计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2022,31(8):230−238 [doi: 10.15888/j.cnki.csa.008660]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

满足 LDP 的多维数据联合分布估计①

褚雪君[1,2], 龙士工[1,2], 刘　海[1,2]

[1](贵州大学 计算机科学与技术学院, 贵阳 550025)
[2](贵州大学 贵州省公共大数据重点实验室, 贵阳 550025)
通信作者: 龙士工, E-mail: 504498990@qq.com

摘　要: 多维数据的发布与分析可以产生巨大的价值, 但在数据收集阶段时常发生隐私泄露的问题. 传统的中心化差分隐私保护方法要求一个完全可信的第三方数据收集者来收集数据, 但在现实中很难找到一个完全可信的第三方数据收集者. 随着属性维度的增加, 数据收集者的求精处理工作 (联合分布的计算) 也成了一个亟待解决的问题. 针对上述问题提出一种适用于多值数据的本地化差分隐私保护算法 (RR-LDP), 引入一元编码和瞬时随机响应技术用来在数据收集阶段保护个人隐私, 降低了通信开销; 在满足 LDP 的情况下, 结合期望最大化 (EM) 算法和 LASSO 回归模型, 提出了高效的多维数据联合分布估计算法 (LREMH). 该算法用 LASSO 回归模型估计初始值, 用 EM 算法进行迭代计算. 理论分析和实验结果表明 LREMH 算法在精度和效率之间取得了平衡.

关键词: 多维数据; 本地化差分隐私; EM 算法; LASSO 回归; 联合分布估计; 隐私保护; 随机响应

引用格式: 褚雪君,龙士工,刘海.满足 LDP 的多维数据联合分布估计.计算机系统应用,2022,31(8):230–238. http://www.c-s-a.org.cn/1003-3254/8660.html

Joint Distribution Estimation for Multidimensional Data Based on LDP

CHU Xue-Jun[1,2], LONG Shi-Gong[1,2], LIU Hai[1,2]

[1](College of Computer Science and Technology, Guizhou University, Guiyang 550025, China)
[2](Guizhou Provincial Key Laboratory of Public Big Data, Guizhou University, Guiyang 550025, China)

Abstract: The release and analysis of multidimensional data can produce great value. However, privacy disclosure often occurs in the data collection phase. The traditional centralized differential privacy protection method requires a completely trusted third-party data collector, which is quite difficult to be found in practice. With the increase in attribute dimensions, the refinement of data collectors (the calculation of joint distribution) has also become an urgent problem to be solved. To address the above problems, this study proposes a localized differential privacy protection algorithm (RR-LDP) for multi-valued data. Unary coding and instantaneous random response technique are introduced to protect personal privacy in the data collection phase, which reduce communication overhead. With the combination of expectation maximization (EM) algorithm and LASSO regression model, the study puts forward an efficient joint distribution estimation algorithm (LREMH) for multidimensional data, which meets the requirement of LDP. The algorithm uses the LASSO regression model to estimate the initial value and employs the EM algorithm for iterative calculation. Theoretical analysis and experimental results show that the LREMH algorithm achieves a balance between accuracy and efficiency.

Key words: multidimensional data; localized differential privacy; expectation maximization (EM) algorithm; LASSO regression; joint distribution estimation; privacy protection; random response

① 基金项目: 国家自然科学基金 (62062020, 62002081)
收稿时间: 2021-11-24; 修改时间: 2021-12-20; 采用时间: 2022-01-05; csa 在线出版时间: 2022-05-30

1 引言

随着移动互联网与大数据的发展, 数据规模也以前所未有的速度不断增长, 数据属性之间的相互关系变得复杂多样, 多维数据已是一种常见的数据发布类型[1]. 在实际应用中, 大量的多维数据被存储在多个分布式组织中, 进行集成后, 这些多维数据将成为做出更好决策和提供高质量服务的宝贵资源.由于数据挖掘和分析技术的提升, 发布多维数据会带来很高的信息价值, 但多维数据中常包含许多隐私信息, 为了保护这些隐私信息在数据发布的过程中不被泄露, 通常会使用差分隐私保护技术. 传统的差分隐私技术将原始数据集中到一个中心服务器, 然后发布满足差分隐私的信息, 通常称其为中心化差分隐私保护 (CDP). 因此中心化差分隐私技术始终基于一个可信的第三方数据收集者并保证不会窃取或泄露用户的敏感信息的前提. 然而想要找到一个真正可信的第三方数据收集者是非常困难的. 鉴此, 在缺少可信的第三方数据收集者的情况下, 本地化差分隐私 (LDP)[2–4] 应运而生, 目前已在业界得到应用. 但多数的本地化差分隐私技术不适用于多维数据, 若直接将其应用于多维数据会造成通信开销较大, 可用性差等问题.

目前, 本地化差分隐私技术已经成为继中心化差分隐私技术之后一种强健的隐私保护模型. 首先, 用户对原始数据进行满足ε–本地化差分隐私的扰动, 然后将其传输给第三方数据收集者, 数据收集者收到扰动后的数据再进行一系列的查询和求精处理, 以得到有效的统计结果. 对本地化差分隐私的研究和应用, 主要考虑以下两个方面问题: (1) 如何设计满足ε–本地化差分隐私的扰动算法; (2) 数据收集者如何对收集到的数据集进行求精处理, 以提高统计结果的可用性. 本文中, 求精处理即通过基本推理和机器学习的方法来捕捉收集到数据集的联合概率分布[5–7] 的过程.

为解决数据收集阶段的隐私泄露, 本地化差分隐私保护通信开销较大以及数据收集者求精处理的问题, 本文提出了 RR-LDP 算法和 LREMH 算法, 主要工作如下:

(1) 提出了一个适用于多维数据的本地差分隐私保护算法 (RR-LDP). 该算法相比直接将 RAPPOR[8] 应用于多维数据上极大地降低了通信开销.

