Privacy-preserving and Optimal Interval Release for Disease Susceptibility

Kosuke Kusano
University of Tsukuba
Tsukuba, Japan
cocuh@mdl.cs.tsukuba.ac.jp

Ichiro Takeuchi
Nagoya Institute of Technology / RIKEN Center for AIP
Nagoya, Japan
takeuchi.ichiro@nitech.ac.jp

Jun Sakuma
University of Tsukuba / JST CREST / RIKEN Center for AIP
Tsukuba, Japan
jun@cs.tsukuba.ac.jp

ABSTRACT

In this paper, we consider the problem of privacy-preserving release of function outputs that take private information as input. Disease susceptibilities are known to be associated with clinical features (e.g., age, sex) as well as genetic features represented by SNPs of individuals. Releasing outputs are not privacy-preserving if the private input can be uniquely identified by probabilistic inference using the outputs. To release useful outputs with preserving privacy, we present a mechanism that releases an interval as output, instead of an output value. We suppose adversaries perform probabilistic inference using released outputs to sharpen the posterior distribution of the target attributes. Then, our mechanism has two significant properties. First, when our mechanism provides the output, the increase of the adversary's posterior on any input attribute is upper-bounded by a prescribed level. Second, under this privacy constraint, the mechanism can provide the narrowest (optimal) interval that includes the true output. Building such a mechanism is often intractable. We formulate the design of the mechanism as a discrete constraint optimization problem so that it is solvable in a practical computation time. We also propose an algorithm to obtain the optimal mechanism based on dynamic programming. After applying our mechanism to release disease susceptibilities of obesity, we demonstrate that our mechanism performs better than existing methods in terms of privacy and utility.

Keywords

Privacy; Genome; Disease Susceptibility; Input Inference; Interval Publication

1. INTRODUCTION

Single nucleotide polymorphisms (SNPs) represent individual differences in DNA sequence variations. Current technological advances in molecular biology enable us to measure numerous SNPs at a reasonable cost. *Genome-wide association studies* (GWAS) have revealed that gene polymorphisms (genetic variation between individuals) have important effects on the constitutions of individuals.

One medical application using SNPs is the prediction of individual susceptibilities to common complex diseases such as diabetes and cardiac infarction. Disease susceptibilities are known to be associated with clinical features (e.g., age, sex, blood pressure, family disease history) and genetic features represented by SNPs of individuals. The variant form of the gene is described as *allele*. The allele whose traits appear with low and high frequency are described as *minor allele* and *major allele*, respectively. Presence (or absence) of genetic variants at specific loci is as input genetic futures. Personalized medical care services based on common disease susceptibilities are becoming an important part of preventive medicine [1, 2, 3].

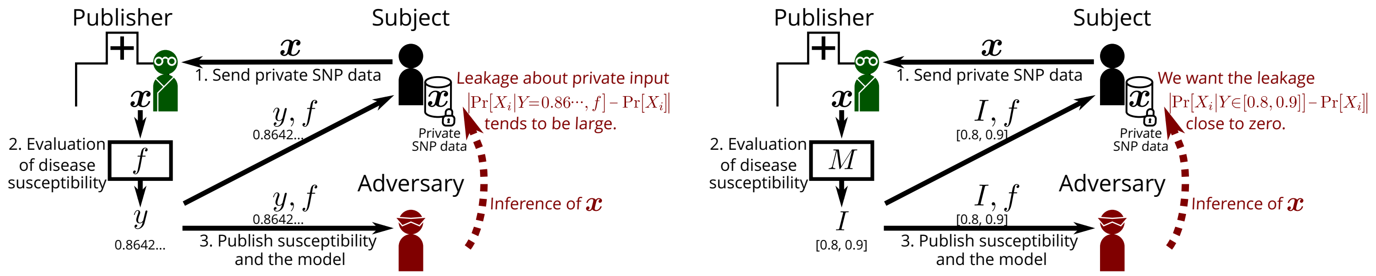
Another important medical application of SNPs is personalized drug administration. Drug sensitivity is known to be affected significantly by personal genomes [7, 11]. Physicians are often forced to undertake trial and error processes to find an appropriate medical treatment that is most effective for their patient. Personalized drug administration helps physicians to select an appropriate medicine that is effective for individual patients, and to adjust proper dosage amounts considering the drug sensitivity of individual patients.

Utilization of personal genomes for medication is extremely beneficial. However, personal genomes can include highly private and sensitive information [13, 14]. Personal genetic information can be used as an identifier that uniquely identifies individuals with very high probability [17]. It also indicates sensitive properties of individuals, such as ethnic descriptions, constitution, and susceptibilities to specific diseases, etc [7, 19]. Consequently, personal genomes must be managed under careful control at every step of collection, storage, and utilization.

In the pursuit of personalized medicine, statistical models are often used. They take personal genomes as input and give some susceptibility or effect information as output. For example, susceptibilities of common diseases are evaluated using logistic regression models. In addition, for personalized drug administration, the proper dosage is determined using a linear regression model. As reported previously [7], one can sometimes infer inputs (personal genomes) from outputs (susceptibilities or effects) using probabilistic inference. In an extreme case, if the function is a one-to-one mapping, then personal genomes can be uniquely identified from out-

ASIA CCS '17 April 02-06, 2017, Abu Dhabi, United Arab Emirates

ACM ISBN 978-1-4503-4944-4/17/04.
DOI: http://dx.doi.org/10.1145/3052973.3053021



(a) Release output without modification

(b) Release output as intervals

Figure 1: Input inference by advesaries.

puts. If such identification can be achieved with a high level of confidence, then model outputs should be treated secretly as sensitive information. It is, therefore, important to guarantee that personal genomes cannot be inferred from function outputs with a high-level confidence. Our study does not necessarily assume the function is injective, but it is used to consider the injective function as the worst case.

This study elucidates the problem of releasing outputs of functions that take private information as input. If outputs can be released with no modification, then the private input might be identified using probabilistic inference (Fig. 1, left). Our objective is to present a mechanism that modifies outputs to be an interval so that the input cannot be inferred with high confidence. Simultaneously, the output must be exact to the greatest extent possible while satisfying privacy constraints (Fig. 1, right).

1.1 Related Work

Attacks to personal genomes by probabilistic inference have been studied extensively in diverse contexts. Ayday et al. considered an inferential attack on SNPs that are in linkage disequilibrium [1] [5]. Humbert et al. considered reconstructing the genomes of the relatives of an individual when the individual genome is observed as prior knowledge [10]. Goodrich et al. exploited the master mind attack, by which the adversary was allowed to issue similarity queries repeatedly to infer the input genome string [8]. Inference of presence (or absence) of any subject in GWAS from GWAS-related statistics has been discussed in relation to statistical testing [9, 18].

The following two studies assessed the problem of input inference from function outputs, which are closely related to our work. Fredrikson et al. presented a study of privacy preservation for personalized dosing of warfarin [7] . The proper dosage (output) is determined using a linear regression model that takes a few genetic markers and demographic features (input). When an attacker can obtain outputs, they pointed out that the attacker can recover patients' genetic markers with a certain level of confidence. To prevent such reconstruction, they used a regression model with differential privacy. They empirically investigated the tradeoff between utility (mortality) and privacy (differential privacy). The authors concluded from an experimental evaluation that differentially private mechanisms did not simultaneously improve genomic privacy while retaining desirable clinical efficacy for warfarin dosing.

Ayday et al. [6] conducted a study of privacy preservation for disease susceptibility evaluation. The susceptibility predictor is built with a multiplicative model [2] that uses genetic features as input. The authors empirically investigated the risk of input inference from disease susceptibilities and introduced an obfuscation method. Their method partitions the output domain into evenly partitioned sections and outputs one section instead of the susceptibility value. Their experimental evaluation reveals that application of their obfuscation method improves the statistical privacy measure (asymmetric entropy). Their obfuscation method improves privacy in a statistical sense but does not necessarily prevent unique identification of input attributes in some cases, as we discuss with experimentally obtained results in later sections.

Existing studies emphasize the risk that private input is learned by releasing outputs, while proactive protection before releasing outputs was not mainly considered, except in one study [7]. Their protection method employs differential privacy, which is recognized as a defacto-standard privacy definition. Differential privacy is known to be resistant against probabilistic inference by adversaries having any prior distribution [12]; however, Fredrikson et al. indicate differential privacy can be problematic, particularly when the output is used for medication. Mechanisms that guarantee differential privacy fundamentally requires randomization of outputs. This property of differentially private mechanisms can severely damage the utility of outputs. As discussed intensively by Fredrikson et al. [7], when the output is used for medical treatment, randomization of outputs is not a feasible option. For medical applications, we want the function output to be deterministic so that physicians can confirm by themselves that the medical treatment following the output is medically safe.

1.2 Our Contribution

The contribution of this paper is two-fold. First, we employed α-obscure privacy for output release. One drawback of differential privacy is that mechanisms that guarantee differential privacy fundamentally require randomization of outputs, as we already mentioned. Our α-obscure privacy

[1] Linkage disequilibrium is the association of frequency of appearance between SNPs. If patient's some SNPs are leaked, some other SNPs can be inferred using probabilistic inference with this association.

[2] Multiplicative model is a disease susceptibility model, which regards the susceptibility as the multiplicative accumulation of the risk of genetic factors.

assumes that prior knowledge is publicly known, and adversaries perform probabilistic inference following the known prior distribution. In this sense, the adversaries that differential privacy assumes are more powerful than those which α-obscure privacy assumes, which means that differential privacy offers more rigorous privacy. However, because of this assumption, our output can be made deterministic. Our mechanism preserves the privacy of outputs not by randomization but by sectionalization of outputs. Also, we can guarantee that the output interval always includes the true output and that the interval of the output can be deterministically confirmed by anyone. Consequently, this type of mechanism is desirable for privacy preservation of medical applications.

Second, we propose the optimal interval release mechanism which satisfies α-obscure privacy (Section 6). In principle, the mechanism that outputs the optimal (i.e., narrowest) interval with satisfying α-obscure privacy is intractable because we need to solve the optimization problem over the infinite set without any convex structure behind it (Problem A). To alleviate this difficulty, we use Theorem 1, which allows us to convert the intractable problem (Problem A) to a set partitioning problem (Problem B), which is solvable with dynamic programming. The time complexity of our mechanism is $O(|\mathbb{T}|^3 d)$; where d is the input dimension and $|\mathbb{T}|$ is the size of the range of target real-valued function. Unfortunately, the computational complexity is exponential; however, this cannot be problematic in many cases because the input features of the target model are screened as a preprocessing. In addition, once our the optimal interval is obtained by our mechanism, the interval can be repeatedly used for any request without spending additional costs.

