Security and Safety, Vol. 1, 2022008 (2022)
https://doi.org/10.1051/sands/2022008
sands.edpsciences.org

Research Article OPEN ACCESS

Social Governance

Implicit privacy preservation: a framework based on data generation

Qing Yang[1,2], Cheng Wang[1,2], Teng Hu[1,2], Xue Chen[1,2], and Changjun Jiang[1,2,*]

[1] *Key Laboratory of Embedded System and Service Computing (Tongji University), Ministry of Education, Shanghai 201804, China*
[2] *National (Province-Ministry Joint) Collaborative Innovation Center for Financial Network Security, Tongji University, Shanghai 201804, China*

Received: 5 March 2022 / Revised: 6 June 2022 / Accepted: 16 June 2022 / Published online: 25 July 2022

Abstract This paper addresses a special and imperceptible class of privacy, called implicit privacy. In contrast to traditional (explicit) privacy, implicit privacy has two essential properties: (1) It is not initially defined as a privacy attribute; (2) it is strongly associated with privacy attributes. In other words, attackers could utilize it to infer privacy attributes with a certain probability, indirectly resulting in the disclosure of private information. To deal with the implicit privacy disclosure problem, we give a measurable definition of implicit privacy, and propose an ex-ante implicit privacy-preserving framework based on data generation, called IMPOSTER. The framework consists of an implicit privacy detection module and an implicit privacy protection module. The former uses normalized mutual information to detect implicit privacy attributes that are strongly related to traditional privacy attributes. Based on the idea of data generation, the latter equips the Generative Adversarial Network (GAN) framework with an additional discriminator, which is used to eliminate the association between traditional privacy attributes and implicit ones. We elaborate a theoretical analysis for the convergence of the framework. Experiments demonstrate that with the learned generator, IMPOSTER can alleviate the disclosure of implicit privacy while maintaining good data utility.

Keywords Privacy preservation, Implicit privacy, Generative adversarial network, Data utility, Data generation

Citation Yang Q, Wang C and Hu T et al. Implicit privacy preservation: a framework based on data generation. Security and Safety 2022; **1**: 2022008. https://doi.org/10.1051/sands/2022008

1 Introduction

Recent years have witnessed a rapid development of data mining techniques [1], which are found popular in a wide variety of fields, from disease prevention [2, 3] and credit evaluation [4, 5] to marketing analysis [6, 7] and anomaly detection [8, 9]. In order to provide accurate services and make effective decisions, both public and private organizations are committed to collecting and analyzing individual data [10]. For example, Walmart collected the shopping basket information of customers and found that beer was the most common commodity bought together with diapers [11]. However, the intentional or accidental disclosure of privacy information [12], such as personal credit information, online transaction history, or medical records, has raised concerns about personal privacy and even caused widespread social panic and significant economic losses. Netflix[1] has released a dataset including movie ratings from 500 000

[*] Corresponding author (email: cjjiang@tongji.edu.cn)
[1] https://www.netflix.com

subscribers. The dataset was intended for researchers of recommender systems. Although the subscribers were anonymous, when using the Internet Movie Database (IMDb)[2] as background knowledge, Narayanan *et al.* [13] were still able to link some records in the Netflix database to known individuals. When the Facebook[3] database was hacked, privacy information from more than 50 million users was leaked, leading to both huge financial loss and severe administrative litigation of the company. Hence, it is pressing to design more effective privacy protection methods that meet both legal requirements and user expectations.

As a matter of fact, a variety of work has been put forward for privacy-preserving issues. These methods govern the explicit use of privacy attributes and can be roughly divided as follows: (1) Data distorting approaches [14, 15], which work by adding noise that obeys a specific distribution to privacy attributes. (2) Data anonymity approaches [16, 17], which map the values of a privacy attribute to a more generalized space to protect privacy information. (3) Data encryption approaches [18–20], which protect privacy attributes by encrypting the privacy information or dealing with them in the ciphertext space.

The above-mentioned privacy-preserving methods are designed for the preservation of traditional privacy attributes. However, there exists another special and imperceptible kind of privacy that could lead to privacy disclosure. It is strongly related to privacy attributes. The study for this privacy starts with a set of examples.

The department store Target[4] often uses customers' shopping history to infer their pregnancy status and sells baby products according to this information. The pregnancy status is an explicit privacy attribute, which represents representative personal confidential and sensitive information. On the contrary, the purchase record is not defined as an explicit privacy attribute, but it is strongly related to the explicit privacy attribute. Therefore, if an attribute or a combination of attributes can potentially infer an explicit privacy attribute, then these attributes are called implicit privacy attributes. As another example, the popularity of Internet medical treatment has prompted users to search for their diseases (the explicit privacy attribute) through search engines, so users' browsing history is closely related to their diseases. Hence, users' browsing history is an implicit privacy attribute that implies users' diseases. Besides, warfarin is a drug for the prevention and treatment of thromboembolic diseases [21]. Patients with different genotypes take different doses of warfarin. There is a strong correlation between the dosage of warfarin and the patient's genotype (the explicit privacy attribute). Therefore, the dosage of warfarin is an implicit privacy attribute which indicates the patient's genotype information. Although traditional privacy protection methods have achieved significant performance in explicit privacy, they cannot guarantee good data utility while eliminating the strong correlation between implicit privacy and explicit privacy attributes [22–24]. Furthermore, implicit privacy attributes are not only associated with explicit privacy attributes, but also with class labels. If the values are changed to a constant or deleted directly, it may lead to low data utility [25]. Therefore, for implicit privacy protection, we need to eliminate the association between implicit and explicit privacy attributes, and at the same time, ensure the association between implicit privacy attributes and class labels of data as much as possible.

Recently, Generative Adversarial Network (GAN) and its improved versions have been proved powerful in a wide range of applications [26–29], including privacy protection. For privacy protection, those GAN-based approaches are mainly proposed for explicit privacy. They can be roughly classified into two categories: (1) Gradient perturbation approaches that add noise into the gradient descent process for model optimization. For example, DPGAN [30] adds noise to the discriminator's gradient in the training procedure to achieve differential privacy guarantees. (2) Output perturbation approaches that add noise into the output of the model. PATE-GAN [31] applies the Private Aggregation of Teacher Ensembles (PATE) framework to GAN and bounds the impact of any single sample on the model to provide rigorous differential privacy guarantees.

Although proved useful, existing studies only consider explicit privacy protection. This paper proposes an ex-ante implicit privacy-preserving framework based on data generation, called IMPOSTER, which is composed of two modules. In particular, the implicit privacy detection module uses normalized mutual information to detect the attributes that are strongly related to explicit privacy attributes. The implicit privacy protection module is constructed by adding a discriminator to GAN equipped with a Variational AutoEncoder (VAE). As a matter of fact, it contains one encoder E, one generator (or decoder) G, and two discriminators D_1 and D_2. As in the standard GAN, G is trained to generate artificial data that simulate

[2] https://www.imdb.com

[3] https://www.facebook.com

[4] https://www.target.com

the distribution of real samples P_r. While D_1 is trained to judge whether the generated data are close to the real data, D_2 is to eliminate the correlation between explicit privacy attributes and implicit ones. The E and G together constitute VAE, which is trained to minimize the reconstruction errors between real samples and fake samples generated by G. When the model converges, the generator produces high-usability data without implicit privacy. It is worth noting that attribute inference attacks [21] require attackers to infer explicit privacy attributes by accessing trained models and non-privacy attributes. This paper focuses on inferring explicit privacy attributes according to the strong correlation between explicit privacy attributes and other attributes, that is, attackers can infer explicit privacy attributes from other attributes with a certain probability, which indirectly causes the privacy information's disclosure. The implicit privacy disclosure problem requires that attackers do not need to access the trained model.

We give a summary of main contributions as follows:

- We address a special and imperceptible class of privacy disclosure problem, called implicit privacy, and give a definition of implicit privacy attributes.
- We propose an ex-ante implicit privacy-preserving framework based on data generation, called IMPOSTER, which contains an implicit privacy detection module and a protection module. Specifically, the former uses normalized mutual information to detect the attributes that are strongly related to privacy attributes. Based on the idea of data generation, the latter equips the GAN framework with an additional discriminator, which is used to eliminate the correlation between explicit privacy attributes and implicit privacy attributes.
- We illustrate the superiority and effectiveness of IMPOSTER by theoretical derivation and experimental results on a public dataset.

The other sections of this work are arranged as follows: Section 2 reviews the related research. Section 3 introduces the preliminaries of the IMPOSTER framework. In Section 4, we detail our proposed framework for implicit privacy preservation based on data generation. In Section 5, we conduct detailed experiments on a public dataset and illustrate the effectiveness of our framework. Finally, in Section 6, we give the conclusion and future work.

2 Related work

We summarize the previous work related to traditional privacy protection approaches.

2.1 Traditional privacy-preserving methods

According to different methods of data processing, the existing privacy-preserving approaches mainly include: data distortion approaches, data encryption approaches, and data anonymity approaches.

Data distortion approaches. Data distortion approaches mainly include: randomization, condensation and differential privacy. Randomization works by injecting random noise into raw data and then publishes disturbed data. Warner *et al.* [32] proposed the randomized response (RR) mechanism that provides plausible deniability for individuals with sensitive information. In order to reconstruct the data distribution without disclosing individual privacy, Charu *et al.* [33] proposed a condensation approach that transforms the original dataset into a new anonymized one, which includes the correlation between different dimensions. Considering the background knowledge attack and differential attack that steal individual privacy information, Dwork *et al.* [34] presented the differential privacy, which provides mathematically provable guarantees for privacy preservation.

