基于稀疏矩阵变换和有界随机扰动的K-Means聚类外包方案

赵韦[1]，谭静文[1]，王焕然[1]，韩帅[1]，杨武[1]，赖明珠[2]
（1.哈尔滨工程大学计算机科学与技术学院，黑龙江 哈尔滨 150001；2.海南师范大学数学与统计学院，海南 海口 571158）

摘　要：针对现有K-Means聚类安全外包方案计算和通信开销高，难以满足实际应用对高效率需求的问题，提出一种基于稀疏矩阵变换和有界随机扰动的隐私保护K-Means聚类外包方案。首先，利用Gram-Schmidt正交化构造稀疏密钥矩阵，实现对明文数据的高效正交变换，有效隐藏明文数据的数值特征；其次，引入服从高斯分布的有界随机扰动，保护明文数据点之间的距离信息，增强用户数据的安全性；最后，结合局部敏感哈希设计近似距离估计方法，在保证聚类准确的前提下降低外包方案的计算开销。理论分析表明，所提方案实现了正确性、安全性和高效性的设计目标。在多个真实数据集上的实验结果表明，相较于现有基于同态加密的K-Means聚类外包方案，所提方案在保持聚类准确的同时，显著降低了计算与通信开销。

关键词：K-Means聚类；矩阵变换；随机扰动；局部敏感哈希；外包计算；隐私保护
中图分类号：TP309
文献标志码：A
DOI：10.11959/j.issn.1000–436x.2026009

K-Means clustering outsourcing scheme based on sparse matrix transformation and bounded random perturbation

Zhao Wei[1], Tan Jingwen[1], Wang Huanran[1], Han Shuai[1], Yang Wu[1], Lai Mingzhu[2]
1. School of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China
2. School of Mathematics and Statistics, Hainan Normal University, Haikou 571158, China

Abstract: To address the problem that existing secure outsourcing schemes for K-Means clustering incur high computational and communication overhead, making them difficult to satisfy the efficiency requirements of practical applications, a privacy-preserving K-Means clustering outsourcing scheme based on sparse matrix transformation and bounded random perturbation was proposed. Firstly, a sparse key matrix was constructed by using Gram-Schmidt orthogonalization to perform efficient orthogonal transformations on plaintext data, effectively hiding the numerical characteristics of the plaintext data. Secondly, bounded random perturbations following a Gaussian distribution were introduced to protect the distance information between plaintext data points, enhancing the security of user data. Finally, an approximate distance estimation method was designed by combining locality sensitive hashing to reduce the computational overhead of the outsourcing scheme under the premise of ensuring clustering accuracy. Theoretical analysis demonstrates that the proposed scheme achieves the design goals of correctness, security and efficiency. Experimental results on multiple real-world datasets show that compared to existing K-Means clustering outsourcing schemes based on homomorphic encryption, the proposed scheme significantly reduces computational and communication overhead while maintaining clustering accuracy.

Keywords: K-Means clustering, matrix transformation, random perturbation, locality sensitive hashing, outsourcing computation, privacy-preserving

收稿日期：2025–11–26；**修回日期：**2026–01–10
通信作者：韩帅，hshuai@hrbeu.edu.cn
基金项目：国家自然科学基金资助项目（No.U22A2036, No.U21B2019, No.62272127, No.62572144）；黑龙江省自然科学基金资助项目（No.TD2022F001, No.LH2024F036）；海南省自然科学基金高层次人才基金资助项目（No.622RC672）
Foundation Items: The National Natural Science Foundation of China (No.U22A2036, No.U21B2019, No.62272127, No.62572144), The Natural Science Foundation of Heilongjiang (No.TD2022F001, No.LH2024F036), The Natural Science Foundation High-level Talents of Hainan (No.622RC672)

0　引言

K-Means聚类算法作为无监督机器学习领域的经典方法，因其原理简单、聚类性能优异而被广泛应用于数据挖掘[1]、图像分割[2]、网络异常检测[3]等诸多场景。其目标是将给定的数据集划分为k个簇，使同一簇内的数据点相似度高，而不同簇间的数据点相似度低[4]。由于该算法通常需要多轮迭代才能收敛至稳定解，K-Means算法在实际应用中面临显著的计算效率瓶颈[5]。在每一轮迭代中，算法需要计算所有数据点到当前各簇中心的欧氏距离[6]。例如，在处理包含100万条128维特征的数据集时，单次迭代需执行约1.28×10^{12}次浮点运算。即使采用高性能计算设备，聚类过程仍需数小时甚至更长时间，这导致资源受限的用户（如移动终端用户或物联网设备）难以独立完成大规模数据的聚类任务[7]。

云计算技术的兴起为破解用户大规模聚类难题开创了新范式[8]。用户通过将计算密集型的K-Means聚类任务外包给拥有强大计算能力的云服务提供商，可显著减轻用户本地的计算负担并加速聚类过程[9]。然而，云服务提供商并非完全可信[10]。若将包含敏感信息的用户数据直接上传至第三方云平台进行计算，会引发严重的隐私泄露风险[11]。因此，设计一种同时兼顾隐私保护和计算效率的K-Means聚类外包方案成为当前亟待解决的问题[12]。

现有研究主要采用差分隐私[13]和同态加密[14]来实现M的安全外包。差分隐私通过向聚类结果注入噪声，将单条记录的增删对聚类结果的影响控制在预设范围内，从而在输出层面提供隐私保障[15]。同态加密支持用户在本地加密数据后上传至云端，云服务器可以对密文执行K-Means聚类，用户解密密文聚类结果后获得与明文聚类相同的结果，实现数据的“可用不可见”[16]。

近年来，差分隐私在K-Means聚类分析中的应用取得了显著进展。针对聚类迭代过程中噪声累积导致聚类效果下降和收敛困难的问题，Li等[15]利用遗传算法优化每轮迭代的隐私预算分配策略，有效提升了聚类算法的实用性。针对隐私预算跨维度分配导致的精度下降问题，Yang等[17]提出了一种基于局部差分隐私的K-Means有界扰动方法。该方法通过对整体记录进行扰动消除了不同维度间的隐私预算分配需求，从而有效缓解了因预算稀释引发的数据效用下降问题。一些研究专注于实际场景下的隐私保护聚类任务需求[13,18]。在多方协作场景下，Zhang等[18]将差分隐私与安全多方计算结合，通过在簇中心更新时添加拉普拉斯噪声并使用Shamir秘密共享机制来保障数据隐私。Ravi等[13]探索了智能电网负载数据的隐私保护问题，通过向簇中心与数据标签添加离散高斯噪声，解决了噪声标签发布带来的隐私泄露风险。为了进一步提升聚类效果，Ni等[19]在每次迭代中向簇中心添加自适应拉普拉斯噪声，利用簇合并来抵消噪声带来的精度下降的影响。上述基于差分隐私的聚类方法主要通过向聚类结果添加噪声来防止个体敏感信息被推断。然而，这些方法依赖明文数据计算，在计算过程中存在泄露风险[20]。此外，这些方案主要对最终输出进行扰动，而聚类算法迭代过程中产生的中间结果可能成为攻击者推断原始数据的突破口[14]。因此，基于差分隐私的聚类方法无法为聚类全过程提供可证明的隐私保障。

同态加密支持在密文数据上执行聚类所需的计算操作，能够为聚类分析全过程提供隐私保障，成为隐私保护K-Means聚类的关键技术[12]。由于同态加密产生的密文不能保持数据点到聚类中心距离的顺序，Liu等[21]利用用户提供的陷门信息来比较加密后的距离。贾春福等[16]利用BGV同态加密[22]实现了聚类的安全外包，并设计了一种密文比较协议。在此基础上，Yuan等[23]融入MapReduce框架，实现密文数据点的相似性度量。一些工作探索了多用户和多云协作完成隐私保护聚类的应用。Ye等[24]采用高效的Paillier同态加密，Wu等[12]采用全同态加密和密文打包技术[25]，在不增加额外开销的情况下实现了并行计算。Tang等[26]提出了一种基于Paillier同态加密的分布式K-means聚类隐私保护算法，通过对加密的聚类中心和数值进行本地明文计算，从而减少了加密状态下的计算量。针对云服务器计算多方密文的难题，Zhang等[14]提出基于多密钥全同态加密[27]的隐私保护K-Means方案，支持对具有不同密钥的用户密文进行聚类分析。为了进一步减少用户的计算负担，Sakellariou等[28]引入可审计服务器的安全模型，将复杂操作卸载到云托管的服务器，但该模型依赖云和用户之间的多次交互，导致通信开销高。为了降低通信开销，Yang

等[29]利用支持密文距离度量的向量同态加密[30]设计了K-Means聚类外包方案，并将其应用于智能电网。然而，现有基于同态加密的聚类外包方法仍面临计算和通信开销大的挑战。昂贵的bootstrapping技术[31]和密文膨胀[32]增加了计算负担和通信成本，制约了基于同态加密的聚类外包方法在资源受限或大规模数据场景下的实际应用。

本文方案与现有K-Means聚类外包方法在技术实现和性能方面的对比如表1所示，从隐私保护范围、聚类精度、用户计算开销、通信开销以及主要不足等方面，对现有代表性的方案进行了系统总结。由表1可知，尽管当前K-Means聚类外包研究取得了较大进展，但仍面临两项关键挑战。一方面，基于差分隐私的方法仅能对最终输出结果提供隐私保护，难以实现聚类全过程的隐私保障，同时噪声添加机制不可避免地影响聚类精度。另一方面，基于同态加密的方法需要用户参与大量中间计算，频繁的交互导致计算和通信开销高昂，限制了其在外包场景中的应用。

为了解决上述挑战，本文提出了一种基于稀疏矩阵变换和有界随机扰动的隐私保护K-Means聚类外包方案。与基于同态加密的方案相比，该方案在确保聚类全过程在密文状态下完成的同时，仅需稀疏矩阵乘法与加法的加密操作和两轮通信，大幅降低了用户计算负担和通信开销，具有较强的实用价值。本文的主要贡献如下。

