第 52 卷　第 6 期　　厦门大学学报(自然科学版)　　Vol. 52　No. 6
2013 年 11 月　　Journal of Xiamen University (Natural Science)　　Nov. 2013
doi:10.6043/j.issn.0438-0479.2013.06.007

基于桶划分的差分隐私直方图发布贪心算法

孙　岚[1*],吴英杰[1],张玺霖[1],谢　怡[2]
(1.福州大学数学与计算机科学学院,福建 福州 350108;2.厦门大学信息科学与技术学院,福建 厦门 361005)

摘要:现有的差分隐私直方图发布技术未能高效处理存在大量低频计数值数据集发布中的隐私保护问题.基于桶划分的思想,提出一种高效的、面向存在大量低频计数值数据集的差分隐私直方图发布贪心算法.算法采用基于邻近桶合并的贪心策略,并利用红黑树对合并过程进行优化.实验对本文算法发布数据的可用性及算法效率与同类算法进行比较分析.实验结果表明,该算法是有效可行的.

关键词:差分隐私;直方图发布;桶划分;贪心算法;红黑树
中图分类号:TP 311　　**文献标志码**:A　　**文章编号**:0438-0479(2013)06-0770-06

在现实生活中,由于数据统计和科学研究的需要,许多研究机构或组织都会对外发布数据,这些待发布的数据中通常都会包含一定的个人敏感信息,因此直接发布数据会导致个人隐私泄露.为了有效地保护个人隐私,近年来,出现了许多隐私保护模型,例如 k-匿名[1]、l-多样性[2]、t-邻近性[3]等等,这些模型大多以匿名为基础,且需要特殊的攻击假设和一定的背景知识,虽然可以在某种程度上保护数据的隐私,但仍具有一定的局限性.为了更好地防御具有背景知识的攻击,Dwork 等[4-7]提出了差分隐私模型,该模型通过向查询结果或分析结果中添加噪声来达到更强的隐私保护目的.直方图发布是差分隐私数据发布的一种常见形式.近年来,一些研究人员在差分隐私直方图发布领域取得了若干有价值的研究成果[8-14].直方图发布的基本思想是根据一个或多个属性将数据表划分成若干不相交的集合,每个集合用其划分意义下的计数值进行标识.显然,在现实生活中,有一些数据集在进行直方图标识时,会出现某些集合中的计数值相对于其他集合中的计数值非常小;而某些集合的计数值又相对非常大的情况.例如在某省个人年收入数据集中,收入高于 100 万的人群就是一个相对非常小的计数值;季节性消费品的销量数据集中,淡季的销量相对于旺季的销量就是一个非常小的计数值.现有针对此类数据集的差分隐私直方图发布算法虽然可以保证数据发布的精度[14],但是执行效率不高.因此,就此类数据集,在保证发布数据质量和可用性的前提下,设计出高效的差分隐私直方图发布算法具有重要的意义.

1　相关工作

Dwork 等[5]最早提出利用差分隐私模型,通过往每个划分集合添加噪声而发布直方图,并利用发布的直方图提供区间计数查询.然而,当查询区间较大时,由于噪声的累加会导致查询结果的噪声过大,从而降低发布数据的可用性.文献[8]提出了基于最优线性无偏估计的差分隐私直方图发布方法,利用一维数据之间的相容性对区间统计查询进行优化,实现了高可用性的数据发布.文献[9]通过小波变换技术,提出了一种提高区间计数查询精度的差分隐私直方图发布方法 Privelet,采用小波变换构造基于区间计数的小波树,通过往小波树添加噪声以满足差分隐私,从而发布可用性较高的数据.文献[10-11]在文献[9]的方法基础上设计出面向任意线性查询的优化方法,解决了非连续区间查询的优化问题.上述文献的研究大多是基于减小绝对误差的思想对查询精度进行优化.文献[12]提出了基于减小相对误差优化查询精度的方法.文献[13]通过对先前区间计数查询进行学习得到区间计数查询的先验概率分布,然后往区间树添加噪声,最后利用带权最小二乘法修复差分隐私区间树的一致

收稿日期:2013-06-09
基金项目:国家自然科学基金项目(61300026);中央高校基本科研业务费专项资金(2010121066);深圳市战略性新兴产业发展专项资金;福州大学科技发展基金项目(2012·XQ-27)
* **通信作者**:lsun@fzu.edu.cn

性,从而提高差分隐私直方图发布区间计数查询的精度. 文献[14] 利用聚类技术,提出了另一种优化区间计数查询的方法 NoiseFirst,其基本思想是对近似相邻的集合进行聚类,使他们成为一个新集合,然后往每个新集合添加噪声,由于构成新集合的子集合数可能不同,故所添加噪声也可能不同.