Data anonymity approaches. Data anonymity approaches mainly include: k-anonymity, l-diversity and t-closeness. Specifically, Sweeney *et al.* [35] proposed the k-anonymity algorithm, which realizes that any sample cannot be differentiated from other $k-1$ samples in the same equivalence class, so as to alleviate privacy leakage caused by linking attacks. Since the k-anonymity does not impose any constraints on sensitive attribute columns, attackers can use homogeneity attacks and background knowledge attacks to discover users' corresponding sensitive data, resulting in privacy disclosure. To overcome the shortcomings of the k-anonymity, Machanavajjhala *et al.* [16] proposed the l-diversity algorithm to guarantee that sensitive attributes have at least l different values in the same equivalence class. In order to defend against similarity attacks, Li *et al.* [17] presented a novel privacy-preserving method named t-closeness, which guarantees that the difference between the distribution of sensitive attribute values in each equivalence class and that in the original dataset shall not exceed a threshold t.

Data encryption approaches. Data encryption approaches are to conceal sensitive data through data encryption technologies. The representative methods include: secure multi-party computing (SMC), homomorphic encryption (HE), and federated learning (FELE). SMC [36] refers to the scenario that multiple participants holding their own private data jointly execute a calculation logic (such as the maximum calculation) and obtain the calculation results in the absence of a reputable third party. Therefore, each participant will not disclose the calculation of their own data. HE [37, 38] takes an encryption algorithm satisfying the properties of ciphertext homomorphic operations. When data are homomorphically encrypted, the algorithm performs particular calculations on the ciphertext, and the results obtained are in the equivalent homomorphism. To perform the same above calculations directly on plaintext data is equal to the decrypted plaintext. FELE [18] is a distributed machine learning technology that breaks data islands. By exchanging encrypted intermediate results, it provides participants with the ability to joint data modeling without privacy disclosure.

2.2 Summary

Traditional privacy-preserving methods have been proved useful in explicit privacy. However, they cannot guarantee good data utility while eliminating the strong correlation between explicit privacy and other attributes. As a matter of fact, traditional privacy protection methods would cause low data utility if used for implicit privacy directly. Therefore, further efforts should be made to protect implicit privacy.

3 Preliminaries

3.1 Randomized response

The Randomized Response (RR) mechanism was first proposed by Warner [32] to provide plausible deniability for individuals responding to sensitive information. For example, consider a questionnaire: "Do you smoke?" For this question, the RR allows respondents to flip an unbiased coin secretly, and respondents tell truth if it comes up heads, otherwise, flip the coin again and answer "Yes" or "No" according to the result of second toss, that is, answer "Yes" if it comes up heads, otherwise, answer "No". This paper uses the k-ary Randomized Response (kRR) mechanism [39] as a benchmark to compare with our proposed framework. Specifically, considering there are n individual records $\mathrm{item}_1, \mathrm{item}_2, \ldots, \mathrm{item}_n$ in a dataset D, each record item_i has some attribute value $s_i \in \mathbb{S}$ regarding an attribute S. $\mathbb{S}$ denotes the value space of an attribute S. $\mathbb{R}$ denotes an output alphabet of sanitized S ($\mathbb{S} = \mathbb{R}$ and $|\mathbb{S}| = k$). We map s_i stochastically to $r_i \in \mathbb{R}$ by Eq (1), *i.e.*, some attribute value s_i remains unchanged with probability $\frac{e^\varepsilon}{|\mathbb{S}|-1+e^\varepsilon}$, and flips to other values in the same attribute value space with probability $\frac{1}{|\mathbb{S}|-1+e^\varepsilon}$. The specific implementation process of the kRR mechanism is shown in Figure 1.

$$Q(r|s) = \begin{cases} \frac{e^\varepsilon}{|\mathbb{S}|-1+e^\varepsilon}, \, s_i = r_i \\ \frac{1}{|\mathbb{S}|-1+e^\varepsilon}, \, \text{otherwise} \end{cases} \tag{1}$$

where ε denotes the privacy budget. In general, the smaller the ε, the higher the level of privacy protection, the lower the data utility.

3.2 GAN and CGAN

As a representative generative model, Generative Adversarial Network (GAN) [40] has the following superiorities: (1) It is not dependent on prior assumptions; (2) It generates synthetic samples similar to the distribution of real samples. GAN produces high-quality output through the mutual game learning of a discriminator D and a generator G. Specifically, G is trained to learn the distribution of real samples

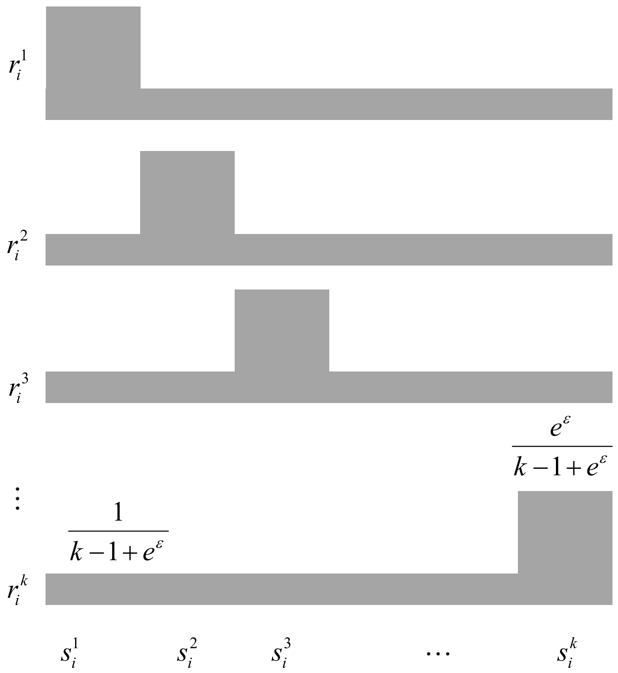

r_i^1
r_i^2
r_i^3
$\vdots$
r_i^k
$\frac{e^{\varepsilon}}{k-1+e^{\varepsilon}}$
$\frac{1}{k-1+e^{\varepsilon}}$
s_i^1
s_i^2
s_i^3
$\cdots$
s_i^k

Figure 1. An illustration of the k-ary Randomized Response (kRR) mechanism

from noise distribution, while D distinguishes synthetic data produced by G from real data distribution. In general, the objective function of D can be expressed as follows:

$$\max_D E_{x\sim P_r}[\log D(x)] + E_{z\sim P_z}[\log(1 - D(G(z)))], \tag{2}$$

where P_r represents the distribution of real samples, $D(x)$ denotes the probability that x obeys P_r rather than the distribution of generated samples P_g, $P_z(z)$ represents a prior distribution of noise variable z, and $G(z)$ represents that G produces synthetic samples from a prior distribution P_z. The objective function of G can be expressed as follows:

$$\min_G E_{z\sim P_z}[\log(1 - D(G(z)))]. \tag{3}$$

Therefore, G and D play the minimax game with a value function $V(G, D)$, which is given by:

$$\min_G \max_D V(G, D) = E_{x\sim P_r}[\log D(x)] + E_{z\sim P_z}[\log(1 - D(G(z)))]. \tag{4}$$

Figure 2 illustrates the structure of GAN. Theoretical analysis shows that GAN aims to minimize the distance between P_g and P_r, and $V(G, D)$ has a global optimal value for $P_r = P_g$ [40].

As an improvement of the traditional GAN, Conditional Generative Adversarial Network (CGAN) [41] takes auxiliary information ζ as a condition to guide G and D to realize a conditional generative model. The objective function of CGAN is given by:

$$\min_G \max_D V(G, D) = E_{x\sim P_r(x|\zeta)}[\log D(x|\zeta)] + E_{z\sim P_z(z|\zeta)}[\log(1 - D(G(z|\zeta)|\zeta))]. \tag{5}$$

Notably, CGAN combines the noise distribution $P_z(z)$ and ζ into a joint latent representation to input into G. Similarly, x and ζ are also input into D.

4 Overview of our approach

4.1 Problem statement

To clarify explicit privacy and implicit privacy, we take the case of the department store Target as an example.

