软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software [doi: 10.13328/j.cnki.jos.007179]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

本地差分隐私频率估计伪数据攻击及防御方法*

王源源[1], 朱友文[1], 吴启晖[2], 王 威[2], 王 箭[1]

[1](南京航空航天大学 计算机科学与技术学院, 江苏 南京 211106)

[2](南京航空航天大学 电子信息工程学院, 江苏 南京 211106)

通信作者: 朱友文, E-mail: zhuyw@nuaa.edu.cn

摘 要: 本地差分隐私被广泛地应用于保护用户隐私的同时收集和分析敏感数据, 但是也易于受到恶意用户的伪数据攻击. 子集选择机制和环机制是具有最优效用的频率估计本地差分隐私方案, 然而, 它们的抗伪数据攻击能力尚缺少深入地分析和评估. 因此, 针对子集选择机制和环机制, 设计伪数据攻击方法, 以评估其抗伪造攻击的能力. 首先讨论随机扰动攻击和随机项目攻击, 然后构建针对子集选择机制和环机制的攻击效用最大化伪数据攻击方法. 攻击者可以利用该攻击方法, 通过假用户向数据收集方发送精心制作的伪数据, 最大化地提高攻击者所选目标值的频率. 理论上严格分析和对比攻击效用, 并通过实验评估伪数据攻击效果, 展示伪数据攻击对子集选择机制和环机制的影响. 最后, 提出防御措施, 可缓解伪数据攻击的效果.

关键词: 本地差分隐私; 伪数据攻击; 防御; 子集选择机制; 环机制

中图法分类号: TP309

中文引用格式: 王源源, 朱友文, 吴启晖, 王威, 王箭. 本地差分隐私频率估计伪数据攻击及防御方法. 软件学报. http://www.jos.org.cn/1000-9825/7179.htm

英文引用格式: Wang YY, Zhu YW, Wu QH, Wang W, Wang J. Data Poisoning Attacks and Defense Methods for Frequency Estimation in Local Differential Privacy. Ruan Jian Xue Bao/Journal of Software (in Chinese). http://www.jos.org.cn/1000-9825/7179.htm

Data Poisoning Attacks and Defense Methods for Frequency Estimation in Local Differential Privacy

WANG Yuan-Yuan[1], ZHU You-Wen[1], WU Qi-Hui[2], WANG Wei[2], WANG Jian[1]

[1](College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China)

[2](College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China)

Abstract: Local differential privacy (LDP) is widely used to collect and analyze sensitive data while protecting user privacy. However, it is vulnerable to data poisoning attacks by malicious users. The k-subset mechanism and the wheel mechanism are LDP schemes with optimal utility for frequency estimation. Yet, their resistance to data poisoning attacks lacks in-depth analysis and evaluation. Therefore, data poisoning attack methods are designed to assess the resistance to data poisoning attacks of both the k-subset mechanism and the wheel mechanism. First, the random perturbed-value attack and random item attack are discussed, and then the maximal gain attack methods against the k-subset mechanism and the wheel mechanism are constructed. The attack methods can be exploited to maximize the frequencies of target items selected by attackers, which is achieved by sending carefully crafted poisoning data to the data collector via fake users. Theoretically, the attack gains are rigorously analyzed and compared, and the effects of data poisoning attacks are experimentally evaluated, demonstrating their impact on the k-subset mechanism and the wheel mechanism. Finally, defensive measures are proposed to mitigate the effects of data poisoning attacks.

Key words: local differential privacy (LDP); data poisoning attack; defense; k-subset mechanism; wheel mechanism

---

* 基金项目: 国家重点研发计划 (2021YFB3100400)
收稿时间: 2023-08-02; 修改时间: 2023-10-12; 采用时间: 2024-03-05; jos 在线出版时间: 2024-08-21

在大数据时代, 隐私保护技术对于收集和分析敏感数据至关重要. 本地差分隐私 (local differential privacy, LDP)[1,2]可以在数据采集过程中为用户隐私信息提供有效保护. LDP 在用户端对用户的原始数据进行编码、扰动, 达到保护隐私的效果. 然后, 用户将扰动后的数据发送给数据收集方, 数据收集方对所有用户的扰动数据进行聚合, 估计出特定计算任务的结果, 例如频率估计值. LDP 模型框架如图 1 所示. 在 LDP 模型框架中, 即使数据收集方不可信, 泄露了扰动后的用户数据, 用户隐私仍然可以得到保护. LDP 被广泛部署在大规模数据收集系统中, 例如, Google 在 Chrome 浏览器中部署了 RAPPOR 机制, 用来收集用户使用 Chrome 浏览器时的默认主页[3]; 苹果公司在 iOS 和 Mac OS 设备中跟踪用户使用 Safari 浏览器访问的网站, 并收集用户对页面加载时自动播放视频的偏好, 在这一过程中利用 LDP 进行隐私保护[4].

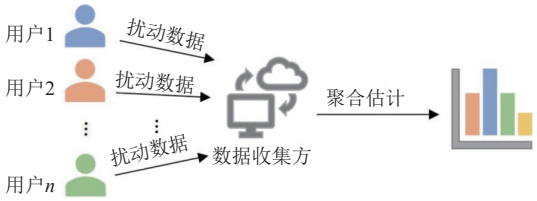


图 1　本地差分隐私模型框架

LDP 协议面临的一个重要问题是可能存在若干恶意用户向数据收集方发送伪造值. 对于数据收集方来说, 伪造值和真实数据的扰动值都是添加噪声后的数据, 所以很难区分恶意用户和普通用户. 因此, 恶意用户精心设计的伪造值可能会对估计结果造成严重影响. 在频率估计 LDP 协议中, 攻击者可以通过伪造若干个用户数据, 使得数据收集方对特定数据项的估算频率比其真实值高, 从而获得不正当利益. 例如, 《纽约时报》(New York Times) 报道了亚马逊旗下众包市场 Mechanical Turk 上的企业雇佣员工, 以每条 25 美分的价格发布假五星级 Yelp 评论[5].

子集选择机制[6,7]和环机制[8]是具有最优效用的 LDP 频率估计机制. 为了对它们抗伪数据攻击能力进行深入评估, 在本文中, 针对它们设计了攻击效用最大化的伪数据攻击方案. 攻击者可以向 LDP 协议中注入假用户, 并利用假用户向数据收集方发送精心设计的伪造数据. 攻击者的目标是提高目标项目的频率估计值. 子集选择机制中用户可以向数据收集方发送 k 个数据, 本文为假用户设计的伪造数据支持尽可能多的目标项目. 在环机制中, 挑选最优的哈希函数 h_m, 哈希函数 h_m 可以使尽可能多的目标项目映射值的覆盖区域存在交集, 从交集中为每个假用户随机选取一个值, 发送给数据收集方. 本文从理论上和实验上证明了, 对于子集选择机制和环机制, 设计的方案可以有效地提高目标项目的频率估计值. 并且, 本文设计了两种防御方法来防御伪数据攻击, 即后处理和限定阈值方法. 后处理利用最值归一化压缩攻击效用. 在限定阈值方法中, 设置频率阈值 τ, 如果某个项目的频率高于阈值 τ, 对该项目进行标记, 将同时含有所有标记项目的用户视为假用户, 排除假用户后再次计算目标项目的频率估计值, 可以减小攻击效用. 实验结果表明, 本文的防御方法可以有效缓解伪数据攻击的负面影响.

概括地说, 本文的主要贡献如下.

- 为了深入评估子集选择机制和环机制两种 LDP 协议的抗伪数据攻击能力, 针对它们设计了攻击效用最大化的伪数据攻击方案.
- 从理论上和实验上评估了对子集选择机制和环机制进行伪数据攻击的有效性, 结果显示本文设计的最大效用攻击可以有效地提高目标项目的频率估计值, 证实了子集选择机制和环机制抗伪数据攻击能力弱.
- 针对 LDP 协议的伪数据攻击提出了防御措施, 实验结果显示可有效缓解伪数据攻击的负面影响.

本文第 1 节介绍 LDP 协议伪数据攻击的相关工作. 第 2 节介绍本文所需的基础知识, 包括子集选择机制和环机制. 第 3 节介绍本文设计的针对子集选择机制和环机制的攻击方法和攻击效用. 第 4 节展示实验结果. 第 5 节介绍针对 LDP 协议伪数据攻击的防御机制. 最后第 6 节总结全文.

1　LDP 机制伪数据攻击相关工作

在机器学习算法中, 攻击者可以利用伪数据攻击操纵数据训练, 通过一定的策略修改原始训练数据集, 或者向

原始数据集中注入污染数据进行操纵攻击, 可以使机器学习分类器的分类边界发生偏移或者改变, 从而主动篡改机器学习模型, 产生错误的输出结果, 造成安全隐患[9]. 例如, 微软开发的与 Twitter 用户交谈的聊天机器人 Tay, 利用社交网络上的对话数据进行训练, 机器人 Tay 在上线 16 h 后被关闭, 因为它受到伪数据攻击后开始提出与种族主义相关的评论[10].

LDP 协议伪数据攻击的相关工作已有进展. 一些工作[11−14]观察到特定的随机响应方案容易受到操纵攻击, 并探讨了潜在的抵御方法. Cao 等人[15]对 kRR[16]、OUE[17]、OLH[17]进行伪数据攻击, 将攻击者选择的项目估计为高频率项目, 操纵数据估计结果. 文献 [15] 攻击的这 3 种 LDP 机制设计较为简单, 现在已经有一些性能更好的 LDP 机制. 另外, Cao 等人[18]还对频繁项识别进行了伪数据攻击. Wu 等人对键值数据的 LDP 协议 (PrivKVM[19]、PCKV-UE[20]、PCKV-GRR[20]) 进行伪数据攻击, 将攻击转化为一个双目标优化问题, 同时最大化攻击者选择的目标键的频率和均值, 弥补了对 LDP 协议中键值数据攻击的空白. 此外 Wu 等人提出了两种针对键值数据 LDP 协议的伪数据攻击的防御措施, 包括基于单类分类器的检测和基于异常得分的检测. Li 等人[21]针对均值估计和方差估计的 LDP 协议 (SR[22]、PM[23]) 进行伪数据攻击, 该攻击可以将估计的均值和方差调整到攻击者所设定的目标值. 与前人攻击不同的是, Li 等人提出的攻击包含输入污染攻击和输出污染攻击. 输入污染攻击可以通过假用户向 LDP 协议中注入虚假的输入数据, 输出污染攻击可以修改受控用户端的 LDP 扰动机制的输出, 它为 LDP 协议中的伪数据攻击提供了一个新的视角.