1)设计了一种新颖的隐私保护K-Means聚类外包方案，通过稀疏正交变换和有界随机扰动对数据加密，使在密文上执行K-Means聚类的结果与明文聚类结果一致，在保证聚类准确性的同时保护了用户数据隐私。用户只需本地执行轻量级的加密操作即可将聚类任务外包至云服务器，不需要参与聚类过程中的计算和通信，显著减轻了用户本地的计算和通信负担。

2) 针对添加扰动会破坏密文域中数据点间欧氏距离的相对顺序从而导致聚类准确性降低的问题，本文推导了确保距离顺序所需添加噪声标准差上界的充要条件，该上界与数据集中数据点之间的距离最小差值密切相关。为了进一步降低用户获取该最小差值的开销，设计了一种基于局部敏感哈希的近似距离计算方法，在保护数据间距离信息的同时避免了用户逐一计算数据集中所有点对距离。

3) 通过理论分析，证明了本文方案满足准确性、高效性和安全性的设计目标。基于4个真实数据集的实验结果表明，该方案的密文聚类结果与明文聚类结果完全一致。此外，用户使用本文外包方案可将本地计算效率提高66.67%以上，在计算和通信开销方面均优于当前性能最佳的基于同态加密的K-Means聚类方法。

1 模型和设计目标

1.1 系统模型

系统模型如图1所示，本文的系统模型包括两个实体：拥有海量数据的用户，以及拥有丰富计算资源的云服务提供商，即云服务器。

1) 用户：明文数据集被存储在用户端。用户负责生成加密所需的密钥，并利用密钥对本地明文数据集进行加密。加密后，用户将密文数据集发送给云服务器。对于云服务器返回的密文聚类结果，用户使用本地保存的密钥进行解密，从而获得真实的聚类结果。

表1 本文方案与现有K-Means聚类外包方法在技术实现和性能方面的对比

方案	主要技术	隐私保护范围	聚类精度	用户计算开销	通信开销	主要不足
文献[13]	差分隐私	输出结果	有损	低	低	仅保护输出结果，聚类过程存在信息泄露风险
文献[15]	差分隐私	输出结果	有损	低	低	仅保护输出结果，聚类过程存在信息泄露风险
文献[17]	局部差分隐私	输出结果	有损	低	低	聚类精度有损，聚类过程存在信息泄露风险
文献[14]	多密钥全同态加密	计算全过程	无损	高	高	密文膨胀严重，计算与通信开销高
文献[16]	BGV全同态加密	计算全过程	无损	高	高	密文膨胀严重，计算与通信开销高
文献[26]	Paillier同态加密	计算全过程	无损	高	高	用户需参与聚类过程计算，计算与通信开销高
文献[29]	向量同态加密	计算全过程	无损	高	低	加解密过程涉及大量矩阵运算，计算开销高
本文方案	矩阵变换/有界扰动	计算全过程	无损	低	低	—

2) 云服务器：云服务器仅接收和处理密文。收到用户发送的密文数据集后对其执行K-Means聚类分析，然后将密文聚类结果返回给用户。在整个过程中，云服务器无法访问密钥或明文信息。

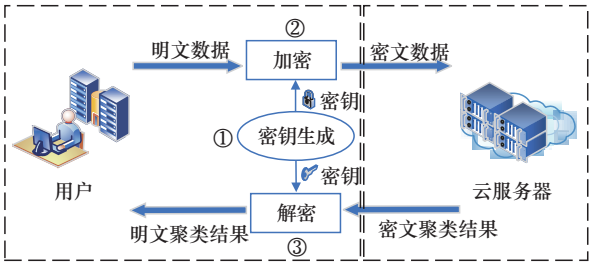

图1　系统模型

1.2　威胁模型

云服务器对于用户来说并不可信。本文假设云服务器的威胁模型是半诚实的[33]，也称为“诚实但好奇”的云服务器模型。在该模型中，云服务器虽然会遵循既定的协议来执行聚类计算，但出于经济利益等动机，云服务器会试图从用户的数据中获取敏感信息。

在本文的威胁模型中，云服务器的攻击目标主要包括：推断用户原始数据的数值特征或统计分布，恢复数据点之间的真实距离关系，破解加密机制以获取密钥或重构明文。关于攻击者的背景知识，本文假设云服务器了解外包方案的所有细节，包括稀疏正交矩阵的构造方法、有界随机扰动的添加机制以及K-Means聚类的计算流程和参数设置。此外，云服务器可以访问用户上传的全部密文数据，以及聚类计算过程中产生的所有中间结果。考虑到现实云环境中可能存在数据泄露或辅助信息获取的情况，本文进一步假设云服务器可能获得部分明文数据及其对应的密文，从而拥有有限数量的明文–密文对。

在上述假设条件下，基于已知明文攻击（known-plaintext attack,KPA）模型[34-35]对本文方案的安全性进行分析。在该模型下，攻击者掌握全部密文、有限数量的明文–密文对及所有中间计算结果，并试图破解密钥或恢复剩余明文数据。云服务器具有多项式时间的计算能力，可以执行离线分析和统计推断。

需要说明的是，本文不考虑云服务器主动偏离协议的恶意行为，例如，篡改聚类计算过程、返回错误结果或实施拒绝服务攻击，同时也不考虑侧信道攻击和物理攻击等超出计算模型范围的威胁。这类攻击可通过可验证计算或可信执行环境等技术进行防护，不属于本文的研究范畴。

1.3　设计目标

在K-Means聚类外包计算场景中，用户通过将计算任务委托给云服务器以减轻本地计算负担，但直接外包明文数据会导致敏感信息泄露。一个理想的安全外包方案必须在保障数据安全的前提下，同时确保聚类结果的准确性与外包过程的计算效率。

现有的K-Means聚类外包方法难以同时满足隐私保护、聚类精度和外包效率的需求。一些方案仅对聚类输出结果提供隐私保护，难以实现聚类全过程的安全保障；另一些方案虽然能够在密文状态下完成聚类计算，但用户需参与大量中间计算与交互，导致计算与通信开销较高，从而削弱了外包计算的实际意义。针对上述问题，本文方案的设计目标主要包括以下3个方面。

1) 准确性。使用外包方案对密文数据集进行K-Means聚类所获得的结果和明文聚类结果相同，避免外包方案对聚类精度产生影响。

2) 安全性。外包方案可以在整个聚类过程中保护用户数据的隐私，防止云服务器获取用户的原始样本数据和样本间的距离信息。本文在已知明文攻击模型下对方案的安全性进行理论分析。

3) 高效性。对用户而言，使用外包方案的计算开销要小于在明文数据下执行K-Means聚类的计算开销，从而降低用户的本地计算负担。

2　预备知识

2.1　K-Means聚类算法

K-Means算法是一种基于划分的聚类算法，通过迭代优化簇内数据点的距离误差来最小化数据点与各自簇中心的距离。K-Means算法的主要参数是最终聚类数目k，其目标是将n个数据点划分为k个簇，使簇内数据点到所属簇中心的距离最小。算法从随机初始化k个簇中心开始，重复执行数据分配与簇中心更新两个过程，直到聚类结果收敛或达到最大迭代次数。K-Means聚类算法的时间复杂度为$O(tknm)$，其中n是数据点的数量，m是特征维度，k是簇数，t是迭代次数。具体的算法步骤如下。

步骤1 从数据集D中随机选择k个数据点作为初始簇中心。

步骤2 对于数据集中每个数据点，计算它到各个簇中心的距离，并将其分配到距离最近的簇中心对应的簇中。

步骤3 对于每个簇，计算该簇中所有数据点的均值向量，并将该均值向量作为新的簇中心。

步骤4 重复步骤2和步骤3，直到簇中心的变化小于特定阈值或者达到最大迭代次数，算法结束。

2.2 混沌系统

混沌系统[36]是指在一个非线性动力系统中，出现随机且不规律的运动，其特点包括不可预测性、不可重复性和高度敏感性。混沌系统高度依赖初始条件的设置。Logistic映射是最典型的混沌系统之一，其定义为

$$U_{t+1}=\alpha U_t\left(1-U_t\right) \tag{1}$$

其中，$\alpha\in(0,4]$，$U_t\in[0,1]$，$t=0,1,2,\cdots,\infty$。当$3.569\,9<\alpha\leqslant 4$时，这个系统是混沌的。

混沌系统的输出类似于随机噪声，具有类噪声特性[37]。随机输出完全由U_t和α决定，其输出结果是确定的。由于混沌映射具有伪随机性，生成的输出值落在具有无限空间的实数域内，因此可以用来生成密钥。本文使用Logistic混沌映射生成密钥，并在此基础上构造稀疏正交矩阵用于加密数据集。

2.3 Gram-Schmidt正交化

Gram-Schmidt正交化是一种将一组线性无关的向量转换为一组正交向量的方法。设$\alpha_1,\cdots,\alpha_m$是$\mathbb{R}^n$中的一组线性无关的向量，令

$$\begin{aligned}\beta_1&=\alpha_1\\ \beta_2&=\alpha_2-\frac{\left\langle\alpha_2,\beta_1\right\rangle}{\left\langle\beta_1,\beta_1\right\rangle}\beta_1\\ &\vdots\\ \beta_m&=\alpha_m-\frac{\left\langle\alpha_m,\beta_1\right\rangle}{\left\langle\beta_1,\beta_1\right\rangle}-\cdots-\frac{\left\langle\alpha_m,\beta_{m-1}\right\rangle}{\left\langle\beta_{m-1},\beta_{m-1}\right\rangle}\beta_{m-1}\end{aligned} \tag{2}$$

则$\beta_1,\cdots,\beta_m$构成一组正交向量。令$e_i=\dfrac{\beta_i}{\left\|\beta_i\right\|}$（$i=1,2,\cdots,m$），则$e_1,\cdots,e_m$是一组标准正交向量且与$\alpha_1,\cdots,\alpha_m$等价。上述从线性无关向量组$\alpha_1,\cdots,\alpha_m$得到正交向量组$\beta_1,\cdots,\beta_m$的过程称为Gram-Schmidt正交化。

3 方案设计

3.1 方案概述

给定一个明文数据集D，它包含n个样本数据和m个属性特征，则该数据集可以表示为矩阵$\boldsymbol{X}\in\mathbb{R}^{n\times m}$。用户为了保护数据隐私需要先将明文矩阵$\boldsymbol{X}$加密，然后将密文矩阵发送给云服务器执行K-Means聚类分析任务。