(2) 结合期望最大化 (EM) 算法和 LASSO 回归模型, 提出了一种高效的多维数据联合分布估计混合算法 (LREMH). 在真实数据集上进行性能评估, 实验结果表明 LREMH 算法在精度和效率之间取得了平衡.

2 相关工作

Erlingsson 等人[8] 提出 RAPPOR 应用随机响应技术和布隆过滤器来实现本地化差分隐私, 并应用在谷歌浏览器上. 苹果的差分隐私团队提出使用 one-hot 编码技术对敏感数据进行编码, 并部署 CMS 算法分析 Safari 中最流行的表情符号和媒体播放偏好. 文献 [9] 通过结合 LDP 与集中式数据模式, 提出具有高可用性的混合模型 BLENDER. 文献 [10] 针对移动设备收集隐私数据问题, 构建了 Harmony 系统, 该系统支持满足 LDP 的统计分析与机器学习功能. 随机响应技术及其变体在收集分布式用户统计数据的安全性方面具有优势, 已成为 LDP 研究的热点. 但目前多数的 LDP 机制并不适用于多维且多值的数据.

Giulia 等人[11] 提出基于 EM 的学习算法从噪声样本空间中估计联合概率分布. 然而它们的方案适用于二维数据, 当维数较高时, 数据的稀疏性会导致很大的效用损失, EM 算法的复杂度也会呈指数级上升. Ren 等人[12] 打破了文献 [11] EM 算法的局限, 将其拓展用于处理多维数据. Li 等人[13] 提出使用 Copula 函数来模拟多维数据的联合分布, 但 Copula 函数不能处理小域的属性. Cormode 等人[14] 将 Hadamard 变换应用于发布本地边缘表, 其优势是节省了通信开销, 但只适用于二进制数据. Zhang 等人[15] 借鉴 PriView[16] 的思想提出 CLMA 方法, 该方法可以在不计算满边际的情况下释放任意方向的边缘表, 并且可以处理非二进制属性. 然而, 除了隐私保护带来的噪声误差外还引入了采样误差.

3 基础知识

3.1 本地化差分隐私 (LDP)

定义 1. ε–本地化差分隐私. 对于一个有N条记录的数据集D, 给定随机算法Q满足ε–本地化差分隐私保护, $Range(Q)$为随机算法Q的取值范围, 那么算法Q在任意两条记录X_1和X_2 ($X_1, X_2 \in D$)上得到相同的输出结果X^*的概率满足:

$$P[Q(X_1) = X^*] \leqslant e^{\varepsilon} P[Q(X_2) = X^*] \tag{1}$$

其中, 概率$P[\cdot]$表示隐私泄露风险, ε表示隐私预算, 代表了隐私保护水平, 其值越小表示不可区分性越大, 隐

私保护等级越高.

性质 1. 序列组合性[17]. 给定数据集D和n个隐私算法$\{Q_1,\cdots,Q_n\}$且算法Q_i $(1\leqslant i\leqslant n)$满足$\varepsilon_i$–本地化差分隐私, 那么$\{Q_1,\cdots,Q_n\}$在$D$上的序列组合满足$\varepsilon$–本地化差分隐私, 其中, $\varepsilon=\sum_{i=1}^{n}\varepsilon_i$.

3.2 随机响应技术

随机响应技术 (randomized response) 是本地化差分隐私保护的主流扰动机制, 旨在调查过程中使用随机化装置, 使被调查者以一个预定的概率p进行诚实的回答, $(1-p)$的概率随意进行回答. 除被调查者以外的任何人均不知道被调查者的回答是否真实, 最后根据概率论的知识计算出敏感问题特征在人群中的真实分布情况的一种调查方法. 假设对n个用户的问答进行统计, 得到患病人数的统计值. 其中真实患病的比例记为π, 假定回答“Yes”的人数为n_1, 回答“No”的人数记为n_2. 根据诚实回答的概率θ可得:

$$\begin{cases} Pr[x_i=\text{“Yes”}]=\pi\cdot p+(1-\pi)\cdot(1-p) \\ Pr[x_i=\text{“No”}]=(1-\pi)\cdot p+\pi\cdot(1-p) \end{cases} \tag{2}$$

为了得到无偏估计, 可以采用极大似然的方法进行估计:

$$L=[\pi\cdot p+(1-\pi)\cdot(1-p)]^{n_1}\cdot[(1-\pi)\cdot p+\pi\cdot(1-p)]^{n-n_1} \tag{3}$$

然后可以求得π的估计$\tilde{\pi}$:

$$\tilde{\pi}=\frac{p-1}{2p-1}+\frac{n_1}{(2p-1)n} \tag{4}$$

则患病的人数$\tilde{n}$可估计为:

$$\tilde{n}=n\cdot\tilde{\pi}=\frac{p-1}{2p-1}n+\frac{n_1}{(2p-1)} \tag{5}$$

4 满足 LDP 的多维数据联合概率分布估计

4.1 设计思路

首先, 根据属性域的大小和每个取值在属性域中的位置将所有变量映射为位串, 得到的位串代表了唯一的原始记录. 然后, 通过随机响应技术进行第一次扰动, 得到的结果称作永久随机响应, 并将其保存在用户本地, 在第三方数据收集者请求数据时, 对永久随机响应的结果再做一次扰动, 得到的结果称作瞬时随机响应, 将瞬时随机响应的结果发送给第三方数据收集者. 最后, 数据收集者聚合收集到的得到随机噪声样本空间, 利用机器学习技术, 可以从中估计联合概率分布, 进行求精处理.

本文所使用的相关符号定义如表 1 所示.