以上有关直方图发布的算法都是针对一般数据集展开研究的,只有文献[14]在实验结果中指出,当用直方图标识的数据集中存在大量低频计数值时,利用 NoiseFirst 算法所发布数据的精度优于其他方法. 然而,由于文献[14]在对近似相邻集合进行聚类时采用的是动态规划策略以保证发布数据精度的全局最优性,而且该动态规划算法的时间效率不高,因此较高的发布数据精度是以牺牲算法时间效率为代价的,尤其是当属性域取值空间较大时,该算法无法在可接受的时间内得到执行结果.

本文拟在差分隐私直方图发布的背景下,针对具有大量低频计数值的数据发布,设计一个基于桶划分的差分隐私直方图发布贪心算法,并通过与同类算法的对比,体现本文算法在保证所发布数据质量和可用性的前提下,在时间效率上具有的明显优势.

2　基础知识与相关定义

定义 1　(兄弟数据表[5])如果数据表 T_1 和 T_2 有且仅有一条记录的值不同,则称 T_1 与 T_2 为兄弟数据表.

定义 2　(ε-差分隐私[5])给定兄弟数据表 T_1 和 T_2,若隐私算法 K 对数据表 T_1 和 T_2 的任意输出 $S\subseteq \mathrm{Range}(K)$ 均满足下面的公式,则称算法 K 满足 ε-差分隐私.

$$\Pr[K(T_1)\in S]\leqslant e^{\varepsilon}\times \Pr[K(T_2)\in S], \quad (1)$$

其中,$\mathrm{Range}(K)$ 表示算法 K 的取值范围. $K(T_1)$、$K(T_2)$分别表示以 T_1、T_2 作为算法 K 的输入而得到的输出. K 是一个随机算法,$\Pr[K(T_1)\in S]$、$\Pr[K(T_2)\in S]$分别表示输出 $K(T_1)$、$K(T_2)$结果为 S 的概率,也表示隐私被披露的风险.

参数 ε 用于衡量隐私保护的强度,其大小由数据管理者设定. 更小的 ε 表示需要添加幅度更大的噪声,从而带来更强的隐私保护. 显然,差分隐私保护强度与发布数据的可用性之间是一对矛盾. 一般地,ε 是一个接近于 1 的公开实数,且满足 $e^{\varepsilon}\approx 1+\varepsilon$,通常取值为 0.01,0.1,1,ln2 和 ln3 等,也可根据实际需要而取值.

最早实现差分隐私的方法是由 Dwork 等提出的拉普拉斯机制[3],该机制也是目前应用最为广泛的机制. 拉普拉斯机制通过添加拉普拉斯噪声实现差分隐私.

定义 3　(拉普拉斯机制[5])拉普拉斯机制通过添加 Laplace 噪声实现差分隐私,其基本思想是将统计数据发布对每一条查询的答复,都视为原数据表 T 通过一个函数 f 得到的映射值 $f(T)$,这些函数构成函数集 F. 往函数集 F 中的每一个 $f(T)$ 添加满足拉普拉斯分布的噪声 η 得到 $f(T)+\eta$. 其中,η 是一个连续型随机变量,其概率密度函数为:

$$p(\eta)=\frac{1}{2\lambda}e^{-\frac{|\eta|}{\lambda}}. \quad (2)$$

由公式(2)可知,拉普拉斯分布的期望值为 0,方差为 $2\lambda^2$. 参数 λ 用于衡量添加噪声的幅度大小,λ 越大,添加噪声幅度越大,隐私保护越强,反之亦然.

除 λ 之外,另一个影响隐私保护强度的因素是敏感度,拉普拉斯机制的敏感度定义如下.