为了保护数据集D中的敏感数据同时提高聚类的计算效率，本文基于稀疏矩阵变换和有界随机扰动的隐私保护K-Means聚类外包方案，以确保在密文数据下K-Means聚类的准确性。方案的核心模块包括稀疏正交矩阵构造和有界随机扰动添加。稀疏正交矩阵构造模块用于生成一个稀疏的正交矩阵，基于该正交矩阵加密明文数据集，以保持数据点间的距离不变。有界随机扰动添加模块通过添加服从高斯分布的有界噪声确保距离大小顺序不变，保护数据点之间的距离隐私。在该模块中，通过引入局部敏感哈希计算所要添加的噪声的上界来降低计算开销。本文方案通过稀疏矩阵乘法和加法实现对整个数据集的加密，保证了明文数据的隐私和聚类结果的准确性，有效提高了K-Means聚类外包的计算效率。

3.2 稀疏正交矩阵构造

为了确保密文空间中聚类结果的准确性，本文利用Gram-Schmidt正交化生成正交矩阵，对明文矩阵$\boldsymbol{X}$进行正交变换以保证数据点间的欧氏距离不变，从而使每个数据点被分配到与明文聚类相同的簇。然而，生成的矩阵通常是稠密矩阵$\boldsymbol{W}$，若利用生成的稠密矩阵$\boldsymbol{W}$对明文矩阵$\boldsymbol{X}$进行加密，则需要计算$\boldsymbol{XW}$。由于$\boldsymbol{X}$和$\boldsymbol{W}$均是稠密矩阵，其矩阵乘法的时间复杂度为$O(n^3)$。为了提高方案的计算效率，本文设计了随机稀疏正交矩阵用于保护明文数据集的隐私。

随机稀疏正交矩阵的构造过程如算法1所示。算法1通过Gram-Schmidt正交化构造了$\left\lfloor\dfrac{m}{2}\right\rfloor$个2×2的正交矩阵（对应算法步骤1)~步骤6)）和一个3×3的正交矩阵（对应算法步骤7)~步骤11)）。若明文数据集的特征数m为偶数，则稀疏正交矩阵$\boldsymbol{W}$的对角线位置是$\dfrac{m}{2}$个2×2的正交矩阵，其余位置的元素均为0，即$\boldsymbol{W}=\mathrm{diag}(\boldsymbol{W}_1,\boldsymbol{W}_2,\cdots,\boldsymbol{W}_{\frac{m}{2}})$，其中$\mathrm{diag}(\cdot)$表示对角矩阵构造函数，用于将给定的2×2的正交矩

阵依次置于矩阵 $\boldsymbol{W}$ 的对角线位置。(对应算法步骤12)和步骤13))。若明文数据集的属性个数 m 为奇数，则稀疏正交矩阵 $\boldsymbol{W}$ 的对角线位置是 $\frac{m-3}{2}$ 个2×2的正交矩阵和一个3×3的正交矩阵，其余位置的元素均为0(对应算法步骤14)~步骤16))。算法1的核心操作是构造多个2×2或3×3的小规模正交矩阵。由于每个小规模正交矩阵的维度是固定的，其生成过程可通过Gram-Schmidt正交化完成，所涉及的向量内积和归一化操作次数均为常数，因此构造一个2×2或3×3的正交矩阵的时间复杂度为 $O(1)$。随着特征维度 m 的增加，需要构造的小规模正交矩阵的数量与 m 呈线性关系，因此算法1的时间复杂度为 $O(m)$。

算法1　稀疏正交矩阵构造算法

输入　明文矩阵 $\boldsymbol{X}$ 的维度 m

输出　稀疏正交矩阵 $\boldsymbol{W}\in\mathbb{R}^{m\times m}$

1)for i=1 to $\left\lfloor\frac{m}{2}\right\rfloor$

2)　随机生成一组线性无关的向量 $\boldsymbol{a}_{i1},\boldsymbol{a}_{i2}\in\mathbb{R}^2$

3)　$\boldsymbol{u}_{i1}=\boldsymbol{a}_{i1},\boldsymbol{w}_{i1}=\frac{\boldsymbol{u}_{i1}}{\|\boldsymbol{u}_{i1}\|}$

4)　$\boldsymbol{u}_{i2}=\boldsymbol{a}_{i2}-\frac{\langle\boldsymbol{a}_{i2},\boldsymbol{u}_{i1}\rangle}{\langle\boldsymbol{u}_{i1},\boldsymbol{u}_{i1}\rangle}\boldsymbol{u}_{i1},\boldsymbol{w}_{i2}=\frac{\boldsymbol{u}_{i2}}{\|\boldsymbol{u}_{i2}\|}$

5)　构造正交矩阵 $\boldsymbol{W}_i=\left[\boldsymbol{w}_{i1},\boldsymbol{w}_{i2}\right]\in\mathbb{R}^{2\times2}$，其中 $i=1,\cdots,\left\lfloor\frac{m}{2}\right\rfloor$

6) end for

7) 随机生成一组线性无关的向量 $\boldsymbol{a}_1,\boldsymbol{a}_2,\boldsymbol{a}_3\in\mathbb{R}^3$

8) $\boldsymbol{u}_1=\boldsymbol{a}_1,w_1=\frac{\boldsymbol{u}_1}{\|\boldsymbol{u}_1\|}$

9) $\boldsymbol{u}_2=\boldsymbol{a}_2-\frac{\langle\boldsymbol{a}_2,\boldsymbol{u}_1\rangle}{\langle\boldsymbol{u}_1,\boldsymbol{u}_1\rangle}\boldsymbol{u}_1,\boldsymbol{w}_2=\frac{\boldsymbol{u}_2}{\|\boldsymbol{u}_2\|}$

10) $\boldsymbol{u}_3=\boldsymbol{a}_3-\frac{\langle\boldsymbol{a}_2,\boldsymbol{u}_1\rangle}{\langle\boldsymbol{u}_1,\boldsymbol{u}_1\rangle}\boldsymbol{u}_1-\frac{\langle\boldsymbol{a}_3,\boldsymbol{u}_2\rangle}{\langle\boldsymbol{u}_2,\boldsymbol{u}_2\rangle}\boldsymbol{u}_2,w_3=\frac{u_3}{\|u_3\|}$

11) 构造正交矩阵 $\boldsymbol{W}_{\left\lfloor\frac{m}{2}\right\rfloor+1}=\left[\boldsymbol{w}_1,\boldsymbol{w}_2,\boldsymbol{w}_3\right]\in\mathbb{R}^{3\times3}$

12) if $\frac{m}{2}$=0

13)　$\boldsymbol{W}$=diag($\boldsymbol{W}_1,\boldsymbol{W}_2,\cdots,\boldsymbol{W}_{\frac{m}{2}}$)

14) else

15)　$\boldsymbol{W}$=diag($\boldsymbol{W}_1,\boldsymbol{W}_2,\cdots,\boldsymbol{W}_{\left\lfloor\frac{m}{2}\right\rfloor+1}$)

16) end if

17) 输出矩阵 $\boldsymbol{W}$

定理1　密钥矩阵 $\boldsymbol{W}$ 是一个正交矩阵。

证明　根据正交矩阵的定义，若 $\boldsymbol{W}\boldsymbol{W}^{\mathrm{T}}=\boldsymbol{I}$，则 $\boldsymbol{W}$ 为正交矩阵。因此，证明 $\boldsymbol{W}$ 右乘它的转置等于单位矩阵即可。

根据算法1，可知矩阵 $\boldsymbol{W}$ 的对角线位置为通过Gram-Schmidt正交化产生的2×2或3×3的正交矩阵。因此，矩阵 $\boldsymbol{W}$ 可以表示为

$$\boldsymbol{W}=\begin{pmatrix}\boldsymbol{w}_1 & & & \\ & \boldsymbol{w}_2 & & \\ & & \ddots & \\ & & & \boldsymbol{w}_{\left\lfloor\frac{m}{2}\right\rfloor}\end{pmatrix} \tag{3}$$

其中，$\boldsymbol{w}_1,\boldsymbol{w}_2,\cdots,\boldsymbol{w}_{\left\lfloor\frac{m}{2}\right\rfloor}$ 为预计算中产生的2×2或3×3的正交矩阵，于是

$$\boldsymbol{W}\boldsymbol{W}^{\mathrm{T}}=\begin{pmatrix}\boldsymbol{w}_1 & & & \\ & \boldsymbol{w}_2 & & \\ & & \ddots & \\ & & & \boldsymbol{w}_{\left\lfloor\frac{m}{2}\right\rfloor}\end{pmatrix}\begin{pmatrix}\boldsymbol{w}_1^{\mathrm{T}} & & & \\ & \boldsymbol{w}_2^{\mathrm{T}} & & \\ & & \ddots & \\ & & & \boldsymbol{w}_{\left\lfloor\frac{m}{2}\right\rfloor}^{\mathrm{T}}\end{pmatrix}=$$
$$\begin{pmatrix}\boldsymbol{w}_1\boldsymbol{w}_1^{\mathrm{T}} & & & \\ & \boldsymbol{w}_2\boldsymbol{w}_2^{\mathrm{T}} & & \\ & & \ddots & \\ & & & \boldsymbol{w}_{\left\lfloor\frac{m}{2}\right\rfloor}\boldsymbol{w}_{\left\lfloor\frac{m}{2}\right\rfloor}^{\mathrm{T}}\end{pmatrix}=\begin{pmatrix}\boldsymbol{I} & & & \\ & \boldsymbol{I} & & \\ & & \ddots & \\ & & & \boldsymbol{I}\end{pmatrix}=\boldsymbol{I} \tag{4}$$