基于上述分析, 本文对两种效用最优的 LDP 协议 (子集选择机制和环机制) 进行伪数据攻击, 根据这两种 LDP 协议的特点, 为假用户设计使攻击效用最大化的伪造数据. 攻击目标是提高选定项目的估计频率, 降低子集选择机制和环机制的整体性能.

2 预备知识与问题定义

这里先给出 LDP 频率机制所考虑的问题场景. 假设有 n 个用户, 每个用户持有一个私有的隐私数据. 用户原始数据的取值范围为$\{1, 2,\ldots, d\}$, 简记为 $[d]$. 数据收集方的任务是估计输入数据取值范围中每个值 $x\in[d]$ 在所有 n 个用户中出现的频率, 即原始数据等于 x 的用户在所有用户中的比例.

LDP 机制可以在保护每个用户数据不泄漏给数据收集方的同时, 使得数据收集方获得高准确度的频率估计结果. 然而, LDP 机制也面临着恶意用户伪数据攻击等问题, 下面简要介绍 LDP 模型和实现机制, 并定义本文所关注的伪数据攻击.

2.1 本地差分隐私

本地差分隐私是差分隐私的变体, 能够在不可信环境中保护用户隐私数据. 用户使用随机化扰动机制来扰动他们的数据, 数据收集方可以根据该随机化扰动机制来估计某些统计量, 例如频率. 本质上, 对于任意两个输入值 v_1 和 v_2, LDP 确保了数据收集方在接收到输出 y 时, 不能分辨出输入是 v_1 还是 v_2. 本地差分隐私定义如下.

定义 1. 本地差分隐私. 一个随机化算法 $R: D\to Y$ 满足 ε-本地差分隐私, 当且仅当, 对任意两个用户的数据 $v_1, v_2\in D$, 以及任意可能的输出 $y\in Y$ 满足如下不等式:

$$\Pr(R(v_1)=y)\leqslant \mathrm{e}^{\varepsilon}\Pr(R(v_2)=y) \tag{1}$$

从定义 1 中可以看出, LDP 中隐私预算 ε 决定隐私保护的强度.

2.2 子集选择机制

子集选择机制[6,7]是一种用于类别型数据频率估计的 LDP 协议, 其对项目频率的估计是无偏估计, 均方误差为 $\Theta\left(\dfrac{\mathrm{e}^{\varepsilon}d}{n(\mathrm{e}^{\varepsilon}-1)^2}\right)$. 相比于其他 LDP 频率估计机制, 子集选择机制估计频率的均方误差数量级最小, 具有最优效用. 在该机制中, 每个用户需要从 d 个数据中随机抽取 k $(1\leqslant k\leqslant d)$ 个数据, 传输给数据收集方, 依照概率 p 决定是否提交自己的真实数据:

$$p=\frac{k\mathrm{e}^{\varepsilon}}{k\mathrm{e}^{\varepsilon}+d-k} \tag{2}$$

依公式 (2) 概率, 如果用户选中自己的真实数据进行提交, 则需要从剩余 $d-1$ 个数值中随机选取 $k-1$ 个不同的数值提交. 如果用户没有选中自己的真实数据, 则将从剩余 $d-1$ 个数值中随机选取 k 个不同的数值提交. 如果直接发送选中的 k 个不同的数值, 通信代价过大. 因此, 每个用户向数据收集方发送一个长为 d 的二进制向量, 用来代表自己选中的数据. 数据收集方根据收到的扰动数据, 推导出项目频率估计值, 频率估计公式如下:

$$q = p\cdot\frac{k-1}{d-1} + (1-p)\cdot\frac{k}{d-1} = \frac{k-p}{d-1} \tag{3}$$

$$\hat{f}_v = \frac{\tilde{f}_v - q}{p-q} \tag{4}$$

其中, $\hat{f}_v$ 表示项目 v 的估计频率, $v\in\{1,2,\ldots,d\}$. $\tilde{f}_v$ 表示用户提交的扰动数据中项目 v 的频率, 数据收集方可以根据用户发送的数据直接进行统计. f_v 表示项目 v 的真实频率, 估计值 $\hat{f}_v$ 是对 f_v 的无偏估计:

$$E\left[\hat{f}_v\right] = \frac{E\left[\tilde{f}_v\right]-q}{p-q} = \frac{f_v p + (1-f_v)q - q}{p-q} = f_v \tag{5}$$

总方差为:

$$\begin{aligned}\sum_{v=1}^{d} Var\left(\hat{f}_v\right) &= \sum_{v=1}^{d} Var\left(\frac{\tilde{f}_v - p}{p-q}\right) = \frac{1}{(p-q)^2}\sum_{v=1}^{d} Var\left(\tilde{f}_v\right)\\ &= \frac{1}{(p-q)^2}\sum_{v=1}^{d}\left[f_v p(1-p) + (1-f_v)q(1-q)\right]\\ &= \frac{1}{(p-q)^2}\left[p(1-p) + (d-1)q(1-q)\right]\\ &= \frac{(d-1)\left[4d\mathrm{e}^{\varepsilon} - (\mathrm{e}^{\varepsilon}+1)^2\right]}{d(\mathrm{e}^{\varepsilon}-1)^2}\end{aligned} \tag{6}$$

2.3 环机制

环机制[8]是一种用于类别型数据和集值数据频率估计的 LDP 协议, 本文采用环机制进行类别型数据的频率估计. 环机制对项目频率的估计是无偏估计, 其均方误差为 $\Theta\left(\frac{\mathrm{e}^{\varepsilon}d}{n(\mathrm{e}^{\varepsilon}-1)^2}\right)$. 相比于其他 LDP 频率估计机制, 环机制估计频率的均方误差数量级最小, 是一种效用最优的 LDP 协议. 同时, 环机制具有通信代价低、计算成本小的优点. 在环机制中, 每个用户的真实数据通过以下 3 个步骤进行处理.

第 1 步, 使用用户 ID 或随机生成的数字作为种子, 通过用户特定的哈希函数将用户的项目 x 映射到 [0.0, 1.0) 范围内的一个数值 v.

第 2 步, 用校准的概率分布 Q 在 [0.0, 1.0) 上随机化数值 v, 得到符合该概率分布的随机变量 y. 概率分布 Q 的定义如公式 (7) ($0.0\leqslant y, v<1.0$), 其中覆盖参数 $w\in(0.0, 0.5)$ 用来控制真/假覆盖概率的覆盖区长度:

$$Q[y|v] = \begin{cases} \dfrac{\mathrm{e}^{\epsilon}}{w\cdot\mathrm{e}^{\epsilon}+(1-w)}, & \text{若 } v\leqslant y<v+w \text{ 或 } 0\leqslant y<v+w-1\\ \dfrac{1}{w\cdot\mathrm{e}^{\epsilon}+(1-w)}, & \text{若 } y\geqslant v+w \text{ 或 } y\leqslant v\end{cases} \tag{7}$$

第 3 步, 从分布 $Q[y|v]$ 中抽取一个值 $z\in[0.0, 1.0)$, 然后将其发送给服务器 (连同种子一起). 根据用户发送的 z 值, 服务器可以估计出每个项目的频率.

形式上, 把用户的真实项目 x 在 [0.0, 1.0) 中的映射值表示为 v, 把覆盖区域 $[v, v+w)$ 或 $[0, v+w-1)$ 表示为 C_v. 用户真实项目 x 的扰动数据 $z\in[0.0, 1.0)$ 在 x 的覆盖区域 C_v 的概率为:

$$P_t = P[z\in C_v|x] = \frac{w\cdot\mathrm{e}^{\varepsilon}}{w\cdot\mathrm{e}^{\varepsilon}+(1-w)} \tag{8}$$

当用户的真实数据为 x' , 假设哈希函数 h 是完美的, 即 $h(x')$ 是均匀随机的, 且与 $h(x)$ 无关, 那么 x' 的扰动数据 $z\in[0.0, 1.0)$ 在 x 的覆盖区域 C_v 的概率为:

$$P_f = E[P[z \in C_v | x']] = w \tag{9}$$

根据真实覆盖概率 P_t 和虚假覆盖概率 P_f, 可以得到 z 在 x 的覆盖区域 C_v 的概率 $P[z \in C_v]$:

$$P[z \in C_v] = f_x \cdot P_t + (1.0 - f_x) \cdot P_f \tag{10}$$

其中, f_x 表示项目 x 的频率, $x \in \{1, \ldots, d\}$. $P[z \in C_v]$ 表示用户提交的扰动数据 z 在区域 C_v 内的概率, 将 $P[z \in C_v]$ 记为 $\tilde{f}_x$, 数据收集方可以根据用户发送的数据直接统计得到 $\tilde{f}_x$. 项目 x 的估计频率 $\hat{f}_x$ 如下:

$$\hat{f}_x = \frac{\tilde{f}_x - P_f}{P_t - P_f} \tag{11}$$

可以验证估计值 $\hat{f}_x$ 是真实项目分布 f_x 的无偏估计:

$$E\left[\hat{f}_x\right] = E\left[\frac{\tilde{f}_x - P_f}{P_t - P_f}\right] = \frac{E\left[\tilde{f}_x\right] - P_f}{P_t - P_f} = \frac{f_x P_t + (1 - f_x) P_f - P_f}{P_t - P_f} = f_x \tag{12}$$