定义 4　(拉普拉斯机制敏感度[5])给定一个函数集 F,若对任何一个函数 $f\in F$ 的输出结果为一个实数,则 F 的敏感度定义为:

$$S(F)=\max_{T_1,T_2}\left(\sum_{f\in F}|f(T_1)-f(T_2)|\right), \quad (3)$$

其中 T_1 和 T_2 为任意一对兄弟数据表.

在许多类型的查询下,敏感度 $S(F)$的值非常小. 以下定理将差分隐私强度参数 ε 同拉普拉斯噪声参数 λ 以及定义 4 的敏感度相关联.

定理 1[5]　设有函数集 F,其敏感度为 $S(F)$,K 为向 F 中每一个函数 f 的输出添加独立噪声的算法. 如果该噪声为参数值取 $S(F)/\varepsilon$ 的拉普拉斯分布,那么算法 K 满足 ε-差分隐私;如果该噪声为参数值取 λ 的拉普拉斯分布,那么算法 K 满足 $(S(F)/\lambda)$-差分隐私.

定理 1 表明,实现差分隐私需要添加的 Laplace 噪声参数 λ 的值取决于敏感度 $S(F)$和差分隐私参数 ε.

3　基于桶划分的差分隐私直方图发布贪心算法

3.1　问题描述

给定一个数据表 T,该数据表包含 n 条记录,每条记录均含有数值型属性 A. 假设属性 A 分别有 m 个不同的取值 $a_1,a_2,\cdots,a_m$,经过统计得到数据表 T 的 n 条记录中,属性 A 分别取 $a_1,a_2,\cdots,a_m$ 的计数值为

$x_1, x_2, \cdots, x_m$,将这些计数值形式化的表示为属性 A 的计数序列 $M=\{x_1, x_2, \cdots, x_m\}$. 实际生活中,一些数据表关于某一属性 A 得到的计数序列中可能存在不少 x_i 的取值非常小. 本文拟针对这样的计数序列,在保证发布数据质量的前提下,设计出高效的差分隐私直方图发布算法.

直方图是一种描述数据统计信息的有效方式,其采用某种划分策略将一个较大的数据集划分成不相交的子集或桶,每个桶用其在某种意义下的计数值表示其特征. 设有一个包含 k 个桶的桶划分方案 $H=\{B_1, B_2, \cdots, B_k\}$ 中,每个桶 $B_j=(l_j, r_j, c_j)$,其中 l_j 和 r_j 分别表示每个桶的左右边界,c_j 为赋给每个桶的值,通常取第 j 个桶中所有 x_i 的平均值. 现将计数序列 $M=\{x_1, x_2, \cdots, x_m\}$ 中的元素分别装入 k 个桶,每个桶内的元素分别满足 $\{x_i \mid l_j \leqslant i \leqslant r_j\}$. 显然,不同的桶划分策略将会对发布数据产生不同的影响,即导致不同的误差. 下面给出桶划分的误差定义.

定义 5 (桶划分误差)给定计数序列 $M=\{x_1, x_2, \cdots, x_m\}$ 和桶划分策略 $H=\{B_1, B_2, \cdots, B_k\}$,根据桶划分策略 H,可以得到一个新的计数序列 $M'=\{x'_1, x'_2, \cdots, x'_m\}$,将 M' 相对 M 的误差称作桶划分策略 H 对于 M 的误差,记作 $\mathrm{Error}(H, M)$.

为了衡量不同桶划分策略的优劣,以平方和误差(sum of squared error,SSE)进行度量. 由于每个桶 B_j 造成的误差 $\mathrm{Error}(B_j)$ 为:

$$\mathrm{Error}(B_j) = \sum_{l_j \leqslant i \leqslant r_j} (c_j - x_i)^2, \tag{4}$$

所以桶划分策略 H 对于 M 的误差 $\mathrm{Error}(H, M)$ 为:

$$\mathrm{Error}(H, M) = \sum_j \sum_{l_j \leqslant i \leqslant r_j} (c_j - x_i)^2. \tag{5}$$

为了尽可能减小桶划分的误差,本文采用贪心策略进行桶的合并. 每次均选择产生误差最小的相邻两个桶进行合并.