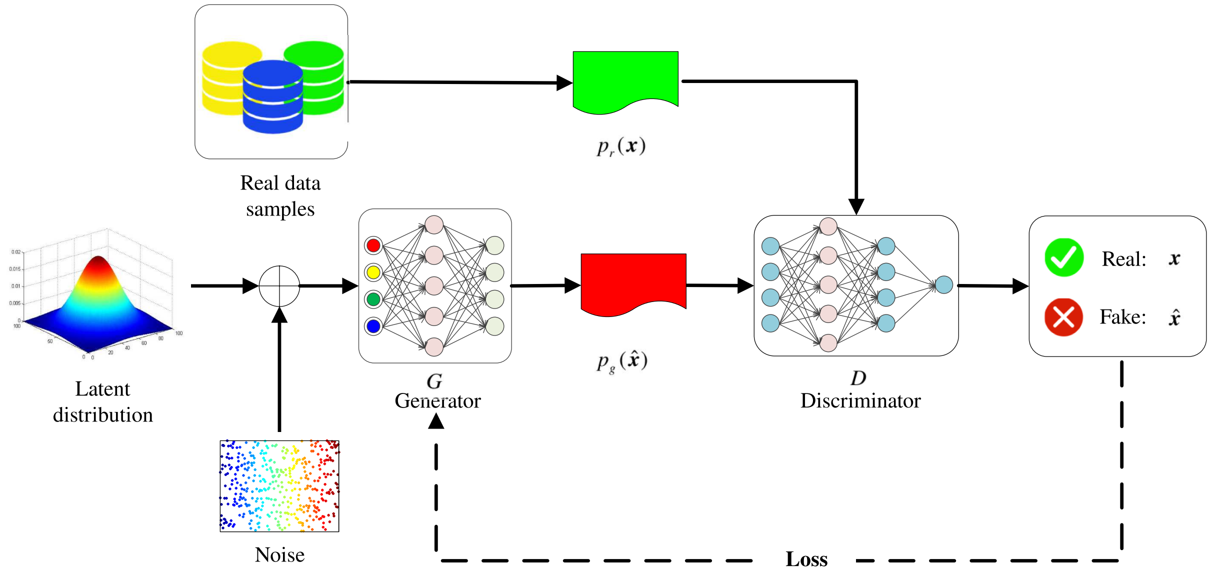


Figure 2. An illustration of GAN

Consider a scenario in which retailers advertise products to customers. Figure 3a shows the use of explicit privacy. Retailers get female customers' medical records illegally, and then decide to recommend product advertisements according to customers' pregnancy status (an explicit privacy attribute). Such explicit privacy disclosure will threaten the life and property of users, such as bullying and credit card fraud. For the disclosure of explicit privacy attributes, we can utilize the current representative and excellent differential privacy (DP) to prevent it. In addition to the explicit privacy mentioned above, there is a special and imperceptible class of non-privacy attribute, called implicit privacy, which is strongly associated with privacy attributes. As shown in Figure 3b, retailers do not directly use the pregnancy status in the medical records, but use the customer's recent purchase records. If the customer often purchases pregnancy-related products P_1 and P_2 recently, retailers can conclude that the customer or one of her family is pregnant, and then recommend pregnancy-related product advertisements to her. If the customer often purchases products O_1 and O_2, retailers will recommend other types of product advertisements to the customer. Here, the purchase record is an implicit privacy attribute, through which we can accurately infer the customer's pregnancy status. In contrast to explicit privacy, implicit privacy is not defined as a privacy attribute, but it strongly correlates with privacy attributes. Attackers can use it to infer explicit privacy indirectly, resulting in a series of privacy disclosure problems. Based on the above case, we define explicit privacy attributes and implicit privacy attributes as follows:

Definition 1. (Explicit privacy attributes, *i.e.*, privacy attributes [42, 43]). In a dataset, attributes that directly represent personal confidential and sensitive information are called explicit privacy attributes (*e.g.*, disease and salary).

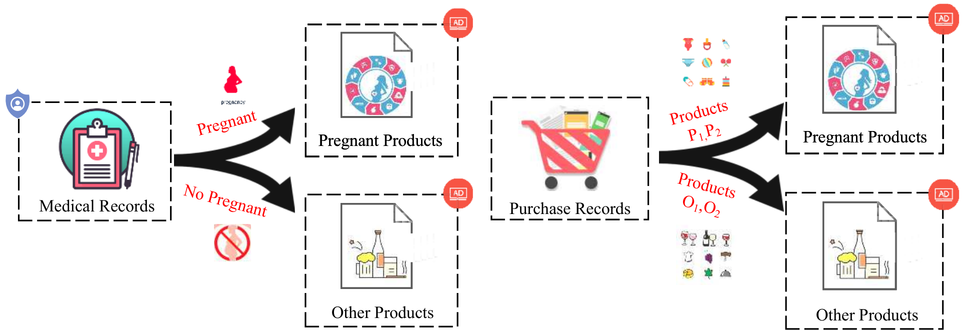
Note that explicit privacy attributes refer to traditional privacy attributes in this paper.

Definition 2. ((θ, β)-implicit privacy attribute). Given a dataset $D = (\boldsymbol{x}, s)$, including the public attributes $\boldsymbol{x}$ and an explicit privacy attribute s (*e.g.*, income, disease status), each attribute in $\boldsymbol{x}_p \subseteq \boldsymbol{x}$ is said to be a (θ, β)-implicit privacy attribute for s with the correlation metric ρ and the performance metric τ if a classification algorithm $f : \boldsymbol{x}_p \to s$ exists, such that

$$\tau(f(\boldsymbol{x}_p), s) \geq \beta, \tag{6}$$

where $\boldsymbol{x}_p = \{x_i | \rho(x_i, s) \geq \theta, x_i \in \boldsymbol{x}\}$, $\rho(*, *)$ is a function that measures the correlation between two attributes, such as the normalized mutual information [44, 45] and Pearson correlation coefficient [46], and τ represents performance measurement of classifiers, such as Accuracy and F1-score.

Here the performance threshold β with higher values indicates the higher prediction performance from $\boldsymbol{x}_p$ to s by f. The selection of β depends on the tolerance of implicit privacy disclosure. With a high risk



(a) Explicit privacy (b) Implicit privacy

Figure 3. A scenario in which retailers recommend product advertisements to customers.

of implicit privacy disclosure tolerated, we can select a larger β to obtain higher data utility. Otherwise, we can select a smaller β to achieve better privacy protection. Similarly, as a correlation threshold, the higher θ indicates that each attribute selected has a higher correlation with s.

Generally speaking, the stronger the correlation between $\boldsymbol{x}_p$ and s, the stronger the predictive ability from $\boldsymbol{x}_p$ to s. For example, in the feature engineering, we also select the features that are strongly related to the class label [47–49]. In a word, our ultimate goal is to eliminate the correlation between explicit and implicit privacy attributes while preserving good data utility.

4.2 Our framework IMPOSTER

As shown in Figure 4, our framework IMPOSTER consists of two modules: (1) the implicit privacy detection module, and (2) the implicit privacy protection module.

4.2.1 Implicit privacy detection module

The explicit privacy attribute can be inferred from other attributes in the dataset, which will also result in the users' privacy disclosure. As shown in Figure 5, attribute x_3 can infer an explicit privacy attribute s with a certain probability, so attribute x_3 is the implicit privacy attribute for s. Therefore, it is necessary to determine the implicit privacy attributes for s in advance and protect them. Commonly used metrics to measure the correlation between two random variables are Pearson correlation coefficient, mutual information (MI), and normalized mutual information (NMI). Pearson's correlation coefficient mainly measures the degree of linear correlation between two random variables. Both MI and NMI can measure the degree of linear correlation or nonlinear correlation between two random variables. Further, NMI is a normalization of the MI score to scale the results between 0 (statistical independence) and 1 (perfect correlation), which reduces the adverse effects of abnormal sample data. Therefore, we use NMI to measure the correlation between the explicit privacy attribute and other attributes. The formula of NMI is as follows:

$$\mathrm{Cor}(x_i, s) = \frac{H_s(x_i|s) + H_s(s|x_i)}{H_s(x_i, s)}, \tag{7}$$

where H_s denotes Shannon entropy. We calculate the relevance between the explicit privacy attribute s and other attributes x_i, and get the attribute set that is strongly related to s. According to Definition 2, we use a classification algorithm f to measure the prediction ability from the attribute set to s, and finally get the implicit privacy attribute set.

4.2.2 Implicit privacy protection module

Based on the idea of data generation, the implicit privacy protection module adds a discriminator into the GAN framework equipped with a VAE model to eliminate the association between implicit privacy attributes and explicit privacy attributes. Although GAN is trained to learn a distribution of synthetic

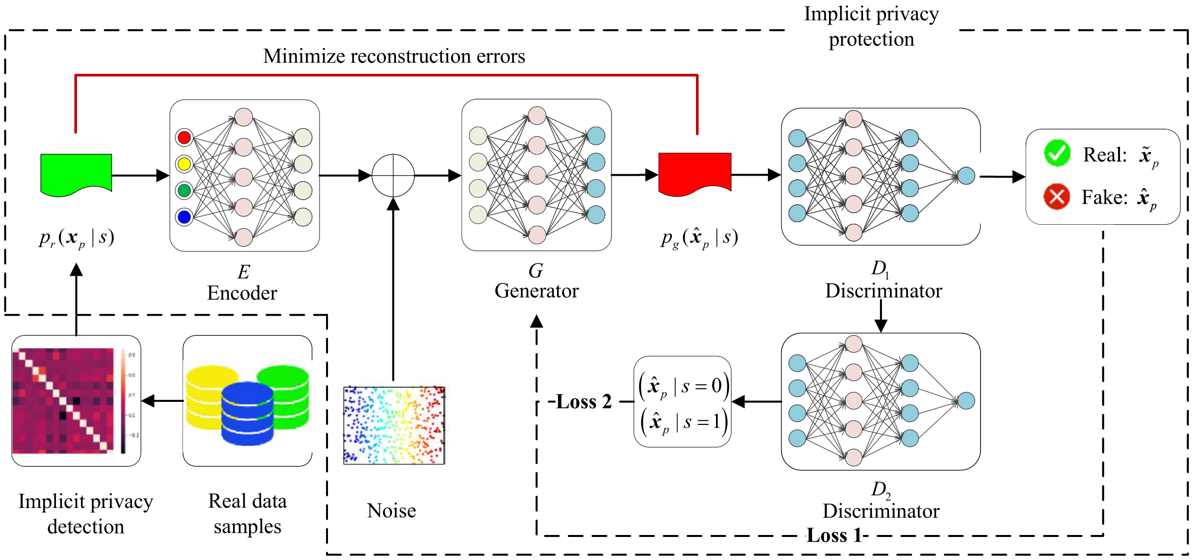

Implicit privacy protection
Minimize reconstruction errors
$p_r(\boldsymbol{x}_p|s)$
E
Encoder
G
Generator
$p_g(\hat{\boldsymbol{x}}_p|s)$
D_1
Discriminator
Real: $\tilde{\boldsymbol{x}}_p$
Fake: $\hat{\boldsymbol{x}}_p$
-Loss 2
$(\hat{\boldsymbol{x}}_p|s=0)$
$(\hat{\boldsymbol{x}}_p|s=1)$
D_2
Discriminator
Loss 1
Implicit privacy detection
Real data samples
Noise