环机制处理类别数据时, 覆盖参数 $w = \dfrac{1}{1 + \mathrm{e}^{\varepsilon}}$, 可以得到 $P_t = \dfrac{1}{2}$ 和 $P_f = \dfrac{1}{1 + \mathrm{e}^{\varepsilon}}$, 因此总方差为:

$$\begin{aligned}
\sum_{x=1}^{d} Var\left(\hat{f}_x\right) &= \sum_{i=1}^{d} Var\left(\frac{\tilde{f}_x - P_f}{P_t - P_f}\right) = \frac{1}{(P_t - P_f)^2} \sum_{x=1}^{d} Var\left(\tilde{f}_x\right) \\
&= \frac{1}{(P_t - P_f)^2} \sum_{x=1}^{d} \left[f_x P_t (1 - P_t) + (1 - f_x) P_f (1 - P_f)\right] \\
&= \frac{1}{(P_t - P_f)^2} \left[P_t (1 - P_t) + (d - 1) P_f (1 - P_f)\right] \\
&= 1 + \frac{4d\mathrm{e}^{\varepsilon}}{(\mathrm{e}^{\varepsilon} - 1)^2}
\end{aligned} \tag{13}$$

2.4 问题定义

本文中, 假设攻击者可以向 LDP 机制中注入一些假用户, 这些假用户可以向数据收集方发送伪造的扰动数据, 如图 2 所示. 这一攻击具有现实意义, 它可以提高非频繁项目的频率, 改变 top-k 频繁项集, 造成安全威胁. 例如, Chrome 浏览器根据用户的浏览量和评分推荐流行主页, 攻击者通过在数据收集过程中注入伪造数据来发起伪数据攻击, 将某钓鱼网站推荐为流行主页, 造成信息泄露的安全隐患; 或者在导航软件中, 系统统计来自用户的路线评分以提供更好的服务, 攻击者向系统发送大量伪造数据, 例如, 为某路况较差的低评分路线发布大量高分评价, 将其推荐为高评分路线, 导致用户体验不佳.

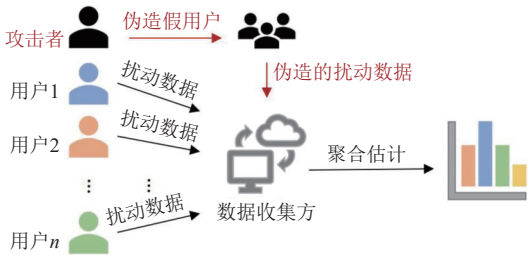


图 2 本地差分隐私伪数据攻击模型

假设系统中的真实用户人数为 n, 攻击者注入 m 个假用户, 因此用户总人数变为 $n+m$. LDP 机制在用户端执行编码和扰动步骤, 攻击者可以获得 LDP 机制的关键细节, 例如, 用户数据域 $[d]$、扰动值的支持集. 攻击者的目标项目集 T 包括 r $(1 \leqslant r \leqslant d)$ 个项目, 表示为 $T = \{t_1, t_2, \ldots, t_r\}$. 攻击者的目标是尽可能大地增加集合 T 中每个项目的频率估计值, 因此攻击者精心制作从假用户发送到数据收集方的扰动值. 用 Y 表示假用户的扰动值集合, Y 的每个元素 y_i 是攻击者为假用户 i 伪造的扰动值. 扰动值 y_i 可以是集合 (如子集选择机制), 也可以是元组 (如环机制).

假设 f_t 表示目标项目 t 的真实频率, $\hat{f}_{t,a}$ 和 $\hat{f}_{t,b}$ 分别表示攻击前和攻击后 LDP 机制对目标项目 t 的频率估计

值, 目标项目 t 的频率增加值表示为:

$$\Delta\hat{f}_t = \hat{f}_{t,b} - \hat{f}_{t,a},\ \forall t \in T \tag{14}$$

频率增加值 $\Delta\hat{f}_t$ 越大, 表示攻击越成功.

攻击的整体效用 G 定义为目标项目的频率增加值期望的总和, 即:

$$G(Y) = \sum\nolimits_{t\in T} E[\Delta\hat{f}_t] \tag{15}$$

攻击者的目标是制作扰动值 Y 以最大化整体效用 G.

3 本地差分隐私攻击方法

LDP 协议伪数据攻击常用方法包括随机扰动值攻击 (random perturbed-value attack, RPA)、随机项目攻击 (random item attack, RIA) 和最大效用攻击 (maximal gain attack, MGA) 这 3 类. RPA 和 RIA 是两个基线攻击. 在 RPA 中, 每个假用户从用户数据域中随机选择一个值, 将该值发送给数据收集方. RIA 考虑了目标项目, 每个假用户从目标项目集中随机选择一个目标项目, 按照扰动规则对该项目进行扰动, 将扰动后的值发送给数据收集方. MGA 的思路是将针对特定目标项目集的攻击转换为最优化问题, 为假用户精心设计扰动值, 使攻击效用最大化, 需要结合具体的 LDP 机制进行设计和构造.

子集选择机制和环机制是具有最优效用的 LDP 协议频率估计方法. 相对于其他 LDP 频率估计机制, 在相同隐私保护等级下, 这两种频率估计方法计算结果的准确度更高. 然而, 目前尚缺少对子集选择机制和环机制的伪数据攻击, 对它们受伪数据攻击的影响程度缺少深入分析. Cao 等人评估了伪数据攻击对 kRR、OUE 和 OLH 的频率估计的有效性[15], 然而他们的主要攻击方法 MGA 具有针对性, 并不适用于子集选择机制和环机制. 因此, 本文设计了针对子集选择机制和环机制的伪数据攻击. 本文首先在子集选择机制和环机制上进行了 RPA 和 RIA 攻击, 评估了攻击效用; 然后, 根据子集选择机制和环机制的特点, 设计了针对这两种机制的 MGA 攻击, 并利用所提攻击方案深入评估了伪数据攻击对子集选择机制和环机制的影响.

3.1 子集选择机制攻击方法

3.1.1 RPA 攻击

每个假用户生成一个长度为 d 的二进制向量, 该向量初始为零向量. 攻击者控制的每个假用户从用户数据域 $\{1, 2,\ldots, d\}$ 中随机选择 k 个值, 即在用户的二进制向量中, 被选中的 k 个值的对应比特设置 1. 这样, 攻击者就伪造了一个随机扰动项目, 假用户将伪造的二进制向量发送给数据收集方. 在 RPA 中, 假用户发送给服务器的数据服从均匀分布, 其将作为参照组, 用来与其他攻击进行对比.

3.1.2 RIA 攻击

RIA 考虑到了攻击者选择的目标项目集 T. 为了提高集合 T 中项目的频率, 攻击者控制每个假用户从集合 T 中随机选择一个项目 t 作为该假用户所持有的项目. 然后, 以公式 (2) 中概率 p 来确定是否向服务器提交项目 t. $[d]\backslash\{t\}$ 表示从用户数据域 $[d]$ 中除去项目 t 后剩余项目的集合. 如果用户提交项目 t, 则从 $[d]\backslash\{t\}$ 中随机选择 k–1 个项目一并传输给数据收集方. 如果用户未提交项目 t, 则从 $[d]\backslash\{t\}$ 中随机选择 k 个项目一并传输给数据收集方.

3.1.3 MGA 攻击

MGA 通过解决以下优化问题来生成每个假用户的扰动值:

$$\arg\max E[F_{y_i}(t)],$$

其中, y_i 是用户 i 发送给服务器的扰动值, 在子集选择机制中, y_i 的表示形式为二进制向量, 其中包含且仅包含 k 个 1. $F_{y_i}(t)$ 是计数函数, 当 y_i 支持项目 t 时, 即 y_i 中项目 t 对应的位值为 1, $F_{y_i}(t)$ 输出 1, 否则输出 0. 通过最大化 $E[F_{y_i}(t)]$ 可以最大化攻击效用, 这将在第 3.2 节详细阐述.

对于子集选择机制, 考虑 $r \leqslant k$ 和 $r > k$ 两种情况.

(1) 当 $r \leqslant k$ 时, 为了使目标项目频率增加值之和最大, 每个假用户向数据收集方发送的 k 个值中包含所有的

目标项目, 即假用户向数据收集方发送的数据包含 r 个目标项目和 $k-r$ 个随机选择的非目标项目. 具体步骤如下.

第①步. 每个假用户初始化一个 d 比特的零向量 V.

第②步. 每个假用户将目标项目集 T 中的每个项目选中, 即集合 T 中每个项目在向量 V 中对应的比特设置为 1.

第③步. 从非目标项目中随机选取 $k-r$ 个项目, 将对应的比特设置为 1.

图 3 展示了 $r \leqslant k$ 时对于子集选择机制 MGA 攻击的数据设计步骤.

图 3 子集选择机制 MGA 攻击的伪造数据示意图

(2) 当 $r > k$ 时, 每个假用户从 r 个目标项目中随机选取 k 个项目, 将这 k 个项目发送给数据收集方, 即假用户从 r 个目标项目中随机选取 k 个项目, 将这 k 个项目对应的比特设置为 1.

最终, 假用户向服务器发送上述方法设计的二进制向量 V, 以最大化地提高目标项目的频率估计值.