定义 6 (相邻桶的距离)相邻桶的距离即相邻两个桶进行合并所产生的误差增益,用 $\mathrm{dis}(B_j, B_{j+1})$ 表示. 该误差增益可以通过下面的公式进行度量.

$$\mathrm{dis}(B_j, B_{j+1}) = \mathrm{Error}(B_j \& B_{j+1}) - \mathrm{Error}(B_j) - \mathrm{Error}(B_{j+1}), \tag{6}$$

其中 $B_j \& B_{j+1}$ 表示桶 B_j 与桶 B_{j+1} 合并后的新桶. $\mathrm{Error}(B_j)$ 和 $\mathrm{Error}(B_{j+1})$ 分别表示桶 B_j 和桶 B_{j+1} 造成的误差,$\mathrm{Error}(B_j \& B_{j+1})$ 表示桶 B_j 与桶 B_{j+1} 合并后得到的新桶产生的误差.

从定义 5 和定义 6 可以看出,不同的桶划分策略会对数据的失真程度有重要的影响,文献[14]依据桶划分的思想设计出了有效的算法解决发布数据的质量问题,但是由于算法采用动态规划策略寻找最优的桶划分方案,算法执行效率较低. 如何在保证发布数据质量的前提下,设计出有效的桶划分策略,提高算法效率是本文研究的核心.

3.2 桶划分的贪心算法

本文采用拉普拉斯机制向计数序列 M 添加噪声得到 M^*,使其满足 ε-差分隐私;然后基于桶划分[14]的思想,对带噪声的计数序列寻找一个合理的桶划分策略. 桶划分阶段的目的是寻找一个有效的桶划分策略 $H=\{B_1, B_2, \cdots, B_k\}$. 将添加了拉普拉斯噪声的计数序列 M^* 中的值 $x_1^*, x_2^*, \cdots, x_m^*$ 分别放入 k 个桶中. 显然 k 的取值范围是 $[1, m]$,k 取值不同会得到不同的桶划分误差. k 取 1 时,即将所有的频率值放入一个桶中,此时,$x_1^*, x_2^*, \cdots, x_m^*$ 均由平均值 $\sum_{i=1}^{m} x_i^* / m$ 取代,所有值的统计频率相同,原数据的失真很大,但是降噪效果较好;k 取 m 时,每个频率值独立为一个桶,$M' = M^*$,此时数据失真最小,但是降噪效果最差. 为了能够保持数据失真程度与降噪效果之间的平衡,需要选取合适的 k 值.

本节算法首先考虑将 M^* 的 m 个数据 $x_1^*, x_2^*, \cdots, x_m^*$ 放入 $k=m$ 个桶中,然后每次选择最邻近的两个桶进行合并. 随着相邻桶的每次合并,桶的数量 k 取值越来越小. 在合并过程中,将 $k=m, \cdots, 2, 1$ 的 m 个不同桶划分策略表示为 $H_m, H_{m-1}, \cdots, H_2, H_1$,这 m 个桶划分策略中将存在一个最优的划分策略. 对不同桶划分策略 H_k 的优劣采用文献[14]中的公式

$$Q = \mathrm{Error}(H_k, M^*) + \frac{4k - 2m}{\varepsilon^2} \tag{7}$$

进行评价. Q 值越小,桶划分策略越好.

在每次进行相邻桶合并时,为了保证选取的两个桶 B_j 与 B_{j+1} 是最相邻的,需要在当前 k 的取值下,对 $j=1, 2, \cdots, k-1$ 的不同取值,依据公式(6)计算并比较相邻桶的距离 $\mathrm{dis}(B_j, B_{j+1})$. 具体算法描述如下:

算法 1:桶划分贪心算法

输入:数据表 T,差分隐私参数 ε.

输出:桶划分优化后的带噪声计数序列 M'.

1) 将数据表 T 映射成长度为 m 的计数序列 M.

2) 往计数序列 M 的每一项添加独立噪声 $\mathrm{Lap}(1/\varepsilon)$,得到矩阵 $\boldsymbol{M}^*$.

3) 设 k 为桶的数量,初始化 $k=m$,把 M^* 的 m 个元素装入 k 个桶中.

4) 计算 $\mathrm{dis}(B_j, B_{j+1})$,贪心寻找最邻近的两个桶,并把这两个桶合并成一个新桶,然后 k——.

5) 重复步骤 4)直到 $k=1$,记录使得 Q 最小的 k 的取值,记为 k_best.

6) 再次执行步骤 3),然后重复执行步骤 4)直到 $k=k$_est,将每个桶内的所有元素分别赋值为桶内元素的平均值,然后将这 m 个新值赋给 M' 输出.

