

3.2 混淆编码和隐私保护 k - d tree 生成

首先,我们对原数据进行混淆编码,混淆编码本质上是一个保距映射(保持相对距离),经过混淆的原始数据实际上进行了变换和缩放,其目的是抵御背景知识攻击,经过混淆编码的向量数据相对位置不发生改变,真实距离转换为相对距离,因此不改变聚类特性.云服务提供商完成聚类挖掘,将挖掘结果返回给用户,用户首先解密聚类结果,然后进行逆混淆编码得到聚类结果.数据发布的过程中,列名同样是需要隐藏的元素,以降低暴力破解的攻击风险.但攻击者确定某一列是血压时,随机选择某一密钥进行解密,当其将数据解密后发现在正常的血压范围里面,随机选择的密钥就有可能是解密密钥;但是如果无法确定选择的列的属性,就很难确定当前选择密钥是否正确.本质上讲,混淆编码过程首先进行一系列的吉文斯旋转,然后进行缩放,进而进行向量的交换混淆,最后进行平移变换. T_{pq} 将一单位矩阵的第 p 行的所有元素与第 q 行互换,这一变换 $T_i(k_i)$,将第 i 行的所有元素乘以非零常数 k_i .

混淆编码过程 $code=key_1m+key_2$,其中, $key_1 = \prod T_{pq} \prod_{i=1}^n T_i(k) \prod G(i, j, \theta)$, $key_2 = [k_1, k_2, \dots, k_{n-1}, k_n]^T$.

在混淆编码的数据上面构建的 k - d tree,需要证明经过混淆编码向量向量仍旧保持了了其相对位置.显而易见,缩放变换、元素交换以及平移变换不改变元素之间的相对位置,下面我们证明一系列的吉文斯变换不会改变向量之间的相对位置.

性质 1. 经过一系列吉文斯变换,向量之间的距离不变.

证明:设向量 $\vec{x} = (x_1, x_2, \dots, x_n)$, $\vec{y} = (y_1, y_2, \dots, y_n)$, 将向量 \vec{x} , \vec{y} 分别执行一轮吉文斯变换,得到 \vec{x}' , \vec{y}' , 即:

$$\begin{cases} \vec{x}' = G(i, j, \theta) \cdot \vec{x} = (x_1, \dots, x_i \cos \theta - x_j \sin \theta, \dots, x_i \sin \theta + x_j \cos \theta, \dots, x_n) \\ \vec{y}' = G(i, j, \theta) \cdot \vec{y} = (y_1, \dots, y_i \cos \theta - y_j \sin \theta, \dots, y_i \sin \theta + y_j \cos \theta, \dots, y_n) \end{cases}$$

则转换过后二者终点的距离:

$$\begin{aligned} d_1(\vec{x}', \vec{y}') &= \sqrt{\sum_{k=1}^n (x'_k - y'_k)^2} \\ &= \sqrt{\sum_{k \neq i, j} (x'_k - y'_k)^2 + (x'_i - y'_i)^2 + (x'_j - y'_j)^2} \\ &= \sqrt{\sum_{k \neq i, j} (x_k - y_k)^2 + [(x_i - y_i) \cos \theta - (x_j - y_j) \sin \theta]^2 + [(x_i - y_i) \sin \theta + (x_j - y_j) \cos \theta]^2} \\ &= \sqrt{\sum_{k \neq i, j} (x_k - y_k)^2 + (x_i - y_i)^2 + (x_j - y_j)^2} \\ &= \sqrt{\sum_{k=1}^n (x_k - y_k)^2} = d_1(\vec{x}, \vec{y}). \end{aligned}$$

综上证明:经过一轮吉文斯变换,向量之间的距离不会发生变换.同理可证:经过一系列吉文斯变换,向量的距离不会发生变化.所以在混淆编码上面构建的 k - d tree,在保证安全的同时保持了其索引特性.

3.3 用户加密发布数据(Enc)

本文中,明文多维数据以向量的形式存储,对于明文高维向量 m ,用户通过高维随机向量生成 r , r 的模应该小于格中向量之间距离的一半.用户使用自己的公钥 K_{pub} 加密 m ,生成的密文二元组 (s_i, t_i) 由两部分组成(第 1 部分 s_i 用于解密过程还原明文数据;第 2 部分 t_i 为 $K_{pub}m$ 的哈希值,主要用于密文状态下的辅助同态计算过程):

$$e_i = (s_i, t_i) = (k_{pub} \cdot m_i + r_i, \text{hash}_M(k_{pub} \cdot m_i)) \quad (1)$$