Figure 4. The framework of IMPOSTER

Algorithm 1 Implicit Privacy Detection Module

```
Require: $\boldsymbol{x} = \{x_1, x_2, \ldots, x_n\}$, attributes set; $s$, privacy attribute; $\theta$, correlation threshold; $P_{set}$, implicit privacy attribute set.
Ensure: $P_{set}$.
initialize $P_{set} = \phi$
for each $x_i \in \boldsymbol{x}$ do
    Calculate normalized mutual information:
        $Cor(x_i, s) = \frac{H_s(x_i|s) + H_s(s|x_i)}{H_s(x_i, s)}$.
    if $Cor(x_i, s) \geq \theta$ then
        $P_{set} \leftarrow x_i$.
    end if
end for
if $\tau(f(P_{set}), s) \geq \beta$, then
    return $P_{set}$.
else
    return $\phi$.
end if
```

samples similar to the distribution of real samples, it is not good at capturing the element-wise errors between synthetic samples and real samples. In order to alleviate this limitation, our model incorporates a VAE [50] to minimize the reconstruction errors between real samples and synthetic samples. VAE includes an encoder E that compresses the original input $\boldsymbol{x}_p$ to a latent representation $z_l \sim E(\boldsymbol{x}_p) = p(z_l|\boldsymbol{x}_p)$ and a decoder G that decompresses z_l to reconstructed output $\hat{\boldsymbol{x}}_p \sim G(z_l) = p(\boldsymbol{x}_p|z_l)$. The total loss function of the VAE includes reconstruction errors and a regularization term, which can be expressed as:

$$\min V_{\text{VAE}} = -E_{p(z_l|\boldsymbol{x}_p)}[\log p(\boldsymbol{x}_p|z_l)] + \text{KL}(p(z_l|\boldsymbol{x}_p)||p(z_l)), \tag{8}$$

where KL(*) refers to the Kullback–Leibler divergence.

After the data pass through the VAE, in order to eliminate the correlation between implicit and explicit privacy attributes, we adopt an improved GAN, which consists of one generator (the decoder of VAE) G and two discriminators D_1 and D_2. The G generates $\hat{\boldsymbol{x}}_p \sim P_g$ from a prior noise distribution P_z to match P_r.

$$\hat{\boldsymbol{x}}_p = G(z|s), z \sim P_z(z). \tag{9}$$

D_1 is a classifier that distinguishes a real sample $\tilde{\boldsymbol{x}}_p = G(E(\boldsymbol{x}_p))$ from a generated faked sample $\hat{\boldsymbol{x}}_p$. Here, s is a binary attribute. Therefore, the discriminator D_2 is also a binary classifier, which is an important part of eliminating the correlation between implicit and explicit privacy attributes. The specific game process of G, D_1 and D_2 will be given in detail later.

Therefore, our improved CGAN sub-module in the implicit privacy protection module is formalized as a minimax game and the value function is given by:

$$\min_{G} \max_{D_1, D_2} V_{\mathrm{CGAN}}(G, D_1, D_2) = V_1(G, D_1) + \lambda V_2(G, D_2), \tag{10}$$

where

$$V_1(G, D_1) = E_{(\tilde{\boldsymbol{x}}_p|s)\sim P_r(\tilde{\boldsymbol{x}}_p|s)}[\log D_1(\tilde{\boldsymbol{x}}_p|s))] + E_{(\hat{\boldsymbol{x}}_p|s)\sim P_g(\hat{\boldsymbol{x}}_p|s)}[\log(1 - D_1(\hat{\boldsymbol{x}}_p|s))], \tag{11}$$

$$V_2(G, D_2) = E_{\hat{\boldsymbol{x}}_p\sim P_g(\hat{\boldsymbol{x}}_p|s=1)}[\log D_2(\hat{\boldsymbol{x}}_p)] + E_{\hat{\boldsymbol{x}}_p\sim P_g(\hat{\boldsymbol{x}}_p|s=0)}[\log(1 - D_2(\hat{\boldsymbol{x}}_p))]. \tag{12}$$

The hyperparameter λ is a trade-off coefficient, which is used to balance data utility and privacy level of generated data.

Similar to the traditional CGAN, the value function V_1 indicates that G and D_1 play a zero-sum game. Specifically, D_1 learns to accurately distinguish between generated samples and real samples, while G learns to generate fake samples similar to real data to fool D_1. In order to make the generated samples contain the information that supports predicting the value of the explicit privacy attributes as little as possible, the second value function V_2 shows that D_2 and G also play a zero-sum game. Specifically, D_2 learns to accurately predict the value of s, while G learns to fool D_2.

The total objective function of the implicit privacy protection module can be formalized as follows:

$$\min_{E,G} \max_{D_1, D_2} V(E, G, D_1, D_2) = V_{\mathrm{CGAN}}(G, D_1, D_2) + V_{\mathrm{VAE}}(E, G). \tag{13}$$

When the reconstruction errors between the real samples and the generated samples in VAE are within an acceptable range, once the implicit privacy protection module converges, the synthetic samples generated by G approximately obey the distribution of real samples, and the correlation between explicit and implicit privacy attributes is eliminated as much as possible.

4.3 Algorithm

Algorithm 1 displays the pseudo code of the implicit privacy detection module. Algorithm 2 displays the pseudo code of the implicit privacy protection module.

Firstly, we sample a minibatch of samples from the output of E and a minibatch of noise samples from P_g to train D_1 and G (from Line 2 to 7). Secondly, we sample a minibatch of samples $(\hat{\boldsymbol{x}}_p|s = 0) \sim P_g(\hat{\boldsymbol{x}}_p|s = 0)$, and sample another batch of samples $(\hat{\boldsymbol{x}}_p|s = 1) \sim P_g(\hat{\boldsymbol{x}}_p|s = 1)$ to train D_2 and G (from Line 8 to 10). Finally, when model converges, we can get G to generate data without implicit privacy.

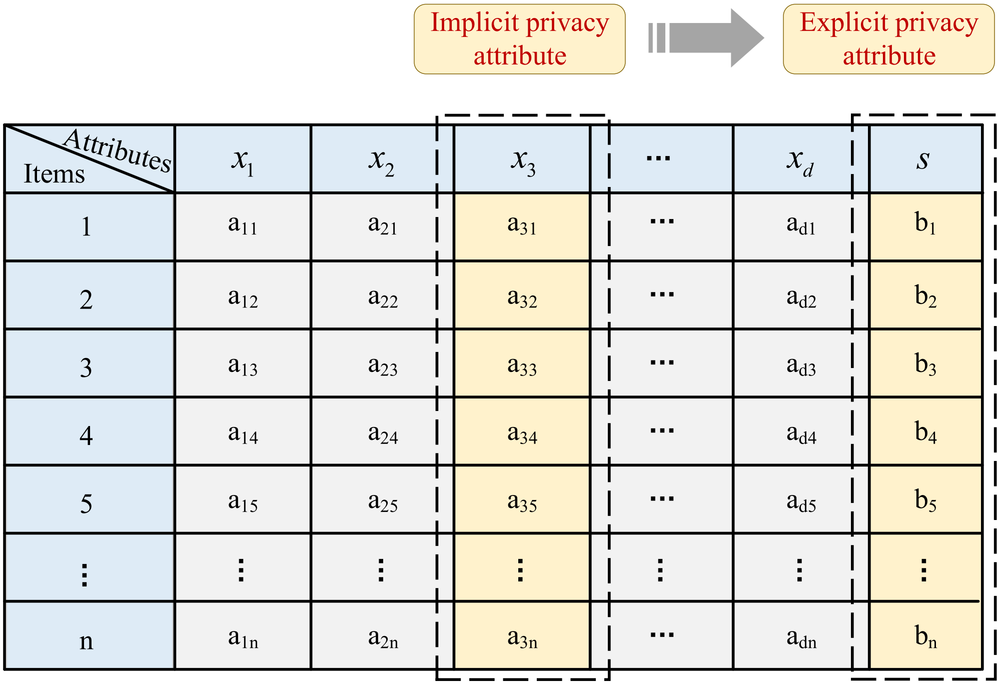
4.4 Theoretical analysis

Different from the traditional GAN, our proposed implicit privacy protection module in IMPOSTER adds an additional discriminator to protect implicit privacy. In addition, we introduce VAE to capture the element-wise errors between synthetic samples and real samples, so as to make the distance between them as close as possible. Therefore, we give a theoretical analysis of the convergence of the implicit privacy protection module when the reconstruction errors between real samples and generated samples in VAE are within an acceptable range.

Proposition 1. Given a fixed encoder E and a fixed generator G, the optimal discriminators D_1^* and D_2^* can be formalized as follows:

$$D_1^*(\boldsymbol{x}_p|s) = \frac{P_r(\tilde{\boldsymbol{x}}_p|s)}{P_r(\tilde{\boldsymbol{x}}_p|s) + P_g(\hat{\boldsymbol{x}}_p|s)}, \tag{14}$$

$$D_2^*(\boldsymbol{x}_p) = \frac{P_g(\hat{\boldsymbol{x}}_p|s = 1)}{P_g(\hat{\boldsymbol{x}}_p|s = 0) + P_g(\hat{\boldsymbol{x}}_p|s = 1)}. \tag{15}$$



Attributes / Items	x_1	x_2	x_3	···	x_d	s
1	a_{11}	a_{21}	a_{31}	···	a_{d1}	b_1
2	a_{12}	a_{22}	a_{32}	···	a_{d2}	b_2
3	a_{13}	a_{23}	a_{33}	···	a_{d3}	b_3
4	a_{14}	a_{24}	a_{34}	···	a_{d4}	b_4
5	a_{15}	a_{25}	a_{35}	···	a_{d5}	b_5
⋮	⋮	⋮	⋮	⋮	⋮	⋮
n	a_{1n}	a_{2n}	a_{3n}	···	a_{dn}	b_n

Figure 5. An illustration of implicit privacy detection

Proof. The theoretical proof is given by the Appendix of the Supporting Information.