表 1 符号定义表

符号	描述
N	数据记录(用户)的数量
X^i	第i个用户的数据记录
x_j^i	X^i的第j个元素
d	数据集中所有属性的个数
A_j	数据集中第j个属性
Ω_j	第j个属性的值域
$\lvert\Omega_j\rvert$	Ω_j值域的大小
ω_j	Ω_j的候选值
s_j^i	将x_j^i经过二进制化处理后的位串
$s_j^i[b]$	位串s_j^i中的第b位
$\hat{s}_j^i$	经过永久随机响应后的s_j^i
$\hat{s}_j^i[b]$	位串$\hat{s}_j^i$中的第b位
$T_j^i[b]$	位串T_j^i中的第b位

4.2 本地差分隐私保护

在本文的本地化差分隐私保护机制设计如算法 1 所示, 其中包含 3 个关键的步骤.

算法 1. RR-LDP 算法

输入: 用户数据记$\{x_j^i, j=1,2,\cdots,d\}$, 属性集$A_j$, 随机翻转概率$f$, 位串长度$|\Omega_j|$

输出: 随机翻转后的位串T^i

1. for $1\leqslant j\leqslant d$
2. 根据取值将x_j^i映射为一个长为$|\Omega_j|$的位串s_j^i
3. 根据f随机翻转s_j^i中的每一位, 得到扰动后的位串$\hat{s}_j^i$
4. end for
5. 对$\hat{s}_j^i$进行瞬时随机扰动, 得到T_j^i
6. 将翻转后的每个位串连接起来得到一个$\sum_{j=1}^{d}|\Omega_j|$位的向量T^i并返回

(1) 假设第i个用户有一个包含d个属性的原始数据记录$X^i=\{x_1^i,x_2^i,\cdots,x_d^i\}$. 用属性$A_j$ $(1\leqslant j\leqslant d)$的值域大小$|\Omega_j|$来确定预定义的位串长度, 每种属性的取值$\omega_j$对应位串中的一位, 进行数据转换时, 将每个属性取值所对应的那一位置 1, 其余位置 0, 即可该数据唯一的位串s_j^i.

(2) 位串s_j^i中的每一位$s_j^i[b]$根据概率$f\in(0,1)$按照式 (6) 进行随机翻转.

$$\hat{s}_j^i[b]=\begin{cases} s_j^i[b], \text{ 概率}=1-f \\ 0, \text{ 概率}=f/2 \\ 1, \text{ 概率}=f/2 \end{cases} \tag{6}$$

上述过程为永久随机响应. 永久随机响应可以保

证用户端相互通信时的隐私安全问题, 抵御纵向攻击. 由于每条记录中所有属性的取值是独立的, 故所得到的二进制位串可唯一代表一条记录.

(3) 初始化一个长度为$|\Omega_j|$的全 0 位串T_j^i, 对$\hat{s}_j^i[b]$根据式 (8) 进行瞬时随机扰动.

$$P(T_j^i[b]=1)=\begin{cases}p, \text{ if } \hat{s}_j^i[b]=0\\ q, \text{ if } \hat{s}_j^i[b]=1\end{cases} \tag{7}$$

其中, $p\in(0,1)$是当$\hat{s}_j^i[b]=0$时, $T_j^i[b]=1$的概率, $q\in(0,1)$是当$\hat{s}_j^i[b]=1$时, $T_j^i[b]=1$的概率.

用户i的每个属性的位串s_j^i经过永久随机响应后得到位串$\hat{s}_j^i$, 再经过瞬时随机响应后得到T_j^i, 将它们连接起来得到一个$\sum_{j=1}^{d}|\Omega_j|$位的向量T^i.

$$T^i=\left[T_1^i[1],\cdots,T_1^i[|\Omega_1|]|\cdots|T_d^i[1],\cdots,T_d^i[|\Omega_d|]\right]$$

根据文献 [8] 的计算可以得到当$T_j^i[b]=1$时, $s_j^i[b]=1$的概率为:

$$q^*=P(T_j^i[b]=1|s_j^i[b]=1)=0.5f(p+q)+(1-f)q \tag{8}$$

当$T_j^i[b]=1$时, $s_j^i[b]=0$ 的概率为:

$$p^*=P(T_j^i[b]=1|s_j^i[b]=0)=0.5f(p+q)+(1-f)p \tag{9}$$

由于服务器每次请求数据时都要做一次瞬时随机响应, 所以服务此每次请求相同的数据得到的结果都是不同的, 此时就可以保证服务器不能通过多次请求数据进行推断攻击.

在本文的 RR-LDP 方案采用一元编码的方式进行二进制转换, 相比于 RAPPOR[8] 所使用布隆过滤器进行二进制转化的方法, 本文映射后的位串长度更小且由于布隆过滤器使用哈希函数进行映射会出现哈希冲突造成映射后的位串冲突, 而 RR-LDP 则不会.

通信开销对比: 假设所有属性的值域都是公开的, 则 RR-LDP 的通信开销最小为$\sum_{j=1}^{d}|\Omega_j|$, 如果直接将 RAPPOR[8] 应用在多维数据上, 它会将其视为一维数据, 此时的通信开销为$\prod_{j=1}^{d}|m_j|$.其中, m_j为布隆过滤器预定义的位串长度.

4.3 隐私分析

定理 1. 在用户端进行的永久随机响应过程满足ε_1−本地化差分隐私, 其隐私保护等级为:

$$\varepsilon_1=2d\ln((2-f)/f) \tag{10}$$

证明: 令S表示用户初始的位串, S'表示经过本地随机翻转的位串. S_1和S_2分别代表两个不同用户的记录, 令它们的条件概率比值记作RR, $RR=P(S'=S^*|S=S_1)/P(S'=S^*|S=S_2)$, 它与隐私保护等级$\varepsilon_1$相关. 由式 (6) 可以得到位串的每一位翻转的概率为$f/2$, 不翻转的概率为$1-f/2$, 由文献 [8] 可得到 $RR_{\max}=((2-f)/f)^2$, 此时的隐私保护等级$\varepsilon_1=2\ln((2-f)/f)$, 其中$f$为随机翻转概率. 根据差分隐私序列组合性质[17], d维数据记录的本地翻转满足ε_1−本地化差分隐私, 其$\varepsilon_1=2d\ln((2-f)/f)$, 其中$d$为原始数据集$D$中属性的个数.