Our mechanism guarantees the following two conditions:

- When the output is provided by the mechanism, the increase of the adversary's belief on any input attribute (α-obscure privacy) is upper-bounded using a level prescribed in the mechanism. Therefore, the mechanism can control the probability with which any input attribute is uniquely identified (privacy guarantee),

- Under the privacy constraint presented above, the mechanism outputs are the narrowest (optimal) interval including the true output (optimality guarantee).

2. PROBLEM FORMULATION

We first define the input inference problem as shown below.

2.1 Output Release

Let $\mathbb{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_d$ be a finite input domain. $\mathbb{Y}$ is an output domain in which $\mathbb{Y}$ is a finite discrete set. The function we consider is defined as $f \in \mathbb{F}$ where $f : \mathbb{X} \to \mathbb{Y}$. Given an input $\mathbf{x} \in \mathbb{X}$, the output $y \in \mathbb{Y}$ is given as $y = f(\mathbf{x})$. We describe the i-th attribute value of $\mathbf{x}$ as $x_i \in \mathcal{X}_i$ and describe the random variables of $\mathbf{x}$, x_i and y respectively as $\mathbf{X}$, X_i and Y.

Two stakeholders appear in the input inference problem: *publisher* and *adversary* (Fig.1). The publisher holds a sensitive private input $\mathbf{x} \in \mathbb{X}$, evaluates $y = f(\mathbf{x})$, and then publishes the output y to the adversary.

The adversary tries to identify the private input using the output obtained from the publisher using probabilistic inference. We assume the adversary has the complete knowledge of the function f and the prior distribution $\Pr[\mathbf{X}]$.

2.2 Input Inference by Adversaries

To learn the private input, the adversary estimates the posterior distribution (conditional distribution of input) $\Pr[X_i|Y = y]$. In some situations, the adversary might be interested in identifying a specific attribute of private inputs, but we assume that the adversary tries to infer every input attribute.

It is noteworthy that inputs are always uniquely determined when f is injective. In other words, if f is injective, the posterior probability $\Pr[X_i = a|Y = y]$ is 1 for some a. In practice, f is rarely injective. If f is not injective, the adversary tries to evaluate the posterior probability $\Pr[X_i = a|Y = y]$ to infer $\mathbf{x}$.

We introduce the probabilistic inference of the posterior distribution used by the adversary. Presuming that the adversary holds the prior $\Pr[\mathbf{X}]$ and obtains y from the publisher, then the adversary tries to infer the input by estimating the posterior $\Pr[X_i = a|Y = y]$. The posterior can be rearranged as

$$\Pr[X_i = a|Y = y] = \frac{\Pr[X_i = a, Y = y]}{\Pr[Y = y]}. \quad (1)$$

Given function $f : \mathbb{X} \to \mathbb{Y}$, the preimage of $\mathcal{Y} \subseteq \mathbb{Y}$ is defiend by

$$f^{-1}[\mathcal{Y}] = \{\mathbf{x} \in \mathbb{X} | f(\mathbf{x}) \in \mathcal{Y}\}.$$

Given the prior $\Pr[\mathbf{X} = \mathbf{x}]$, the denominator in Eq. 1 is evaluated as

$$\begin{aligned}\Pr[Y = y] &= \sum_{\mathbf{x} \in \{\mathbf{x} \in \mathbb{X} | f(\mathbf{x}) = y\}} \Pr[\mathbf{X} = \mathbf{x}] \\ &= \sum_{\mathbf{x} \in f^{-1}[\{y\}]} \Pr[\mathbf{X} = \mathbf{x}].\end{aligned}$$

In a similar manner, the numerator in Eq. 1 is evaluated as

$$\Pr[X_i = a, Y = y] = \sum_{\mathbf{x} \in \{\mathbf{x} \in \mathbb{X} | f(\mathbf{x}) = y \wedge x_i = a\}} \Pr[\mathbf{X} = \mathbf{x}].$$

The posterior is therefore derived as shown below

$$\Pr[X_i = a|Y = y] = \frac{\sum_{\mathbf{x} \in \{\mathbf{x} | f(\mathbf{x}) = y \wedge x_i = a\}} \Pr[\mathbf{X} = \mathbf{x}]}{\sum_{\mathbf{x} \in f^{-1}[\{y\}]} \Pr[\mathbf{X} = \mathbf{x}]}. \quad (2)$$

In principle, the probabilistic inference of inputs from outputs for general functions is intractable because the size of the input domain is exponential in the input dimension. For this study, we presume that the computational resources of the adversary are unlimited, which indicates that the adversary can evaluate $\Pr[X_i|Y = y]$ using exhaustive evaluation of $y = f(\mathbf{x})$ for every $\mathbf{x} \in \mathbb{X}$.

2.3 Mechanism

To prevent the adversary from estimating the private input, the publisher can modify or abstract the output so that the inputs are not uniquely identified. We designate an algorithm that modifies outputs for reducing the risk of input inference as a *mechanism.*

In the following sections, we restrict the output domain to the real. Also, we suppose mechanisms output real continuous intervals as the abstraction of function outputs. Thus, a mechanism is defined as a map $\mathcal{M} : \mathbb{R} \to \mathbb{I}$ where $\mathbb{I}$ is the set of real continuous intervals. The performance of mechanism is measured by privacy and utility, which are defined in the following subsections.

2.4 Privacy Measure

We define α-obscure privacy as the measure of the input inference risk of mechanisms. Intuitively, the probabilistic inference of the posterior helps the adversary to infer the private input if the posterior obtained as a result of inference is significantly different from the prior. A similar idea is presented in [15] as the *Uniformative Principle.* The notion of α-obscure privacy is defined based on this intuition.

Definition 1. *Let* $\Pr[X_i = x_i]$ *be the prior distribution of* x_i. *Let* $y = \mathcal{M}(\mathbf{x})$ *be the output of function* $\mathcal{M} : \mathbb{X} \to \mathbb{I}$. *Then* $\mathcal{M}$ *is* α*-obscure private with respect to the* i*-th input attribute if, for all* $a \in \mathcal{X}_i$ *and for all* $\mathbf{x} \in \mathbb{X}$,

$$|\Pr[X_i = a | Y = y] - \Pr[X_i = a]| \le \alpha.$$

Actually, α is equal to 0 if the posterior is equal to the prior, which indicates the inference gives no information about the input to the adversary. On the other hand, if the posterior is significantly different from the prior, then α takes a value greater than 0, which indicates that the inference gives information that helps the adversary to infer the private input. For any mechanism $\mathcal{M}$, α is bounded with the prior probabilities as Eq. 3

$$0 \le \alpha \le \max_{a \in \mathcal{X}_i} \max\{\Pr[X_i = a], 1 - \Pr[X_i = a]\}. \quad (3)$$

The proof, which is almost obvious, is omitted here.

The α-obscure privacy measures the privacy breach on only a single input attribute. We emphasize the necessity of devoting attention to α-obscure privacy for all attributes to assess the entire privacy breach. Additionally, we remark that we cannot compare α of different attributes directly. Actually, the α of an attribute is dependent on the prior probability of the attribute. The prior probabilities of attributes are all different. To compare the inference risk of two or more attributes, one must see both α and the prior probabilities.

The following condition similar to α-obscure privacy has been proposed to judge a privacy breach [16].

$$\frac{\Pr[X_i = a | Y = y]}{\Pr[X_i = a]} \le \gamma, \Pr[X_i = a] < \delta, \Pr[X_i = a | Y = y] < \sigma.$$

The following discussion in this paper fully works with the privacy definition above, but unless otherwise stated, we use α-obscure privacy as the privacy measure.

2.5 Utility Measure

The risk of inference will be low if there exist two or more inputs that give the same interval as its output. Hereinafter, we measure the risk of inference by α of the α-obscure privacy of the mechanism.

If the mechanism always returns a constant value or interval for any input, the adversary cannot obtain any information from the output, and the posterior remains unchanged from the prior. This is preferable in terms of privacy. However, as one might expect, such outputs are useless. To release meaningful outputs, one must consider both the privacy measure and the utility measure.

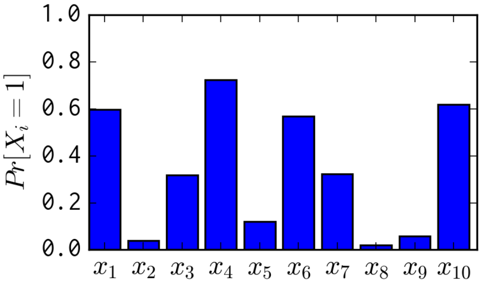


Figure 2: The prior (frequency) distribution of input attribute used for disease susceptibility models of obesity.

The output of a mechanism is more useful if the width of the interval is smaller, as long as the interval includes the output corresponding to the true input. The utility of a mechanism is therefore defined by a negative expected interval width as

$$\begin{aligned} \text{utility}(\mathcal{M}) &= -\mathrm{E}_{\mathbf{x} \in \mathbb{X}}[|\mathcal{M}(\mathbf{x})|] \\ &= -\sum_{\mathbf{x} \in \mathbb{X}} |\mathcal{M}(\mathbf{x})| \Pr[\mathbf{X} = \mathbf{x}], \end{aligned}$$

where $|\cdot|$ denotes the width of an interval.

Section 6 presents the design of a mechanism that guarantees privacy in terms of α-obscure privacy and which simultaneously achieves higher utility.

3. DISEASE SUSCEPTIBILITY MODELS

As an important example of input inference from outputs, we assess the risk of personal genome inference from disease susceptibility release. Disease susceptibility is known to be affected by both genetic factors and clinical factors.

Susceptibility models we used in this study are built with a dataset collected for an epidemiological study of chronic kidney disease [19]. We use the same data to build models of obesity. The dataset collected by the study includes 442 SNP profiles and 10 clinical features[3] of 5202 subjects.

We consider the susceptibility models of obesity. For evaluation of the susceptibility of obesity, 10 SNPs out of 442 SNPs are selected as the genetic features by hypothetical testing. In addition, the ten clinical features mentioned above are used for susceptibility modeling. The risk model is built with logistic regression using the selected SNPs and clinical features.