因此，密钥矩阵 $\boldsymbol{W}$ 为正交矩阵。证毕。

定理2　正交变换能够保持变换前后向量之间的欧氏距离不变。

证明　设 f 是空间 V 上的正交变换，对于任意向量 $\boldsymbol{a}$，$\boldsymbol{b}\in V$，有

$$\|f(\boldsymbol{a})-f(\boldsymbol{b})\|^2=\|f(\boldsymbol{a}-\boldsymbol{b})\|^2=f(\boldsymbol{a}-\boldsymbol{b})f(\boldsymbol{a}-\boldsymbol{b})=(\boldsymbol{a}-\boldsymbol{b})(\boldsymbol{a}-\boldsymbol{b})=\|\boldsymbol{a}-\boldsymbol{b}\|^2 \tag{5}$$

因此，经过正交变换后与变换前的向量之间的距离相等。证毕。

3.3　有界随机扰动添加

虽然正交变换能够隐藏明文数据集 D 的数值特征，但也泄露了数据点之间的距离信息。假设经过矩阵变换后的数据集为 X'，一种保护距离信息的方法是将 X' 乘以一个随机数 γ，这样距离也变化了 γ 倍。然而，攻击者可以通过求距离的最大公约数获

得γ。为了保护数据点之间的距离隐私，本文通过向X'中添加可控的高斯噪声，实现对距离信息的扰动处理。

K-Means聚类划分簇的依据是数据点与簇中心之间的相对距离顺序，即与簇中心距离最近的数据点划分到该簇中。因此，只要添加的噪声能够保持密文空间中数据点到簇中心距离的大小顺序不变，就能够保证聚类结果的准确性。由于簇中心对于用户来说是未知的，本文通过定理3，将点到簇中心的距离转化为点到簇内其他点的距离之和。

定理3 设$\boldsymbol{x}_i'$为矩阵变换后的数据点（即向量），对于任意簇S_j，存在常数C使

$$\|\boldsymbol{x}_i' - \boldsymbol{c}_j\|^2 = \frac{1}{|S_j|}\sum_{\boldsymbol{x}_k' \in S_j}\|\boldsymbol{x}_i' - \boldsymbol{x}_k'\|^2 - C \tag{6}$$

其中，$|S_j|$表示第j个簇S_j中包含的数据点个数，$\boldsymbol{x}_k'$是簇S_j中的数据点，$\boldsymbol{c}_j = \frac{1}{|S_j|}\sum_{\boldsymbol{x}_k' \in S_j}\boldsymbol{x}_k'$为簇中心，常数$C = \frac{1}{|S_j|}\sum_{\boldsymbol{x}_k' \in S_j}\left\|\boldsymbol{x}_k'\right\|^2 - \left\|\boldsymbol{c}_j\right\|^2$。

证明 经过矩阵变换后，数据点$\boldsymbol{x}_i'$与簇中心$\boldsymbol{c}_j$之间的距离平方可以表示为

$$\left\|\boldsymbol{x}_i' - \boldsymbol{c}_j\right\|^2 = \left\|\boldsymbol{x}_i'\right\|^2 - 2\boldsymbol{x}_i'^{\mathrm{T}}\boldsymbol{c}_j + \left\|\boldsymbol{c}_j\right\|^2 \tag{7}$$

由于簇中心为簇内数据点的均值$\boldsymbol{c}_j = \frac{1}{|S_j|}\sum_{\boldsymbol{x}_k' \in S_j}\boldsymbol{x}_k'$，将$\boldsymbol{c}_j$代入式(8)中可得

$$2\boldsymbol{x}_i'^{\mathrm{T}}\boldsymbol{c}_j = \frac{1}{|S_j|}\sum_{\boldsymbol{x}_k' \in S_j}2\boldsymbol{x}_i'^{\mathrm{T}}\boldsymbol{x}_k' \tag{8}$$

数据点$\boldsymbol{x}_i'$与簇S_j中其他数据点之间的距离平方之和为

$$\sum_{\boldsymbol{x}_k' \in S_j}\left\|\boldsymbol{x}_i' - \boldsymbol{x}_k'\right\|^2 = |S_j|\left\|\boldsymbol{x}_i'\right\|^2 - 2\boldsymbol{x}_i'^{\mathrm{T}}\sum_{\boldsymbol{x}_k' \in S_j}\boldsymbol{x}_k' + \sum_{\boldsymbol{x}_k' \in S_j}\left\|\boldsymbol{x}_k'\right\|^2 \tag{9}$$

式(9)两边同时除以$|S_j|$，可得

$$\frac{1}{|S_j|}\sum_{\boldsymbol{x}_k' \in S_j}\left\|\boldsymbol{x}_i' - \boldsymbol{x}_k'\right\|^2 = \left\|\boldsymbol{x}_i'\right\|^2 - 2\boldsymbol{x}_i'^{\mathrm{T}}\boldsymbol{c}_j + \frac{1}{|S_j|}\sum_{\boldsymbol{x}_k' \in S_j}\left\|\boldsymbol{x}_k'\right\|^2 \tag{10}$$

综上所述，数据点$\boldsymbol{x}_i'$与簇中心$\boldsymbol{c}_j$之间的距离平方满足以下关系，即

$$\begin{aligned}&\left\|\boldsymbol{x}_i' - \boldsymbol{c}_j\right\|^2 = \\ &\left\|\boldsymbol{x}_i'\right\|^2 - 2\boldsymbol{x}_i'^{\mathrm{T}}\boldsymbol{c}_j + \frac{1}{|S_j|}\sum_{\boldsymbol{x}_k' \in S_j}\left\|\boldsymbol{x}_i'\right\|^2 - \\ &\frac{1}{|S_j|}\sum_{\boldsymbol{x}_k' \in S_j}\left\|\boldsymbol{x}_i'\right\|^2 + \left\|\boldsymbol{c}_j\right\|^2 = \\ &\frac{1}{|S_j|}\sum_{\boldsymbol{x}_k' \in S_j}\left\|\boldsymbol{x}_i' - \boldsymbol{x}_k'\right\|^2 - C\end{aligned} \tag{11}$$

证毕。

根据定理3，得到如下推论。

推论1 对于数据集X'，若数据点$\boldsymbol{x}_i'$满足$\sum_{\boldsymbol{x}_k' \in S_j}\left\|\boldsymbol{x}_i' - \boldsymbol{x}_k'\right\|^2 < \sum_{\boldsymbol{x}_k' \in S_l}\left\|\boldsymbol{x}_i' - \boldsymbol{x}_k'\right\|^2 + |S_j|C_j - |S_l|C_l$，则$\left\|\boldsymbol{x}_i' - \boldsymbol{c}_j\right\|^2 < \left\|\boldsymbol{x}_i' - \boldsymbol{c}_l\right\|^2$。

定理3和推论1表明，数据点到簇中心的距离平方$\left\|\boldsymbol{x}_i' - \boldsymbol{c}_j\right\|^2$可以由该点到簇内所有点的距离平方和$\sum_{\boldsymbol{x}_k' \in S_j}\left\|\boldsymbol{x}_i' - \boldsymbol{x}_k'\right\|^2$线性表示。这意味着在聚类任务中，质心距离的比较可以等价转化为点对距离和的比较，不需要显式计算簇中心，仅通过点对距离即可隐式反映质心距离关系。根据这种关系，即使在簇中心未知的情况下，仍能通过添加噪声扰动，保持加密数据中距离和的相对顺序，从而保证聚类结果的正确性。

为了确保添加噪声后数据点间的距离顺序保持不变，需要严格约束噪声的范围。本文将通过理论推导建立噪声上界与距离顺序保持条件之间的定量关系，以确保聚类结果的准确性。设经过矩阵变换后的两个向量$\boldsymbol{x}_i'$和$\boldsymbol{x}_j'$之间的欧氏距离为$d_{ij} = \left\|\boldsymbol{x}_i' - \boldsymbol{x}_j'\right\|$，则添加噪声后的距离为$d_{ij}' = \left\|\left(\boldsymbol{x}_i + \boldsymbol{e}_i\right) - \left(\boldsymbol{x}_j + \boldsymbol{e}_j\right)\right\| = \left\|\boldsymbol{x}_i - \boldsymbol{x}_j + \left(\boldsymbol{e}_i - \boldsymbol{e}_j\right)\right\|$，其中，$\boldsymbol{e}_i$和$\boldsymbol{e}_j$分别为对$\boldsymbol{x}_i'$和$\boldsymbol{x}_j'$添加的噪声向量。对于任意两对数据点$(\boldsymbol{x}_i, \boldsymbol{x}_j)$和$(\boldsymbol{x}_k, \boldsymbol{x}_l)$，若$d_{ij} > d_{kl}$，则需保证$d'_{ij} > d'_{kl}$。根据三角不等式，距离和噪声向量之间满足

$$\begin{aligned}\left|d_{ij}' - d_{ij}\right| &\leqslant \left\|\boldsymbol{e}_i - \boldsymbol{e}_j\right\| \\ \left|d_{kl}' - d_{kl}\right| &\leqslant \left\|\boldsymbol{e}_k - \boldsymbol{e}_l\right\|\end{aligned} \tag{12}$$

因此，保持距离大小顺序的充要条件是

$$\left\|\boldsymbol{e}_i - \boldsymbol{e}_j\right\| + \left\|\boldsymbol{e}_k - \boldsymbol{e}_l\right\| < d_{ij} - d_{kl} \tag{13}$$

基于上述充要条件，将距离顺序保持问题转化为对噪声添加范围的约束问题。该转化涉及两个关键因素：1)保证扰动后的距离差不超过原始距离差；2)基于矩阵变换后空间X'的特性确定噪声的最大允许强度。为此，引入噪声范数上界ε和矩阵变换后空间的最小点对距离差$\delta_{\min}$，通过建立ε和$\delta_{\min}$之间的定量关系并结合高斯噪声的统计特性，来推导噪声标准差σ的理论上界。