定理 2. 在用户端进行的瞬时随机响应过程满足ε_2−本地化差分隐私, 其隐私保护等级为:

$$\varepsilon_2=\log\left(\frac{q^*(1-p^*)}{p^*(1-q^*)}\right) \tag{11}$$

证明过程与定理 1 类似, 详见文献 [8]. 因为相同的转换是由所有用户独立完成的, 所以上述本地化差分隐私保护适用于所有分布式用户.

4.4 基于期望最大化算法 (EM) 的联合分布估计算法

EM 算法是在存在缺失或不完整数据的情况下获得最大似然估计的常用方法. 它特别适合于 RAPPOR[8] 这种只收集它们的噪声表示且真实值未知的应用中. 文献 [12] 中的 EM 算法主要分为以下 3 步:

第 1 步: 初始化, $P(\omega_1\omega_2\cdots\omega_k)=1/(\prod_{j=1}^{k}|\Omega_d|)$, 设置均匀分布分布作为初始的先验概率;

第 2 步: 更新, 根据式 (6) 得到原始位串的每一位$s_j^i[b]$翻转的概率为$f/2$, 不翻转的概率为$1-f/2$. 通过比较属性域与$\hat{s}_j^i$可得条件概率$P(\hat{s}_j^i|\omega_j)$; 由于本地化随机翻转是独立进行的, 其联合条件概率分布$P(\hat{s}_1^i\hat{s}_2^i\cdots\hat{s}_k^i|\omega_1\omega_2\cdots\omega_k)=\prod_{j=1}^{k}P(\hat{s}_j^i|\omega_j)$可以通过组合每个属性来计算, 得到一个特定的位串组合的所有条件分布, 通过贝叶斯定理计算它们对应的后验概率$P_t(\omega_C|\hat{s}_C^i)=P_t(\omega_C)\cdot P(\hat{s}_C^i|\omega_C)/\Sigma_{\omega_C}P_t(\omega_C)\cdot P(\hat{s}_C^i|\omega_C)$, 其中$P_t(\omega_C)=P_t(\omega_1\omega_2\cdots\omega_k)$为第 t 次迭代时的k 维联合概率;

第 3 步: 迭代, 得到后验概率后, 通过计算后验概率的平均值来更新先验概率. 在下一次迭代中利用更新后的先验概率计算后验概率. 用上述方法进行迭代直至收敛, $maxP_t(\omega_1\omega_2\cdots\omega_k)-maxP_{t-1}(\omega_1\omega_2\cdots\omega_k)\geqslant\delta$.

其中k为指定属性, 如$A_1,A_2,\cdots,A_k$, C为k的索引集, $C=\{1,2,\cdots,k\}$. $A_j=\omega_j$或$x_j=\omega_j$都用ω_j来表示.

EM 算法具有较高的精度, 但对初始值比较敏感, 当初始值选择合适时, 上述方法能达到较好的收敛效果. 然而文献 [12] 将联合分布初始化为均匀分布, 显然

不是最优的, 当属性个数k增大时, 由于$\Omega_1\times\Omega_2\times\cdots\times\Omega_k$中所有组合的样本空间爆发性增长, 算法的复杂度也会急剧上升, 阻碍良好的收敛. 同时多维数据呈现出的稀疏性也会带来较大的误差, 从而导致最终的估计达不到所需的效用.

4.5 基于 LASSO 回归的联合分布估计算法

LASSO 回归最早由 Tibshirani 于 1996 年提出[18], 文献 [8] 将它和最小二乘法用于收到噪声样本后的解码工作. 如第 4.1 节所述, 位串是原始记录的唯一代表. 随机翻转后, 本地用户会产生大量不同程度的噪声样本. 此时, 可以利用$\vec{y}=M\beta$来估计噪声样本空间的联合分布, 其中 M 是预测变量, $\vec{y}$是响应变量, β是回归系数向量, 这里的目的是估计 M 上的分布, 而不是原来的域. 响应变量$\vec{y}$可以根据已知的随机翻转概率f, 从位串中估计出来. 因此, 唯一的问题就是求出一个准确的回归系数β. 基于 LASSO 回归的联合分布估计主要包含以下几步:

第 1 步: 在服务器接收到所有经过随机翻转的位串后, 对其值为 1 的位进行计数, 记为$\hat{y}_i[b]=\sum_{i=1}^{N}\hat{s}_j^i[b]$;

第 2 步: 根据随机翻转概率f, 扰动前位串中每一位为 1 的真实计数$y_i[b]$可以被估算为$y_i[b]=(\hat{y}_i[b]-fN/2)/(1-f)$, 这些计数构成响应向量$\vec{y}$, 其长度为$\sum_{i=1}^{k}m_j$, 其中$m_j$为布隆过滤器预定义的位串长度;

第 3 步: 假设在域Ω_j上所有的哈希函数为$H_i(\Omega_j)=\{H_i(\omega)|\forall\omega\in\Omega_j\}$, 那么可以得到任意维度的候选集$M=[H_1(\Omega_1)\times H_2(\Omega_2)\times\cdots\times H_k(\Omega_k)]$;

第 4 步: 对响应向量$\vec{y}$和候选矩阵 M 拟合一个 LASSO 回归模型, 然后选择非零系数作为每个候选串对应的频率. 通过将系数向量$\vec{\beta}$按顺序重构为k维矩阵, 并除以N, 就可以得到k维联合分布.