M 为哈希矩阵,矩阵的秩为 1,见公式(2). $M_{hash}(K_{pub} \cdot m)$ 的哈希过程是单向的,对于该哈希等式的求解,满足条件的解有无数多个.对于同一个数据集, M_{hash} 的取值应满足公式(3),云服务提供商才可以计算密文距离:

$$M_{hash} = \begin{bmatrix} h_1 & h_2 & \dots & h_{n-1} & h_n \\ 0 & 0 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{n \times n} \quad (h_i \in Z) \quad (2)$$

$$\text{rank}([M_{hash}; M_{\otimes}]) = n \quad (3)$$

3.4 用户解密还原挖掘结果(Dec)

云服务提供商将挖掘结果返回给用户,此时,挖掘结果是密文状态的,防止了云服务提供商的非授权访问.用户得到挖掘结果需要进行如下操作,最终获得明文挖掘结果:

$$v = \text{round}(S_i K_{pri}^{-1}) \cdot K_{pri}$$

$$\text{code} = v \cdot K_{pub}^{-1}$$

用 Babai 算法计算最近向量 v 最接近 s_i ,再次计算 $v \cdot K_{pub}^{-1}$,即得到明文编码 code .

逆混淆编码过程还原混淆编码数据为原始数据:

$$m = \text{key}_1^T (\text{code} - \text{key}_2)$$

3.5 云服务提供商距离的密文计算

在隐私保护聚类挖掘的过程中,云服务提供商需要计算密文之间的距离.对于一对密文向量 (s_1, t_1) 及 (s_2, t_2) ,密文的计算过程如下:

$$s_1 = K_{pub} \cdot m_1 + r_1, t_1 = \text{hash}_M(K_{pub} \cdot m_1) \quad (4)$$

$$s_2 = K_{pub} \cdot m_2 + r_2, t_2 = \text{hash}_M(K_{pub} \cdot m_2) \quad (5)$$

等式(4)减等式(5),得:

$$s_1 - s_2 = K_{pub} \cdot (m_1 - m_2) + r_1 - r_2 \quad (6)$$

$$t_1 - t_2 = M_{hash} \cdot (K_{pub} \cdot m_1) - M_{hash} \cdot (K_{pub} \cdot m_2) = M_{hash} \cdot (K_{pub} \cdot (m_1 - m_2)) \quad (7)$$

对公式(6)两边分别乘以 $\bar{p} = \{l_1, l_2, \dots, l_{n-1}, l_n\}$,由定义 4 扩展的高维向量积的性质可得等式(8):

$$\bar{p} \cdot (S_1 - S_2) = \bar{p} \otimes (K_{pub} \cdot (m_1 - m_2)) = M_{\otimes} \cdot (K_{pub} \cdot (m_1 - m_2)) \quad (8)$$

\bar{p} 与 K_{pub} 对于云服务提供商已知,又由等式(8)结合等式(7)可以求出 $K_{pub} \cdot (m_1 - m_2)$, K_{pub} 对于云端已知,进而可以得到 $(m_1 - m_2)$,最终可以求出 m_1 和 m_2 的欧式距离.

$\bar{p} \cdot (S_1 - S_2)$ 对于云服务提供商已知,且云服务提供商掌握 \bar{p} ,但是在等式(8)中, \bar{p} 生成的矩阵的值为 0,所以依靠密文二元组的第 1 部分无法求解出 $K_{pub} \cdot (m_1 - m_2)$.当直线取值满足等式(3)时,由等式(7)和等式(8)可以求解出唯一的 $K_{pub} \cdot (m_1 - m_2)$,进而求解出唯一的 $(m_1 - m_2)$,最终求得欧氏距离.

算法 2. *computeDistance*.

Input:密文元组 e_1, e_2 ;

Output:元组之间的欧式距离 dis_{e_1, e_2} .

1: $(S_1 - S_2)$

2: $\bar{p} \cdot (S_1 - S_2)$

3: 等式(7)和等式(8)可求解出 $K_{pub} \cdot (m_1 - m_2)$

4: 左乘 K_{pub}^{-1} 即可得到 $m_1 - m_2$

5: $dis_{e_1, e_2} = \text{computeEuclidean}(m_1 - m_2)$

6: return dis_{e_1, e_2} ;

3.6 均值密文计算

对于明文 m_1, m_2, \dots, m_n ,加密处理后如下所示:

$$(e_1, e_2, \dots, e_n) = \left\{ \begin{array}{l} e_1 = (K_{pub} \cdot m_1 + r_1, hash_M(K_{pub} \cdot m_1)) \\ e_2 = (K_{pub} \cdot m_2 + r_2, hash_M(K_{pub} \cdot m_2)) \\ \dots \\ e_n = (K_{pub} \cdot m_n + r_n, hash_M(K_{pub} \cdot m_n)) \end{array} \right\}.$$

密文均值计算公式:

$$e_{mid} = \left\{ \sum_{i=1}^n s_i / n, \sum_{i=1}^n t_i / n \right\}.$$

性质 2. 密文状态下均值运算具有同态性质.