令ε为噪声向量的范数上界，即对于任意i,j有$\left|\boldsymbol{e}_i - \boldsymbol{e}_j\right| \leqslant \varepsilon$，则$2\varepsilon \leqslant d_{ij} - d_{kl}$。令$\delta_{\min}$为矩阵变换后空间中所有点对距离差的最小绝对值，即$\delta_{\min}=\min\left\{\left|d_{ij} - d_{kl}\right|\right\}$。那么，保持距离大小顺序的充要条件是$\varepsilon < \dfrac{\delta_{\min}}{2}$。基于高斯噪声的三西格玛定律，对于$\boldsymbol{e}_i \sim N(0,\sigma^2 I)$，有$\boldsymbol{e}_i - \boldsymbol{e}_j \sim N(0,\sigma^2 I)$，其范数上界满足$P\left(\left\|\boldsymbol{n}_i - \boldsymbol{n}_j\right\| \leqslant 3\sqrt{2}\,\sigma\right) \approx 99.73\%$。为了确保$\varepsilon = 3\sqrt{2}\sigma \leqslant \dfrac{\delta_{\min}}{2}$成立，噪声标准差须满足$\sigma \leqslant \dfrac{\delta_{\min}}{6\sqrt{2}}$。因此，只要计算出$\delta_{\min}$，即可确定最大允许的噪声强度$\sigma$。然而，直接计算矩阵变换后空间$X'$中所有数据点对的距离差值$\delta_{\min}$需要枚举$O(n^2)$对点，时间复杂度为$O(n^2 d)$，这对于大规模数据集的计算代价过高，从而失去外包的意义。为此，本文基于局部敏感哈希设计了一种近似计算方法，在容忍可控误差的前提下高效估算$\delta_{\min}$。其核心思想是通过哈希函数将距离较近的数据点以高概率映射到相同的桶中，而距离较远的数据点映射到不同的桶中，从而快速筛选候选近邻点对，避免全局计算。基于局部敏感哈希计算$\delta_{\min}$的步骤如下。

1) 选择局部敏感哈希函数族

对于欧氏距离，选择p-稳定分布哈希函数

$$h(\boldsymbol{x}') = \left\lfloor \frac{\boldsymbol{a}\boldsymbol{x}' + b}{w} \right\rfloor \tag{14}$$

其中，$\boldsymbol{a}$是各分量独立采样自标准正态分布的随机向量，b是均匀随机偏移，w是桶宽，用于控制哈希粒度。

2) 构建哈希表

创建l个独立的哈希表，每个表使用一组独立的哈希函数$g_i = \left(h_1, h_2, \cdots, h_k\right)$。对于数据集中的每个数据点$\boldsymbol{x}'$，计算其在所有$l$个哈希表中的桶编号$g_i(\boldsymbol{x}')$，并存入对应的桶。

3) 查询近似最小距离

对于每个数据点$\boldsymbol{x}'$，在所有的哈希表中找到与$\boldsymbol{x}'$同桶的候选点集合C，计算$\boldsymbol{x}$与C中所有点的真实距离，记录这些距离中的最小值$d_{\min}$。对所有点重复此过程，最终取全局最小值$\delta'_{\min} = \min\{\, d_{\min}(\boldsymbol{x}') \mid \boldsymbol{x}' \in \boldsymbol{X}' \}$作为$\delta_{\min}$的估计值。

3.4　方案整体流程

针对数据集D即矩阵$\boldsymbol{X}$的安全性保护，方案主要包括3个步骤，分别是密钥生成、数据加密和云计算。方案流程如图2所示，对于明文数据集，用户依次执行密钥生成与数据加密，随后将密文数据集和参数k外包至云服务器进行K-Means聚类计算，最终获取聚类结果。以下是对3个步骤的细节描述。

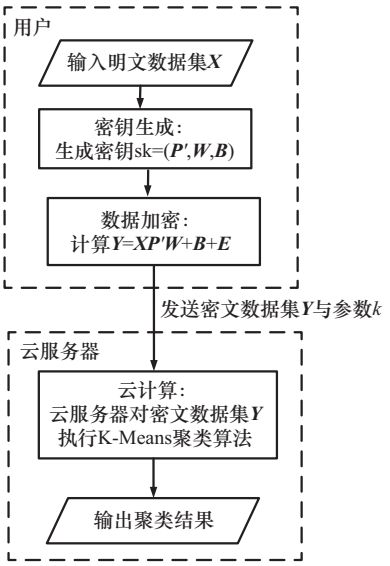

图2　方案流程

步骤 1　密钥生成。首先，用户生成随机矩阵$\boldsymbol{P}$，矩阵$\boldsymbol{P}$的对角线元素为−1或1，其余位置的元素为0，用Fisher-Yates洗牌算法[38]将矩阵$\boldsymbol{P}$按行或者按列进行随机排序，得到密钥矩阵$\boldsymbol{P}'$（对应算法步骤1~8）。然后选择一个混沌Logistic映射$U_{t+1}=\alpha U_t(1-U_t)$，其中$3.569\,9<\alpha\leqslant 4$，通过该Logistic映射产生一套伪随机数$\{\beta_1, \beta_2, \cdots, \beta_m\}$。令向量$\boldsymbol{\beta}=[\beta_1, \beta_2, \cdots, \beta_m]$，复制$n$个$\boldsymbol{\beta}$得到矩阵$\boldsymbol{B}\in\mathbb{R}^{n\times m}$（对应算法步骤9)~步骤13)）。最后通过算法1生成一个随机的稀疏正交矩阵$\boldsymbol{W}$（对应算法2步骤14)）。综上所述，密钥sk=($\boldsymbol{P}'$,$\boldsymbol{W}$,$\boldsymbol{B}$)。密钥生成的细节如算法2所示。算法2的计算开销主要由以下3个部分组成：1)生成置换矩阵$\boldsymbol{P}'$，由于该矩阵仅包含m个非零元

素，其置换操作可在线性时间内完成，时间复杂度为$O(m)$；2)通过Logistic混沌映射生成m个伪随机数，用于构造矩阵$\boldsymbol{B}$，时间复杂度为$O(m)$；3)调用算法1生成稀疏正交矩阵$\boldsymbol{W}$，时间复杂度为$O(m)$。因此，算法2的整体时间复杂度为$O(m)$。

算法2 密钥生成算法

输入 矩阵$\boldsymbol{X}$的维度n和m

输出 密钥sk=($\boldsymbol{P}'$,$\boldsymbol{W}$,$\boldsymbol{B}$)

1)生成一个全零矩阵$\boldsymbol{P}\in\mathbb{R}^{m\times m}$

2) for i=1 to m

3) for j=1 to m

4) if i=j

5) 矩阵$\boldsymbol{P}$中的元素p_{ij}=−1或1

6) end if

7) end for

8) end for

9) 利用Fisher-Yates洗牌算法随机排列矩阵$\boldsymbol{P}$的行顺序或列顺序得到置换后的矩阵$\boldsymbol{P}'$

10) 初始化混沌映射$U_{t+1}=\alpha U_t(1-U_t)$，其中3.569 9<$\alpha$≤4

11) 利用映射$U_{t+1}=\alpha_1 U_t(1-U_t)$生成一组长度为$m$的伪随机序列$\{\beta_1,\beta_2,\cdots,\beta_m\}$

12) 构造向量$\boldsymbol{\beta}=[\beta_1,\beta_2,\cdots,\beta_m]$

13) 将$\boldsymbol{\beta}$复制n次，构造矩阵$\boldsymbol{B}=[\boldsymbol{\beta},\boldsymbol{\beta},\cdots,\boldsymbol{\beta}]^{\mathrm{T}}\in\mathbb{R}^{m\times n}$

14) 根据算法1生成稀疏正交矩阵$\boldsymbol{W}$

15) 输出密钥sk=($\boldsymbol{P}'$,$\boldsymbol{W}$,$\boldsymbol{B}$)

步骤2 数据加密。为了确保加密后数据点之间的欧氏距离不变，用户首先使用密钥sk对明文矩阵$\boldsymbol{X}$进行等距变换，得到矩阵$\boldsymbol{X}'$为

$$\boldsymbol{X}'=\boldsymbol{X}\boldsymbol{P}'\boldsymbol{W}+\boldsymbol{B} \tag{15}$$

其中，$\boldsymbol{P}'$为置换矩阵，$\boldsymbol{W}$为稀疏正交矩阵，$\boldsymbol{B}$为平移矩阵。虽然保证加密数据集中数据点之间的距离不变可以确保聚类结果与明文数据集的聚类结果完全一致，但同时数据点之间的距离信息也被泄露了。为了保护数据点之间的距离信息，通过向矩阵$\boldsymbol{X}'$中添加能够保持距离的大小顺序不变的噪声矩阵$\boldsymbol{E}$来隐藏距离隐私，同时也进一步增强了明文矩阵$\boldsymbol{X}$的安全性。密文矩阵$\boldsymbol{Y}$为

$$\boldsymbol{Y}=\boldsymbol{X}'+\boldsymbol{E} \tag{16}$$

其中，$\boldsymbol{E}$中的每个元素均服从高斯分布$N(0,\sigma^2)$。在加密过程中，参数k不变。完成加密操作后，用户将$\boldsymbol{Y}$和k发送给云服务器。

步骤3 云计算。云服务器接收到$\boldsymbol{Y}$和k后，对密文矩阵$\boldsymbol{Y}$执行K-Means算法得到每个点所属的簇编号，并将簇标签返回给用户。用户直接使用簇标签，不需要对计算结果进行解密操作便可获得真实的聚类结果。

4 性能分析

4.1 准确性分析

定理4 本文方案可以实现明文数据集X的与密文数据集Y的K-Means聚类结果的一致性。

证明 K-Means聚类的结果是由簇中心到数据点之间的欧氏距离决定的。因此，若密文空间中数据点之间的距离顺序与明文空间中数据点之间的距离顺序一致，聚类结果也将保持一致。

设明文矩阵$\boldsymbol{X}$中任意两个数据点$\boldsymbol{x}_i$和$\boldsymbol{x}_j$的欧氏距离为$d=\left\|\boldsymbol{x}_i-\boldsymbol{x}_j\right\|$，密文矩阵$\boldsymbol{Y}=\boldsymbol{X}\boldsymbol{P}'\boldsymbol{W}+\boldsymbol{B}+\boldsymbol{E}$中对应的两个数据点$\boldsymbol{y}_i$和$\boldsymbol{y}_j$的距离为

$$d'=\left\|\left(\boldsymbol{x}_i\boldsymbol{P}'\boldsymbol{W}+\boldsymbol{\beta}+\boldsymbol{e}_i\right)-\left(\boldsymbol{x}_j\boldsymbol{P}'\boldsymbol{W}+\boldsymbol{\beta}+\boldsymbol{e}_j\right)\right\|=\left\|\left(\boldsymbol{x}_i-\boldsymbol{x}_j\right)\boldsymbol{P}'\boldsymbol{W}+\left(\boldsymbol{e}_i-\boldsymbol{e}_j\right)\right\| \tag{17}$$