Let $\mathbf{x} \in \{0,1\}^d$ denote the genetic features and $\mathbf{x}_c \in \mathbb{R}^{d'}$ signify clinical features of a subject. The i-th input attribute value x_i of $\mathbf{x}$ is given as a Boolean value, which denotes whether or not the i-th SNP is the major (or minor) allele. $\mathbf{x}_c$ represents the clinical feature vector. All features are represented as binaries, integers, or real values.

Disease susceptibility $r \in \mathbb{R}$ is predicted using the following logistic regression

$$r = \sigma\left(\mathbf{w}^\intercal \mathbf{x} + \mathbf{w}_c^\intercal \mathbf{x}_c\right), \quad (4)$$

[3] Clinical features include age, sex, BMI, smoking history, blood creatinine, medical history of diabetes mellitus, hypertriglyceridemia, hypoalphalipoproteinemia, hyperbetalipoproteinemia, high blood pressure

where $\mathbf{w} \in \mathbb{R}^d, \mathbf{w}_c \in \mathbb{R}^{d'}$ are the model parameters and $\sigma(\cdot)$ donates the logistic sigmoid function.

Fig. 2 presents the prior probabilities $\Pr[X_i = 1]$ of 10 SNP features $(x_1, x_2, \ldots, x_{10})$ used in the disease susceptibility model for obesity.

4. PERSONAL GENOME INFERENCE FROM DISEASE SUSCEPTIBILITY RELEASE

As already described in Section 2, we assume that the adversary has full access to the model (e.g., $\mathbf{w}, \mathbf{w}_c$) and the prior distribution (e.g., the frequency distribution of the SNPs $\Pr[\mathbf{X}]$). When the susceptibility evaluation is used for medication, it is natural to assume that the model is publicly released. In addition, the frequency distributions (including correlations) of the SNPs are widely shared for research purposes. In this problem, we further presume that the adversary can learn the clinical features of targets (e.g., $\mathbf{x}_c$) as prior knowledge because we can expect that adversary can collect such common features of targets easily. In summary, as the background knowledge, we assume that the adversary has prior $\Pr[\mathbf{X}]$, model parameters $\mathbf{w}, \mathbf{w}_c$, and clinical features $\mathbf{x}_c$. In this setting, given the disease susceptibility r of a subject, the adversary attempts to identify SNPs of the subject by probabilistic inference.

By rearranging Eq. 4, we have $\mathbf{w}^\intercal \mathbf{x} = \sigma^{-1}(r) - \mathbf{w}_c^\intercal \mathbf{x}_c$. Because the adversaries have $r, \mathbf{w}_c$ and $\mathbf{x}_c$, the adversary can readily evaluate $\mathbf{w}^\intercal \mathbf{x}$. Consequently, letting

$$f(\mathbf{x}) = \mathbf{w}^\intercal \mathbf{x},$$

the problem of input inference is reduced to estimation of the posterior

$$\Pr\left[X_i \middle| \mathbf{X} \in f^{-1}\left[\{\mathbf{w}^\intercal \mathbf{x}\}\right]\right]$$

where $\mathbf{w}^\intercal \mathbf{x}$ is given as an output. The posterior probability is readily obtained by evaluation of Eq. 2.

We experimentally evaluated the posterior distribution of all possible genetic features when disease susceptibilities (obesity) were released with no modification. The significant digits of the model parameter $\mathbf{w}$ were changed from six to one. When the size of the input domain is fixed, the function f tends to be more injective as the number of significant digits increases.

The results are presented in Table 1. Figures in the table represent the rate of inputs (SNPs) that are identified uniquely when the outputs (disease susceptibilities) are revealed. From the result, it is apparent that if the significant digits of the model parameter are greater than five, then f becomes injective and reveals private genetic features completely. Even with one significant digit, more than 50 % of inputs are uniquely identified for three out of ten genetic features. In any settings shown in Table 1, the probability that unique identification occurs is not zero. This means that in terms of α-obscure privacy, α always reaches the upper bound.

In view of the fact that a model with one-digit precision can be useless for many medical applications, it is apparently difficult to balance utility and privacy when the disease susceptibility is revealed with no modification or abstraction.

5. RELEASING DISEASE SUSCEPTIBILITY WITH EQUALLY PARTITIONED INTERVAL

5.1 Equally Partitioning Mechanism

In the previous section, we experimentally demonstrated that the release of disease susceptibilities causes unique identification of SNPs even when the significant digits of the model parameters are few. To reduce the inference risk, abstraction of disease susceptibilities before release is necessary.

Ayday et al. introduced a mechanism that partitions the output domain evenly and which outputs the interval including the output value [6]. We briefly summarize the mechanism herein. Let $[t_{\min}, t_{\max}]$ be the output domain. The equally partitioning mechanism divides the output domain into n disjoint intervals. Given output t, the mechanism returns the interval that includes the output. The disjoint interval I_k is defined as

$$I_k = \begin{cases} \left[\frac{(n-k+1)t_{\min}+(k-1)t_{\max}}{n}, \frac{(n-k)t_{\min}+kt_{\max}}{n}\right) & (1 \le k < n) \\ \left[\frac{t_{\min}+(n-1)t_{\max}}{n}, t_{\max}\right] & (k = n) \end{cases}$$

where k is the index of the interval. We describe the set of all disjoint intervals as $\mathbf{I}_n = \{I_k\}_{k=1}^n$.

The rounding function round_n is defined as

$$\mathrm{round}_n(t) = I \text{ where } I \in \mathbf{I}_n \text{ such that } t \in I.$$

The equally partitioning mechanism $\mathcal{M}_{\mathrm{round}_n} : \mathbb{X} \to \mathbb{I}$ is defined as presented below.

$$\mathcal{M}_{\mathrm{round}_n}(\mathbf{x}) = \mathrm{round}_n \circ f(\mathbf{x}).$$

5.2 Utility and Privacy of Equally Partitioning Mechanism

As the partitioning number increases, the output interval width decreases, and the output becomes more exact. For instance, when $[t_{\min}, t_{\max}] = [0, 1]$ and $n = 5$, the interval set includes five intervals:

$$\left[0, \frac{1}{5}\right), \left[\frac{1}{5}, \frac{2}{5}\right), \left[\frac{2}{5}, \frac{3}{5}\right), \left[\frac{3}{5}, \frac{4}{5}\right), \left[\frac{4}{5}, 1\right].$$

When the input is 0.3, the rounding function returns $\left[\frac{1}{5}, \frac{2}{5}\right)$.

The utility is evaluated as

$$\mathrm{utility}(\mathcal{M}_{\mathrm{round}_n}) = -\frac{t_{\max} - t_{\min}}{n}.$$

The utility is markedly improved by increasing the partitioning number n.

The privacy guaranteed by the equally partitioning mechanism is data-dependent. We experimentally evaluated the risk of unique identification with the disease susceptibility of obesity, $y = f(\mathbf{x})$. Fig. 3 portrays the distribution of $f(\mathbf{x})$ for all possible points in the input domain. In the figure above, we divide $\mathbb{X}$ into two sets: $\mathbf{x} \in \mathbb{X}$ with $x_2 = 0$ and $\mathbf{x} \in \mathbb{X}$ with $x_2 = 1$. In the figure below, $\mathbb{X}$ is divided into two sets in the same manner with respect to x_6. The vertical lines in the figures display intervals when the output domain is partitioned into five sections. Given $\mathbf{x}$, the mechanism outputs one of these five intervals. The cross (green) and circle (blue) points in the figure above (resp. below) show the distribution of $f(\mathbf{x})$ when $x_2 = 1$ and $x_2 = 0$ (resp. $x_6 = 1$ and $x_6 = 0$).

Table 1: The rate of genetic features identified uniquely from the disease susceptibilities.

significant digit of $\mathbf{w}$	input attribute									
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
1	0.0441	0.4559	0.0441	0.0441	0.2500	0.1912	0.3382	0.6618	0.6618	0.6765
2	0.4886	0.7188	0.5341	0.5312	0.5625	0.4659	0.4886	0.6875	0.5483	0.4688
3	0.9650	0.8576	0.9813	0.9627	0.9405	0.9463	0.9533	0.8623	0.9592	0.9405
4	0.9950	1.0000	0.9821	0.9821	0.9861	0.9861	0.9950	1.0000	0.9821	0.9950
5	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
6	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

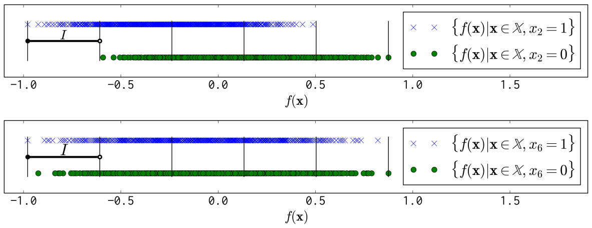


Figure 3: Distribution of $f(\mathbf{x})$ (obesity) for all possible points in the input domain. In the figure above, $\mathbb{X}$ is divided into two sets: $\mathbf{x} \in \mathbb{X}$ with $x_2 = 1$ and $\mathbf{x} \in \mathbb{X}$ with $x_2 = 0$. Also, $f(\mathbf{x})$ of $\mathbf{x}$ in the former and latter set are denoted respectively by blue cross points and green circle points. In the figure below, the distributions of $f(\mathbf{x})$ are shown with respect to x_6 in the same manner.

As the figure shows, it is readily apparent that when the leftmost interval I is released by the mechanism, one can uniquely identify that x_2 is 1 because this interval includes no cross (green) point. The same holds for the rightmost interval. x_2 is always 0 if the rightmost interval is released.

In contrast, as the figure below shows, the release of none of the intervals cause unique identification of x_6. This difference is explainable by the model parameter. Looking at details of the model parameter, w_2, the model parameter corresponding to x_2, has the largest absolute value, as $w_2 = -0.38$. However, the model parameter corresponding to x_6 has a smaller absolute value, as $w_6 = -0.05$. From these, it is apparent that x_2 has a greater impact on the outputs than x_6, and that x_2 is uniquely identified much more easily than x_6. In Figure Fig. 3, since there exists an interval that causes unique identification of x_2, α of α-obscure privacy w.r.t. x_2 reaches upper bound whereas α of α-obscure privacy w.r.t. x_6 does not. This means that the partitioning number of this mechanism can improve the privacy, but it can control neither α nor the probability of unique identification.

To conclude, the partitioning number should be set to less than four for this model to avoid this unique identification. The variation of the outputs of the original model is $|\mathbb{Y}| = 2^{10}$. Abstraction of the outputs by four sections might be acceptable if the outputs are disease susceptibilities. However, if the outputs are the dosage amount of personalized medicine, then four-level categorization can be too abstract.