证明:明文的中心点为 $m_{mid} = (\sum_{i=1}^n m_i) / n$, 由于 $(\sum_{i=1}^n r_i) / n$ 仍为直线上的点, 所以中心点的计算具有同态性质. 密文状态下的计算结果与直接对均值进行隐私保护的数据发布处理, 结果是一致的.

$$s_{mid} = (s_1 + s_2 + \dots + s_n) / n = K_{pub} \left(\sum_{i=1}^n m_i / n \right) + \sum_{i=1}^n r_i / n = K_{pub} \left(\sum_{i=1}^n m_i / n \right) + r = K_{pub} \cdot m_{mid} + r,$$

$$t_{mid} = (t_1 + t_2 + \dots + t_n) / n = hash_M \left(K_{pub} \left(\sum_{i=1}^n m_i / n \right) \right) = hash_M (K_{pub} \cdot m_{mid}).$$

3.7 性能分析和安全分析

与现有的方法相比, 由于本文采用基于格的困难问题的批处理方式进行数据发布, 与基于大素数难分解等困难问题进行数据发布的方式相比性能较好. 密文 k -mean 算法的时间复杂度是 $O(n \log(n))$, 密文层次聚类的时间复杂度为 $O(n^2)$, 密文 DBSCAN 时间复杂度 $O(n)$. 由于本文构建了隐私保护的 kd-tree, 密文挖掘过程中处理性能有很大的提升.

最短向量问题(the shortest vector problem, 简称 SVP)是寻找一个格 L 中最短的非零向量. 即, 寻找一个 $v \in L$ 满足其欧几里德范数 $\|v\|$ 最小. 最接近向量问题(the closest vector problem, 简称 CVP)是对于一个非格 L 中的向量 w , 在格中寻找一个向量 v , 使得 $\|w-v\|$ 最小. CVP 和 SVP 都是 NP 完备问题, 因此求解起来十分困难. 因此, 这两个问题都是可以作为密钥体制的基础的. 论文中对于明文 m , 其密文 $S_i = K_{pub} \cdot m_i + r_i$, 其中, 公钥 K_{pub} 是一组劣质级, 云端掌握 S_i 和 K_{pub} , 求解明文 m_i 必须计算出 $K_{pub} \cdot m_i$. $K_{pub} \cdot m_i$ 求解是一个最接近向量求解问题, 这是一个 CVP 问题, 可以被证明是 NP 问题.

其次, 算法每次加密元组使用的 r_i 都是随机, 因此, 即使是同一明文, 两次不同的加密结果也不一样, 从而避免了统计攻击. 由于加密过程中随机数 r 的存在, 攻击者即使获得若干组明文密文对, 也无法通过其对应关系获取用户的隐私数据明文信息, 或者依靠对应关系计算出用户私钥. 因此, 本算法可以抵御选择明文攻击(chosen-plaintext attack). 本文算法与 RBT^[15]的平移旋转相比安全性更高, PBT 中, 数据的安全性依赖于 θ , 明文与密文有一一对应关系, 当泄露一组明文密文对时, θ 就会泄露, 所以 PBT 中的隐私保护处理方法不能抵御选择明文攻击.

当敌手 A 通过公钥无法计算得到算法私钥, 该算法即具有不可攻破性. 而本文方法对于敌手 A 掌握公开参数 K_{pub} , 求解 K_{pri} 需要求解最短向量问题, 该问题是 NP 问题, 因此, 本文的算法具有不可攻破性. 密钥取值位数在生成密钥的时候可以调节, 当密钥长度是 1024 位, 暴力破解需要 2^{1024} 次枚举, 安全性得到了一定的保证.

4 基于 k - d tree 的隐私保护聚类挖掘方法(PPCM)

4.1 基于密文的 k - d tree

隐私保护的聚类挖掘过程中, 图 4 所示的 k -mean 需要求解出指定范围中的元素, 图 5 所示的 dbscan 需要求解核心点. 该问题可以转化为求解 KNN 问题, 即, k 最近邻是否在指定半径内. 图 6 所示的层级聚类需要求解最近邻问题, 事实上就是 KNN 问题的特殊情况. 构建高效的高维密文索引, 可以高效地进行范围查询和 KNN 查询, 是个亟需解决的困难问题. 本文在混淆编码的基础上构建了密文 k - d tree, 用于聚类挖掘中的性能优化. 用户将数据上传到云服务提供商之前, 构建密文 k - d tree (如图 7 所示).