3.2 子集选择机制攻击效用

3.2.1 攻击效用评估模型

在子集选择机制中, 根据公式 (4) 可以得到目标项目 t 的频率估计值 $\hat{f}_t$, 根据公式 (14) 可以得到目标项目 t 的频率增加值 $\Delta\hat{f}_t$. 进一步, 可以将 $\Delta\hat{f}_t$ 展开为:

$$\Delta\hat{f}_t = \hat{f}_{t,b} - \hat{f}_{t,a} = \frac{\tilde{f}_{t,b}-q}{p-q} - \frac{\tilde{f}_{t,a}-q}{p-q} = \frac{\frac{1}{n+m}\sum_{i=1}^{n+m} F_{y_i}(t) - q}{p-q} - \frac{\frac{1}{n}\sum_{i=1}^{n} F_{y_i}(t) - q}{p-q} = \frac{\sum_{i=n+1}^{n+m} F_{y_i}(t)}{(n+m)(p-q)} - \frac{m\sum_{i=1}^{n} F_{y_i}(t)}{n(n+m)(p-q)} \tag{16}$$

其中, y_i 是用户 i 发送给服务器的扰动数据. $F_{y_i}(t)$ 是计数函数, 当 y_i 支持项目 t 时, $F_{y_i}(t)$ 输出 1, 否则输出 0. 子集选择机制中, 项目的频率估计是无偏估计, 即 $E[\hat{f}_t] = f_t$, f_t 是项目 t 的真实频率, 因此, 有以下等式:

$$\sum_{i=1}^{n} E[F_{y_i}(t)] = n(f_t(p-q)+q) \tag{17}$$

目标项目 t 的频率增加值的期望为:

$$E[\Delta\hat{f}_t] = \frac{\sum_{i=n+1}^{n+m} E[F_{y_i}(t)]}{(n+m)(p-q)} - \frac{m\sum_{i=1}^{n} E[F_{y_i}(t)]}{n(n+m)(p-q)} \tag{18}$$

公式 (18) 中的第 2 项只取决于真实用户, 为了简单起见, 将第 2 项表示为常数 c_t. 根据公式 (17) 可以得到:

$$c_t = \frac{m(f_t(p-q)+q)}{(n+m)(p-q)} \tag{19}$$

因此, 攻击的整体效用如下:

$$G = \sum_{t\in T} E[\Delta\hat{f}_t] = \frac{\sum_{i=n+1}^{n+m}\sum_{t\in T} E[F_{y_i}(t)]}{(n+m)(p-q)} - c \tag{20}$$

其中, $c = \sum_{t\in T} c_t = \frac{m(f_T(p-q)+rq)}{(n+m)(p-q)}$, $f_T = \sum_{t\in T} f_t$, c 与假用户发送到数据收集方的扰动值无关. 攻击的整体效用 G 取决于计数函数的期望值 $E[F_{y_i}(t)]$.

3.2.2 攻击效用

结论 1. 子集选择机制 RPA 攻击整体效用为 $\frac{mrk}{(n+m)(p-q)d} - c$.

证明: RPA 从用户数据域{1, 2,…, d}中随机选择 k 个项目, 因此, 每个项目被选中发送给数据收集方的概率是 k/d. 可以计算出计数函数的期望值, 如下所示:

$$E[F_{y_i}(t)] = \Pr(F_{y_i}(t) = 1) = \frac{k}{d} \tag{21}$$

代入公式 (20) 得到攻击整体效用:

$$G = \sum\nolimits_{t \in T} E[\Delta \hat{f}_t] = \frac{mrk}{(n+m)(p-q)d} - c \tag{22}$$

结论 2. 子集选择机制 RIA 攻击整体效用为 $\frac{m[p+(r-1)q]}{(n+m)(p-q)} - c$.

证明: RIA 从目标项目集 T 中随机选择一个项目 t, 再根据概率 p 来确定是否向服务器提交项目 t. 因此, 项目 t 被提交给数据收集方的概率为 $\frac{1}{r} \cdot p + \left(1 - \frac{1}{r}\right) \cdot q$. 可以计算出计数函数的期望值, 如下所示:

$$E[F_{y_i}(t)] = \Pr(F_{y_i}(t) = 1) = \frac{1}{r} \cdot p + \left(1 - \frac{1}{r}\right) \cdot q \tag{23}$$

代入公式 (20) 得到攻击整体效用:

$$G = \sum\nolimits_{t \in T} E[\Delta \hat{f}_t] = \frac{m[p+(r-1)q]}{(n+m)(p-q)} - c \tag{24}$$

结论 3. 当 $r \leqslant k$ 时, 子集选择机制 MGA 攻击整体效用为 $\frac{mr}{(n+m)(p-q)} - c$; 当 $r > k$ 时, 攻击整体效用为 $\frac{mk}{(n+m)(p-q)} - c$.

证明: 子集选择机制的 MGA 存在 $r \leqslant k$ 和 $r > k$ 两种情况.

当 $r \leqslant k$ 时, 每个假用户向数据收集方发送的数据包含 r 个目标项目和 $k-r$ 个随机选择的非目标项目, 因此, 每个目标项目被选中的概率为 1, $E[F_{y_i}(t)] = 1$. 根据公式 (20) 得到攻击整体效用:

$$G = \sum\nolimits_{t \in T} E[\Delta \hat{f}_t] = \frac{mr}{(n+m)(p-q)} - c \tag{25}$$

当 $r > k$ 时, 每个假用户从 r 个目标项目中随机选取 k 个项目, 向数据收集方提供的 k 个项目全是目标项目, 因此每个目标项目被选中的概率为 k/r, $E[F_{y_i}(t)] = k/r$. 根据公式 (20) 得到攻击整体效用:

$$G = \sum\nolimits_{t \in T} E[\Delta \hat{f}_t] = \frac{mk}{(n+m)(p-q)} - c \tag{26}$$

根据公式 (2)、公式 (3) 可以得到 p, q 的具体值, 用 $\beta = \frac{m}{n+m}$ 表示假用户比例, 将 p, q, β 代入攻击整体效用 G 中, 可以得到攻击效用的具体表示, 如表 1 所示.

表 1 子集选择机制攻击效用分析

攻击类型	RPA	RIA	MGA
攻击效用	$\beta\left(\frac{r}{d} - f_T\right)$	$\beta(1-f_T)$	$\beta\left[r\left(1+\frac{(d-1)}{k(\mathrm{e}^{\varepsilon}-1)}\right)-f_T\right]$ $(r \leqslant k)$或$\beta\left[\frac{r\mathrm{e}^{\varepsilon}+(d-r-1)(k\mathrm{e}^{\varepsilon}+d-k)}{(d-k)(\mathrm{e}^{\varepsilon}-1)}-f_T\right]$ $(r>k)$

本文分析 3 种攻击效用的大小. 在表 1 中, 不难看出 $\frac{r}{d} < 1$, $\beta > 0$ 且 $f_T < 1$, 可得 $\beta\left(\frac{r}{d} - f_T\right) < \beta(1-f_T)$. 因此 RPA 攻击效用小于 RIA 攻击效用. 对于 MGA 攻击, 当 $r \leqslant k$ 时, $1 + \frac{(d-1)}{k(\mathrm{e}^{\varepsilon}-1)} > 1$, $r\left(1+\frac{(d-1)}{k(\mathrm{e}^{\varepsilon}-1)}\right) > 1$, $\beta\left[r\left(1+\frac{(d-1)}{k(\mathrm{e}^{\varepsilon}-1)}\right)-f_T\right] > \beta(1-f_T)$, 因此 $r \leqslant k$ 时, MGA 攻击效用大于 RIA 攻击效用. $r > k$ 时, 将分式 $\frac{r\mathrm{e}^{\varepsilon}+(d-r-1)(k\mathrm{e}^{\varepsilon}+d-k)}{(d-k)(\mathrm{e}^{\varepsilon}-1)}$ 的分子 $[r\mathrm{e}^{\varepsilon}+(d-r-1)(k\mathrm{e}^{\varepsilon}+d-k)]$ 减去分母 $(d-r)(\mathrm{e}^{\varepsilon}-1)$, 化简后可以得到 $(d-r)[(k-1)\mathrm{e}^{\varepsilon}+(d-k)]$. 已知 $d-r>0$, $k-1 \geqslant 0$, $d-k>0$, 可以判断出 $(d-r)[(k-1)\mathrm{e}^{\varepsilon}+(d-k)] > 0$, 那么 $\frac{r\mathrm{e}^{\varepsilon}+(d-r-1)(k\mathrm{e}^{\varepsilon}+d-k)}{(d-k)(\mathrm{e}^{\varepsilon}-1)} > 1$. 因为 $\beta > 0$ 且 $f_T < 1$, 所以 $\beta\left[\frac{r\mathrm{e}^{\varepsilon}+(d-r-1)(k\mathrm{e}^{\varepsilon}+d-k)}{(d-k)(\mathrm{e}^{\varepsilon}-1)}-f_T\right] > \beta(1-f_T)$. 因此 $r > k$ 时, MGA 攻击效用大于 RIA 攻击效用. 综上, 可以得到攻击效用的大小比较: $G_{\mathrm{MGA}} > G_{\mathrm{RIA}} > G_{\mathrm{RPA}}$.

从表 1 中可以看出, 3 种攻击的主要影响因素包括用户数据域大小 d 、隐私预算 ε 、假用户比例 β 、目标项目个数 r 等. 本文具体分析效果最好的 MGA 攻击受参数影响时攻击效用的变化: 当用户数据域大小 d 增大时, 对于 $r \leqslant k$ 的子集选择机制, 因为 d 在 MGA 攻击效用公式的分子中, 因此, 当 d 增大时, 攻击效用增大; 对于 $r > k$ 的子集选择机制, 在 MGA 攻击效用公式中, d 在分子的最高次幂为 2 次, 在分母的最高次幂为 1 次, 因此, 当 d 增大时, MGA 攻击效用增大. 当隐私预算 ε 增大时, 对于 $r \leqslant k$ 的子集选择机制 MGA 攻击效用减小. 当假用户比例 β 增大或者目标项目个数 r 增大时, MGA 攻击效用都会增大. 第 4.3.1 节实验验证了这些参数对攻击效用的影响.

3.3 环机制攻击方法

3.3.1 RPA 攻击

对环机制进行 RPA 攻击时, 每个假用户从用户数据域 $\{1, 2, \ldots, d\}$ 中随机选择一个项目 t, 通过用户特定的哈希函数将用户的项目 t 映射到 [0.0, 1.0) 范围内的一个数值 v , 从覆盖区域 C_v 中选择一个值发送给数据收集方. RPA 攻击随机选择用户数据域中的项目, 因此假用户发送的数据服从均匀分布, 攻击效果较差, 可以作为参照组与另外两种攻击效用进行对比.