根据定理1，$\boldsymbol{P}'$和$\boldsymbol{W}$均为正交矩阵，可得

$$(\boldsymbol{P}'\boldsymbol{W})^{\mathrm{T}}(\boldsymbol{P}'\boldsymbol{W})=\boldsymbol{W}^{\mathrm{T}}(\boldsymbol{P}')^{\mathrm{T}}\boldsymbol{P}'\boldsymbol{W}=\boldsymbol{W}^{\mathrm{T}}\boldsymbol{I}\boldsymbol{W}=\boldsymbol{W}^{\mathrm{T}}\boldsymbol{W}=\boldsymbol{I} \tag{18}$$

因此，$\boldsymbol{P}'\boldsymbol{W}$为正交矩阵。根据定理2，有

$$d'=d+\left\|\boldsymbol{e}_i-\boldsymbol{e}_j\right\| \tag{19}$$

由三角不等式和噪声约束条件$\left\|\boldsymbol{e}_i-\boldsymbol{e}_j\right\|\leqslant 3\sqrt{2}\sigma$可知$-3\sqrt{2}\sigma\leqslant\left\|y_i-y_j\right\|\leqslant 3\sqrt{2}\sigma$。当$\sigma\leqslant\frac{\delta_{\min}}{6\sqrt{2}}$时，若$\left\|\boldsymbol{x}_i-\boldsymbol{x}_j\right\|>\left\|\boldsymbol{x}_k-\boldsymbol{x}_l\right\|$，则$\left\|\boldsymbol{y}_i-\boldsymbol{y}_j\right\|-\left\|\boldsymbol{y}_k-\boldsymbol{y}_l\right\|\geqslant\delta_{\min}-6\sqrt{2}\sigma\geqslant 0$。虽然添加噪声后的距离改变了，但距离的大小顺序不变。由于K-Means的簇划分仅依赖数据点到簇中心的最近邻关系，因此密文数据集$\boldsymbol{Y}$与明文数据集$\boldsymbol{X}$的聚类结果一致。

综上所述，本文方案可以实现准确性的设计目标。证毕。

4.2 安全性分析

定理5 假设存在一个概率多项式时间敌手A，本文方案可以抵抗已知明文攻击。

证明 上述定理即证明敌手A在知道加密算

法、密文数据 $\boldsymbol{Y}$ 和一些明文数据的情况下，推断出密钥 sk 和明文数据 $\boldsymbol{X}$ 的概率可以忽略不计。

假设敌手获得明文矩阵 $\boldsymbol{X}\in\mathbb{R}^{n\times m}$ 中的一些元组 $\boldsymbol{T}\in\mathbb{R}^{t\times m}$，即 t 个明文−密文对 $(\boldsymbol{T},\boldsymbol{Y}_T)$，其中 $\boldsymbol{Y}_T$ 为明文子集 $\boldsymbol{T}$ 对应的密文。敌手试图通过建立方程组 $\boldsymbol{Y}_T=\boldsymbol{TP'W}+\boldsymbol{B}_T+\boldsymbol{E}_T$ 来求解 sk 或恢复 $\boldsymbol{X}$。方程组中包含的方程个数为 tm，未知数包括 $\boldsymbol{W},\boldsymbol{B}_T$ 和噪声矩阵 $\boldsymbol{E}_T$。首先，密钥 $\boldsymbol{W}$ 中有 $2m$ 个非零未知数，$\boldsymbol{B}_T$ 中的未知数即随机数 $\{\beta_1,\beta_2,\cdots,\beta_m\}$，共有 m 个未知数。噪声矩阵 $\boldsymbol{E}_T$ 中的未知数个数为 tm，所以方程组中共有 $tm+3m$ 个未知数。由于未知数的个数大于方程组的个数，因此敌手无法获得密钥 sk 或恢复 $\boldsymbol{X}$。若敌手尝试暴力破解攻击，根据密钥生成算法，$\boldsymbol{P}$ 中的非零元素由 m 个-1 或 1 的数组成，共有 2^m 种可能性，对 $\boldsymbol{P}$ 进行列置换共有 $m!$ 种可能性，所以产生密钥 $\boldsymbol{P'}$ 的可能性共有 $m!2^m$ 种。此外，$\boldsymbol{\beta}$ 由混沌映射生成，初始值落在具有无限空间的实数域内，在初始值未知的情况下，攻击者无法获取 $\boldsymbol{\beta}$。如果敌手 A 发起穷举攻击来猜测密钥，其成功的概率可以忽略不计。因此，云服务器无法根据密文 $\boldsymbol{Y}$ 和一些明文数据恢复明文 $\boldsymbol{X}$ 和密钥 sk。

综上所述，本文方案可以实现安全性的设计目标。证毕。

4.3　计算复杂度分析

定理 6　本文方案的计算复杂度是 $O(mn)$，其中 n 是数据点的数量，m 是数据特征的数量。

证明　这里的计算复杂度指用户的计算开销，用户的计算开销包括以下两个部分。

1) 密钥生成。在这个阶段，用户需要生成密钥 sk=$(\boldsymbol{P'},\boldsymbol{W},\boldsymbol{B})$，根据密钥生成算法，$\boldsymbol{P'}$ 的生成需要先产生 m 个−1 或 1 的数值，然后用 Fisher-Yates 洗牌算法对 $\boldsymbol{X}$ 进行随机排列，计算复杂度为 $O(m)$。由于用户通过预计算生成 2×2 或 3×3 的正交矩阵，因此基于这些小规模的正交矩阵构造密钥 W 的计算复杂度可以忽略不计。密钥 B 的产生需要通过混沌系统产生一套伪随机数 $\{\beta_1,\beta_2,\cdots,\beta_m\}$，计算复杂度为 $O(m)$。因此，密钥生成的计算复杂度是 $O(m)$。

2) 数据加密。在这个阶段，用户需要先计算 $\boldsymbol{XP'W}$。由于 $\boldsymbol{P'}$ 和 $\boldsymbol{W}$ 均为稀疏矩阵，因此计算 $\boldsymbol{XP'W}$ 的复杂度为 $O(nm)$。接着，用户计算 $\boldsymbol{XP'W}+\boldsymbol{B}+\boldsymbol{E}$，计算复杂度为 $O(mn)$。为了使添加的噪声能够保持距离的大小顺序不变，需要利用局部敏感哈希计算数据点之间的最小距离，计算复杂度为 $O(lhmn)$，其中，h 为哈希函数的个数，l 为哈希表的个数。由于 l、h 均为常数，数据加密的计算复杂度是 $O(mn)$。

综上所述，用户使用本文方案的计算复杂度是 $O(mn)$，而用户直接执行 K-Means 聚类的计算复杂度为 $O(tkmn)$。因此，本文方案实现了高效性的设计目标。证毕。

5　实验评估

5.1　实验环境和数据集

为了验证本文方案的实际性能，在一台配置为 AMD Ryzen 7 3700X 处理器（3.6 GHz）和 48 GB 内存的计算机上进行了实验评估。其中，数据加密模块由 C 语言实现，K-Means 聚类模块使用 Python 语言实现。相比云服务器的配置，用户端的配置通常更低。为了公平地比较云服务器与用户的计算开销，所有的实验均在同一台计算机上执行。每组实验重复进行 3 次，最终结果取平均值以减少误差。密钥 sk 在区间（−100,100）内随机采样，哈希函数的个数 h 与哈希表的个数 l 均设置为 3。

实验选取了 UCI 机器学习库中 4 个广泛用于聚类分析任务的经典数据集，数据集编号按照其数据量递增排列，具体如下。数据集 D_1 是一个包含 178 条记录的葡萄酒数据集[39]，每条记录包括酒精、苹果酸等 13 项化学成分特征。数据集 D_2 是一个包含 440 条记录的客户年度消费数据集[40]，每条记录包括生鲜、牛奶等 7 类商品的消费信息。数据集 D_3 是一个包含 1 941 条记录的带钢缺陷数据集[41]，每条记录涵盖传送带长度、钢板厚度等 28 种特征属性。数据集 D_4 是一个包含 5 456 条记录的机器人导航数据集[42]，每条记录包含超声检查结果、超声波传感器读数等 25 个导航状态相关特征。

为了评估本文方案在计算效率与通信开销方面的性能，选取了 4 个当前在隐私保护聚类中表现优异的代表性方案作为对比基线，具体如下。

1) 文献[16]提出了一种基于 BGV 同态加密的聚类外包方案，适用于 K-Means 和 DBSCAN 等聚类算法。该方案通过设计编码预处理策略在保证聚类精度的同时降低了计算开销，并构建了密文比较协议以实现密文间的距离大小比较。

2) 文献[29]利用支持向量运算的向量同态加

密[29]来保护明文数据，并设计了一种基于向量同态加密的密文距离比较方法，减少了 K-Means 聚类外包过程中的通信开销。

3) 文献[14]基于多密钥全同态加密[27]实现了支持多用户密钥环境下的 K-Means 聚类外包，并构建了安全的平方欧几里得距离、比较、最小值和平均值等操作协议，提升了聚类计算的灵活性。

4) 文献[26]提出了一种基于 Paillier 同态加密的分布式 K-means 聚类隐私保护算法，通过对加密的聚类中心和数值进行本地明文计算，从而减少了用户的计算开销。

5.2　准确性评估

在对本文方案进行理论准确性分析的基础上，进一步通过实验对明文数据与密文数据下的 K-Means 聚类结果进行了分析。为了验证密文数据聚类结果与明文数据聚类结果的一致性，实验采用如下流程。首先，随机生成 3 组不同的密钥，分别对同一明文数据集进行加密，得到 3 个密文数据集。然后，分别对 3 个密文数据集执行 K-Means 聚类，并将每次聚类结果与明文聚类结果进行比对。若所有样本在密文数据下的簇划分均与明文数据的划分结果完全一致，则认为该轮聚类的准确率为 100%。本文方案在不同数据集下的聚类准确率如表 2 所示。