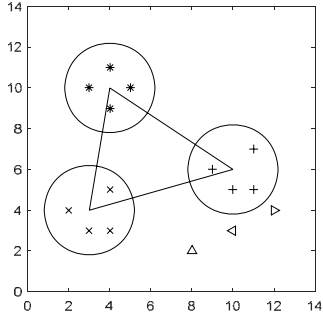


Fig.4 Range query problem of k -means

图4 k -means 范围查询问题

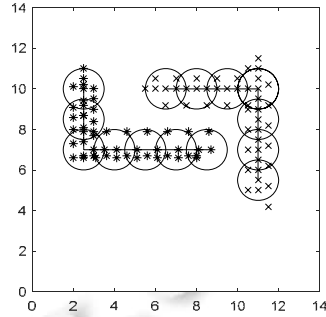


Fig.5 KNN problem of DbSCAN

图5 DbSCAN 的 KNN 问题

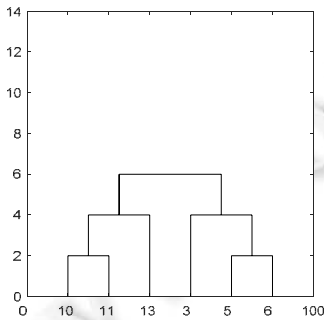


Fig.6 Nearest neighbor problem of hierarchical clustering

图6 层次聚类最近邻问题

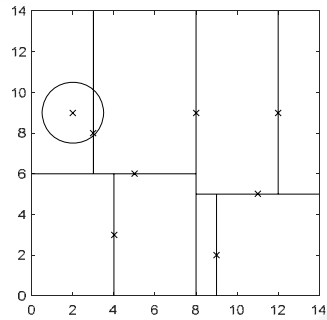


Fig.7 Two-dimensional ciphertext k - d tree example

图7 二维密文 k - d tree 示例

k - d 树最近邻搜索的过程如下:从根节点开始,递归地往下移.往左还是往右的决定方法与插入元素的方法一样(如果输入点在分区面的左边则进入左子节点,在右边则进入右子节点).一旦移动到叶节点,将该节点当做目前最佳点.解开递归,并对每个经过的节点运行下列步骤:如果目前所在点比目前最佳点更靠近输入点,则将其变为目前最佳点.检查另一边子树有没有更近的点,如果有,则从该节点往下找.当根节点搜索完毕后,完成最近邻搜索.

4.2 隐私保护聚类挖掘

首先,用户生成自己的公私钥对 K_{pri}, K_{pub} 以及公开参数向量 $\vec{p} = \{l_1, l_2, \dots, l_{n-1}, l_n\}$;然后,对数据进行隐私保护的数据处理,并将处理结果上传到云端,云端调用密文中心点计算算法和密文距离计算算法进行数据挖掘;最终,将多种挖掘算法的结果返回给用户,避免使用单一算法挖掘难以发现潜在的挖掘聚簇的问题.用户端使用保持距离的解密算法还原挖掘的最终明文结果.

用户预处理过程中,用户将元组使用第 3.3 节中的算法将隐私数据进行加密处理后,结果见表 5.

Table 5 Post-release data

表 5 发布数据

Id	Age	Weight	Heartrate
1237	$S(75), T(75)$	$S(80), T(80)$	$S(63), T(63)$
3420	$S(79), T(79)$	$S(64), T(64)$	$S(55), T(55)$
2543	$S(67), T(67)$	$S(52), T(52)$	$S(70), T(70)$
4461	$S(62), T(62)$	$S(58), T(58)$	$S(76), T(76)$

上传到云服务提供商,同时还需要上传的是加密过程中产生随机数向量的生成参数 \vec{p} .对于云端隐私保护的数据挖掘结果,用户可以使用第 3.4 节的算法得到明文数据.当用户可以判断聚类类型时,用户可以直接指定

挖掘算法;否则,云端使用多种聚类挖掘算法进行处理,返回聚类结果给用户。

云端收到用户的聚类分析请求后,根据聚类请求选择合适的聚类算法进行处理。隐私保护的聚类处理核心问题是密文数据点距离计算问题和均值密文计算问题。PPK-means 反复迭代计算密文均值和密文距离,直至簇的均值不发生变化,即:两次迭代计算均值之间的距离小于某一阈值 δ ,见算法 4。求解过程中,试用 $k-d$ tree 进行优化,以质心为中心、两个质心的连线距离的一般为半径,寻找此范围里面的向量,从而大幅度减少了距离计算的时间开销。基于层级的聚类需要求解距离最近的元组,该问题是 KNN 问题的特例,使用 $k-d$ tree 减少了距离运算,每个元组查找最近邻的时间复杂度由 $O(n)$ 降低为 $O(\log n)$ 。dbscan 需要求解向量是否为核心点,然后连接核心点,此时,利用隐私保护 $k-d$ tree 时间复杂度为 $O(\log n)$ 。

算法 4. 隐私保护的聚类算法 PPK-means.