3.3.2 RIA 攻击

RIA 攻击考虑了目标项目集 T, 从目标项目集 T 中为假用户选取项目, 扰动后发送给数据收集方, 这样设计的数据可以提高目标项目的频率估计值. 具体来说, 攻击者控制每个假用户从目标项目集 T 中随机选择一个项目 t , 按照环机制的扰动规则, 先通过用户 i 特定的哈希函数 h_i 将用户的项目 t 映射到 [0.0, 1.0) 范围内的一个数值 v_i , 再依照概率分布抽取一个值 $y_i \in [0.0, 1.0)$, 将 y_i 发送给数据收集方.

3.3.3 MGA 攻击

对环机制进行 MGA 攻击时, 仍然需要考虑以下优化问题来生成每个假用户的扰动值 y_i :

$$\arg\max E[F_{y_i}(t)],$$

其中, y_i 是用户 i 发送给服务器的扰动值, 在环机制中, y_i 的表示形式为元组 (s, z), s 是哈希函数的种子, z 是用户数据的扰动值. $F_{y_i}(t)$ 是计数函数, 当 y_i 支持项目 t 时, 即 y_i 在 t 的覆盖区域中, $F_{y_i}(t)$ 输出 1, 否则输出 0. 如果每个假用户向数据收集方发送的数据都在目标项目的覆盖区域内, 可以使攻击效用达到最大化.

假用户的扰动数据设计步骤如下.

第①步. 使用哈希函数将所有目标项目映射到 [0.0, 1.0) 范围内, 例如, 用户 i 使用哈希函数 h_i 将目标项目 t_1, t_2, t_3 映射为 t'_1, t'_2, t'_3 , 其中 $t'_1, t'_2, t'_3 \in [0.0, 1.0)$.

第②步. 在目标项目覆盖区域的交集中随机选取一个值, 将该值和哈希函数种子一同发送给服务器. 如图 4 所示, t'_1, t'_2, t'_3 的覆盖区域是 $C_{t'_1}, C_{t'_2}, C_{t'_3}$, 在 $C_{t'_1}, C_{t'_2}, C_{t'_3}$ 的交集区域选取一个值 z_i 作为用户 i 的扰动值, 发送给服务器.

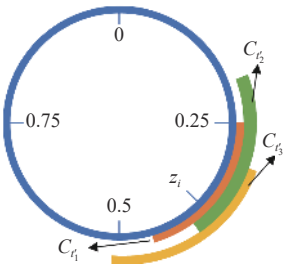


图 4 环机制 MGA 攻击的伪造数据示意图

为了实现进一步优化, 本文从哈希函数集 H 中选取一个哈希函数 h_m , h_m 可以使最多的目标项目哈希后的值覆盖区域存在交集. 具体来说, r 个目标项目 $t_1, t_2, \ldots, t_r$ 的覆盖区域表示为 $C_{t_1}, C_{t_2}, \ldots, C_{t_r}$, 在最优情况下, 每个假用户发送给服务器的扰动数据可以增加所有目标项目的频率估计值, 即假用户 j 发送的扰动值 $z_j \in C_{t_1} \cap C_{t_2} \cap \ldots \cap C_{t_r}$.

3.4 环机制攻击效用

3.4.1 攻击效用评估模型

在环机制中, 根据公式 (11) 可以得到项目 t 的估计频率 $\hat{f}_t$, 并且根据公式 (14) 可以得到目标项目 t 的频率增加值 $\Delta\hat{f}_t$, 因此, 进一步扩展 $\Delta\hat{f}_t$:

$$\Delta\hat{f}_t=\hat{f}_{t,b}-\hat{f}_{t,a}=\frac{\tilde{f}_{t,b}-P_f}{P_t-P_f}-\frac{\tilde{f}_{t,a}-P_f}{P_t-P_f}=\frac{\frac{1}{n+m}\sum_{i=1}^{n+m}F_{y_i}(t)-P_f}{P_t-P_f}-\frac{\frac{1}{n}\sum_{i=1}^{n}F_{y_i}(t)-P_f}{P_t-P_f}=\frac{\sum_{i=n+1}^{n+m}F_{y_i}(t)}{(n+m)(P_t-P_f)}-\frac{m\sum_{i=1}^{n}F_{y_i}(t)}{n(n+m)(P_t-P_f)} \tag{27}$$

其中, y_i 是用户 i 发送给服务器的扰动数据. $F_{y_i}(t)$ 是计数函数, 当 y_i 支持项目 t 时, $F_{y_i}(t)$ 输出 1, 否则输出 0. 环机制中, 项目的频率估计是无偏估计, 即 $E[\hat{f}_t]=f_t$, f_t 是项目 t 的真实频率, 因此, 可以得到公式 (28):

$$\sum_{i=1}^{n}E[F_{y_i}(t)]=n(f_t(P_t-P_f)+P_f) \tag{28}$$

目标项目 t 的频率增加值的期望为:

$$E[\Delta\hat{f}_t]=\frac{\sum_{i=n+1}^{n+m}E[F_{y_i}(t)]}{(n+m)(P_t-P_f)}-\frac{m\sum_{i=1}^{n}E[F_{y_i}(t)]}{n(n+m)(P_t-P_f)} \tag{29}$$

与子集选择机制效用评估模型相同, 公式 (27) 中的第 2 项只取决于真实用户, 将第 2 项表示为常数 c_t. 根据公式 (28) 可以得到:

$$c_t=\frac{m(f_t(P_t-P_f)+P_f)}{(n+m)(P_t-P_f)} \tag{30}$$

因此, 攻击的整体效用如公式 (31) 所示:

$$G=\sum_{t\in T}E[\Delta\hat{f}_t]=\frac{\sum_{i=n+1}^{n+m}\sum_{t\in T}E[F_{y_i}(t)]}{(n+m)(P_t-P_f)}-c \tag{31}$$

其中, $c=\sum_{t\in T}c_t=\frac{m(f_T(P_t-P_f)+rP_f)}{(n+m)(P_t-P_f)}$, $f_T=\sum_{t\in T}f_t$, c 与假用户发送到数据收集方的扰动值无关. 攻击的整体效用 G 取决于计数函数的期望值 $E[F_{y_i}(t)]$.

3.4.2 攻击效用

结论 4. 环机制 RPA 攻击整体效用为 $\frac{mr[P_t+(d-1)P_f]}{(n+m)(P_t-P_f)d}-c$.

证明: RPA 从用户数据域{1, 2,…, d}中随机选择一个值发送给数据收集方, 因此, 每个目标项目 t 被选中的概率为 $1/d$, y_i 支持项目 t 的概率为 $\frac{1}{d}\cdot P_t+\left(1-\frac{1}{d}\right)\cdot P_f$. 可以计算出计数函数的期望值, 如公式 (32) 所示:

$$E[F_{y_i}(t)]=\Pr(F_{y_i}(t)=1)=\frac{1}{d}\cdot P_t+\left(1-\frac{1}{d}\right)\cdot P_f \tag{32}$$

根据公式 (32), 可以得到攻击整体效用:

$$G=\sum_{t\in T}E[\Delta\hat{f}_t]=\frac{mr[P_t+(d-1)P_f]}{(n+m)(P_t-P_f)d}-c \tag{33}$$

结论 5. 环机制 RIA 攻击整体效用为 $\frac{m[P_t+(r-1)P_f]}{(n+m)(P_t-P_f)}-c$.

证明: 在进行 RIA 攻击时, 用户 i 从目标项目集 T 中随机选择一个项目 t, 进行扰动后向服务器发送扰动值 y_i, y_i 支持项目 t 的概率为 $\frac{1}{r}\cdot P_t+\left(1-\frac{1}{r}\right)\cdot P_f$. 因此, 可以计算出计数函数的期望值, 如下所示:

$$E[F_{y_i}(t)]=\Pr(F_{y_i}(t)=1)=\frac{1}{r}\cdot P_t+\left(1-\frac{1}{r}\right)\cdot P_f \tag{34}$$

代入公式 (31) 得到攻击整体效用:

$$G=\sum_{t\in T}E[\Delta\hat{f}_t]=\frac{m[P_t+(r-1)P_f]}{(n+m)(P_t-P_f)}-c \tag{35}$$

结论 6. 环机制 MGA 攻击整体效用为 $\dfrac{mr}{(n+m)(P_t-P_f)}-c$.

证明: 环机制的 MGA 攻击中, 本文选取一种最理想情况. 从哈希函数集 H 中选取一个哈希函数 h_m, 使用哈希函数 h_m 将所有目标项目进行映射, 所有目标项目的覆盖区域存在交集, 从交集中随机选取一个值, 发送给数据收集方. 因此, 向数据收集方发送的扰动值 y_i 支持项目 t 的概率为 1, 即 $E[F_{y_i}(t)]=1$, 代入公式 (31) 得到攻击整体效用:

$$G=\sum\nolimits_{t\in T}E[\Delta\hat{f}_t]=\frac{mr}{(n+m)(P_t-P_f)}-c \tag{36}$$

根据公式 (8)、公式 (9) 可以得到 P_t,P_f 的具体值, 用 $\beta=\dfrac{m}{n+m}$ 表示假用户比例, 将 P_t,P_f,β 代入攻击整体效用 G 中, 可以得到攻击效用的具体表示, 如表 2 所示.

表 2　环机制攻击效用分析

攻击类型	RPA	RIA	MGA
攻击效用	$\beta\left(\dfrac{r}{d}-f_T\right)$	$\beta(1-f_T)$	$\beta\left(\dfrac{2re^{\varepsilon}}{e^{\varepsilon}-1}-f_T\right)$

本文理论上分析了 3 种攻击效用的大小. 已知 $\dfrac{r}{d}<1$, $\beta>0$ 且 $f_T<1$, 可得 $\beta\left(\dfrac{r}{d}-f_T\right)<\beta(1-f_T)$. 因此 RPA 攻击效用小于 RIA 攻击效用. 对于环机制 MGA 攻击, 将分式 $\dfrac{2re^{\varepsilon}}{e^{\varepsilon}-1}$ 的分子分母相减, 可以得到 $2re^{\varepsilon}-(e^{\varepsilon}-1)=(2r-1)e^{\varepsilon}+1$. 因为 $r\geqslant 1$, 所以 $(2r-1)e^{\varepsilon}+1>0$, 则 $\dfrac{2re^{\varepsilon}}{e^{\varepsilon}-1}>1$. 已知 $\beta>0$ 且 $f_T<1$, 可以判断出 $\beta\left(\dfrac{2re^{\varepsilon}}{e^{\varepsilon}-1}-f_T\right)>\beta(1-f_T)$, 因此环机制 MGA 攻击效用大于 RIA 攻击效用. 综上, 可以得到攻击效用大小满足 $G_{\mathrm{MGA}}>G_{\mathrm{RIA}}>G_{\mathrm{RPA}}$.