3.3　时间复杂度分析

算法 1 的时间复杂度主要由以下几个步骤决定:将数据表 T 映射成计数序列 M,该步骤需要 $O(n)$ 的时间复杂度;向计数序列 M 添加拉普拉斯噪声得到 M^* 需要 $O(m)$ 的时间复杂度;针对 m 种不同的桶划分策略,每次的桶划分策略 H_k 都要通过枚举法寻找最邻近的两个桶并使用线性表合并这两个桶,时间复杂度为 $O(m^2)$. 所以算法 1 的时间复杂度为 $O(n+m^2)$.

3.4　基于红黑树的桶划分算法优化

算法 1 中使用枚举法寻找最邻近的两个桶,并使用线性表将找到的最邻近两个桶进行合并,这两个过程的时间复杂度均为 $O(m)$. 可利用两颗红黑树分别对这两个步骤进行优化.

红黑树是一种支持在对数时间 $O(\log m)$ 实现插入、删除、查询以及返回最小值的树形结构. 为了便于描述,现将用于最近距离查询的红黑树称为距离红黑树(distance red-black tree,DRBT),将用于桶合并操作的红黑树称为合并红黑树(merge red-black tree,MRBT).

算法 2:查询以及合并最邻近桶过程的优化

输入:DRBT, MRBT.

输出:更新后的 DRBT,更新后的 MRBT.

1) 从 DRBT 返回最小值,得到要合并的桶为 B_j 与 B_{j+1},从 DRBT 删除 B_j 与 B_{j+1} 的合并误差增益.

2) 如果 $j>1$,从 MRBT 查找 B_{j-1},在 DRBT 中删除 B_{j-1} 与 B_j 的合并误差增益(因为 B_j 将要和 B_{j+1} 合并). 同理如果 $j+1<m$,在 DRBT 中删除 B_{j+1} 与 B_{j+2} 的合并误差增益.

3) 从 MRBT 删除 B_{j+1}(B_{j+1} 已被 B_j 合并).

4) 如果 $j>1$,从 MRBT 查找 B_{j-1},在 DRBT 中插入 B_{j-1} 与 B_j 的合并误差增益. 同理如果 $j<m$,在 DRBT 中插入 B_j 与 B_{j+1} 的合并误差增益.

算法 2 中利用红黑树对寻找最邻近两个桶以及将这两个桶进行合并的过程进行优化后,使得这两个过程时间复杂度降至 $O(\log m)$,因此优化之后的算法的时间复杂度为 $O(n+m\log m)$.

4　实验结果与分析

实验采用 3 种不同数据集中不同的属性作为实验数据,这些数据集及属性分别是巴西人口统计数据集中的年龄属性 Age、美国人口统计数据集家庭版中每个家庭的月煤气消费量属性 GASP 和月租金属性 ARNTP、美国人口统计数据集个人版中的个人收入属性 PINCP. 每个数据集中包括的记录数 n 及属性的不同取值 m 如表 1 所示.

表 1　数据集规模

Tab. 1　The size of the data set

数据集	Age	GASP	GRNTP	PINCP
n	100079675	1981693	1981693	3597186
m	102	648	5413	15551

实验环境:Intel ® Pentium ® Dual2.70 GHz,2 GB 内存,Windows XP 操作系统,Visual C++6.0. 采用 C++中 set 模板实现红黑树,并利用 MATLAB 作出实验图表. 为了便于描述,用 CNFG(closest neighbor first greedy)表示本文的桶划分算法.

4.1　数据发布精度

为了说明 CNFG 能够有效地保证发布数据的可用性,将 CNFG 算法同文献[14]的 NoiseFirst 算法及 Dwork 提出的差分隐私拉普拉斯噪声机制[5]进行发布数据精度的比较. 实验中,差分隐私参数 ε 分别取 0.01,0.1 和 1,采用 SSE 衡量计数序列中每个元素的误差. 为了使实验的结果更具一般性,每种算法运行 50 次,取 50 次结果的平均值作为最终结果.

由于文献[14]的 NoiseFirst 算法时间复杂度为 $O(m^3)$,而数据集 GRNTP 和 PINCP 中对应属性的 m 取值较大,因此对于这两个数据集,NoiseFirst 算法无法在可接受的时间内得到运行结果,故图 1 中没有对应的图形显示.