6. RELEASING DISEASE SUSCEPTIBILITY WITH OPTIMAL INTERVAL

In this section, we propose a mechanism that is optimal in the sense of the utility defined in Section 2 under the constraints of α-obscure privacy. The optimal interval mechanism is defined as a mechanism that outputs the narrowest interval with the guarantee that the input inference risk (α-obscure privacy) is kept lower than acceptable levels specified by the publisher. The mechanism we present in this section can explicitly restrict α of α-obscure privacy as constraints. In the following, α denotes the privacy measure of intervals provided by the mechanism. On the other hand, $\tilde{\alpha}$ represents the constraint on the privacy measure, which is given as the input of the mechanism before releasing the intervals. We call α privacy measure and $\tilde{\alpha}$ privacy budget. We formulate the problem of finding the optimal interval mechanism as a constraint maximization problem. Then we derive an algorithm that derives the optimal interval mechanism.

6.1 Mechanism Design by Constraint Optimization

Two constraints are necessary for the mechanism to output useful intervals. First, given x, the output interval of the mechanism should include $f(\mathbf{x})$. This constraint is necessary because false intervals can be released without this constraint. This constraint is represented by the following expression.

$$\forall_{\mathbf{x}\in\mathbb{X}}, f(\mathbf{x}) \in \mathcal{M}(\mathbf{x})$$

Second, we require that all intervals released by the mechanism be mutually disjoint. For example, let $f(\mathbf{x}_1) = 1.2$ and $f(\mathbf{x}_2) = 1.2$ be the outputs of $\mathbf{x}_1$ and $\mathbf{x}_2$. Also, let $[0, 2] = \mathcal{M}(\mathbf{x}_1)$ and $[1, 3] = \mathcal{M}(\mathbf{x}_2)$ be the corresponding output intervals of the mechanism. This mechanism satisfies the first constraint. However, the mechanism outputs different intervals for same true outputs; $f(\mathbf{x}_1) = f(\mathbf{x}_2)$. If this happens, one might misunderstand that the second output shows higher susceptibility than the first. To avoid this misunderstanding, we introduce the second constraint, which requires that all the intervals be mutually disjoint:

$$\forall_{\mathbf{x},\mathbf{x}'\in\mathbb{X},\mathcal{M}(\mathbf{x})\neq\mathcal{M}(\mathbf{x}')}, \mathcal{M}(\mathbf{x}) \cap \mathcal{M}(\mathbf{x}') = \emptyset$$

Let $\mathbb{I}$ be the set of intervals over the output domain of f. The mechanism is defined as a map from an input to an interval, $\mathcal{M} : \mathbb{X} \to \mathbb{I}$. Putting the privacy constraint and the two constraints for the intervals all together, we can formulate the optimization problem A to ascertain the optimal interval. Let $\mathbb{M}$ be the set of mapping $\mathbb{X} \to \mathbb{I}$.

$$\begin{aligned} \underset{\mathcal{M}\in\mathbb{M}}{\text{maximize}} \quad & \text{utility}\,(\mathcal{M}) \\ \text{subject to} \quad & \forall_{1\leq i\leq d}, \mathcal{M} \text{ is } \tilde{\alpha}_i\text{-obscure private} \\ & \forall_{\mathbf{x},\mathbf{x}'\in\mathbb{X},\mathcal{M}(\mathbf{x})\neq\mathcal{M}(\mathbf{x}')}, \mathcal{M}(\mathbf{x}) \cap \mathcal{M}(\mathbf{x}') = \emptyset \\ & \forall_{\mathbf{x}\in\mathbb{X}}, f(\mathbf{x}) \in \mathcal{M}(\mathbf{x}). \end{aligned} \tag{A}$$

Because $\mathbb{M}$ is an infinite set and because this has no structure behind, this optimization problem is intractable. To solve this problem, we introduce the notions of the α-obscure set and α-obscure private function.

Definition 2. *Let $\mathcal{S} \subseteq \mathbb{X}$. $\mathcal{S}$ is an α-obscure set with respect to the i-th attribute if, for all $a \in \mathcal{X}_i$,*

$$|\Pr[x_i = a | \mathbf{X} \in \mathcal{S}] - \Pr[x_i = a]| \leq \alpha.$$

Let π be a partition of $\mathbb{X}$. We define function $\phi_\pi : \mathbb{X} \to \pi$ with partition π as

$$\phi_\pi(\mathbf{x}) = \mathcal{S} \text{ where } \mathcal{S} \in \pi \text{ such that } \mathbf{x} \in \mathcal{S}.$$

The following theorem characterizes mechanisms with α-obscure privacy by partitions for which the elements are all α-obscure sets.

Theorem 1. *Let π be a partition of $\mathbb{X}$. Then, ϕ_π is an α-obscure private function with respect to the i-th attribute if and only if every element of π is an α-obscure set with respect to the i-th attribute.*

Proof. The preimage of $\mathcal{S} \in \pi$ with ϕ_π is

$$\phi_\pi^{-1}[\{\mathcal{S}\}] = \mathcal{S}.$$

When ϕ_π is α-obscure private with respect to the i-th attribute, the following holds.

$$\begin{aligned} &\forall_{a\in\mathcal{X}_i}\forall_{\mathbf{x}\in\mathbb{X}}, |\Pr[X_i = a|Y = \phi_\pi(\mathbf{x})] - \Pr[X_i = a]| \leq \alpha \\ \Leftrightarrow &\forall_{a\in\mathcal{X}_i}\forall_{\mathcal{S}\in\pi}, \left|\Pr\left[X_i = a\middle|\mathbf{X} \in \phi_\pi^{-1}[\{\mathcal{S}\}]\right] - \Pr[X_i = a]\right| \leq \alpha \\ \Leftrightarrow &\forall_{a\in\mathcal{X}_i}\forall_{\mathcal{S}\in\pi}, |\Pr[X_i = a|\mathbf{X} \in \mathcal{S}] - \Pr[X_i = a]| \leq \alpha \end{aligned}$$

By Definition 2, the last statement means that $\mathcal{S}$ is α-obscure set with respect to the i-th attribute for any $\mathcal{S} \in \pi$, which concludes the proof. □

Accordingly, we can obtain an $\tilde{\alpha}$-obscure private function by designing π, a partition of $\mathbb{X}$, so that all elements of π are $\tilde{\alpha}$-obscure sets. ϕ_π is useful as a mechanism with the guarantee of $\tilde{\alpha}$-obscure privacy, but it outputs a set of $\mathbf{x}$, not interval. To transform ϕ_π to a mechanism that outputs an interval, we apply a function $\psi : \mathcal{P}(\mathbb{X}) \to \mathbb{I}$ to ϕ_π where $\mathcal{P}(\mathbb{X})$ is the power set of $\mathbb{X}$:

$$\psi(\mathcal{S}) = \left[\min_{\mathbf{x}\in\mathcal{S}} f(\mathbf{x}), \max_{\mathbf{x}\in\mathcal{S}} f(\mathbf{x})\right].$$

Using ϕ_π and ψ, we can define a mechanism $\mathcal{M}_\pi$ that outputs an interval as

$$\mathcal{M}_\pi(\mathbf{x}) = \psi \circ \phi_\pi(\mathbf{x}).$$

By Theorem 1, the problem of finding an $\tilde{\alpha}$-obscure mechanism is reduced to the problem of finding a partition for which the elements are all $\tilde{\alpha}$-obscure sets. We can next reformulate Problem A as the following optimization problem of $\mathcal{M}_\pi$:

$$\begin{aligned} \underset{\pi\in\Pi}{\text{maximize}} \quad & \text{utility}(\mathcal{M}_\pi) \\ \text{subject to} \quad & \forall_i \forall_{\mathcal{S}\in\pi}, \mathcal{S} \text{ is } \tilde{\alpha}_i\text{-obscure set} \\ & \forall_{\mathcal{S},\mathcal{S}'\in\pi,\mathcal{S}\neq\mathcal{S}'}, \psi(\mathcal{S}) \cap \psi(\mathcal{S}') = \emptyset. \end{aligned} \tag{B}$$

One can show that the solution to Problem A is equivalent to the solution to Problem B. Let $\mathcal{M} : \mathbb{X} \to \mathbb{I}$ be a mechanism that satisfies the first constraint (privacy constraint) in Problem A. Because $\mathcal{M}$ satisfies the privacy constraint in Problem A, if we group all $\mathbf{x} \in \mathbb{X}$ by $\mathcal{M}(\mathbf{x})$, then it always forms a partition π of $\mathbb{X}$ so that each element of π is $\tilde{\alpha}$-obscure set by Theorem 1.

Given such π, let us consider a mechanism that outputs an interval that includes all values in $\{f(\mathbf{x})|\mathbf{x} \in \phi_\pi(\mathbf{x})\}$ to satisfy the third constraint in Problem A. There are infinitely many mechanisms satisfying the condition, but the mechanism with the largest utility among such mechanisms is readily determined as $\mathcal{M}_\pi(\mathbf{x})$ because ψ outputs the narrowest interval that covers all elements in $\{f(\mathbf{x})|\mathbf{x} \in \phi_\pi(\mathbf{x})\}$.

Noting that enumeration of $\mathcal{M} \in \mathbb{M}$ with the first constraint in Problem A is equivalent to enumeration of partition π of $\mathbb{X}$ for which elements are $\tilde{\alpha}$-obscure sets, the solution to Problem B is equivalent to that of Problem A.