本文具体分析 MGA 攻击受参数 d、ε、β、r 影响时攻击效用的变化: 环机制的 MGA 攻击效用中不含参数 d, 所以环机制的 MGA 攻击效用不受 d 大小变化的影响. 当假用户比例 β 增大或者目标项目个数 r 增大时, 环机制 MGA 攻击效用都会增大. 第 4.3.2 节实验验证了这些参数对攻击效用的影响.

4　实验结果与分析

4.1　实验设置与运行环境

实验采用合成数据集 SynData、IPUMS 数据集.

- SynData 满足均匀分布, 数据包含 10 000 个真实用户, 用户数据域是 [1, 100].
- IPUMS 是美国历年人口普查数据集[24], 实验选择 2010 年加利福尼亚州的数据, 按照 2.5% 的比例采样, 使用其中的区号属性, 数据中包含 1 048 575 个用户和 205 个区号.

数据集的具体信息见表 3.

表 3　实验数据集

数据集	SynData	IPUMS
真实用户人数	10 000	1 048 575
数据域大小 d	100	205
假用户比例 β	0.1	0.1
目标项目数量 r	10	20
隐私预算 ε	1.0	1.0
子集选择机制中用户提交数据个数 k	27	55

实验平台是 8 核 AMD R7-5800h、16 GB 内存、Windows 11 系统, 代码采用 Python 实现.

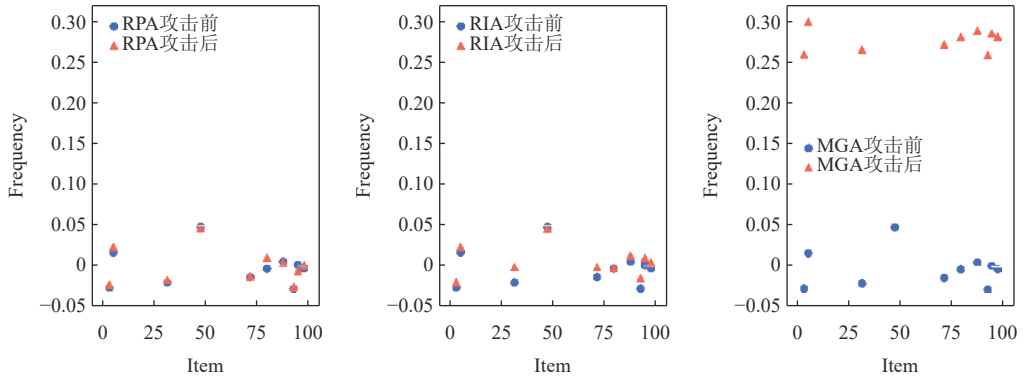
对于子集选择机制中用户提交数据个数 k 的设置, 文献 [6] 指出子集选择机制中最优子集的大小为 $\left\lfloor\dfrac{d}{1+e^{\varepsilon}}\right\rfloor$ 或 $\left\lceil\dfrac{d}{1+e^{\varepsilon}}\right\rceil$. 参考这一结论, 当数据域大小 $d=100$, 隐私预算 $\varepsilon=1.0$ 时, 设置 $k=\left\lfloor\dfrac{100}{1+e^{1.0}}\right\rfloor=27$; 当数据域大小 $d=$

205, 隐私预算 $\varepsilon = 1.0$ 时, 设置 $k = \left\lfloor \frac{205}{1+e^{1.0}} \right\rfloor = 55$. 根据实验数据集参数, 本文评估了 $r \leqslant k$ 时子集选择机制伪数据攻击的效用情况. 这里用 $g_{\text{s-RPA}}$、$g_{\text{s-RIA}}$ 和 $g_{\text{s-MGA}}$ 表示实验中子集选择机制 RPA 攻击、RIA 攻击和 MGA 攻击的实际效用; 用 $g_{\text{w-RPA}}$、$g_{\text{w-RIA}}$ 和 $g_{\text{w-MGA}}$ 表示实验中环机制 RPA 攻击、RIA 攻击和 MGA 攻击的实际效用.

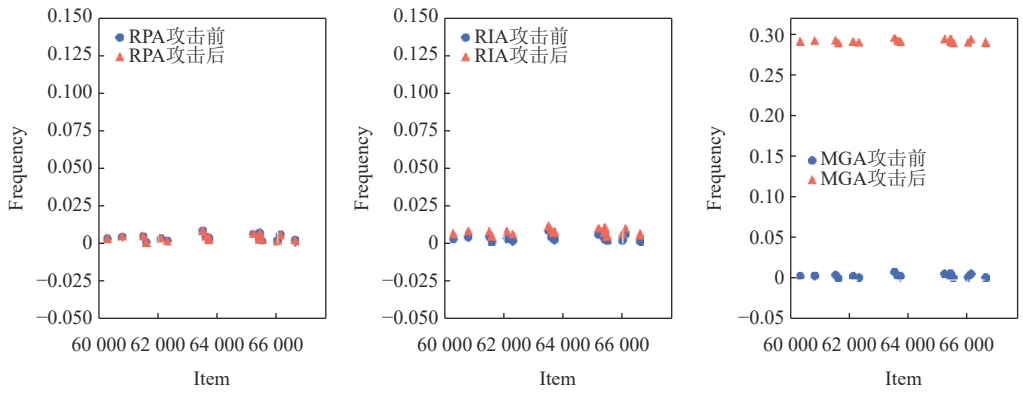
4.2 攻击实验结果

4.2.1 子集选择机制攻击结果

本文在 SynData 数据集和 IPUMS 数据集上进行了子集选择机制伪数据攻击实验, 图 5 显示了 3 种攻击前后目标项目的估计频率. 从图 5 中可以看出, RPA 攻击和 RIA 攻击前后目标项目频率估计结果相差不大, 因此, RPA 攻击和 RIA 攻击效果较差. MGA 攻击后目标项目的频率估计结果提高效果比较明显, 攻击效果较好. 为了更好地展示实验结果, 本文没有对频率估计得到的负值进行处理. 表 4 显示了子集选择机制攻击效用的具体结果, 和第 3.2 节理论分析一致, $g_{\text{s-MGA}} > g_{\text{s-RIA}} > g_{\text{s-RPA}}$.



(a) SynData上攻击子集选择机制



(b) IPUMS上攻击子集选择机制

图 5 子集选择机制攻击结果

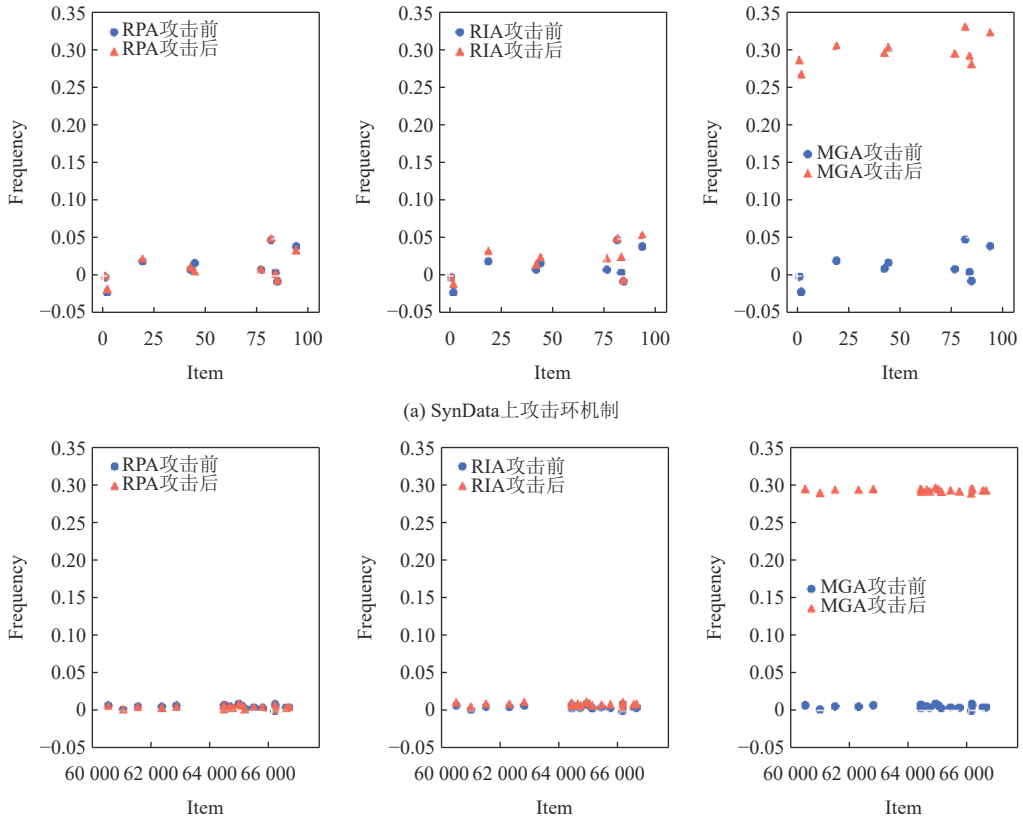
表 4 子集选择机制攻击效用

数据集	$g_{\text{s-RPA}}$	$g_{\text{s-RIA}}$	$g_{\text{s-MGA}}$
SynData	0.022	0.083	2.839
IPUMS	0.003	0.081	5.734

4.2.2 环机制攻击结果

图 6 显示了在 SynData 数据集和 IPUMS 数据集上进行环机制伪数据攻击的实验结果. 同样, 没有对频率估计得到的负值进行处理. 表 5 显示了环机制攻击效用的具体结果, 在 SynData 数据集上, $g_{\text{w-RPA}} = 0.024$, $g_{\text{w-RIA}} =$

0.123, $g_{\text{w-MGA}} = 2.875$; 在 IPUMS 数据集上, $g_{\text{w-RPA}} = -0.002$, $g_{\text{w-RIA}} = 0.084$, $g_{\text{w-MGA}} = 5.744$, 同样可得 RPA 攻击和 RIA 攻击效果较差, MGA 攻击效果较好, 即 $g_{\text{s-MGA}} > g_{\text{s-RIA}} > g_{\text{s-RPA}}$, 和理论分析相符.