输入:数据集 C 、簇的个数 K 和用户指定的阈值、迭代次数 num ;

输出:聚簇 C_i 。

1: choose c_1, c_2, \dots, c_n as initial centroid

2: Do {

3: $generateSearchPath()$;

4: $rangeQuery()$;

5: 调用第 3.5 节距离的密文计算方法,将每个点指派到最近的中间点,形成 K 个簇。

6: 调用第 3.6 节密文的中心点计算方法,重新计算每个簇的中间点

7: $i++$;

8: }while (质心发生变化大于指定的阈值 δ or $i < num$)

基于格的保持距离的加密算法可以精确地计算出向量之间的相对距离以及中心点与各个向量值间的相对距离。对于 k -means 而言,本文的发布方法可以准确地求解均值和密文条件下数据之间的距离,因此可以保证挖掘的正确性。对于层次聚类而言,本文算法求解的距离具有保持距离相对大小的特性,因此可以选择出当前相距最小的两个元素。对于 DNSCAN 而言,本文的隐私保护数据发布方法可以精确地求解 ϵ 邻域中的核心点。

综上所述,数据经过基于格的保持距离数据处理方法处理,上传到云端可以进行聚类挖掘过程中涉及的所有同态密文计算,因此可以保证挖掘的正确性。

5 实验结果与分析

5.1 实验环境

实验环境:Window 7 操作系统;CPU intel B950 2.10GHz;4.00GB 内存。编程环境:Matlab7.0.11 编译器。

实验数据:本文的实验数据选用加州大学尔湾分校的机器学习数据库中的心脏心律失常数据集的真实数据^[26],针对真实数据集规模有限的问题,本文利用正态分布生成了模拟数据集,使用模拟数据集进行较大规模的实验。参照 RBT 方法^[15],论文选择数据集中的 3 列数据:年龄(age)、体重(weight)和心率(heart rate)。实验中,随机选择 10 000,20 000,30 000 和 40 000 条记录规模分别进行了实验分析。衡量优质基的阈值参数采用 $\gamma=0.9$,加密过程批处理维度 3。

论文研究了一种隐私保护的数据聚类挖掘方法,引入挖掘准确率用于度量隐私保护数据挖掘算法与明文条件下数据挖掘算法的一致性,见公式(9)(隐私保护的聚类方法准确率与对应的普通明文条件下的数据挖掘结果进行比较,准确率值越高,表明隐私保护的聚类数据分析方法分析结果越精确):

$$rate_{accuracy} = \sum_{i=1}^k \frac{|C_i \cap S_i|}{|C_i|} \times \frac{|C_i|}{|C|} \quad (9)$$

其中, C 为数据集, C_i 为明文条件下得到的第 i 个数据聚类, S_i 为与之对应的隐私保护数据挖掘聚类方法得到的第 i 个密文数据聚类。本文设计的准确率度量方法中,考虑了聚类大小权重因素($|C_i|/|C|$)的影响。

本文设计实验,从两个方面对提出的隐私保护聚类数据挖掘方法进行评估,即,挖掘算法的性能和精确度。

5.2 实验结果分析

本文针对基于格的保持距离的加密算法进行了密钥生成、加密解密性能以及密文计算的时间开销的实验测试,实验结果证明,本文提供的方法切实可行.与现有的方法相比,聚类挖掘的精度优于现有方法,用户发布数据时在时间开销有所增加,但是在可接受范围内.实验分别针对基于格的保持距离的加密算法和聚类挖掘算法进行了分析比较.

5.2.1 基于格的保持距离的加密算法性能分析

密钥生成性能实验结果如图 8 所示,用户生成密钥的时间开销主要是生成公钥的时间,私钥的生成时间忽略不计.一次公钥私钥对生成时间约为 8ms.对于初始化密钥过程中的转换矩阵计算取值,实验结果表明,进行 30 次初等变换即可.由图 9 可以看出,加密的时间要远远超过解密的时间.这是因为加密过程中符合条件的随机数的生成占用了一定的时间.平均一次加密的时间为 0.4ms.如实验数据所示,本文提出的算法与现有的方法相比性能更好.主要是由于一次可以加密或解密一个元组中多列.由图 10 可以看出:40 000 次的密文距离计算只需要 67s,40 000 次的中间点求解计算只需要 67s,聚类挖掘所需要的云端的密文计算开销在可接收范围内.考虑到实际应用中云计算服务器性能强,实际应用中可以获得更好的隐私数据挖掘效率.