从图 1 可以看出,对于数据集 Age,采用了桶划分思想的 CNFG 算法和 NoiseFirst 算法相对于 Dwork 提出的差分隐私拉普拉斯噪声机制而言,所发布数据的误差并没有减小,反而在某些 ε 取值下略有增加. 究其原因,Age 是一个分布相对较均匀的数据集,不存在之前文中描述的低频情况,所以 CNFG 算法和 NoiseFirst 算法应用于 Age 数据集的数据发布无法体现算法的优势. 其他 3 个数据集中,对应属性的某些取值都有频率较低的现象,尤其是 PINCP 数据集中这种现象最为严重,因此将 CNFG 算法用于这 3 个数据集的数据发布所造成的数据误差有明显改善,且低频现象越明显的,误差的改善情况越好.

NoiseFirst 采用的动态规划算法保证了桶划分策

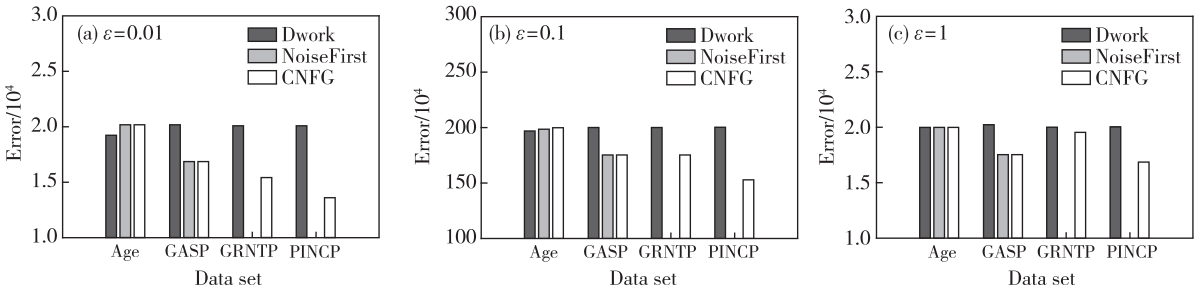

图 1 不同 ε 取值下 3 种算法的平均误差

Fig. 1 The average error of three algorithm under different ε

表 2 不同 ε 取值下 CNFG 算法和 NoiseFirst 算法的运行时间

Tab. 2 Execution time between CNFG and NoiseFirst under different ε s

Data set	Age(m=102)			GASP(m=648)		
	0.01	0.1	1	0.01	0.1	1
NoiseFirst	0.00532	0.00468	0.00376	1.02842	1.02154	0.99262
CNFG	0.00064	0.00030	0.00094	0.01188	0.01010	0.00858
Data set	GRNTP(m=5413)			PINCP(m=15551)		
	0.01	0.1	1	0.01	0.1	1
NoiseFirst	—	—	—	—	—	—
CNFG	0.10990	0.10148	0.08958	0.33658	0.35566	0.31848

略的最优性,使得该算法具有较好的数据发布精度. 在图 1 中,对于数据集 GASP,采用贪心策略的 CNFG 算法和 NoiseFirst 算法所发布数据的误差相当,由此可见,本文的 CNFG 算法同样具有较好的数据发布精度.

4.2 时间效率分析

表 2 为不同 ε 取值下 CNFG 算法和 NoiseFirst 算法在 4 个数据集下的执行时间. 为了更好地比较不同算法之间的效率差异,表中的运行时间不包含原数据表映射为计数序列所消耗的时间.

从表 2 可以看出,当 m 的值小于1 000时,CNFG 算法和 NoiseFirst 算法都有较高的时间效率,且 CNFG 算法比 NoiseFirst 算法效率更高. 随着 m 的增大,两个算法运行的时间差距变大. 当 m 的值大于 5 000时,CNFG 算法依然能够保持在较短的时间内得到运行结果,而 NoiseFirst 已经无法在可接受的时间内运行出结果.

5 结 论

本文针对存在大量低频计数值数据集的差分隐私直方图发布,提出了一种基于桶划分的贪心算法. 算法通过将频率相近的相邻数据划分在一个桶内,并对同一个桶内的数据以平均值取代来提高噪声,以达到平衡差分隐私发布数据精度和隐私保护程度的目的. 基于真实数据集的仿真实验结果表明,本文算法可在保证发布数据质量及可用性的前提下拥有较高的执行效率.