Note that the training objective for D_1 is to estimate whether $\boldsymbol{x}_p$ comes from P_r or P_g. D_2 is used to eliminate the association between the explicit privacy attribute s and implicit privacy attributes $\boldsymbol{x}_p$. Given a fixed encoder E, the optimal discriminators D_1^* and D_2^*, Eq (13) can be changed as follows:

$$\begin{aligned} C(G) = & \max_{D_1,D_2} V_{\mathrm{CGAN}}(G, D_1, D_2) + V_{\mathrm{VAE}}(E, G) \\ = & -(2+\lambda)\log 4 + 2 \times JS(P_r(\tilde{\boldsymbol{x}}_p|s)||P_g(\hat{\boldsymbol{x}}_p|s)) \\ & + 2\lambda \times JS(P_g(\hat{\boldsymbol{x}}_p|s=0)||P_g(\hat{\boldsymbol{x}}_p|s=1)) + V_{\mathrm{VAE}}(E, G). \end{aligned} \tag{16}$$

The detailed derivation process is described in the Appendix of the Supporting Information.

The objective function of VAE includes cross entropy and Kullback–Leibler divergence. For the Eq (16), since Jensen–Shannon divergence, Kullback–Leibler divergence and cross entropy are convex functions [51], $C(G)$ can converge to a global minimum. Therefore, as for $C(G)$, we give a theorem as follows:

Theorem 1. Given a fixed encoder E, the optimal discriminators D_1^* and D_2^*, there exists a global minimum for the function $C(G)$.

Proof. The detailed proof is described in the Appendix of the Supporting Information.

5 Experimental evaluation

We evaluate the effectiveness of our proposed framework IMPOSTER from the following aspects: (1) whether the generated synthetic data eliminate the correlation between explicit privacy attributes and implicit privacy attributes; (2) whether the generated synthetic data preserve good data utility; (3) parameter sensitivity analysis.

5.1 Dataset

We evaluate our proposed privacy-preserving framework IMPOSTER on a real-world dataset. We present some details and statistics of the dataset as follows:

Algorithm 2 Implicit Privacy Protection Module

```
Require: $m$, batch size; $n_{ite}$, number of training iterations; $e$, $g$, $d_1$, $d_2$, parameters of $E$, $G$, $D_1$ and $D_2$, respectively.
Ensure: Implicit privacy-preserving generator $G$.
for $t = 0, \dots, n_{ite}$ do
    Sample $m$ real samples:
        $\{\boldsymbol{x}_p^{(1)}, \dots, \boldsymbol{x}_p^{(m)}\} \sim P_r(\boldsymbol{x}_p|s).$
    Generate $m$ samples from encoder $E$:
        $\{z_l^{(1)}, \dots, z_l^{(m)}\} = E(\{\boldsymbol{x}_p^{(1)}, \dots, \boldsymbol{x}_p^{(m)}\}).$
    Generate $m$ samples from decoder $G$:
        $\{\tilde{\boldsymbol{x}}_p^{(1)}, \dots, \tilde{\boldsymbol{x}}_p^{(m)}\} = G(\{(z_l|s)^{(1)}, \dots, (z_l|s)^{(m)}\}).$
    Sample $m$ noise samples:
        $\{z^{(1)}, \dots, z^{(m)}\} \sim P_g(\hat{\boldsymbol{x}}_p|s).$
    Update $D_1$ by ascending its stochastic gradient:
        $\nabla_{d_1} \frac{1}{m} \sum_{i=1}^{m} \left[ \log D_1(G((\tilde{\boldsymbol{x}}_p|s)^{(i)}) + \log(1 - D_1(G((z)^{(i)}))) \right].$
    Update $G$ by descending its stochastic gradient:
        $\nabla_g \frac{1}{m} \sum_{i=1}^{m} \log(1 - D_1(G(z^{(i)}))).$
    Sample a minibatch of m samples:
        $\{(\hat{\boldsymbol{x}}_p|s=0)^{(1)}, \dots, (\hat{\boldsymbol{x}}_p|s=0)^{(m)}\} \sim P_g(\hat{\boldsymbol{x}}_p|s=0),$
    and sample another minibatch of m samples:
        $\{(\hat{\boldsymbol{x}}_p|s=1)^{(1)}, \dots, (\hat{\boldsymbol{x}}_p|s=1)^{(m)}\} \sim P_g(\hat{\boldsymbol{x}}_p|s=1).$
    Update $D_2$ by ascending its stochastic gradient:
        $\nabla_{d_2} \frac{1}{2m} \sum_{i=1}^{2m} \log[D_2(\hat{\boldsymbol{x}}_p^{(i)}) + \log(1 - D_2(\hat{\boldsymbol{x}}_p^{(i)}))].$
    Update $G$ by descending its stochastic gradient:
        $\nabla_g \frac{1}{2m} \sum_{i=1}^{2m} \log[D_2(\hat{\boldsymbol{x}}_p^{(i)}) + \frac{1}{m} \sum_{i=1}^{m} \log(1 - D_2(\hat{\boldsymbol{x}}_p^{(i)}))].$
    Update $E$ and $G$ according to Eq (8).
end for
return $G$;
```

UCI adult dataset[5]**.** The dataset contains 48 842 instances. Each instance contains 7 numerical and 7 categorical variables, and the class label represents whether the annual income exceeds $50k.

5.2 Evaluation metrics

This paper adopts several metrics to verify the performance of our framework. These metrics are listed as follows.

Accuracy. Accuracy measures the proportion of the number of rightly predicted samples to the entire number of predicted samples and can be expressed as follows:

$$\text{Accuracy} = \frac{TN + TP}{TP + TN + FP + FN}, \tag{17}$$

[5] https://archive.ics.uci.edu/ml/datasets/adult

where TP refers to true positive, FP refers to false positive, TN refers to true negative, and FN refers to false negative.

F1-score. Both precision and recall are important performance metrics in classification problems. However, they are a pair of opposite metrics. In general, the higher the recall, the lower the precision. In order to comprehensively consider these two indicators, we adopt F1-score to measure the prediction performance of classifiers. The F1-score denotes a weighted average of the recall and precision. It can be given by:

$$\text{F1-score} = \frac{2 \cdot \text{recall} \cdot \text{precision}}{\text{recall} + \text{precision}}, \tag{18}$$

where precision $= \frac{TP}{FP+TP}$ and recall $= \frac{TP}{TP+FN}$.

5.3 Experimental setting

For the purpose of exploring the effectiveness of our framework, we adopt several state-of-the-art and representative classifiers: XGBoost [52], gradient boosting decision tree (GBDT) [53, 54], random forest (RF) [55], multi-layer perceptron classifier (MLP) [56] and logistic regression (LR) [57]. We train several classifiers on two different settings.

- Setting A: classifiers are trained on real samples and tested on real samples.
- Setting B: classifiers are trained on generated samples and tested on generated samples.

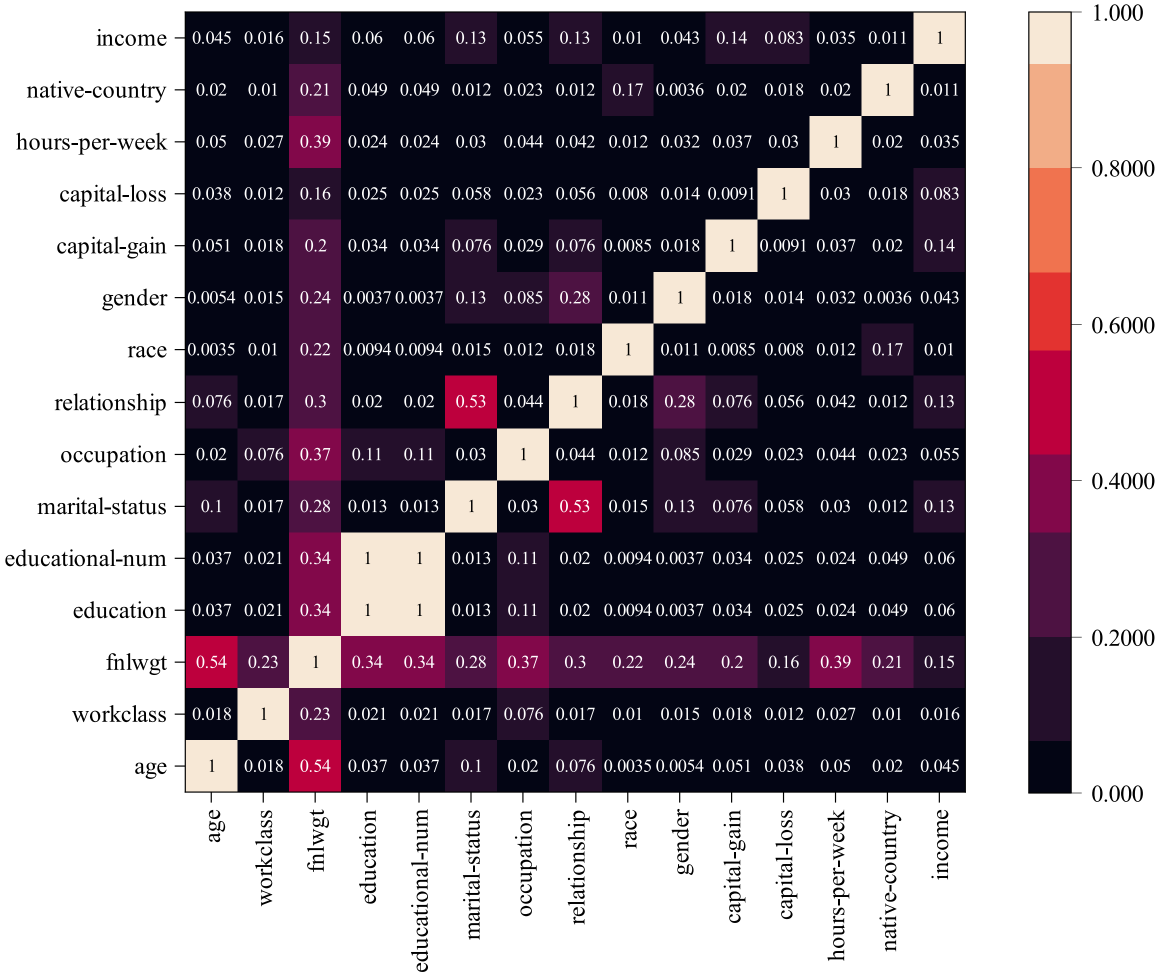
We are mainly concerned with two comparisons. On the one hand, compared with setting A, if the classifiers trained on synthetic data have poor prediction performance for the explicit privacy attribute on synthetic data (setting B), our framework IMPOSTER is able to eliminate the correlation between implicit and explicit privacy attributes. On the other hand, compared with setting A, if the classifiers trained on synthetic data have good prediction performance for the class label on synthetic data (setting B), the generated synthetic data can capture the corresponding relationship between attributes and labels, and the association between attributes, that is, data utility.