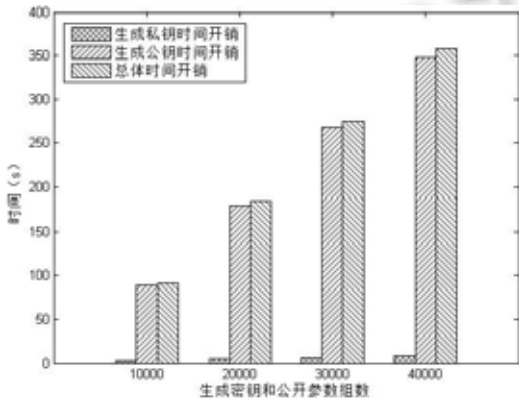


Fig.8 Key generation time overhead

图 8 密钥生成时间开销

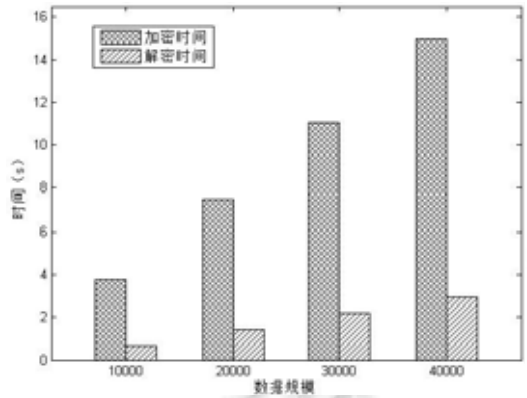


Fig.9 Encryption and decryption time overhead

图 9 加密解密时间开销

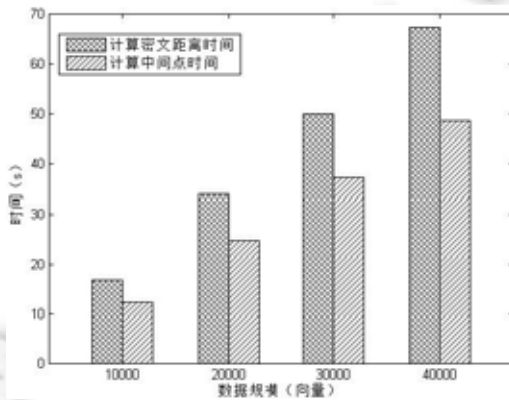


Fig.10 CSP calculation time overhead

图 10 云端密文计算时间开销

5.2.2 隐私保护的聚类挖掘方法性能与准确率分析

本文提出的聚类挖掘方法包括用户预处理部分时间开销、云服务提供商挖掘部分时间开销以及与现有方

法挖掘准确率的比较.

从图 11 可以看出:与现有的方法相比,本文 PPCM 发布数据的时间较高,但是在可以接受的范围内.主要原因是发布算法基于格的困难为题,提升了隐私保护数据的安全性而带来的时间开销.

从图 12 可以看出:密文 k -mean 算法的时间复杂度是 $O(n\log(n))$,密文层次聚类的时间复杂度为 $O(n^2)$,密文 DBSCAN 时间复杂度 $O(n)$.实验结果表明:实际情况中, k -mean 并非严格按照理论分析的时间复杂度产生时间开销,由于数据集的差异以及初始中心点的选择不同,PPk-mean 算法的时间开销会与理论分析有所差异.从结果可以看到:隐私保护的层次挖掘算法性能最差,PPDBscan 性能最好.一方面,这与选取的数据及有关关系;另一方面,3 种算法针对不同聚类类型的数据集各有优缺点,并无可比性.很多情况下,用户需要执行多种挖掘方法,以便发现数据集中潜在的聚类.

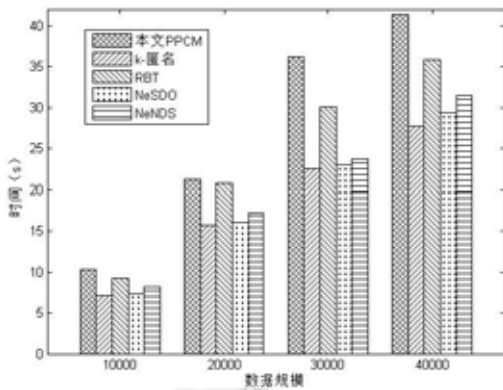


Fig.11 User preprocessing time overhead
图 11 用户预处理时间开销

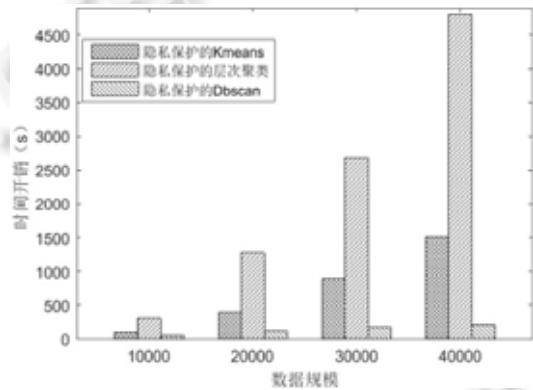


Fig.12 Privacy protection clustering mining time overhead
图 12 隐私保护的聚类挖掘时间开销

从图 13 可以看出:本文提出的算法与目前准确率最高的 RBT 算法的准确率相当,但是更加安全.在极端情况下,由于 k 取值的不同, k 匿名挖掘甚至于不能达到实验中的正确率.由于不同的挖掘算法侧重点不同,对于同一数据返回的结果相差较大,本文的准确率针对 3 种方法分别进行比较,即:原始数据的 k -means 结果与密文环境下 PPk-means 的结果进行对比,计算精确率.