6.2 Partitioning Algorithm

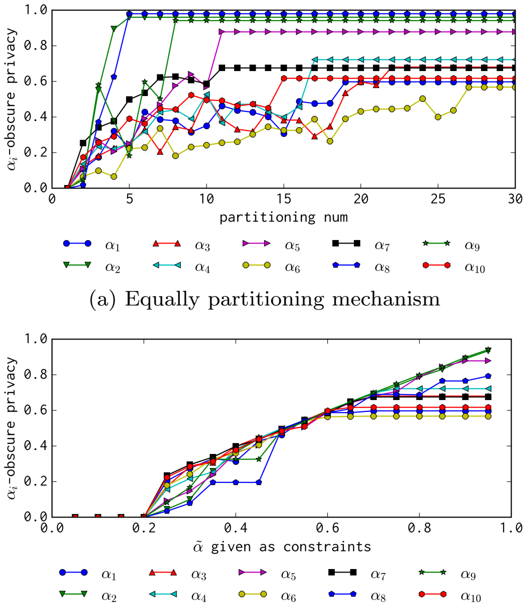
Algorithm 1 Partitioning Algorithm

```
Require: Input dimension:$d$, real-valued function:$f$, privacy
    budget:$\{\tilde{\alpha}_i\}_{1\leq i\leq d}$
Ensure: optimal partition of $\mathbb{X}$: $\pi^*$
$m_\pi[0] \leftarrow \emptyset$
$m_{\text{exist}}[0] \leftarrow$ true
for $i : 1, \cdots, |\mathbb{T}|$ do
    $\pi_i^* \leftarrow \emptyset$
    $\text{u}_i^* \leftarrow -\infty$
    $\text{exist}_i \leftarrow$ false
    for $j : 0, \cdots, i-1$ do
        $T_{i,j} \leftarrow \{t_k\}_{j+1\leq k\leq i}$
        $S_{i,j} \leftarrow f^{-1}[T_{i,j}]$
        if $\forall_{1\leq k\leq d}, S_{i,j}$ is $\tilde{\alpha}_k$-obscure set on $k$-th attribute
    $\wedge m_{\text{exist}}[j]$ then                ▷ eliminating non-private partition
            $\pi_{i,j} \leftarrow \{S_{i,j}\} \cup m_\pi[j]$
            $\text{u}_{i,j} \leftarrow \sum_{\mathcal{S}\in\pi_{i,j}} \sum_{\mathbf{x}\in\mathcal{S}} -\Pr[\mathbf{X} = \mathbf{x}]\,|\psi(\mathcal{S})|$
            if $\sigma_i^* = \emptyset \vee \text{u}_i^* < \text{u}_{i,j}$ then
                $\pi_i^* \leftarrow \pi_{i,j}$
                $\text{u}_i^* \leftarrow \text{u}_{i,j}$
                $\text{exist}_i \leftarrow$ true
            end if
        end if
    end for
    $m_\pi[i] \leftarrow \pi_i^*$
    $m_{\text{exist}}[i] \leftarrow \text{exist}_i$
end for
$\pi^* \leftarrow m_\pi[|\mathbb{T}|]$
```

We propose an algorithm (Alg. 1) to solve Problem B. In the algorithm, $\mathbb{T}$ denotes the image of $\mathbb{X}$ with f, i.e., $\mathbb{T} = \{f(\mathbf{x})|\mathbf{x} \in \mathbb{X}\}$. Also, t_i represents the i-th largest value in $\mathbb{T}$. This algorithm takes the input dimension d, real-valued function f, the privacy budget $\{\tilde{\alpha}_i\}_{1\leq i\leq d}$, and outputs the optimal partition π^* of $\mathbb{X}$, which induces the optimal interval mechanism that satisfies at least $\tilde{\alpha}_i$-obscure privacy for any i.

We distinguish $\tilde{\alpha}_i$ as the privacy budget given as constraints and α_i as the privacy measure assessed by the obtained mechanism. The solution to Problem B guarantees that the mechanism satisfies at least $\tilde{\alpha}_i$-obscure privacy, but the realization of privacy guarantee is not tight, i.e., $\tilde{\alpha}_i \geq \alpha_i$.

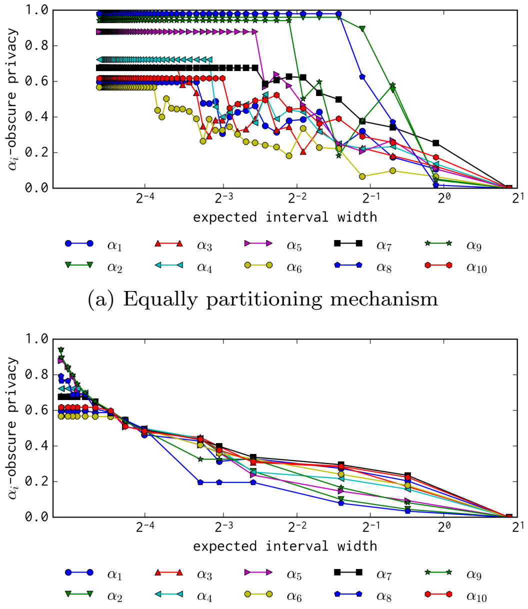
This algorithm solves the set partitioning problem of $\mathbb{T}$ using dynamic programming. In the algorithm, m stands for the dynamic programming table. $m_\pi[i]$ includes the optimal partition of $f^{-1}\left[\{t_k\}_{1\leq k\leq i}\right]$ and the elements of the partition are $\tilde{\alpha}_l$-obscure set for all attribute l if $m_{\text{exist}}[i]$ is true. The algorithm searches the $\tilde{\alpha}_l$-obscure set and the



(a) Equally partitioning mechanism

(b) Optimal interval mechanism

Figure 4: α-obscure privacy vs. (partitioning number n or privacy budget $\tilde{\alpha}$, applying to f of obesity. The mechanism achieves better privacy if α_i is small.



(a) Equally partitioning mechanism

(b) Optimal interval mechanism

Figure 5: α-obscure privacy vs. expected interval width, applying to f of obesity. The mechanism achieves better privacy if α_i is small. The output of mechanism has high utility if the expected interval width is small.

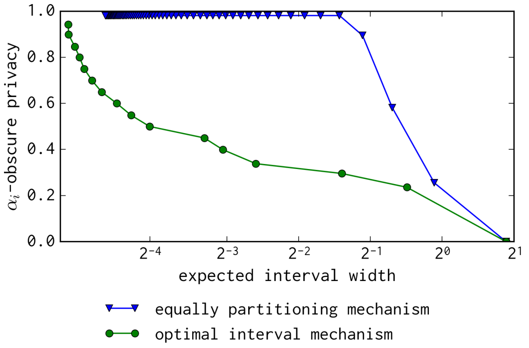


Figure 6: Expected interval width vs. the worst α among all attributes (obesity). The mechanism achieves better privacy if α_i is small. The output of mechanism has high utility if the expected interval width is small.

optimal partition of $f^{-1}\left[\{t_k\}_{1\leq k\leq i}\right]$ by increasing i in the loop starting from line 3. If such a partition exists, then the partition is memorized at line 20, and the flag is set to true at line 21. The flag is set to false at line 21 if it does not exist.

At line 4-6, variables are initialized. In the loop starting from line 7, the algorithm divides t_k as $\{t_k\}_{1\leq k\leq j}$ and $\{t_k\}_{j+1\leq k\leq i}$. If $S_{i,j} = f^{-1}\left[\{t_k\}_{j+1\leq k\leq i}\right]$ is an $\tilde{\alpha}_l$-obscure set and $m_{\text{exist}}[j]$ is true, letting $\pi_{i,j} = \{S_{i,j}\} \cup m_\pi[j]$, then $\pi_{i,j}$ is the set of $\tilde{\alpha}_l$-obscure set, and $\pi_{i,j}$ is a partition of $f^{-1}\left[\{t_k\}_{1\leq k\leq i}\right]$ (line 10-11). The algorithm calculates utility $u_{i,j}$ at line 12, and compares the utility to the intermediate solution π_i^* (line 12). The algorithm updates the intermediate solution (line 14-16) if partition $\pi_{i,j}$ has better utility than π_i^*. After the loop of j, the algorithm finds the $\tilde{\alpha}_l$-obscure set and the optimal partition π_i^* in $f^{-1}\left[\{t_k\}_{1\leq k\leq i}\right]$. By increasing i, the algorithm searches π_i^* and stores π_i^* in $m_\pi[i]$ in the same manner. Finally, the algorithm finds $m_\pi[\ |\mathbb{T}|\ ]$ that is optimal and $\tilde{\alpha}_l$-obscure partition of $\mathbb{T}$.

The computational complexity of this algorithm is $O\left(|\mathbb{T}|^3 d\right)$. Because $|\mathbb{T}|$ is exponential in d, the size of the input domain and the computation complexity is exponential in d.

7. DISCUSSION

This section presents an experimental comparison of the equally partitioning mechanism and the optimal interval mechanism in terms of the following two aspects. First, we compare the utility and the privacy of the two mechanisms changing the control parameters of each mechanism. Second, we present the intervals obtained using the optimal interval mechanism and compare them with the equally partitioning mechanism. Third, we compare the utility of our optimal partitioning mechanism and the Laplace mechanism that guarantee differential privacy with various privacy budgets.

As already described in Section 2, we assume the adversary uses the prior $\Pr[\mathbf{X}]$ as the background knowledge. Given numerous input attributes, it is difficult to evaluate the prior empirically from a limited number of samples in reality. In the experiments, we assumed that input attributes are mutually independent and approximate the prior by

$$\Pr[\mathbf{X} = \mathbf{x}] \simeq \prod_{1 \leq i \leq d} \Pr[X_i = x_i].$$

If the adversary has more precise background knowledge (e.g. joint probabilities on two input attributes), then we can use more advanced estimation methods such as the Markov-chain Monte-Carlo method or the belief propagation method. Our discussions can be readily applied to such advanced estimation methods without loss of generality, but we employ the posterior estimation of Eq. 2 for this study.

7.1 Comparison of Privacy and Utility

This section presents an experimental comparison of the equally partitioning mechanism and the optimal interval mechanism in terms of α-obscure privacy and utility, changing control parameters: partitioning number n for the equally partitioning mechanism and $\tilde{\alpha}$ for the optimal interval mechanism.

Let d represent the number of input attributes. To evaluate α of the equally partitioning mechanism, we changed the partitioning number from 1 to 30 and assessed α_i for each attribute. Fig. 4, Fig. 5, and Fig. 6 represent results of the susceptibility model of obesity.

Fig. 4a shows the partitioning number vs. the value of α_i for each genetic feature when the equally partitioning mechanism is used. The upper bound of α_i relies on the prior probability of the genetic feature as shown by Eq. 3. As the partitioning number increases, α_i of each genetic feature also increases non-monotonically[4]. We remark that the equally partitioning mechanism cannot control α_i directly, which makes it difficult to control the risk of unique identification. In Fig. 4a, α_i of x_2 and x_8 reach the upper bound when the partitioning number is five. Therefore, unique identification of x_2 and x_8 occurs if the partitioning number is equal to or less than four. In addition, this cannot be known without empirical evaluation.