4.6 LREMH 算法

基于 EM 的算法在样本足够的情况下, 可以表现出良好的收敛性, 但也会产生很高的复杂度. 其高复杂度是因为它迭代扫描用户的数据, 并构建一个先验分布表, 其大小为$N\cdot\prod|\Omega_j|$. 然而, 在多维情况下, Ω_j的组合是非常稀疏的, 且有很多零项. 同时, 由于 EM 对初始值的选择敏感, 均匀分配的初始值会导致收敛速度较慢. 然而基于 LASSO 回归的联合分布估计方法可以有效地解决由于多维数据的稀疏性导致的过拟合和效率慢的问题, 但与基于 EM 的算法相比, 精度略有下降.

为了在精度和效率之间取得平衡, 本文提出了 LREMH 算法, 该算法首先用基于 LASSO 回归的方法估计初始值, 这样得到的初始值会比均匀分布的初始值更加精确, 同时对基于 EM 算法的收敛性有积极的改进作用, 然后根据 LASSO 回归模型计算出冗余候选项, 并消除他们, 从而提高计算效率, 最后使用 EM 算法进行迭代计算得到一个较为准确的估计值.

算法 2. LREMH 算法

输入: A_j, $|\Omega_j|$, f, 索引集C, T^i
输出: $P_0(\omega_1\omega_2\cdots\omega_k)$

1. for each $j\in C$ do
2. for each b=1, 2, ⋯, $|\Omega_j|$ do
3. 计算$\hat{y}_j[b]=\sum_{i=1}^{N}T_j^i[b]$
4. 计算$y_j[b]=[\hat{y}_j[b]-N(p+0.5fq-0.5fp)]/(1-f)(q-p)$
5. end for
6. end for
7. 令 $\vec{y}=[y_1[1],\cdots,y_1[|\Omega_1|]|\cdots|y_k[1],\cdots,y_k[|\Omega_k|]]$
8. 令 $M=[(\Omega_1)\times(\Omega_2)\times\cdots(\Omega_k)]$
9. 计算$\vec{\beta}=Lasso(M,\vec{y})$ /*使用回归分析计算初始值*/
10. 返回$P_0(\omega_1\omega_2\cdots\omega_k)=\vec{\beta}/N$
11. 令$C'=\{x|x\in C,P_0(x)=0\}$
12. for each i=1,⋯,N do
13. for each j=1,⋯,k do
14. for each b=1, ⋯, $|\Omega_j|$ do
15. if $T_j^i[b]$=1
16. 计算$P(T_j^i|\omega_j)=\prod_{b=1}^{|\Omega_j|}q*^{\left|T_j^i[b]\cdot\omega_j[b]\right|}p*^{\left|T_j^i[b]-\omega_j[b]\right|}$
17. else
18. 计算$P(T_j^i|\omega_j)=\prod_{b=1}^{|\Omega_j|}(1-q*)^{\left|T_j^i[b]-\omega_j[b]\right|}(1-p*)^{\left|T_j^i[b]-\omega_j[b]\right|}$
19. end if
20. end for
21. end for
22. if $\omega_1\omega_2\cdots\omega_k\in C'$
23. $P(T_1^i\cdots T_k^i|\omega_1\cdots\omega_k)=0$
24. else
25. 计算$P(T_1^i\cdots T_k^i|\omega_1\cdots\omega_k)=\Pi_{j\in C}P(T_j^i|\omega_j)$
26. end if
27. end for
28. 初始化 t=0 /*迭代次数*/
29. repeat
30. for each i=1,⋯,N do
31. for each $\omega_c\in(\Omega_1)\times(\Omega_2)\times\cdots(\Omega_k)$ do
32. $P_t(\omega_c|T_C^i)=P_t(\omega_c)\cdot P(T_C^i|\omega_c)/\Sigma_{\omega_C}P_t(\omega_c)\cdot P(T_C^i|\omega_c)$
33. end for
34. end for
35. 令$P_{t+1}(\omega_c)=\sum_{i=1}^{N}P_t(\omega_c|T_C^i)/N$
36. 更新 $t=t+1$
37. 直到$maxP_t(\omega_1\omega_2\cdots\omega_k)-maxP_{t-1}(\omega_1\omega_2\cdots\omega_k)\geqslant\delta$
38. 返回$P(A_c)=P_t(\omega_c)$

本文提出的 LREMH 算法主要包含以下 3 个步骤:

第 1 步: 计算初始值, 根据永久随机响应翻转概率 f 和瞬时随机响应的翻转概率 p 和 q, 使用基于 LASSO 回归的联合分布方法计算初始值 (第 1–10 行).

第 2 步: 消除冗余项, 利用基于 LASSO 回归的联合分布估计方法得到联合分布为 0 的属性并消除他们 (第 11, 22–23 行).

第 3 步: 更新迭代, 根据永久随机响应翻转概率 f 和瞬时随机响应的翻转概率 p 和 q, 使用基于 EM 的联合分布估计算法通过组合每个属性来计算得到一个特定的位串组合的所有条件分布, 通过贝叶斯定理计算它们对应的后验概率. 得到后验概率后, 通过计算后验概率的平均值来更新先验概率. 在下一次迭代中利用更新后的先验概率计算后验概率. 使用上述方法进行迭代直至收敛 (第 12–37 行).

上述 LREMH 算法具有两个优势:

(1) 回归分析能够非常有效地选择稀疏的候选项. 因此, EM 算法可以只计算这些稀疏候选项上的条件概率, 而不是所有候选项上的条件概率, 从而降低了时间和空间复杂度.

(2) EM 算法对初值比较敏感, 尤其是在候选空间稀疏的情况下. 回归分析可以对联合分布产生较好的初始估计. 相对于均匀赋值的初值, 使用回归分析生成的初值可以进一步加快 EM 算法的收敛速度.