5.4 Correlation elimination

In this subsection, we conduct elaborate experiments to illustrate whether our framework IMPOSTER is able to eliminate the correlation between implicit and explicit privacy attributes. Specifically, we first use the implicit privacy detection module to explore the correlation between the explicit privacy attribute and other attributes in the original dataset. Figure 6 illustrates the normalized mutual information between corresponding attributes in Adult. Here, we treat “gender” as an explicit privacy attribute and set $\theta = 0.01$. We then adopt implicit privacy attributes to construct classifiers to infer the attribute “gender” on setting A and B. From Table 1, we can observe that, compared with setting A, all the classifiers trained on synthetic data generated by IMPOSTER have a significantly decreased performance in predicting the explicit privacy attribute (setting B). Therefore, our framework IMPOSTER is able to eliminate the correlation between implicit and explicit privacy attributes.

5.5 Data utility

In order to evaluate whether our framework can guarantee the data utility, we use the synthetic dataset generated by IMPOSTER to predict the class label “income” on setting A and B. From Table 1, we can see that, compared with setting A, even though the performance of all classifiers trained on setting B in predicting the class label decreases slightly, the prediction accuracy of training on data generated by IMPOSTER is higher than 81%. This result reflects that synthetic samples generated by IMPOSTER have captured the relationship between attributes and class labels well. Therefore, our framework can guarantee data utility while protecting implicit privacy.



	age	workclass	fnlwgt	education	educational-num	marital-status	occupation	relationship	race	gender	capital-gain	capital-loss	hours-per-week	native-country	income
income	0.045	0.016	0.15	0.06	0.06	0.13	0.055	0.13	0.01	0.043	0.14	0.083	0.035	0.011	1
native-country	0.02	0.01	0.21	0.049	0.049	0.012	0.023	0.012	0.17	0.0036	0.02	0.018	0.02	1	0.011
hours-per-week	0.05	0.027	0.39	0.024	0.024	0.03	0.044	0.042	0.012	0.032	0.037	0.03	1	0.02	0.035
capital-loss	0.038	0.012	0.16	0.025	0.025	0.058	0.023	0.056	0.008	0.014	0.0091	1	0.03	0.018	0.083
capital-gain	0.051	0.018	0.2	0.034	0.034	0.076	0.029	0.076	0.0085	0.018	1	0.0091	0.037	0.02	0.14
gender	0.0054	0.015	0.24	0.0037	0.0037	0.13	0.085	0.28	0.011	1	0.018	0.014	0.032	0.0036	0.043
race	0.0035	0.01	0.22	0.0094	0.0094	0.015	0.012	0.018	1	0.011	0.0085	0.008	0.012	0.17	0.01
relationship	0.076	0.017	0.3	0.02	0.02	0.53	0.044	1	0.018	0.28	0.076	0.056	0.042	0.012	0.13
occupation	0.02	0.076	0.37	0.11	0.11	0.03	1	0.044	0.012	0.085	0.029	0.023	0.044	0.023	0.055
marital-status	0.1	0.017	0.28	0.013	0.013	1	0.03	0.53	0.015	0.13	0.076	0.058	0.03	0.012	0.13
educational-num	0.037	0.021	0.34	1	1	0.013	0.11	0.02	0.0094	0.0037	0.034	0.025	0.024	0.049	0.06
education	0.037	0.021	0.34	1	1	0.013	0.11	0.02	0.0094	0.0037	0.034	0.025	0.024	0.049	0.06
fnlwgt	0.54	0.23	1	0.34	0.34	0.28	0.37	0.3	0.22	0.24	0.2	0.16	0.39	0.21	0.15
workclass	0.018	1	0.23	0.021	0.021	0.017	0.076	0.017	0.01	0.015	0.018	0.012	0.027	0.01	0.016
age	1	0.018	0.54	0.037	0.037	0.1	0.02	0.076	0.0035	0.0054	0.051	0.038	0.05	0.02	0.045

Figure 6. The normalized mutual information between corresponding attributes in Adult

Table 1. Accuracy and F1-score of predicting "gender" (the explicit privacy attribute) and "income" (the class label) in classifiers on setting A and B

	Setting A				Setting B			
	Predict "gender"		Predict "income"		Predict "gender"		Predict "income"	
$\theta = 0.01$, $\lambda = 1$	Accuracy	F1-score	Accuracy	F1-score	Accuracy	F1-score	Accuracy	F1-score
XGBoost	0.8455	0.8266	0.8331	0.7567	0.6440	0.4444	0.8326	0.7298
GBDT	0.8439	0.8269	0.8320	0.7538	0.6726	0.4021	0.8229	0.7176
RF	0.8426	0.8221	0.8319	0.7564	0.6509	0.4414	0.8295	0.7349
MLP	0.8448	0.8237	0.8311	0.7648	0.6732	0.5417	0.8324	0.7169
LR	0.8474	0.8301	0.8332	0.7532	0.6732	0.4024	0.8138	0.7030

5.6 Comparative experiment

To verify the superiority of IMPOSTER, we use the traditional privacy protection method kRR mechanism, a state-of-the-art and representative algorithm in differential privacy, to protect implicit privacy. Specifically, given a privacy budget ε, we keep the value of each implicit privacy attribute unchanged with high probability and flip it to another value in the same attribute value space with low probability. Then, we utilize the data disturbed by kRR to train the GBDT classifier to predict the class label and the explicit privacy attribute, respectively.

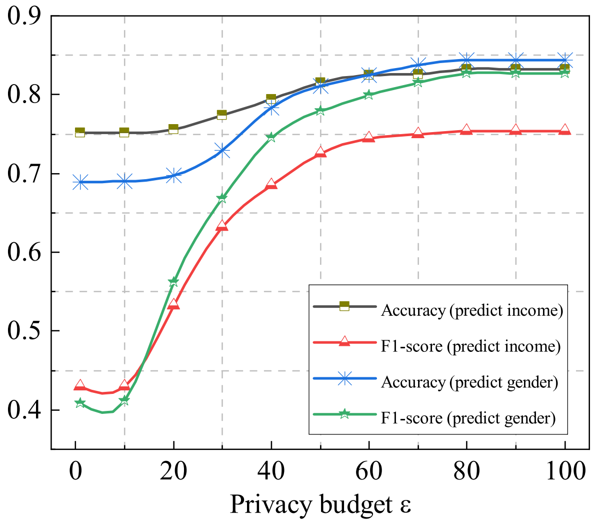


Figure 7. The prediction performance of data perturbed by the kRR mechanism on the class label and the explicit privacy attribute, respectively. The x-axis denotes different values of privacy budget ε, and the y-axis denotes F1-score and accuracy

Figure 7 indicates the prediction performance of data perturbed by the comparative method kRR mechanism on the class label and the explicit privacy attribute. From Figure 7, we can observe that, with the privacy budget ε changing from 1 to 100, the prediction performance of disturbed implicit privacy attributes for the class label and the explicit privacy attribute increases together. By comparing the data of Table 1 and Figure 7, when the accuracy of the predicted explicit privacy attribute reaches 68.91% on the data disturbed by kRR, the accuracy of the predicted class label is only 75.17%. However, the accuracy of the predicted explicit privacy attribute and class label is 67.26% and 82.29%, respectively on the data generated by IMPOSTER. We can conclude that, the highest prediction performance for "gender" and "income" does not exceed the prediction performance on the original data, either on data disturbed by the kRR mechanism or on the data generated by IMPOSTER. However, compared with the kRR mechanism, our proposed framework can maintain better data utility while providing the same privacy-preserving level.