In Fig. 4a, α_i of x_2 and x_8 increase rapidly as the number of partitions increases. It reaches the upper bound when the partitions are five. In contrast, for example, α of x_6 increases slowly and reaches the upper bound when the number of partitions is 27. This is because the model parameters of x_2 and x_8 have large absolute values while that of x_6 is close to zero. When the absolute value is large, it strongly affects the output values which mades input inference easy. Arranging the absolute values of the model parameters for obesity in descending order, we have $w_2, w_8, w_9, w_7, w_5, w_{10}, w_4, w_1, w_3$ and w_6. It is apparent that α_i reaches the upper bound almost in the same order.

Fig. 5a shows the relation between expected interval width (negative utility) and α_i of the equally partitioning mechanism with changing the partitioning number. As the figure shows, the privacy and utility share a tradeoff relation.

To evaluate α_i of the optimal interval mechanism, we changed $\tilde{\alpha}$ of the privacy constraints from 0.0 to 1.0 and solved the Problem B. For each resulting solution (each optimal interval), we evaluated α_i of each genetic feature. Since the optimal interval mechanism is given as the solution to Problem B, we remark that privacy measure α_i of intervals provided by the resulting mechanism is always less than privacy budget $\tilde{\alpha}$. In our problem formulation, $\tilde{\alpha}_i$ of each genetic feature can be controlled individually, but we simplify $\tilde{\alpha}_i$ so that α_i for all genetic features are equal to or less than a specified value $\tilde{\alpha}$.

Fig. 4b portrays the plot of $\tilde{\alpha}$ vs. α_i for each genetic feature when the optimal interval mechanism is used. From this figure, one can confirm that the optimal interval mechanism satisfies $\tilde{\alpha}$-obscure privacy on all genetic features. Differently from the equally partitioning mechanism, α_i (horizontal axis) never exceeds $\tilde{\alpha}$ given as the constraint (vertical axis). Since α_i is constrained by $\tilde{\alpha}$, no genetic feature can be uniquely identified as long as $\tilde{\alpha}$ is less than the upper bound of Eq. 3.

To compare Fig. 5a and Fig. 5b, we draw the worst α_i and expected interval width in Fig. 6. As the Fig. 6 shows, the optimal interval mechanism dominates the equally partitioning mechanism in term of utility and privacy.

7.2 Comparison of Obtained Intervals

In this section, we compare the utility of the two mechanisms with almost identical privacy level.

We first generated intervals by the equally partitioning mechanism ($k = 6, 8, 10$), and evaluated the utility(expected interval width) and α_i for each input attribute x_i. The values of α_i are found in the caption of Figs. 8a, 8c, and 8e.

Next, we generated the optimal interval using Algorithm 1. To compare the utility of the optimal interval mechanism with the equally partitioning mechanism fairly, in terms of α-obscure-privacy, $\tilde{\alpha}_i$ are determined as

$$\tilde{\alpha_i} = \begin{cases} \max_{a \in \mathcal{X}_i} \max\{\Pr[X_i = a], 1 - \Pr[X_i = a]\} - 0.01, \\ \qquad \text{if input attribute } i \text{ is uniquely identified} \\ \qquad \text{when equally partitioning mechanism is used,} \\ \alpha_i \text{ of the equally partitioning mechanism, o.w.} \end{cases}$$

By doing so, the resulting intervals are guaranteed to achieve equal or stronger privacy while avoiding unique identification than the intervals generated by the equally partitioning mechanism.

The resulting optimal partitions are presented in Figs. 8b, 8d, and 8f. The utility and α_i of the optimal interval are presented in the caption, too. According to the value of α_i and utility, it is apparent that the optimal partitioning mechanism achieves stronger privacy and provides better utility simultaneously. Vertical lines show the intervals generated by the mechanism. The optimal interval mechanism apparently provides narrower (that is, more exact) outputs than the equally partitioning mechanism.

Regarding the interval of the equally partitioning mechanism, the leftmost (resp. rightmost) interval includes points with $x_2 = 1$ (resp. points $x_2 = 0$) only. Therefore, x_2 can be identified uniquely by the mechanism outputs. For all results of the optimal interval mechanism, it is apparent that all intervals include both points. Consequently, the adversary cannot uniquely identify any input attribute by the mechanism outputs.

To conclude, the optimal interval mechanism has better performance than the equally partitioning mechanism in terms of privacy and utility. In addition, the optimal interval mechanism can specifically control α_i for each input

[4] α_i increases non-monotonically because $\{f(\mathbf{x}) | \mathbf{x} \in \mathbb{X}\}$ is discrete and $I' \subset I$ does not always hold for some $I \in \mathbf{I}_n$ and $I' \in \mathbf{I}_{n+1}$. If the partitioning number n is set to 2^k, α-obscure privacy increases monotonically because $I' \subset I$ always holds for any $I \in \mathbf{I}_{2^k}$ and any $I' \in \mathbf{I}_{2^{k+1}}$.

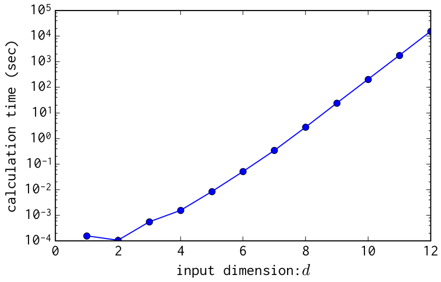


Figure 7: Calculation time of Alg. 1. We implemented the application on Rust language [4]. The average execution time of 100 trials are plotted.

attribute. This control helps us to avoid unique identification of inputs.

7.3 Empirical Analysis of Computational Efficiency

We analyze the computational efficiency of the proposed algorithm empirically using synthetic datasets. In these experiments, we used a linear function $f(\mathbf{x}) = \mathbf{w}^\intercal \mathbf{x}$ for which the input dimension is d and the input domain $\mathbb{X}$ is $\{0,1\}^d$. Here, $\mathbf{w}$ was set as a random vector drawn from the uniform distribution over $[0,1]^d$. The significant digits of $\mathbf{w}$ were set to 10. For each i, the prior probability $\Pr[X_i = 1]$ was set as a random value drawn from the uniform distribution over $[0,1]$. In addition, α_i was chosen from the uniform distribution over $[0,1]$. We randomly generate $\mathbf{w}, \alpha$ and prior probabilities for each trial and evaluated the computation time necessary to build the optimal partitioning π^* with the change of the input dimension as $d = 1, ..., 12$. We implemented the algorithm with Rust language [4]. The program was run on a personal computer with an Intel Xeon 2.60 GHz CPU and 32GB RAM. No parallelization technique was used.

In Fig 7, we present the average computation time of 100 trials. As the figure shows, the calculation time is exponential in d. When the input dimension is 12, the calculation time is about 15,400 s ($\simeq$ 4 h).

In general, the number of SNPs is quite large (e.g. 3 million). We do not claim that our algorithm always works in a practical time when the input dimension is large. However, the number of SNPs that are fundamentally necessary for medical applications is not typically that large because only a limited number of genetic features typically affect on our constitutions or traits. In the drug administration example [7], only two SNPs were used to evaluate the proper dosage of warfarin. In the disease susceptibility models of obesity used for these experiments, 10 SNPs, were used as inputs after a proper screening process. In addition, once the optimal interval is built, the mechanism can provide an output of the model using the interval repeatedly. Considering these points, there remains room for improvement of the computation time. However, our algorithm can be sufficiently practical with real-world applications.

7.4 Empirical Comparison of Utility to Differential Privacy

We compare our optimal interval mechanism to the Laplace mechanism with the disease susceptibility models. Our mechanism guarantees $\tilde{\alpha}$-obscure privacy while the Laplace mechanism guarantees ϵ-differential privacy. Since the privacy definitions are not the same, the outputs are not directly comparable. Instead, we compare the utility of two mechanisms. Given $\tilde{\alpha}$ and inputs, our mechanism outputs intervals. For the intervals, we evaluate the probability that the differentially private mechanism returns output within the intervals for various ϵ. If the probability is high, the output distribution of the differentially private mechanism is close to the interval of our mechanism. In contrast, if the probability is low, the output of the differentially private mechanism distributes broadly outside of the interval. We call this probability inclusion probability.

First, we introduce a differential privacy mechanism for disease susceptibility models. Since the input is a binary vector, we define the adjacent datasets as two vectors whose attribute values are the same except one attribute. The global sensitivity of model f is given as $\max_i |w_i|$. Based on the sensitivity method, the Laplace mechanism $\mathcal{M}_\epsilon$ adds Laplace noise $\mathrm{Lap}(0,b)$ to predicted susceptibility $f(\mathbf{x})$. The range of the susceptibility is bounded. To avoid the output of the Laplace mechanism takes a value outside of the range, the output is rounded as a postprocess. This rounding process is called cutoff.

$$\mathcal{M}_\epsilon(\mathbf{x}) = \max\left\{\min\left\{\bar{y}, t_{\max}\right\}, t_{\min}\right\}$$
$$\text{where } \bar{y} = f(\mathbf{x}) + \mathrm{Lap}\left(0, \frac{\max_i |w_i|}{\epsilon}\right)$$

Fig. 9a portrays the inclusion probability that the output of the Laplace mechanism is contained in the output interval the optimal interval when α and ϵ is changed.

$$\begin{aligned}&\mathrm{E}_{\mathbf{x}\in\mathbb{X}}\left[\Pr\left[\mathcal{M}_\epsilon(\mathbf{x}) \in \mathcal{M}_\pi(\mathbf{x})\right]\right]\\&= \sum_{\mathbf{x}\in\mathbb{X}} \Pr\left[\mathcal{M}_\epsilon(\mathbf{x}) \in \mathcal{M}_\pi(\mathbf{x})\right] \Pr\left[\mathbf{X} = \mathbf{x}\right]\end{aligned}$$

In Fig. 9a, when $\alpha = 0.2$, the inclusion probability is almost equal to 1.0 for all ϵ. This means that the output of the both mechanisms almost distributes in the same region with probability 1. Now that differential privacy offers stronger privacy guarantee than α-obscure privacy, the differentially private mechanism is preferred in the setting. In the same figure, when $\alpha = 0.4$, the inclusion probability is less than 0.1 for $\epsilon = 0.1$. This means that the output of the 0.1-differentially private mechanism is located the outside of the interval of the 0.4-optimal interval mechanism with probability 0.9. In this setting, we can see that the 0.4-optimal interval mechanism gives much better utility than 0.1-differentially private mechanism.