4.7 满足本地化差分隐私证明

证明. 在 LREMH 算法中, 所有的输入数据的都是经过 RR-LDP 算法处理后的数据, 且 LREMH 算法的整个流程中没有任何操作引入其他隐私保护和随机扰动, RR-LDP 算法的永久随机响应和瞬时随机响应根据定理 1 和定理 2 证得分别满足 ε_1–本地化差分隐私和 ε_2–本地化差分隐私, 根据本地化差分隐私的性质 1 (序列组合性) 可证得, LREMH 算法满足 ε–本地化差分隐私, 其中 $\varepsilon=\varepsilon_1+\varepsilon_2$.

5 实验与分析

5.1 实验环境

实验中使用了两个真实数据集, NLTCS 和 Adult. NLTCS 数据集来自美国护理调查中心, 包含 21 574 名残疾人不同时间段的活动. 成人数据集来自 1994 年美国人口普查, 包含 45 222 个居民的个人信息, 如性别、工资和教育水平. 在预处理中对一些连续域进行了离散化处理并删除了一些缺省值.

实验中所使用的软硬件参数如下:

(1) 操作系统: Windows 10;

(2) 硬件参数: IntelCore i5, 2.0 GHz CPU, 4 GB;

(3) 编译环境及工具: Python 2.7, PyCharm.

本文分别从数据集 NLTCS 和数据集 Adult 中采样了 20% 的数据和 10% 的数据. 算法的效率是通过计算估计时间和估计的精度来衡量的. 每组实验运行 10 次, 并报告平均运行时间. 为了测量精度, 本文使用了两个数据集上的平均变异距离 (AVD) 来量化估计的联合分布 $P(\omega)$ 和原始联合分布 $Q(\omega)$ 之间的接近程度.

$$Dist_{AVD}(P,Q)=0.5\Sigma_{\omega\in\Omega}|P(\omega)-Q(\omega)| \quad (12)$$

为了快速收敛, 将收敛间隙设置为 0.001, 在瞬时响应中通常取 q=0.75, p=0.5.

5.2 安全性分析

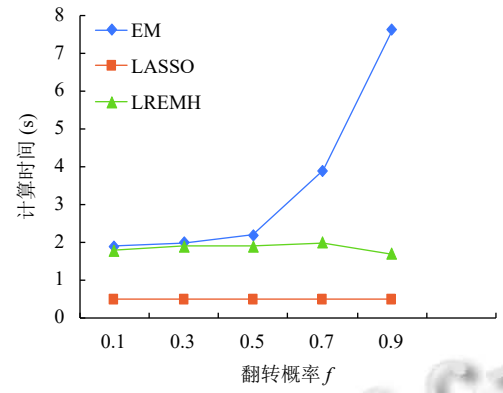
根据第 4.7 节的结论, LREMH 算法是满足 ε–本地化差分隐私, 又根据本地化差分隐私的两个性质得到 $\varepsilon=\varepsilon_1+\varepsilon_2=2d\ln((2-f)/f)+\log[q^*(1-p^*)/p^*(1-q^*)]$, 由于本次实验的 p 和 q 是定值, 故 ε 的大小只与 d (数据记录的数量) 和 f (永久随机响应的翻转概率) 有关, 而两个数据集的数据记录数 d 也是确定的, 所以本次实验的安全性只与 f 相关. 从本地化差分隐私的定义可以看出 ε 越小, e^{ε} 就越小, 数据记录之间的差距就越小, 就越难分辨, 安全性就越强, 反之则 ε 的值越大, 安全性就越弱, 本次实验的 f 取 (0, 1), 根据式 (10) 可以看出随着 f 的增大 ε 的值会减小, 安全性会增强.

5.3 估计效率对比

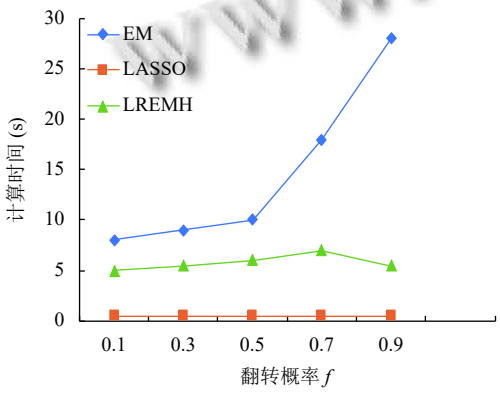
(1) NLTCS 数据集: 如图 1, 对于任一维度 k, LASSO 回归始终比 EM 算法和 LREMH 算法快, 尤其是当 k 较大时. 由于 LASSO 回归的时间复杂度主要受用户数量的影响, 所以当 k 增大时, LASSO 回归的计算时间增长缓慢, 而 EM 算法的计算时间增长较快是因为 EM 算法必须反复扫描每个用户的位串, 同时, 固定的收敛精度会有更多的迭代从而导致 EM 算法的时间消耗随着 f 的增加而增加. 相比之下, LASSO 回归可以更有效地估计联合分布. 因为 LASSO 回归的初始估计可以大大减少候选属性空间和所需的迭代次数, 所以 LREMH 算法的复杂度要比 EM 算法小.

(2) Adult 数据集: 如图 2, EM 算法在低维 k=2 的情

况下以可接受的复杂度运行. 当k=5 时, EM 算法的时间复杂度急剧增加了几倍. 当k进一步增加时, 在 120 s 内没有返回任何结果. 然而, LASSO 回归只需要几秒的时间.