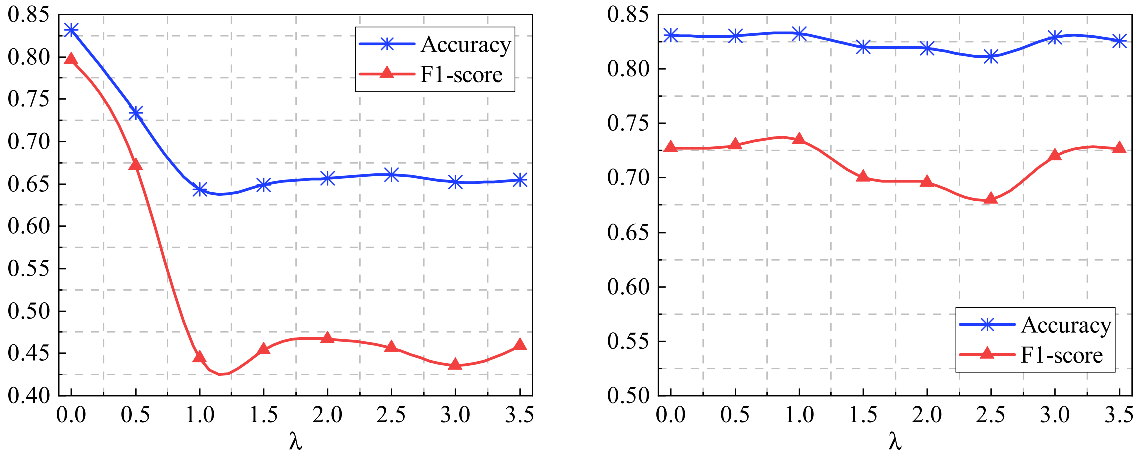
5.7 Parameter sensitivity analysis

In this part, we evaluate the sensitivity of parameters θ and λ, respectively. Except for the parameters explored, all the other parameters take default values. As a correlation threshold, the larger θ indicates that each attribute selected has a higher correlation with the explicit privacy attribute. We first get the candidate set of implicit privacy attributes by varying θ and then train an XGBoost classifier to evaluate the accuracy and F1-score of the predicted "gender" (the explicit privacy attribute) and "income" (the class label) on settings A and B, as shown in Table 2. We can observe that with the decrease in θ, the accuracy and F1-score of the predicted "gender" and "income" tend to increase on setting A, *i.e.*, smaller values of θ offer higher accuracy and F1-score. On setting B, however, with the decrease in θ, the accuracy and F1-score of the predicted "gender" are lower than 67% and 46%, respectively, while the accuracy and F1-score of the predicted "income" are slightly lower than those on setting A under the same θ. Therefore, our framework IMPOSTER can maintain good data utility while protecting implicit privacy effectively.

The trade-off coefficient λ is an important parameter of IMPOSTER, which is used to balance the data utility and privacy level of synthetic data. We evaluate how parameter λ affects the synthetic datasets generated by IMPOSTER in terms of two dimensions: correlation elimination and data utility. For correlation elimination, when λ becomes larger, the framework IMPOSTER tends to generate synthetic data with a lower correlation between implicit and explicit privacy attributes. In essence, the game of G and D_1 is to generate synthetic data similar to the original data, including capturing the correlation between attributes, which limits the extent to which IMPOSTER can eliminate the correlation. Therefore, when parameter λ increases to a certain degree, the accuracy and F1-score show fluctuations in a certain

Table 2. Accuracy and F1-score of predicting "gender" (the explicit privacy attribute) and "income" (the class label) with different θ on setting A and B

	Setting A				Setting B			
	Predict "gender"		Predict "income"		Predict "gender"		Predict "income"	
θ	Accuracy	F1-score	Accuracy	F1-score	Accuracy	F1-score	Accuracy	F1-score
0.002	0.8521	0.8347	0.8415	0.7693	0.6549	0.4566	0.8377	0.7564
0.01	0.8455	0.8266	0.8331	0.7567	0.6440	0.4444	0.8326	0.7298
0.02	0.8424	0.8237	0.8257	0.7441	0.6524	0.4464	0.8227	0.7205
0.1	0.8021	0.7668	0.7771	0.6401	0.6615	0.4443	0.7687	0.5733



(a)

(b)

Figure 8. The parameter sensitivity of IMPOSTER with different λ, where the x-axis denotes different values of trade-off coefficient λ, whereas the y-axis denotes F1-score and accuracy. (a) Correlation elimination (performance of the predicted explicit privacy attribute "gender") in synthetic datasets from IMPOSTER with different λ. (b) Data utility (performance of the predicted class label "income") in synthetic datasets from IMPOSTER with different λ

interval. This can be observed in Figure 8a, where we train an XGBoost classifier on the synthetic data generated by IMPOSTER to predict the explicit privacy attribute "gender". For data utility, we adopt the synthetic data generated by IMPOSTER to construct an XGBoost classifier to predict the class label "income". Figure 8b shows the performance curves of the predicted class label "income" with different λ. From Figure 8b, we can observe that with the increase of λ, accuracy and F1-score keep relatively steady only with a slight fluctuation. Observations from Figures 8a and b illustrate that IMPOSTER can get rid of the correlation between implicit and explicit privacy attributes while preserving data utility. Note that when λ is around 1, our framework IMPOSTER achieves the best performance, and when $\lambda = 0$, it degenerates to CGAN equipped with VAE, which cannot be used to remove data correlation between implicit and explicit privacy attributes.

6 Conclusion and future work

This paper addresses a special and imperceptible class of privacy, called implicit privacy, and then proposes an ex-ante implicit privacy-preserving framework based on data generation, called IMPOSTER, which consists of implicit privacy detection and protection modules. Specifically, the former uses normalized mutual information to detect attributes strongly related to privacy attributes. The latter equips the standard GAN framework with an additional discriminator, which is used to eliminate the association between explicit and implicit privacy attributes. Experimental results demonstrate that IMPOSTER can learn a generator producing data without implicit privacy while preserving good data utility.

In future work, on the one hand, we will adopt the Rényi entropy [58] to explore the correlation between multi-attributes and explicit privacy attributes, which is an open and interesting question. On the other hand, we will apply the proposed IMPOSTER framework to address the implicit privacy issue of time series data in the financial risk control scenario, generate data that eliminate the implicit privacy as much as possible to meet user expectations and regulatory requirements, and replace the real data with the generated data to train the financial anti-fraud model and improve the robustness of financial risk control systems.

Conflict of Interest

The authors declare that they have no conflict of interest.

Data Availability

The data supporting the finding of this study are publicly available in UCI Machine Learning Repository at https://archive.ics.uci.edu/ml/datasets/adult.

Authors' Contributions

Qing Yang contributed to the theoretical development, experimental design and manuscript drafting. Cheng Wang made a significant contribution to the theoretical development and revising it for intellectual content. Teng Hu contributed to the experimental design and revised the content of the manuscript. Xue Chen contributed to revising and polishing the content of the manuscript. Changjun Jiang made a significant contribution to the theoretical development and jointly wrote this paper.

Acknowledgements

We are very grateful to all editors and reviewers for their constructive comments. In addition, we thank Jipeng Cui for revising and polishing the content of the manuscript.

Funding

This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFB2100801, in part by the National Natural Science Foundation of China (NSFC) under Grant 61972287, and in part by the Fundamental Research Funds for the Central Universities under Grant 22120210524.

Supporting Information

The supporting information is available online at https://sands.edpsciences.org/10.1051/sands/2022008/olm. The supporting information is published as submitted, without typesetting or editing. The responsibility for scientific accuracy and content remains entirely with the authors.

References

[1] Han J, Pei J and Kamber M. Data Mining: Concepts and Techniques. The Netherlands: Elsevier, 2011.
[2] Jia JS, Lu X and Yuan Y et al. Population flow drives spatio-temporal distribution of COVID-19 in China. Nature 2020, **582**: 389–94.
[3] Park Y and Ho JC. Tackling overfitting in boosting for noisy healthcare data. IEEE Trans Knowl Data Eng 2021; **33**: 2995–3006.
[4] Wang W, Lesner C and Ran A et al. Using small business banking data for explainable credit risk scoring. Proc AAAI Conf Artif Intell 2020; **34**: 13396–401.
[5] Liu Y, Ao X and Zhong Q et al. Alike and unlike: Resolving class imbalance problem in financial credit risk assessment. In: Proc. 29th ACM Int. Conf. Inf. Knowl. Manag. Virtual Event, Ireland, Oct. 19–23, 2020, 2125–8.
[6] De Montjoye YA, Radaelli L and Singh VK et al. Unique in the shopping mall: On the reidentifiability of credit card metadata. Science 2015; **347**: 536–9.
[7] Zhang L, Shen J and Zhang J et al. Multimodal marketing intent analysis for effective targeted advertising. IEEE Trans Multim 2022; **24**: 1830–43.
[8] Deng A and Hooi B. Graph neural network-based anomaly detection in multivariate time series. Proc AAAI Conf Artif Intell 2021; **35**: 4027–35.
[9] Hu W, Gao J and Li B et al. Anomaly detection using local kernel density estimation and context-based regression. IEEE Trans Knowl Data Eng 2018; **32**: 218–33.
[10] Wu FJ and Luo T. Crowdprivacy: Publish more useful data with less privacy exposure in crowdsourced location-based services. ACM Trans Priv Secur 2020; **23**: 6:1–25.
[11] Holt JD and Chung SM. Efficient mining of association rules in text databases. In: Proc. 1999 ACM CIKM Int. Conf. Inf. Knowl. Manag., Kansas City, Missouri, USA, Nov. 2–6, 1999, 234–42.
[12] Sankar L, Rajagopalan SR and Poor HV. Utility-privacy tradeoffs in databases: An information-theoretic approach. IEEE Trans Inf Forensics Secur 2013, **8**: 838–52.
[13] Narayanan A and Shmatikov V. Robust de-anonymization of large sparse datasets. In: 2008 IEEE Symp. Secur. Priv. (SP), Oakland, CA, USA, May 18–22, 2008, 111–25.