Fig. 9b depicts the same inclusion probability, but the x-axis is set to the expected interval width of the optimal interval mechanism. In this figure, for example, we can see that when the expected interval width of the intervals that the optimal interval mechanism outputs are 0.4 (between 2^{-1} and 2^{-2}), the inclusion probability that 0.1-differentially private mechanism gives outputs in the interval with length 0.4 is less than 0.2.

As we see from the both figures, the outputs of differentially private mechanisms distribute broadly with high probability even with large ϵ, and useful outputs are obtained with very low probability. Thus, particularly when large deviation from the true output is not acceptable, the optimal interval mechanism with α-obscure privacy can be an alternative.

8. CONCLUSION

As described in this paper, we defined the input inference problem in which the input domain is finite and discrete. Also, we introduced α-obscure privacy as the measure of the input inference risk. Taking genome-based disease susceptibility predictors as an example, we demonstrated that disclosure of raw susceptibility values is equivalent in some cases to publishing the input value. Using an existing obfuscation method (equally partitioning mechanism), we confirmed that the input can be uniquely determined even when the output is obfuscated with a large interval.

We show that designing a mechanism that achieves α-obscure privacy is equivalent to a constrained optimization problem of partitioning of the input domain. Based on this understanding, we designed a mechanism that releases optimal intervals instead of output values. The computational complexity of our algorithm is exponential in the input dimension; improvement of the optimization algorithm remains as our future work.

In this paper, we considered interval release for a single function while interval release for multiple functions is often required for real-world applications (e.g., disease susceptibilities to several common diseases). In such a case, the mechanism needs to output a cuboid that ensures α-obscure privacy. Most of the discussions related to Theorem 1 hold for the multiple output case; however, the optimal cuboid release is a far more difficult problem than the optimal interval release problem. Efficient solutions for optimal cuboid release remains as future work.

9. ACKNOWLEDGMENT

The authors would like to thank Yoshiji Yamada who collected data used for the experiments. The work is supported by JST CREST (Advanced Core Technologies for Big Data Integration) and JSPS KAKENHI 16H02864.

10. REFERENCES

[1] 23andMe - DNA Genetic Testing & Analysis. https://www.23andme.com/.

[2] Counsyl | DNA screening for the important moments in life. https://www.counsyl.com/.

[3] Pathway Genomics. https://www.pathway.com/.

[4] The Rust Programming Language. https://www.rust-lang.org/.

[5] E. Ayday, J. L. Raisaro, and J.-P. Hubaux. Personal use of the genomic data: privacy vs. storage cost. In *2013 IEEE Global Communications Conference (GLOBECOM '13)*, pages 2723–2729, 2013.

[6] E. Ayday, J. L. Raisaro, J.-P. Hubaux, and J. Rougemont. Protecting and evaluating genomic privacy in medical tests and personalized medicine. In *Proceedings of the 12th ACM Workshop on Privacy in the Electronic Society (WPES '13)*, pages 95–106, New York, NY, USA, 2013. ACM.

[7] M. Fredrikson, E. Lantz, S. Jha, S. Lin, D. Page, and T. Ristenpart. Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing. In *23rd USENIX Security Symposium (USENIX Security 14)*, pages 17–32, 2014.

[8] M. T. Goodrich. The mastermind attack on genomic data. In *2009 30th IEEE Symposium on Security and Privacy (S&P '09)*, pages 204–218. IEEE, 2009.

[9] N. Homer, S. Szelinger, M. Redman, D. Duggan, W. Tembe, J. Muehling, J. V. Pearson, D. A. Stephan, S. F. Nelson, and D. W. Craig. Resolving individuals contributing trace amounts of DNA to highly complex mixtures using high-density SNP genotyping microarrays. *PLoS Genet*, 4(8):e1000167, 2008.

[10] M. Humbert, E. Ayday, J.-P. Hubaux, and A. Telenti. Addressing the concerns of the Lacks family: quantification of kin genomic privacy. In *Proceedings of the 20th ACM SIGSAC conference on Computer & communications security (CCS '13)*, pages 1141–1152. ACM, 2013.

[11] J. A. Johnson, L. Gong, M. Whirl-Carrillo, B. F. Gage, S. A. Scott, C. M. Stein, J. L. Anderson, S. E. Kimmel, M. T. Lee, M. Pirmohamed, M. Wadelius, T. E. Klein, and R. B. Altman. Clinical Pharmacogenetics Implementation Consortium Guidelines for CYP2C9 and VKORC1 genotypes and warfarin dosing. *Clin. Pharmacol. Ther.*, 90(4):625–629, Oct 2011.

[12] S. P. Kasiviswanathan and A. Smith. On the 'semantics' of differential privacy: A bayesian formulation. *Journal of Privacy and Confidentiality*, 6(1):1, 2014.

[13] Z. Lin, A. B. Owen, and R. B. Altman. Genomic research and human subject privacy. *Science*, 305(5681):183–183, 2004.

[14] J. E. Lunshof, R. Chadwick, D. B. Vorhaus, and G. M. Church. From genetic privacy to open consent. *Nature Reviews Genetics*, 9(5):406–411, 2008.

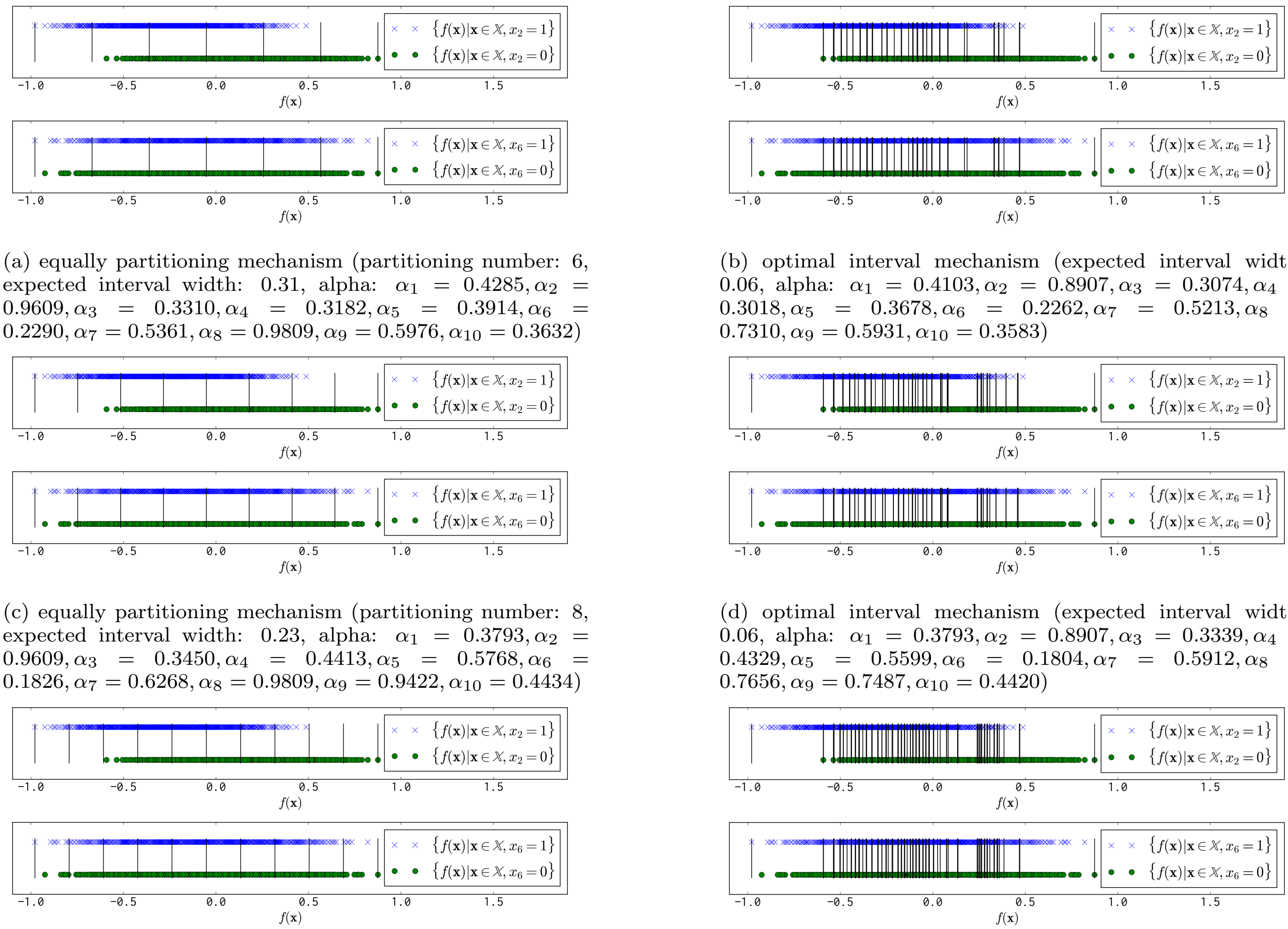
[15] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam. l-diversity: Privacy beyond k-anonymity. *ACM Trans. Knowl. Discov. Data*, 1(1), Mar. 2007.

[16] V. Rastogi, D. Suciu, and S. Hong. The boundary between privacy and utility in data publishing. In *Proceedings of the 33rd international conference on Very large data bases (VLDB '07)*, pages 531–542. VLDB Endowment, 2007.

[17] P. Vos, R. Hogers, M. Bleeker, M. Reijans, T. Van de Lee, M. Hornes, A. Friters, J. Pot, J. Paleman, M. Kuiper, et al. Aflp: a new technique for DNA fingerprinting. *Nucleic acids research*, 23(21):4407–4414, 1995.

[18] R. Wang, Y. F. Li, X. Wang, H. Tang, and X. Zhou. Learning your identity and disease from research papers: information leaks in genome wide association study. In *Proceedings of the 16th ACM conference on Computer and communications security (CCS '09)*, pages 534–544. ACM, 2009.

[19] T. Yoshida, K. Kato, T. Fujimaki, K. Yokoi, M. Oguri, S. Watanabe, N. Metoki, H. Yoshida, K. Satoh, Y. Aoyagi, Y. Nishigaki, M. Tanaka, Y. Nozawa, G. Kimura, and Y. Yamada. Association of genetic variants with chronic kidney disease in Japanese individuals. *Clin J Am Soc Nephrol*, 4(5):883–890, May 2009.