(a) NLTCS, k=3



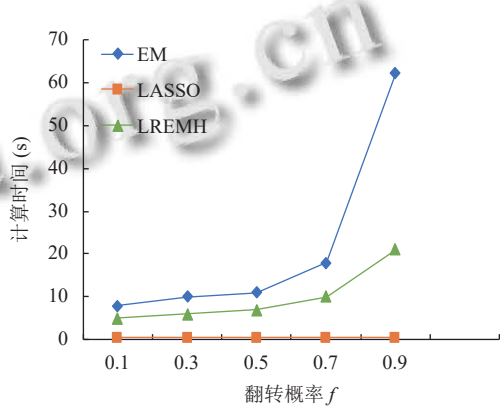
(b) NLTCS, k=6

图 1 NLTCS 数据集估计效率对比

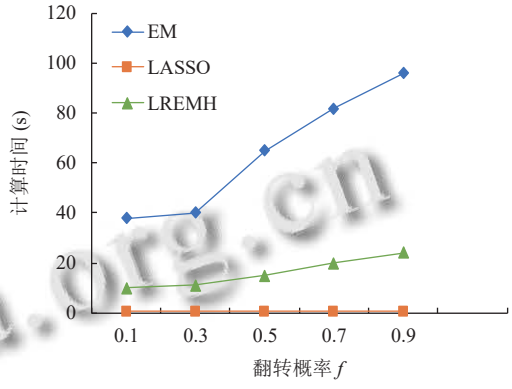
5.4 估计精度对比

(1) NLTCS 数据集: 如图 3, 当f很小时, EM 算法的 AVD 误差很小, 但当f增大时, 它会急剧增大, 高达 0.28. 相比之下, 即使f= 0.9, LASSO 回归的 AVD 误差也保持在 0.1 左右. 当f较大时, LASSO 回归的 AVD 误差与 EM 算法相当, 甚至更好. 这是因为 LASSO 回归在从M和$\vec{y}$估计系数时对f不敏感. 由于 EM 算法扫描每条记录的位串, 所以它对f很敏感, 并且容易得到某些局部最优值. 相比之下, LREMH 算法在 LASSO 回归和 EM 算法之间实现了更好的权衡. 例如, 当f值较小时, 它的 AVD 误差小于 LASSO 回归; 当f值较大时, 它的 AVD 误差优于 EM 算法.

(2) Adult 数据集: 如图 4, 当k= 2 时, LASSO 回归的 AVD 误差几乎不随f变化, 因为回归分析对f不敏感. 而 EM 算法的 AVD 误差而随f逐渐增大. 当f很大时, LASSO 回归的趋势非常接近 EM 算法. 因为 LREMH 算法比 EM 算法运行的快得多且估计精度于 EM 算法相差不大, 所以它实现了精度和效率之间的平衡. 另外, 当k= 5 时, 估计误差也增大. 而 LREMH 算法可以在 LASSO 回归和 EM 算法之间进一步平衡, 因为当k更大时, 候选集将会更稀疏, 而 LREMH 算法可以有效地减少候选集的冗余和迭代次数.



(a) Adult, k=2

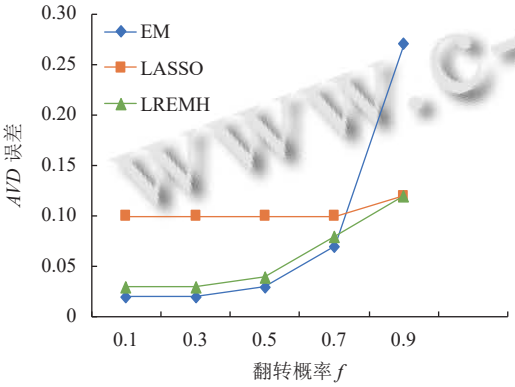


(b) Adult, k=5

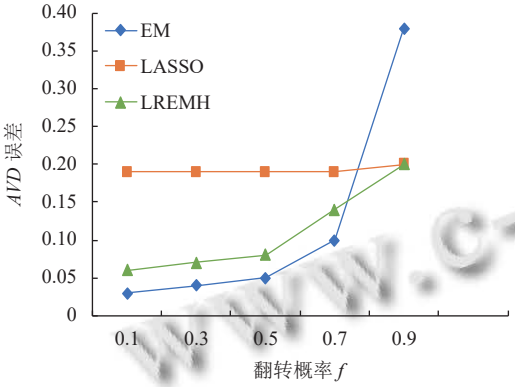
图 2 Adult 数据集估计效率对比

根据图 1, 图 2 可以看出 LREMH 算法的估计效率随着属性维度和f的增加而缓慢下降, 但始终处于 EM 算法和 LASSO 回归算法两者之间. 当f>0.7 时 EM 算法的效率急剧下降, 这是因为 EM 算法对f很敏感, 并且容易得到某些局部最优值. 由于 LREMH 算法使用 LASSO 回归快速的估计初始值, LREMH 算法对f的敏感度要低于 EM 算法, 同时使用 LASSO 回归估计的初始值要比直接使用均匀分布的初始值更加精确, 很好的解决的 EM 算法对初始值敏感的问题, 而且有效地减少了迭代的次数, 这使得 LREMH 算法的估计效率

一直比 EM 算法高. 根据图中可以看出, LREMH 算法的估计精度随着属性维度和f的增加而下降, 但也在大部分情况下处于 EM 算法和 LASSO 回归算法两者之间, 当属性的维度增大时, 需要计算的候选集会变得更加稀疏, 所以 EM 算法的误差会随着维度和f的增加而增加, 由于 LASSO 回归只进行一次回归分析的估计, 并没有像 EM 算法一样进行迭代, 所以其估计的精度在大多数情况下不如 EM 算法, 但 LREMH 算法在估计初始值的同时会消除冗余候选项, 解决的初值估计问题, 减少了迭代次数, 降低了得到局部最优的概率, 所以 LREMH 算法在效率和精度之间取得了均衡.