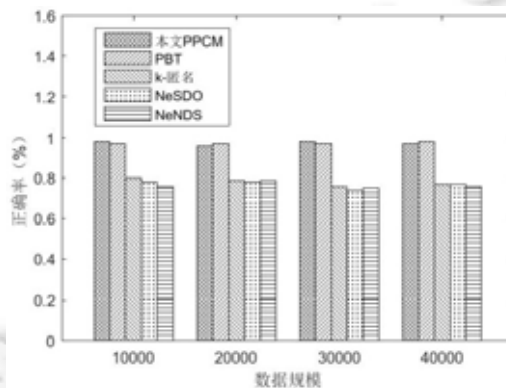


Fig.13 Privacy protection mining algorithm accuracy
图 13 隐私保护挖掘算法准确率

与现有的方法相比,本文的方法在增强安全的同时,很好地保证了挖掘结果的精度和挖掘的效率.虽然牺牲了一定效率,但保证了隐私数据的精度和安全性.对于用户而言,挖掘精度对于用户至关重要,挖掘结果的精度

出现偏差,很可能对于用户决策造成不可弥补的损失.

6 总结与展望

本文主要研究了云计算环境下支持隐私保护的隐私数据加密发布方法以及密文隐私数据的挖掘方法.首先,本文提出了一种用于隐私数据发布的基于格的保持距离的加密算法,在该方法的基础上,设计了聚类挖掘过程中需要的密文计算方法,可以对密文数据进行处理,处理的结果保持同态特性.云服务提供商在保障用户隐私的条件下,有能力进行聚类挖掘,最终将挖掘结果返回给用户.隐私保护的聚类挖掘方法提供了3种聚类挖掘机制,用户可以根据自己的数据特征,选择有效的聚类挖掘算法进行云端隐私数据挖掘操作.

References:

- [1] Ni WW, Chen G, Chong ZH, Wu YJ. Privacy-Preserving data publication for clustering. *Journal of Computer Research and Development*, 2012,49(5):1095–1104. (in Chinese with English abstract)
- [2] Ding Y, Wang HM, Shi PC, Wu QB, Dai HD, Fu HY. Trusted Cloud Service. *Chinese Journal of Computers*, 2015,38(1):133–149 (in Chinese with English abstract). [doi: 10.3724/SP.J.1016.2015.00133]
- [3] Wu XD, Zhu XQ, Wu GQ, Ding W. Data mining with big data. *IEEE Trans. on Knowledge and Data Engineering*, 2014,26(1): 97–107. [doi: 10.1109/TKDE.2013.109]
- [4] Agrawal R, Srikant R. Privacy-Preserving data mining. *ACM Sigmod Record*, 2000,29(2):439–450. [doi: 10.1145/342009.335438]
- [5] Zhou SG, Li F, Tao YF, Xiao XK. Privacy preservation in data applications: A survey. *Chinese Journal of Computers*, 2009,32(5): 847–861 (in Chinese with English abstract). [doi: 10.3724/SP.J.1016.2009.00847]
- [6] Mohammed N, Chen R, Fung BCM, Yu PS. Differentially private data release for data mining. In: *Proc. of the 17th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*. ACM Press, 2011. 493–501. [doi: 10.1145/2020408.2020487]
- [7] Allard T, Hébrail G, Massegli F, Pacitti E. Chiaroscuro: Transparency and privacy for massive personal time-series clustering. In: *Proc. of the 2015 ACM SIGMOD Int'l Conf. on Management of Data*. ACM Press, 2015. 779–794. [doi: 10.1145/2723372.2749453]
- [8] Vlachos M, Schneider J, Vassiliadis VG. On data publishing with clustering preservation. *ACM Trans. on Knowledge Discovery from Data (TKDD)*, 2015,9(3):23. [doi: 10.1145/2700403]
- [9] Giannotti F, Lakshmanan LVS, Monreale A, Pedreschi D, Wang H. Privacy-Preserving mining of association rules from outsourced transaction databases. *Systems Journal, IEEE*, 2013,7(3):385–395. [doi: 10.1109/JSYST.2012.2221854]
- [10] Fong PK, Weber-Jahnke JH. Privacy preserving decision tree learning using unrealized data sets. *IEEE Trans. on Knowledge and Data Engineering*, 2012,24(2):353–364. [doi: 10.1109/TKDE.2010.226]
- [11] Johnson A, Shmatikov V. Privacy-Preserving data exploration in genome-wide association studies. In: *Proc. of the 19th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining*. ACM Press, 2013. 1079–1087. [doi: 10.1145/2487575.2487687]
- [12] Zhang XY, Liu C, Nepal S, Pandey S, Chen JJ. A privacy leakage upper bound constraint-based approach for cost-effective privacy preserving of intermediate data sets in cloud. *IEEE Trans. on Parallel and Distributed Systems*, 2013,24(6):1192–1202. [doi: 10.1109/TPDS.2012.238]
- [13] Nguyen P. Cryptanalysis of the goldreich-goldwasser-halevi cryptosystem from crypto'97. In: *Proc. of the Annual Int'l Cryptology Conf. Berlin, Heidelberg: Springer-Verlag*, 1999. 288–304. [doi: 10.1007/3-540-48405-1_18]
- [14] Yi X, Zhang Y. Equally contributory privacy-preserving k -means clustering over vertically partitioned data. *Information Systems*, 2013,38(1):97–107. [doi: 10.1016/j.is.2012.06.001]
- [15] Oliveira SRM, Zaiane OR. Achieving privacy preservation when sharing data for clustering. In: *Proc. of the Workshop on Secure Data Management in Conjunction with VLDB. Berlin, Heidelberg: Springer-Verlag*, 2004. 67–82. [doi: 10.1007/978-3-540-30073-1_6]
- [16] Islam MZ, Brankovic L. Privacy preserving data mining: A noise addition framework using a novel clustering technique. *Knowledge-Based Systems*, 2011,24(8):1214–1223. [doi: 10.1016/j.knsys.2011.05.011]