表 2　本文方案在不同数据集下的聚类准确率

数据集	数据数量/个	属性数量/个	聚类簇数	准确率
D_1	178	13	3	100%
D_2	440	7	2~5	100%
D_3	1 941	28	7	100%
D_4	5 456	25	4	100%

由表 2 可以看出，在 3 组实验中，4 个真实数据集均达到 100% 的聚类准确率。其中，数据集 D_2 为无标签的数据集，分别设定聚类簇数 k 为 2、3、4、5；其余数据集为有标签的数据集，k 值对应实际类别数。实验结果说明，无论 k 的取值如何，本文方案在保持数据隐私的同时，均能够准确复现明文聚类结果，确保了外包方案对最终聚类准确性的零损失，具有良好的稳定性和适应性。以数据集 D_1 为例，通过主成分分析算法将其降维至二维以便可视化聚类效果。数据集 D_1 在明文数据和密文数据下的 K-Means 聚类结果分别如图 3 和图 4 所示。从图 3 和图 4 可以看出，两者的聚类簇划分一致，各类数据点在二维空间中的分布保持高度一致，未出现误分或簇边界重叠的情况。该结果不仅直观地进一步验证了聚类准确率，也说明本文方案在保持数据隐私的前提下，能够完整地保留数据的聚类结构信息，适用于准确性要求高的隐私保护聚类任务。

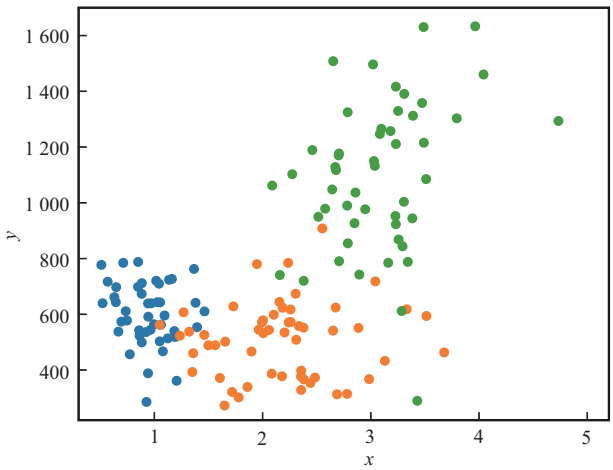

图 3　数据集 D_1 在明文数据下的 K-Means 聚类结果

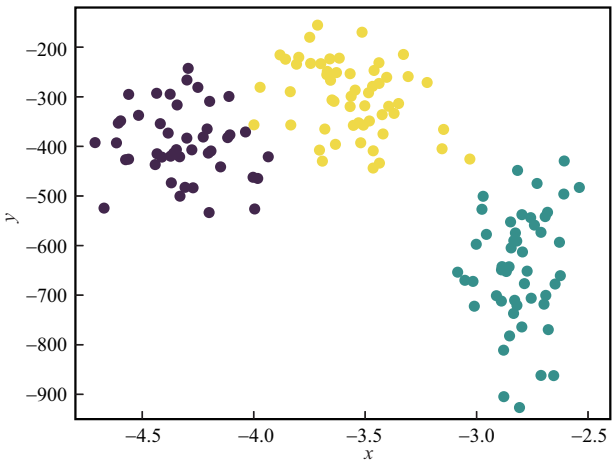

图 4　数据集 D_1 在密文数据下的 K-Means 聚类结果

5.3　高效性评估

1) 对本文方案的计算效率评估

为了验证本文方案的高效性，实验测量并比较了以下 3 种情况的运行时间：用户使用本文方案消耗的时间，即用户加密消耗的时间；用户本地运行明文 K-Means 聚类消耗的时间；云服务器使用本文方案在密文数据下执行 K-Means 聚类所消耗的时间。不同操作下的运行时间如表 3 所示。为了更直观地展示表 3 中的数据，对其进行可视化处理，3 种情形下消耗的时间对比如图 5 所示。

表3　不同操作下的运行时间

数据集	用户使用本文方案消耗的时间/s	对明文数据聚类消耗的时间/s	对密文数据聚类消耗的时间/s
D_1	0.006	0.022	0.016
D_2	0.012	0.036	0.031
D_3	0.046	0.139	0.144
D_4	0.104	0.318	0.337

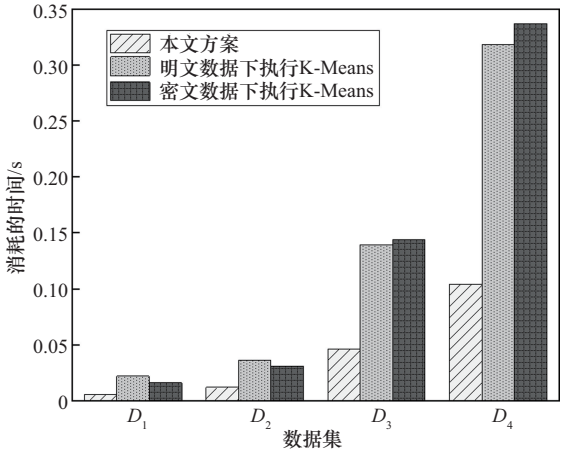

图5　3种情形下消耗的时间对比

用户外包聚类任务的目的是减轻本地的计算负担。由图5可以看出，用户使用本文外包方案所消耗的时间明显低于其本地直接运行K-Means聚类算法消耗的时间。随着数据集规模的增大，本文方案消耗时间的增长幅度最小，表明本文方案为用户节省了大量的本地计算资源，实现了高效性的设计目标。此外，在相同硬件平台上，使用本文方案在密文数据下执行K-Means聚类消耗的时间与在明文数据下执行K-Means聚类消耗的时间非常接近。由于所有的实验均在相同配置的计算机上执行，而现实世界中云服务器的配置通常高于用户终端，因此云服务器使用本文方案在密文数据下执行K-Means聚类消耗的时间会比用户在明文数据下执行K-Means聚类消耗的时间更短。实验结果表明本文方案在密文数据下运行K-Means聚类的效率较高，避免了因密文扩张导致的效率大幅度降低问题。

计算效率加速比结果如表4所示。由表4可以看出，用户本地运行K-Means聚类消耗的时间与使用本文方案消耗的时间的比值均大于或等于3，这意味着对用户而言，本地运行K-Means聚类消耗的时间是本文方案消耗的时间的3倍以上，使用本文方案显著降低了用户的计算负担。此外，明文K-Means聚类消耗的时间与密文K-Means聚类消耗的时间的比值接近1，表明在密文状态下运行K-Means聚类算法的计算开销并未因加密而增加。这确保了本文方案不仅大幅节省了用户的计算开销，也不会给云端服务器带来额外的计算负担，能够充分发挥外包计算的效率优势。

2) 与现有方案的性能对比

为了进一步验证本文方案的高效性，在计算开销和通信开销两个方面将其与现有性能最优的基于同态加密的外包方案即文献[14]、文献[16]、文献[26]、文献[29]进行了比较。由于这些对比方案的计算和内存消耗较高，在4个数据集中能够处理的数据集规模最大为D_1，因此选用D_1作为性能比较的数据集。

针对方案的计算开销比较，本文使用用户本地和云服务器在外包计算过程中消耗的时间作为衡量指标。用户外包消耗的时间对比如图6所示。由图6可知，本文方案在所有对比方法中耗时最短，具备最高的计算效率，具体原因分析如下。文献[14]与文献[16]方案均以BGV同态加密为密码原语，需要依次加密明文数据集中的所有元素，因此计算时间最长。文献[26]方案在聚类过程中大量采用明文计算，仅在统计结果阶段引入Paillier同态加密进行保护，显著减少了同态运算次数，因此计算时间相对较短。文献[29]方案采用向量同态加密，需要对明文数据集中的每个向量依次进行加密。尽管相较于逐元素的加密方式有所优化，但仍不可避免地产

表4　计算效率加速比结果

数据集	$\frac{\text{用户本地运行K-Means聚类消耗的时间}}{\text{用户使用本文方案消耗的时间}}$	$\frac{\text{明文K-Means聚类消耗的时间}}{\text{密文K-Means聚类消耗的时间}}$
D_1	3.667	1.375
D_2	3.000	1.161
D_3	3.021	0.965
D_4	3.058	0.944

生一定的加密开销。相比之下，本文方案将明文数据集作为一个整体进行加密，仅需要一次加密操作。本文特别设计了稀疏正交矩阵作为密钥，加密过程仅涉及稀疏矩阵乘法计算，极大地降低了用户本地的计算负担，因此计算效率更高。

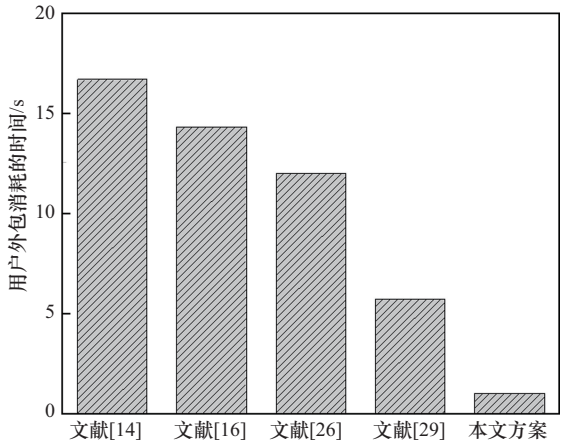

图6　用户外包消耗的时间对比

为了进一步比较本文方案和对比方法在用户端计算开销方面的差异，分别统计了用户在密钥生成、加密、解密3个阶段的计算时间，用户在不同阶段中消耗的时间对比如图7所示。从图7中可以观察到，文献[14]与文献[16]方案在3个阶段中的计算时间均明显高于其他对比方法。其主要原因在于，这两种方案均使用BGV全同态加密，需要对数据集中大量元素进行逐一加密，并在加密阶段涉及复杂的大整数模运算和密钥切换操作，从而导致用户端计算开销显著增加。文献[26]方案仅在统计结果阶段引入Paillier同态加密，其余计算过程主要在明文状态下完成，有效减少了同态运算的次数与复杂度，因此在密钥生成、加密和解密3个阶段的耗时均相对较短。文献[29]方案采用向量同态加密，以向量为基本加密单元进行数据保护，相较于逐元素的加密方式在一定程度上降低了加密与解密的计算复杂度，因此其用户端计算时间亦保持在较低水平。本文方案利用稀疏正交矩阵和有界随机扰动对数据进行加密，避免了复杂的同态加密操作。加密与解密过程主要由稀疏矩阵乘法构成，计算复杂度低、实现效率高，因此在密钥生成、加密和解密3个阶段的计算耗时均为最少。这一结果与图6中用户总体计算开销的结论保持一致，证明了本文方案在用户端计算效率方面的优势。