(a) equally partitioning mechanism (partitioning number: 6, expected interval width: 0.31, alpha: $\alpha_1 = 0.4285, \alpha_2 = 0.9609, \alpha_3 = 0.3310, \alpha_4 = 0.3182, \alpha_5 = 0.3914, \alpha_6 = 0.2290, \alpha_7 = 0.5361, \alpha_8 = 0.9809, \alpha_9 = 0.5976, \alpha_{10} = 0.3632$)

(b) optimal interval mechanism (expected interval width: 0.06, alpha: $\alpha_1 = 0.4103, \alpha_2 = 0.8907, \alpha_3 = 0.3074, \alpha_4 = 0.3018, \alpha_5 = 0.3678, \alpha_6 = 0.2262, \alpha_7 = 0.5213, \alpha_8 = 0.7310, \alpha_9 = 0.5931, \alpha_{10} = 0.3583$)

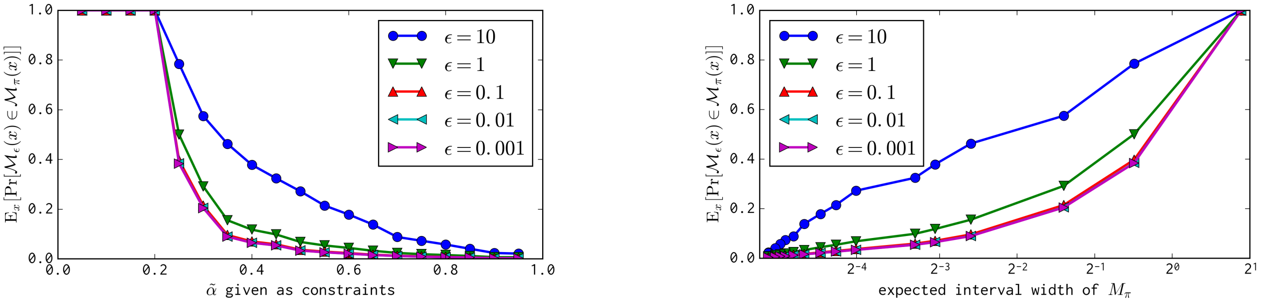
(c) equally partitioning mechanism (partitioning number: 8, expected interval width: 0.23, alpha: $\alpha_1 = 0.3793, \alpha_2 = 0.9609, \alpha_3 = 0.3450, \alpha_4 = 0.4413, \alpha_5 = 0.5768, \alpha_6 = 0.1826, \alpha_7 = 0.6268, \alpha_8 = 0.9809, \alpha_9 = 0.9422, \alpha_{10} = 0.4434$)

(d) optimal interval mechanism (expected interval width: 0.06, alpha: $\alpha_1 = 0.3793, \alpha_2 = 0.8907, \alpha_3 = 0.3339, \alpha_4 = 0.4329, \alpha_5 = 0.5599, \alpha_6 = 0.1804, \alpha_7 = 0.5912, \alpha_8 = 0.7656, \alpha_9 = 0.7487, \alpha_{10} = 0.4420$)

(e) equally partitioning mechanism (partitioning number: 10, expected interval width: 0.19, alpha: $\alpha_1 = 0.3515, \alpha_2 = 0.9609, \alpha_3 = 0.5212, \alpha_4 = 0.5275, \alpha_5 = 0.5695, \alpha_6 = 0.2426, \alpha_7 = 0.5873, \alpha_8 = 0.9809, \alpha_9 = 0.9422, \alpha_{10} = 0.4968$)

(f) optimal interval mechanism (expected interval width: 0.04, alpha: $\alpha_1 = 0.3504, \alpha_2 = 0.8907, \alpha_3 = 0.5170, \alpha_4 = 0.5110, \alpha_5 = 0.5351, \alpha_6 = 0.2411, \alpha_7 = 0.5369, \alpha_8 = 0.7310, \alpha_9 = 0.8892, \alpha_{10} = 0.4852$)

Figure 8: Equally partitioned intervals (left) and optimal intervals (right). Optimal intervals are designed to avoid unique identification with satisfaction of at least the same α-obscure privacy of equally partitioned intervals.



(a) Privacy budget $\tilde{\alpha}$ vs. the inclusion probability (obesity). If the inclusion probability is low, the output of the optimal interval mechanism has higher utility than the output of the differential privacy mechanism. If the $\tilde{\alpha}$ is small, the optimal interval mechanism guarantees stronger privacy. If the ϵ is small, the Laplace mechanism guarantees stronger privacy.

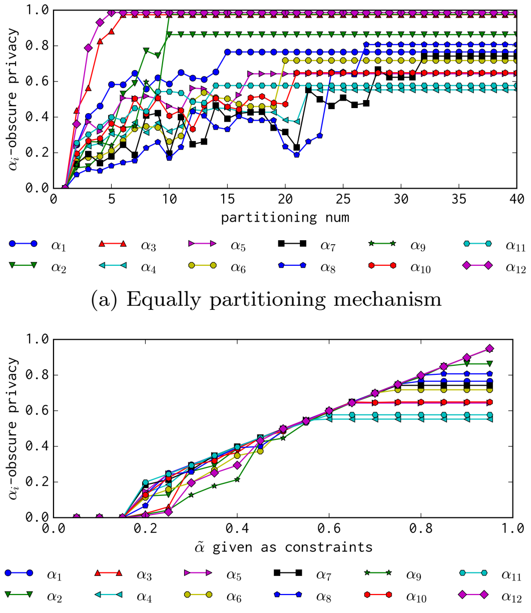
(b) The expected interval width vs. the inclusion probability (obesity), with changing $\tilde{\alpha}$ of the optimal interval mechanism. If the inclusion probability is low, the output of the optimal interval mechanism has higher utility than the output of the differential privacy mechanism. If the expected interval width is small, the output of the optimal interval mechanism has higher utility. If the ϵ is small, the Laplace mechanism guarantees stronger privacy.

Figure 9: The inclusion probability that the outputs of the differentially private mechanism are contained in the interval provided by the optimal interval mechanism.

APPENDIX

A. ADDITIONAL EXPERIMENTS IN CEREBRAL HEMORRHAGE

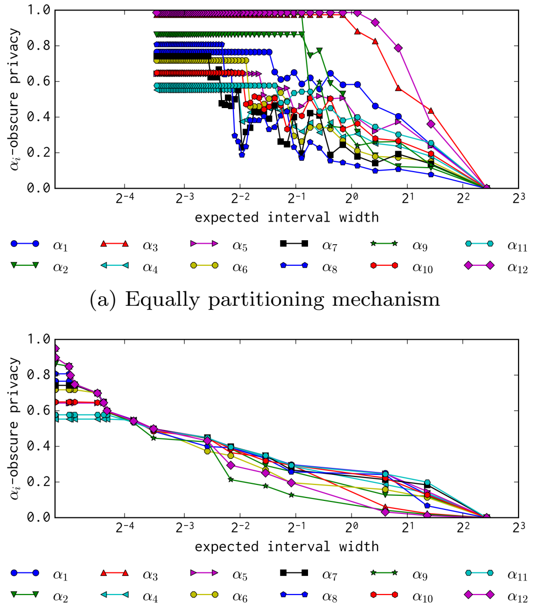
We experimentally evaluated privacy and utility of the optimal interval mechanism and equally partitioned mechanism with the susceptibility model of cerebral hemorrhage (Fig. 10-12). Also, comparison to differential privacy in the same model is shown in Fig. 13. We can confirm that our mechanism also achieves better utility-privacy trade-off with the susceptibility model of cerebral hemorrhage, too.



(a) Equally partitioning mechanism

(b) Optimal interval mechanism

Figure 10: α-obscure privacy vs. partitioning number n or privacy budget $\tilde{\alpha}$, applying to f of cerebral hemorrhage. The mechanism achieves better privacy if α_i is small.



(a) Equally partitioning mechanism

(b) Optimal interval mechanism

Figure 11: α-obscure privacy vs. expected interval width, applying to f of cerebral hemorrhage, changing controll parameters. The mechanism achieves better privacy if α_i is small. The output of mechanism has high utility if the expected interval width is small.

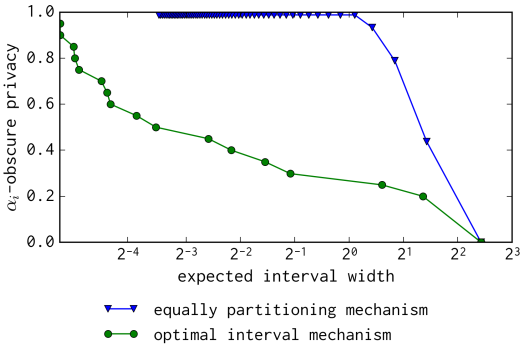
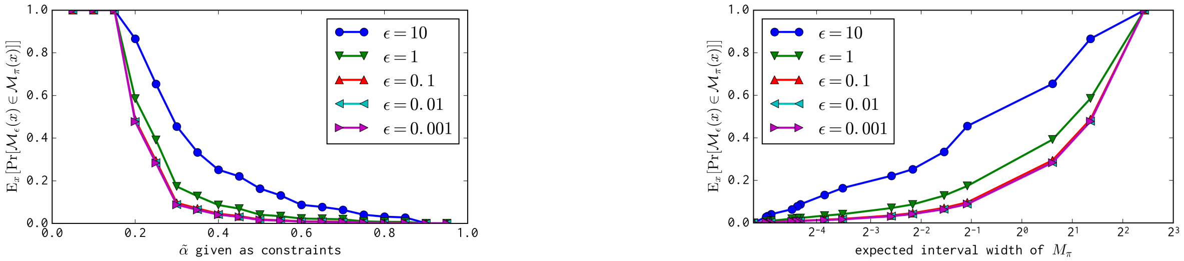


Figure 12: Expected interval width vs. the worst α among all attributes (cerebral hemorrhage). The mechanism achieves better privacy if α_i is small. The output of mechanism has high utility if the expected interval width is small.



(a) privacy budget $\tilde{\alpha}$ vs. the inclusion probability (cerebral hemorrhage). If the inclusion probability is low, the output of optimal interval mechanism has higher utility than the output of differential privacy mechanism. If the $\tilde{\alpha}$ is small, optimal interval mechanism guarantee stronger privacy.

(b) The expected interval width vs. the inclusion probability (cerebral hemorrhage), changing $\tilde{\alpha}$ of optimal interval mechanism. If the inclusion probability is low, the output of optimal interval mechanism has higher utility than the output of differential privacy mechanism. If the expected interval width is small, the output of optimal interval mechanism has high utility.

Figure 13: The inclusion probability that the outputs of differentially private mechanism are contained in the interval provided by the optimal interval mechanism, changing ϵ and $\tilde{\alpha}$.