvisor: A Privacy Advisor for Sharing Hashtags. In: *Proceedings of the 2018 Web Conference (WWW)*. ACM, 2018, 287–296.

[157] Zhang, Y., Humbert, M., Surma, B., Manoharan, P., Vreeken, J., and Backes, M. CTRL+Z: Recovering Anonymized Social Graphs. *CoRR abs/1711.05441* (2017).

[158] Zhang, Y., Hu, W., Xu, W., Chou, C. T., and Hu, J. Continuous authentication using eye movement response of implicit visual stimuli. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 4 (2018), 177.

[159] Zhou, X., Peng, B., Li, Y. F., Chen, Y., Tang, H., and Wang, X. To release or not to release: evaluating information leaks in aggregate human-genome data. In: *Proceedings of the 16th European Symposium on Research in Computer Security (ESORICS)*. 2011, 607–627.

[160] Zhu, T., Li, G., Zhou, W., and Philip, S. Y. Differentially private data publishing and analysis: a survey. *IEEE Transactions on Knowledge and Data Engineering* 29, 8 (2017), 1619–1638.