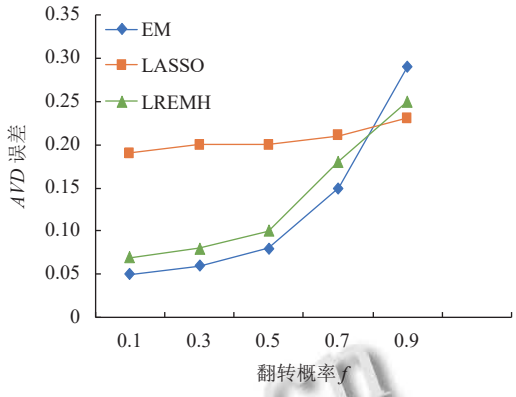
(a) NLTCS, k=3



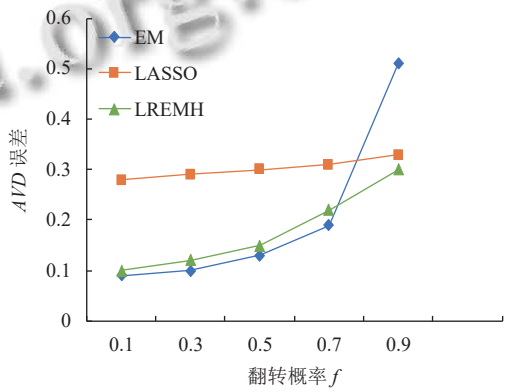
(b) NLTCS, k=6

图 3 NLTCS 数据集估计精度对比

并且由于f的值直接影响了隐私保护等级, 当f增加时, 隐私保护的等级就越高, 也就导致了随着f的增加, 3 个算法估计的效率和精度都随之下降, 从图 3 和图 4 中可以直观看出当f<0.7 时, LREMH 算法的估计效率和精度在 LASSO 回归算法和 EM 算法中取得了良好的均衡.



(a) Adult, k=2



(b) Adult, k=5

图 4 Adult 数据集估计精度对比

6 总结与展望

多维数据的联合概率分布估计对于数据发布具有重要作用, 本文提出的 LREMH 算法在满足本地化差分隐私的情况下, 结合期望最大化算法和回归分析方法消除冗余候选项, 用回归分析估计初始联合分布, 然后用期望最大化算法进行迭代计算, 直至收敛. 通过实验验证 LREMH 算法在精度和效率之间取得了平衡. 下一步工作将会围绕如何学习到与原始数据最为拟合的概率图模型, 如何进一步提高发布数据的可用性等方面的问题进行研究.

参考文献

1 马苏杭, 龙士工, 刘海, 等. 面向高维数据发布的个性化差分隐私算法. 计算机系统应用, 2021, 30(4): 131–138. [doi: 10.15888/j.cnki.csa.007870]

2 Chen R, Li HR, Qin AK, *et al*. Private spatial data aggregation in the local setting. 2016 IEEE 32nd International Conference on Data Engineering (ICDE).

Helsinki: IEEE, 2016. 289–300.
3 Duchi JC, Jordan MI, Wainwright MJ. Local privacy and statistical minimax rates. 2013 51st Annual Allerton Conference on Communication, Control, and Computing (Allerton). Monticello: IEEE, 2013. 1592–1592.
4 Kairouz P, Oh S, Viswanath P. Extremal mechanisms for local differential privacy. Proceedings of the 27th International Conference on Neural Information Processing Systems. Montreal: ACM, 2014. 2879–2887.
5 Ye M, Barg A. Optimal schemes for discrete distribution estimation under locally differential privacy. IEEE Transactions on Information Theory, 2018, 64(8): 5662–5676. [doi: 10.1109/TIT.2018.2809790]
6 Nie YW, Wang SW, Yang W, *et al*. Classification learning from private data in heterogeneous settings. 23rd International Conference on Database Systems for Advanced Applications. Gold Coast: Springer, 2018. 577–585.
7 Yilmaz E, Al-Rubaie M, Chang JM. Locally differentially private naive Bayes classification. arXiv: 1905.01039, 2019.
8 Erlingsson Ú, Pihur V, Korolova A. Rappor: Randomized aggregatable privacy-preserving ordinal response. Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security. Scottsdale: ACM, 2014. 1054–1067.
9 Avent B, Korolova A, Zeber D, *et al*. Blender: Enabling local search with a hybrid differential privacy model. Proceedings of the 26th USENIX Security Symposium. Vancouver: USENIX, 2017. 747–764.
10 Nguyên TT, Xiao XK, Yang Y, *et al*. Collecting and analyzing data from smart device users with local differential privacy. arXiv: 1606.05053, 2016.
11 Giulia F, Vasyl P, Úlfar E. Building a rappor with the unknown: Privacy-preserving learning of associations and data dictionaries. Proceedings on Privacy Enhancing Technologies, 2016, 2016(3): 41–61. [doi: 10.1515/popets-2016-0015]
12 Ren XB, Yu CM, Yu WR, *et al*. LoPub: High-dimensional crowdsourced data publication with local differential privacy. IEEE Transactions on Information Forensics and Security, 2018, 13(9): 2151–2166. [doi: 10.1109/TIFS.2018.2812146]
13 Li HR, Xiong L, Jiang XQ. Differentially private synthesization of multi-dimensional data using copula functions. Advances in Database Technology: Proceedings. International Conference on Extending Database Technology, 2014, 2014: 475–486. [doi: 10.5441/002/edbt.2014.43]
14 Cormode G, Kulkarni T, Srivastava D. Marginal release under local differential privacy. Proceedings of the 2018 International Conference on Management of Data. Houston: ACM, 2018. 131–146.
15 Zhang ZK, Wang TH, Li NH, *et al*. CALM: Consistent adaptive local marginal for marginal release under local differential privacy. Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. Toronto: ACM, 2018. 212–229.
16 Qardaji W, Yang WN, Li NH. Priview: Practical differentially private release of marginal contingency tables. Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data. Snowbird: Association for Computing Machinery, 2014. 1435–1446. [doi: 10.1145/2588555.2588575]
17 McSherry F. Privacy integrated queries: An extensible platform for privacy-preserving data analysis. Communications of the ACM, 2010, 53(9): 89–97. [doi: 10.1145/1810891.1810916]
18 Tibshirani R. Regression shrinkage and selection via the LASSO. Journal of the Royal Statistical Society: Series B (Methodological), 1996, 58(1): 267–288.

(校对责编: 孙君艳)