[14] Li S, Ji X and You W. A personalized differential privacy protection method for repeated queries. In: 2019 IEEE 4th Int. Conf. Big Data Anal. (ICBDA), Suzhou, China, Mar. 15–18, 2019, 274–280.

[15] Dwork C and Roth A. The algorithmic foundations of differential privacy. Found Trends Theor Comput Sci 2014; **9**: 211–407.

[16] Machanavajjhala A, Kifer D and Gehrke J et al. *L*-diversity: Privacy beyond *k*-anonymity. ACM Trans Knowl Discov Data 2007; **1**: 3.

[17] Li N, Li T and Venkatasubramanian S. t-closeness: Privacy beyond k-anonymity and l-diversity. In: Proc. 23rd Int. Conf. Data Eng., The Marmara Hotel, Istanbul, Turkey, Apr. 15–20, 2007, 106–15.

[18] Yang Q, Liu Y and Chen T et al. Federated machine learning: Concept and applications. ACM Trans Intell Syst Technol 2019; **10**: 12:1–19.

[19] Mohassel P and Zhang Y. Secureml: A system for scalable privacy-preserving machine learning. In: 2017 IEEE Symp. Secur. Priv. (SP), San Jose, CA, USA, May 22–26, 2017, 19–38.

[20] Chen H, Dai W and Kim M et al. Efficient multi-key homomorphic encryption with packed ciphertexts with application to oblivious neural network inference. In Proc. 2019 ACM SIGSAC Conf. Comput. Commun. Secur., CCS 2019, London, UK, Nov. 11–15, 2019, 395–412.

[21] Fredrikson M, Lantz E and Jha S et al. Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing. In: Proc. 23rd USENIX Secur. Symp., San Diego, CA, USA, Aug. 20–22, 2014, 17–32.

[22] Krause A and Horvitz E. A utility-theoretic approach to privacy and personalization. In: Proc. Twenty-Third AAAI Conf. Artif. Intell., Chicago, Illinois, USA, July 13–17, 2008, Vol. 8, 1181–8.

[23] Gross R, Airoldi E and Malin B et al. Integrating Utility into Face De-identification. Berlin, Heidelberg: Springer, 2006.

[24] Yang Q, Wang C and Wang C et al. Fundamental limits of data utility: A case study for data-driven identity authentication. IEEE Trans Comput Soc Syst 2020; **8**: 398–409.

[25] Datta A, Fredrikson M and Ko G et al. Use privacy in data-driven systems: Theory and experiments with machine learnt programs. In: Proc. 2017 ACM SIGSAC Conf. Comput. Commun. Secur., CCS 2017, Dallas, TX, USA, Oct. 30–Nov. 03, 2017, 1193–210.

[26] Tseng BW and Wu PY. Compressive privacy generative adversarial network. IEEE Trans Inf Forensics Secur 2020; **15**: 2499–513.

[27] Kim H, Park J and Min K et al. Anomaly monitoring framework in lane detection with a generative adversarial network. IEEE Trans Intell Transp Syst 2020; **22**: 1603–15.

[28] Ruffino C, Hérault R and Laloy E et al. Pixel-wise conditioned generative adversarial networks for image synthesis and completion. Neurocomputing 2020; **416**: 218–30.

[29] Zhang K, Zhong G and Dong J et al. Stock market prediction based on generative adversarial network. Proc Comput Sci 2019; **147**: 400–6.

[30] Xie L, Lin K and Wang S et al. Differentially private generative adversarial network, arXiv preprint arXiv:1802.06739, 2018.

[31] Jordon J, Yoon J and van der Schaar M. PATE-GAN: Generating synthetic data with differential privacy guarantees. In: 7th Int. Conf. Learn. Represent., ICLR 2019, New Orleans, LA, USA, May 6–9, 2019.

[32] Warner SL. Randomized response: A survey technique for eliminating evasive answer bias. J Am Stat Assoc 1965; **60**: 63–9.

[33] Aggarwal CC and Philip SY. A Condensation Approach to Privacy Preserving Data Mining. Berlin, Heidelberg: Springer, 2004.

[34] Dwork C. Differential privacy. In: Autom. Lang. Program. 33rd Int. Colloq. ICALP 2006, Venice, Italy, Jul. 10–14, 2006, Proc. Part II, 2006, 1–12.

[35] Sweeney L. k-anonymity: A model for protecting privacy. Int J Uncertain Fuzziness Knowl Based Syst 2002; **10**: 557–70.

[36] Zhao C, Zhao S and Zhao M et al. Secure multi-party computation: Theory, practice and applications. Inf Sci 2019; **476**: 357–72.

[37] Elgamal T. A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Trans Inf Theor 1985; **31**: 469–72.

[38] Ullah S, Li XY and Hussain MT et al. Kernel homomorphic encryption protocol. J Inf Secur Appl 2049; **48**: 102366.

[39] Kairouz P, Oh S and Viswanath P. Extremal mechanisms for local differential privacy. J Mach Learn Res 2016; **17**: 492–542.

[40] Goodfellow I, Pouget-Abadie J and Mirza M et al. Generative adversarial nets. In: Adv. Neural Inf. Process. Syst. 27: Annu. Conf. Neural Inf. Process. Syst. 2014, Montreal, Quebec, Canada, Dec. 8–13, 2014, 2672–2680.

[41] Mirza M and Osindero S. Conditional generative adversarial nets, arXiv preprint arXiv:1411.1784, 2014.

[42] Torra V. Data Privacy: Foundations, New Developments and the Big Data Challenge. Heidelberg: Springer, 2017.

[43] Xu L, Jiang C and Wang J et al. Information security in big data: Privacy and data mining. IEEE Access 2014; **2**: 1149–76.

[44] Estévez PA, Tesmer M and Perez CA et al. Normalized mutual information feature selection. IEEE Trans Neural Netw 2009; **20**: 189–201.

[45] Jaynes ET. Information theory and statistical mechanics. Phys Rev 1957; **106**: 620.

[46] Benesty J, Chen J and Huang Y et al. Pearson correlation coefficient. Berlin, Heidelberg: Springer, 2009.

[47] Nargesian F, Samulowitz H and Khurana U et al. Learning feature engineering for classification, In: Proc. Twenty-Sixth Int. Jt. Conf. Artif. Intell., IJCAI 2017, Melbourne, Australia, Aug. 19–25, 2017, 2529–35.

[48] Datta A, Sen S and Zick Y. Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems. In: 2016 IEEE Symp. Secur. Priv. (SP), San Jose, CA, USA, May 22–26, 2016, 598–617.

[49] Hild II KE, Erdogmus D and Torkkola K et al. Feature extraction using information-theoretic learning. IEEE Trans Pattern Anal Mach Intell 2006; **28**: 1385–92.

[50] Kingma DP and Welling M. Auto-encoding variational bayes. In: Bengio Y and LeCun Y, editors, 2nd Int. Conf. Learn. Represent., ICLR 2014, Ban, AB, Canada, Apr. 14–16, 2014, Conf. Track Proc., 2014.
[51] Menéndez ML, Pardo JA and Pardo L et al. The jensen-shannon divergence. J Frankl Inst 1997; **334**: 307–18.
[52] Chen T and Guestrin C. XGBoost: A scalable tree boosting system. In: Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, San Francisco, CA, USA, Aug. 13–17, 2016, 785–94.
[53] Friedman JH. Greedy function approximation: A gradient boosting machine. Annal Stat 2001; **29**: 1189–232.
[54] Li Q, Wen Z and He B. Practical federated gradient boosting decision trees. Proc AAAI Conf Artif Intell 2020; **34**: 4642–9.
[55] Breiman L. Random forests. Mach Learn 2001; **45**: 5–32.
[56] Anthony M and Bartlett PL. Neural Network Learning: Theoretical Foundations. Cambridge: Cambridge University Press, 2009.
[57] Pan X and Xu Y. A safe feature elimination rule for L1-regularized logistic regression. IEEE Trans Pattern Anal Mach Intell 2021.
[58] Fehr S and Berens S. On the conditional Rényi entropy. IEEE Trans Inf Theor 2014; **60**: 6801–10.

Qing Yang is now a Ph.D. student at the Department of Computer Science and Technology, Tongji University in Shanghai, China. His research interests include data mining, data privacy and identity authentication.

Cheng Wang received his Ph.D. degree from the Department of Computer Science and Technology, Tongji University in 2011. He is currently a professor at the Department of Computer Science and Technology, Tongji University. His research interests include cyberspace security and intelligent information service.

Teng Hu is now a Ph.D. student at the Department of Computer Science and Technology, Tongji University in Shanghai, China. His research interests include data mining, machine learning and fraud detection.

Xue Chen is now a Ph.D. student at the Department of Computer Science and Technology, Tongji University in Shanghai, China. Her research interests include data privacy and machine learning.

Changjun Jiang received his Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 1995. He is currently the Leader of the Key Laboratory of the Ministry of Education for Embedded System and Service Computing, Tongji University, Shanghai, China. He is an academician of the Chinese Academy of Engineering, an IET Fellow and an Honorary Professor with Brunel University London, Uxbridge, England. He has been the recipient of one international prize and seven prizes in the field of science and technology.