(a) SynData上攻击环机制

(b) IPUMS上攻击环机制

图 6 环机制攻击结果

表 5 环机制攻击效用

数据集	$g_{\text{w-RPA}}$	$g_{\text{w-RIA}}$	$g_{\text{w-MGA}}$
SynData	0.024	0.123	2.875
IPUMS	−0.002	0.084	5.744

4.3 参数对攻击效果影响分析

4.3.1 参数对子集选择机制攻击效用的影响

本文使用 SynData 数据集测试子集选择机制中参数 d (项目域中项目数量), ε (隐私预算), β (假用户的比例), r (目标项目的数量) 对 3 种攻击整体效用的影响. 图 7 可以看出, 用户数据域增大时, MGA 攻击整体效用增大. 从第 3.2 节理论分析得到, MGA 的攻击效用中用户数据域 d 在分子位置, 因此, 当用户数据域增大时, MGA 攻击效用增大. 隐私预算 ε 增大, MGA 整体效用下降. 从第 3.2 节理论分析得到 MGA 的攻击效用中, 隐私预算 ε 在分母位置, 因此, 当隐私预算 ε 增大时, MGA 攻击效用减小. 假用户比例 β 增大时, MGA 攻击效用增大. 因为假用户比例 β 增大, 攻击者操纵更多的假用户向数据收集方发送假数据, 攻击效果更加显著, MGA 攻击效用增大. 目标项目数量 r 增大时, 假用户向服务器发送的数据中目标项目比例增大, 因此 MGA 攻击效用增大. RPA 攻击和 RIA 攻击效用太小, 随参数变化不明显.

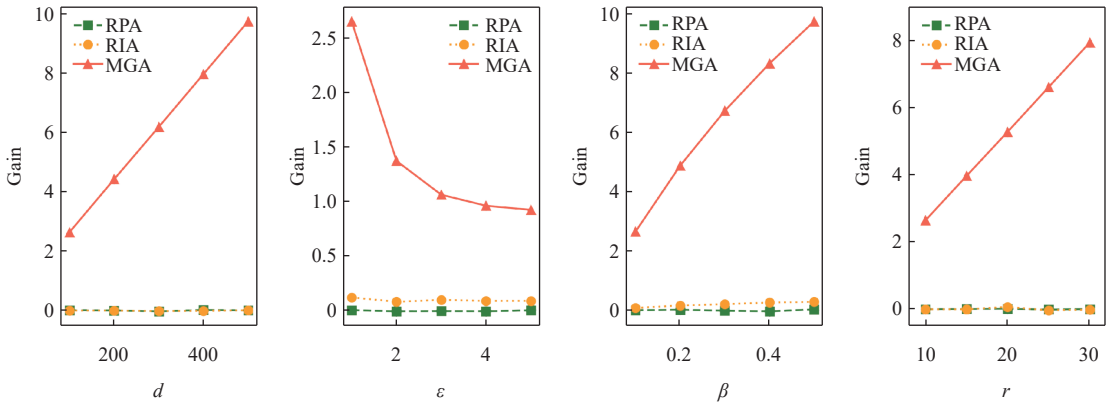


图 7　子集选择机制参数对攻击效用的影响

4.3.2　参数对环机制攻击效用的影响

对于环机制, 从图 8 可知, 用户数据域 d 增大时, 攻击整体效用不变. 因为环机制可以通过哈希函数把用户数据域映射到 [0.0, 1.0) 之间, 因此用户数据域 d 对 MGA 攻击整体效用无影响, 第 3.4 节理论分析得到的攻击效用不含参数 d, 也可证明该结论. 隐私预算 ε 增大, MGA 效用下降. 与子集选择机制 MGA 攻击相同, 假用户比例 β 和目标项目数量 r 增大时, 环机制 MGA 攻击效用增大. RPA 攻击和 RIA 攻击效用太小, 随参数变化不明显.

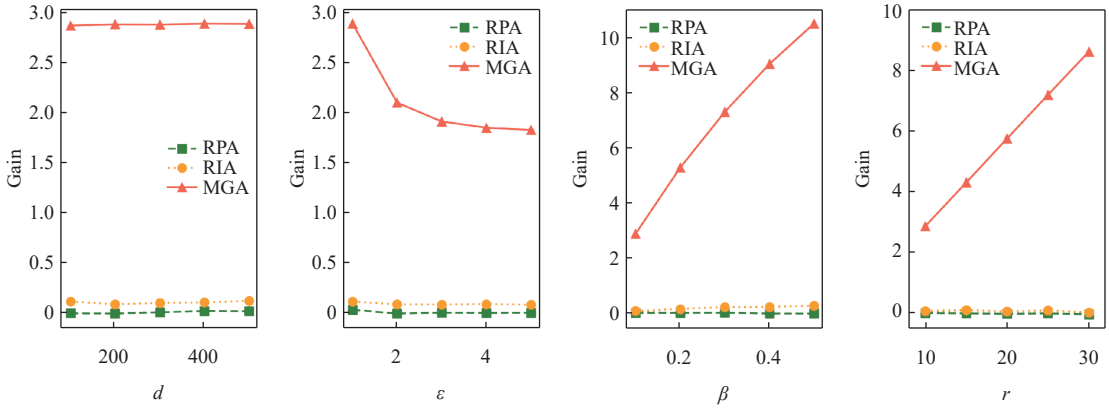


图 8　环机制参数对攻击效用的影响

5　攻击防御方法

5.1　后处理

从第 4.2 节的实验结果可看到, LDP 机制估计的项目频率存在负值. 可以对项目频率的估计值进行处理, 使每个项目的频率估计值都非负, 且项目频率的估计值之和为 1. 数据收集方首先按照 LDP 机制估计每个项目 v 的频率 $\hat{f}_v$, 然后数据收集方找到最小的项目估计频率 $\hat{f}_{\min}$, 对每个项目 v 的频率估计值进行校准, 即 $\bar{f}_v = \dfrac{\hat{f}_v - \hat{f}_{\min}}{\sum_v (\hat{f}_v - \hat{f}_{\min})}$, 其中 $\bar{f}_v$ 是校准的频率. 攻击整体效用是由攻击后和攻击前目标项目的校准频率之差来计算的. 这样也可以降低频率增加值, 达到一定的防御效果, 但该防御方法不能识别假用户, 也不能识别攻击者的目标项目.

实验评估了后处理防御方法的有效性, 采用 SynData 数据集, 参数设置与第 4.1 节相同, 实验重复 100 次, 结果取平均值, 经过后处理的攻击效用如表 6 所示. 表 6 中第 4 列显示了采用后处理前后攻击效用的差值, 可以得

出, MGA 攻击效用大幅度下降, 后处理方法降低目标项目频率估计增加值效果较好. 对于 RPA 攻击和 RIA 攻击, 采取后处理方法后攻击整体效用也比原始攻击效用小. 我们注意到, 子集选择机制 RPA 攻击经过后处理之后, 攻击效用变为负值. 由于未对攻击前的估计频率进行后处理, 仅对攻击后的估计频率进行后处理, 因此, 当攻击效用比较小的时候, 可能出现目标项目后处理之后的估计频率之和小于攻击前估计频率之和, 即后处理攻击效用为负值.

表 6　后处理攻击效用

攻击类型	原始的攻击效用	后处理的攻击效用	差值
子集选择机制RPA	0.0019	–0.0009	0.0028
子集选择机制RIA	0.0837	0.0195	0.0642
子集选择机制MGA	2.6458	0.3553	**2.2905**
环机制RPA	0.0031	0.0019	0.0012
环机制RIA	0.0803	0.0252	0.0551
环机制MGA	2.8682	0.4393	**2.4289**

5.2　限定阈值方法

限定阈值方法仅适用于防御子集选择机制伪数据攻击. 子集选择机制中, 用户向服务器发送长为 d 的二进制向量, 对所有用户发送的二进制向量, 服务器统计向量中每个位出现 1 的次数. 进一步, 本文设置阈值 τ. 如果某个位出现 1 的次数高于阈值 τ, 对该位进行标记. 被标记的位对应的项目视为攻击者选择的目标项目, 同时将含有所有标记项目的用户视为假用户, 排除假用户的扰动数据后再次计算目标项目的频率估计值, 更加接近真实的频率估计值. 该方法具有一定的局限性, 在用户数据服从均匀分布的情况下, 防御效果更好.

实验评估了限定阈值方法的有效性, 采用 SynData 数据集, 参数设置和第 4.1 节相同. 真实用户人数 $n = 10000$, 假用户人数 $m = 1000$, 每个用户从数据域的 100 个数据中选择 27 个数据进行提交. 实验首先进行采样, 随机选取用户总人数 20% 的扰动数据, 统计每个项目出现的次数. 根据估算, 每个项目平均出现 $(10000 + 1000) \times 20\% \times 27 \div 100 = 594$ 次, 因此阈值要略高于该值. 实验中, 间隔选取阈值来测定不同阈值下的防御效果. 在筛选出含有标记项目的假用户后, 排除假用户计算目标项目的频率估计值和攻击效用. 图 9 显示了设置不同阈值时, 子集选择机制 MGA 攻击效用的变化, 可以看出随着阈值增大, MGA 攻击效用减小, 当阈值大于 640 后, MGA 攻击效用可降为 0, 限定阈值防御方法效果较好.