- [17] Zhu Y, Liu L. Optimal randomization for privacy preserving data mining. In: Proc. of the 10th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. ACM Press, 2004. 761–766. [doi: 10.1145/1014052.1014153]
- [18] Chong ZH, Ni WW, Liu TT, Zhang Y. A privacy-preserving data publishing algorithm for clustering application. Journal of Computer Research and Development, 2010,47(12):2083–2089 (in Chinese with English abstract).
- [19] Li YP, Chen MH, Li QW, Zhang W. Enabling multilevel trust in privacy preserving data mining. IEEE Trans. on Knowledge and Data Engineering, 2012,24(9):1598–1612. [doi: 10.1109/TKDE.2011.124]
- [20] Navarro-Arribas G, Torra V, Erola A, Castella-Roca J. User k -anonymity for privacy preserving data mining of query logs. Information Processing & Management, 2012,48(3):476–487. [doi: 10.1016/j.ipm.2011.01.004]
- [21] Xiao X, Yi K, Tao Y. The hardness and approximation algorithms for l -diversity. In: Proc. of the 13th Int'l Conf. on Extending Database Technology. ACM Press, 2010. 135–146. [doi: 10.1145/1739041.1739060]
- [22] Dwork C. A firm foundation for private data analysis. Communications of the ACM, 2011,54(1):86–95. [doi: 10.1145/1866739.1866758]
- [23] Mohan P, Thakurta A, Shi E, Song D, Culler DE. GUPT: Privacy preserving data analysis made easy. In: Proc. of the 2012 ACM SIGMOD Int'l Conf. on Management of Data. ACM Press, 2012. 349–360. [doi: 10.1145/2213836.2213876]
- [24] Parameswaran R, Blough DM. Privacy preserving data obfuscation for inherently clustered data. Int'l Journal of Information and Computer Security, 2008,2(1):4–26. [doi: 10.1504/IJICS.2008.016819]
- [25] Nissim K, Raskhodnikova S, Smith A. Smooth sensitivity and sampling in private data analysis. In: Proc. of the 39th Annual ACM Symp. on Theory of Computing. ACM Press, 2007. 75–84. [doi: 10.1145/1250790.1250803]
- [26] Blake CL, Merz CJ. UCI repository of machine learning databases. Irvine: University of California, Department of Information and Computer Science, 1998. <http://www.ics.uci.edu/~mllearn/MLRepository.html>

附中文参考文献:

- [1] 倪巍伟,陈耿,崇志宏,吴英杰.面向聚类的数据隐藏发布研究.计算机研究与发展,2012,49(5):1095–1104.
- [2] 丁滢,王怀民,史佩昌,吴庆波,戴华东,富弘毅.可信云服务.计算机学报,2015,38(1):133–149. [doi: 10.3724/SP.J.1016.2015.00133]
- [5] 周水庚,李丰,陶宇飞,肖小奎.面向数据库应用的隐私保护研究综述.计算机学报,2009,32(5):847–861. [doi: 10.3724/SP.J.1016.2009.00847]
- [18] 崇志宏,倪巍伟,刘腾腾,张勇.一种面向聚类的隐私保护数据发布方法.计算机研究与发展,2010,47(12):2083–2089.



崔一辉(1981 -),男,陕西西安人,硕士,主要研究领域为应用密码学,云安全,隐私保护.



史成良(1994 -),男,学士,主要研究领域为应用密码学,云安全,隐私保护.



宋伟(1978 -),男,博士,副教授,主要研究领域为应用密码学,云安全,隐私保护.



程芳权(1985 -),男,博士,主要研究领域为应用密码学,云安全,隐私保护.



王占兵(1992 -),男,硕士,主要研究领域为应用密码学,云安全,隐私保护.