参考文献:

[1] Sweeney L. k-anonymity: a model for protecting privacy [J]. International Journal on Uncertainty, Fuzziness and Knowledge Based Systems, 2002, 10(5): 557-570.

[2] Machanavajjhala A, Gehrke J, Kifer D, et al. l-diversity: privacy beyond k-anonymity[J]. ACM Transactions on Knowledge Discovery from Data (TKDD), 2007, 1(1): 3.

[3] Li N, Li T. t-closeness: privacy beyond k-anonymity and l-diversity[C]// Proc of the 23rd International Conference on Data Engineering (ICDE). Istanbul, Turkey: IEEE, 2007: 106-115.

[4] Dwork C. Differential privacy[C]// Proc of the 33rd International Colloquium on Automata, Languages and Programming (ICALP). Venice, Italy: Springer Berlin Hei-

delberg,2006:1-12.

[5] Dwork C,McSherry F,Nissim F K,et al. Calibrating noise to sensitivity in private data analysis[C]// Proc of the 3rd Theory of Cryptography Conference (TCC). New York, USA:Springer Berlin Heidelberg,2006:363-385.

[6] Dwork C. Differential privacy:a survey of results[C]// Proc of the 5th Theory and Applications of Models of Computation (TAMC). Xi′an, China: Springer Berlin Heidelberg,2008:1-19.

[7] Dwork C,Smith A. Differential privacy for statistics:what we know and what we want to learn[J]. Journal of Privacy and Confidentiality,2009,1(2):135-154.

[8] Hay M,Rastogi V,Miklau G,et al. Boosting the accuracy of differentially private histograms through consistency [C]// Proc of the 36th Conference of Very Large Databases (VLDB). Istanbul, Turkey: VLDB Endowment, 2010:1021-1032.

[9] Xiao X,Wang G,Gehrke J. Differential privacy via wavelet transforms[J]. IEEE Transactions on Knowledge and Data Engineering (TKDE),2011,23(8):1200-1214.

[10] Li C,Hay M,Rastogi V,et al. Optimizing linear counting queries under differential privacy[C]// Proc of the 29th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems(PODS). Indianapolis,USA: ACM,2010:123-134.

[11] Li C, Miklau G. An adaptive mechanism for accurate query answering under differential privacy[C]// Proc of the 38th Conference of Very Large Databases(VLDB). Istanbul,Turkey:VLDB Endowment,2012:514-525.

[12] Xiao X,Bender G,Hay M,et al. iReduct:differential privacy with reduced relative errors[C]// Proc of the 2011 ACM SIGMOD International Conference on Management of Data (SIGMOD). Athens:ACM,2011:229-240.

[13] Peng S,Yang Y,Zhang Z,et al. DP-tree:indexing multidimensional data under differential privacy[C]// Proc of the 2012 ACM SIGMOD International Conference on Management of Data (SIGMOD). Scottsdale: ACM, 2012:864.

[14] Xu J,Zhang Z,Xiao X,et al. Differentially private histogram publication[C]// Proc of IEEE 28th International Conference on Data Engineering(ICDE). Washington, DC:IEEE,2012:32-43.

Greedy Algorithm Based on Bucket Partitioning for Differentially Private Histogram Publication

SUN Lan[1*],WU Ying-jie[1],ZHANG Xi-lin[1],XIE Yi[2]

(1. College of Mathematics and Computer Science,Fuzhou University,Fuzhou 350108,China;
2. School of Information Science and Engineering,Xiamen University,Xiamen 361005,China)

Abstract:Existed methods of differential privacy histogram publication cannot efficiently protect the privacy of data sets which have a large amount of small counts for different values of certain attribute. We proposed an efficient greedy algorithm to handle the differential privacy histogram publish problem of this particular data sets. The greedy algorithm adopted the merging strategies of closest neighbor barrel,and optimized the merging procedure using the red-black tree. Many simulation experiments were launched to compare the proposed algorithm with similar algorithms in terms of the availability of published data and the efficiency of algorithm. The simulation results verify that the greedy algorithm is efficient.

Key words:differential privacy;histogram publication;bucket partitioning;greedy algorithm;red-black tree