进一步, 本文还对比了子集选择机制 MGA 攻击下无防御、后处理、限定阈值方法的防御效果, 图 10 显示了实验结果. 可以观察到, 随着隐私预算 ε 变大, 无防御时攻击效用减小, 后处理和限定阈值两种防御方法均能将攻击效用降低至较小水平, 防御效果较好. 并且, 限定阈值防御方法能够将攻击效用下降至接近 0, 防御效果优于后处理.

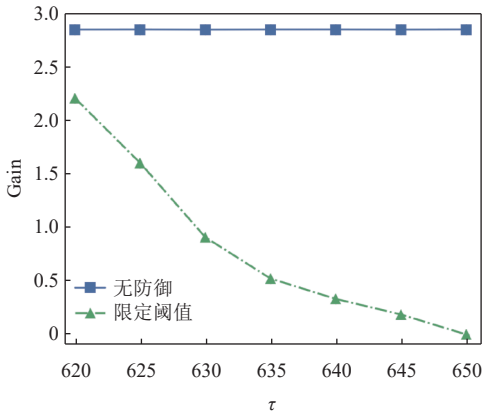


图 9　限定阈值防御结果

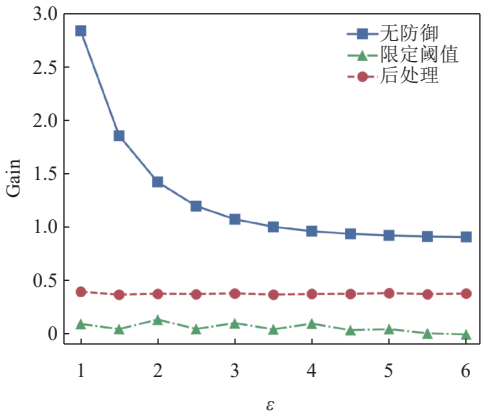


图 10　防御结果比较

6 总 结

针对子集选择机制和环机制两种效用最优的 LDP 协议, 本文设计了攻击效用最大的伪数据攻击方案, 并通过攻击方案展示了子集选择机制和环机制易于遭受伪数据攻击的负面影响. 攻击者可以向 LDP 机制中注入假用户, 向服务器发送伪造的数据, 达到显著提高目标项目频率估计值的目的. 本文通过理论分析和实验评估证实了所设计攻击方案的有效性. 最后, 本文提出了针对伪数据攻击的防御方法. 未来的工作主要包括深入分析伪数据攻击对其他各种 LDP 机制的影响, 以及设计安全高效的防御措施来应对伪数据攻击.

References:

[1] Dwork C. Differential privacy: A survey of results. In: Proc. of the 5th Int'l Conf. on Theory and Applications of Models of Computation. Xi'an: Springer, 2008. 1–19. [doi: 10.1007/978-3-540-79228-4_1]

[2] Evfimievski A, Gehrke J, Srikant R. Limiting privacy breaches in privacy preserving data mining. In: Proc. of the 22nd ACM SIGMOD-SIGACT-SIGART Symp. on Principles of Database Systems. San Diego: ACM, 2003. 211–222. [doi: 10.1145/773153.773174]

[3] Erlingsson Ú, Pihur V, Korolova A. RAPPOR: Randomized aggregatable privacy-preserving ordinal response. In: Proc. of the 2014 ACM SIGSAC Conf. on Computer and Communications Security. Scottsdale: ACM, 2014. 1054–1067. [doi: 10.1145/2660267.2660348]

[4] Cormode G, Jha S, Kulkarni T, Li NH, Srivastava D, Wang TH. Privacy at scale: Local differential privacy in practice. In: Proc. of the 2018 Int'l Conf. on Management of Data. Houston: ACM, 2018. 1655–1658. [doi: 10.1145/3183713.3197390]

[5] Luca M, Zervas G. Fake it till you make it: Reputation, competition, and Yelp review fraud. Management Science, 2016, 62(12): 3412–3427. [doi: 10.1287/mnsc.2015.2304]

[6] Ye M, Barg A. Optimal schemes for discrete distribution estimation under locally differential privacy. IEEE Trans. on Information Theory, 2018, 64(8): 5662–5676. [doi: 10.1109/TIT.2018.2809790]

[7] Wang SW, Huang LS, Nie YW, Zhang XY, Wang PZ, Xu HL, Yang W. Local differential private data aggregation for discrete distribution estimation. IEEE Trans. on Parallel and Distributed Systems, 2019, 30(9): 2046–2059. [doi: 10.1109/TPDS.2019.2899097]

[8] Wang SW, Qian YQ, Du JC, Yang W, Huang LS, Xu HL. Set-valued data publication with local privacy: Tight error bounds and efficient mechanisms. Proc. of the VLDB Endowment, 2020, 13(8): 1234–1247. [doi: 10.14778/3389133.3389140]

[9] Jagielski M, Oprea A, Biggio B, Liu C, Nita-Rotaru C, Li B. Manipulating machine learning: Poisoning attacks and countermeasures for regression learning. In: Proc. of the 2018 IEEE Symp. on Security and Privacy (SP). San Francisco: IEEE, 2018. 19–35. [doi: 10.1109/SP.2018.00057]

[10] Shanghai Observer. "The Most Evil AI Ever"? This chatbot has learned to curse, causing serious moral and ethical controversy. 2022. https://export.shobserver.com/baijiahao/html/498201.html

[11] Ambainis A, Jakobsson M, Lipmaa H. Cryptographic randomized response techniques. In: Proc. of the 7th Int'l Workshop on Theory and Practice in Public Key Cryptography. Singapore: Springer, 2004. 425–438. [doi: 10.1007/978-3-540-24632-9_31]

[12] Moran T, Naor M. Polling with physical envelopes: A rigorous analysis of a human-centric protocol. In: Proc. of the 25th Int'l Conf. on the Theory and Applications of Cryptographic Techniques. St. Petersburg: Springer, 2006. 88–108. [doi: 10.1007/11761679_7]

[13] Cheu A, Smith A, Ullman J. Manipulation attacks in local differential privacy. In: Proc. of the 2021 IEEE Symp. on Security and Privacy (SP). San Francisco: IEEE, 2021. 883–900. [doi: 10.1109/SP40001.2021.00001]

[14] Kato F, Cao Y, Yoshikawa M. Preventing manipulation attack in local differential privacy using verifiable randomization mechanism. In: Proc. of the 35th Annual IFIP WG 11.3 Conf. on Data and Applications Security and Privacy XXXV. Calgary: Springer, 2021. 43–60. [doi: 10.1007/978-3-030-81242-3_3]

[15] Cao XY, Jia JY, Gong NZ. Data poisoning attacks to local differential privacy protocols. In: Proc. of the 30th USENIX Security Symp. Berkeley: USENIX, 2021. 947–964.

[16] Kairouz P, Oh S, Viswanath P. Extremal mechanisms for local differential privacy. The Journal of Machine Learning Research, 2016, 17(1): 492–542. [doi: 10.5555/2946645.2946662]

[17] Wang TH, Blocki J, Li NH, Jha S. Locally differentially private protocols for frequency estimation. In: Proc. of the 26th USENIX Security Symp. Vancouver: USENIX, 2017. 729–745.

[18] Wu YJ, Cao XY, Jia JY, Gong NZ. Poisoning attacks to local differential privacy protocols for key-value data. In: Proc. of the 31st USENIX Security Symp. Boston: USENIX, 2022. 519–536.

[19] Ye QQ, Hu HB, Meng XF, Zheng HD. PrivKV: Key-value data collection with local differential privacy. In: Proc. of the 2019 IEEE Symp. on Security and Privacy (SP). San Francisco: IEEE, 2019. 317–331. [doi: 10.1109/SP.2019.00018]

[20] Gu XL, Li M, Cheng YQ, Xiong L, Cao Y. PCKV: Locally differentially private correlated key-value data collection with optimized utility. In: Proc. of the 29th USENIX Security Symp. Berkeley: USENIX, 2020. 967–984.

[21] Li XG, Li NH, Sun WH, Gong NZ, Li H. Fine-grained poisoning attack to local differential privacy protocols for mean and variance estimation. In: Proc. of the 32nd USENIX Security Symp. Anaheim: USENIX, 2023. 1739–1756.

[22] Duchi JC, Jordan MI, Wainwright MJ. Minimax optimal procedures for locally private estimation. Journal of the American Statistical Association, 2018, 113(521): 182–201. [doi: 10.1080/01621459.2017.1389735]

[23] Wang N, Xiao XK, Yang Y, Zhao J, Hui SC, Shin H, Shin J, Yu G. Collecting and analyzing multidimensional data with local differential privacy. In: Proc. of the 35th IEEE Int'l Conf. on Data Engineering (ICDE). Macao: IEEE, 2019. 638–649. [doi: 10.1109/ICDE.2019.00063]

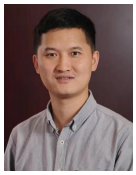
[24] IPUMS census database. 2022. http://kdd.ics.uci.edu/databases/ipums/ipums.html

附中文参考文献:

[10] 上观. "史上最邪恶 AI"? 这个聊天机器人学会骂人, 引严重道德伦理争议. 2022. https://export.shobserver.com/baijiahao/html/498201.html

王源源(2000－), 女, 博士生, 主要研究领域为本地差分隐私.

王威(1990－), 男, 博士, 研究员, CCF 专业会员, 主要研究领域为空天地一体化网络, 电磁频谱安全, 区块链.



朱友文(1986－), 男, 博士, 教授, 博士生导师, CCF 专业会员, 主要研究领域为数据安全, 隐私计算, 密码学.

王箭(1968－), 男, 博士, 教授, 博士生导师, CCF 高级会员, 主要研究领域为信息安全, 应用密码学, 隐私保护.

吴启晖(1970－), 男, 博士, 教授, 博士生导师, 主要研究领域为认知信息论, 电磁空间频谱智能管控, 天地一体化信息网络, 无人机集群智能通信.