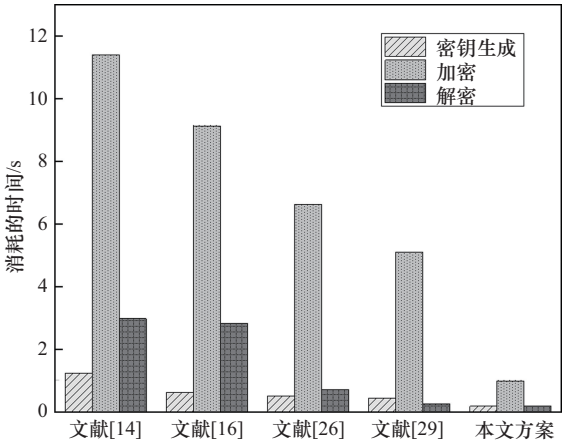

图7　用户在不同阶段中消耗的时间对比

本文还比较了云服务器在执行外包方案时的计算开销，实验结果如图8所示。文献[14]与文献[16]方案在云服务器端均需基于BGV全同态加密执行密文运算，并在聚类过程中频繁调用密文比较协议，其密文乘法与重线性化操作计算代价较高，因而服务器耗时最长。文献[26]方案主要利用Paillier同态加密完成加法聚合操作，不需要进行密文距离计算，因而整体耗时较短。文献[29]采用向量同态加密进行密文距离计算，能够在一定程度上提高密文计算效率，使云服务器消耗的时间相对较低，但仍明显高于在明文状态下直接执行K-Means聚类的时间。相比上述方案，本文方案仅需云服务器对密文数据执行K-Means聚类，不需要在聚类过程中进行加解密或其他安全协议的调用，因此云服务器消耗的时间最短。实验结果表明，本文方案在显著减轻用户端计算负担的同时，并未增加云服务器的计算开销，有效提升了方案的整体计算效率。

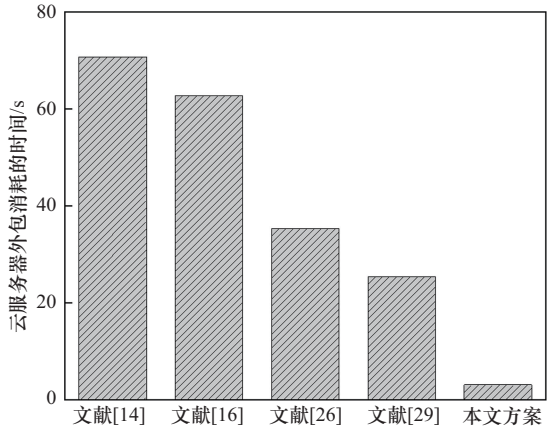

图8　云服务器外包消耗的时间对比

将用户与云服务器之间传输的数据总量作为衡量指标，本文方案与现有方案的通信数据量对比如图9所示。由图9可以看出，本文方案所产生的通信数据量最少，原因分析如下。由于同态加密不能确保密文距离的大小不变，文献[14]与文献[16]方案均设计了对应的密文比较协议，在对密文数据集进行K-Means聚类的过程中，每次比较距离大小都需要调用密文比较协议，密文比较协议的实现需要用户和云服务器之间进行交互，频繁的数据交换显著增加了通信负担，因此这两个方案的通信开销相对较大。文献[26]方案将距离计算与簇划分过程置于明文状态下完成，仅在统计结果阶段对局部统计引入Paillier同态加密进行保护，从而避免了在密文状态下进行距离比较及相关安全协议的调用。用户与云服务器之间仅需传输加密后的统计数据和聚类中心更新信息，传输数据量远小于原始数据集，从而有效降低了通信开销。文献[29]方案基于向量同态加密设计了密文比较协议，该协议不需要用户和云服务器之间进行交互，但仍然需要传输一个参数矩阵以实现该协议，因此通信开销相对较小。相比之下，本文方案支持在密文数据中直接进行距离比较，既不需要用户和云服务器之间进行交互，也不需要额外传输任何参数；整个通信过程仅需传输一次密文数据集及最终聚类结果，通信数据量显著减少，展示了本文方案在通信资源利用方面的优势。

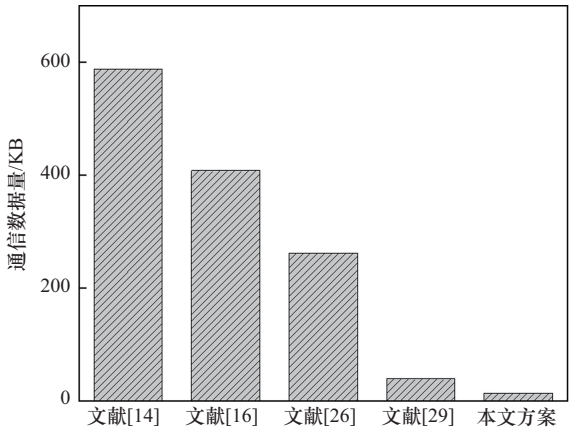

图9　本文方案与现有方案的通信数据量对比

综上所述，本文隐私保护K-Means聚类外包方案在准确性和高效性方面均表现出优越性能。本文方案在不降低聚类精度的前提下，显著降低了用户的计算开销和通信负担。此外，本文方案在密文状态下执行K-Means聚类的效率与明文环境相当，有效缓解了传统加密方案中因密文扩张造成的性能瓶颈。这些实验结果不仅验证了理论分析的正确性，也充分体现了本文方案在实际应用中的可行性与优势。

5.4　消融实验

为验证本文方案中关键组件对安全性、聚类准确率及用户计算开销的影响，本节设计了消融实验。实验通过对方案中基于混沌系统的伪随机数生成，基于Gram-Schmidt正交化的稀疏正交矩阵构造，以及基于局部敏感哈希的有界随机扰动等关键组件进行逐一替换或移除，对比分析不同方法下性能的变化情况。由于数据集D_4规模最大，更有利于放大各组件在计算开销上的差异，因此实验在数据集D_4上进行。

实验共设置5组对比方案，以体现各组件的作用。方案1即本文完整方案，采用混沌系统生成伪随机数，构造稀疏正交矩阵作为加密密钥，并在加密过程中引入基于局部敏感哈希的有界随机扰动。方案2将混沌系统替换为传统的伪随机数生成器，实验中采用经典的梅森旋转算法[43]生成随机序列，其余设置保持不变。方案3将稀疏正交矩阵替换为稠密正交矩阵，该稠密矩阵通过Gram-Schmidt正交化生成，其余设置保持不变。方案4不引入基于局部敏感哈希的随机扰动，仅使用稀疏正交矩阵对数据进行加密。方案5添加随机扰动，但不利用局部敏感哈希计算扰动上界，而是基于全局距离计算扰动上界，其余设置保持不变。

消融实验结果如表5所示，给出了上述5种方案下的安全性、聚类准确率以及用户计算时间的统计结果。由表5可以看出，5种方案的聚类结果均与明文K-Means聚类结果完全一致，聚类准确率均达到100%，说明各组件的移除或替换不会影响聚类的正确性。然而，在安全性方面，方案4存在明显缺陷。由于正交变换能够保持数据点之间的欧氏距离不变，当未引入随机扰动时，攻击者可通过分析密文数据点之间的距离关系推断明文距离信息。移除有界随机扰动后，加密前后数据点间的距离值如图10所示，所有散点严格分布于$y=x$直线上，表明加密前后距离保持一致，导致数据点间距离信息被泄露。相比之下，其余4种方案通过引入扰动有效破坏了这一线性关系，保障了距离信息的安全性，可以抵抗已知明文攻击。

表5　消融实验结果

方案	随机数生成方式	正交矩阵	有界随机扰动	局部敏感哈希	安全性	聚类准确率	用户计算时间/ s
方案1	混沌系统	稀疏	√	√	已知明文攻击	100%	0.104
方案2	伪随机数生成器	稀疏	√	√	已知明文攻击	100%	0.135
方案3	混沌系统	稠密	√	√	已知明文攻击	100%	0.318
方案4	混沌系统	稀疏	×	×	泄露距离信息	100%	0.064
方案5	混沌系统	稀疏	√	×	已知明文攻击	100%	1.272

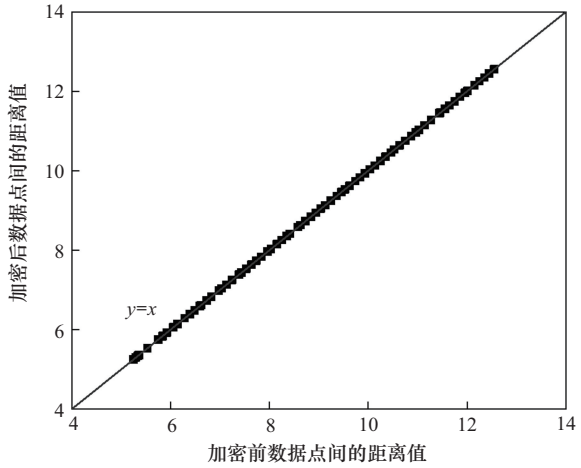

图10　移除有界随机扰动后加密前后数据点间的距离值

在用户计算开销方面，方案4消耗的时间最少，其余依次为方案1、方案2、方案3，方案5消耗的时间最多。方案4由于未引入扰动，因此整体计算时间最少。方案1由于同时引入基于混沌系统的随机数生成、稀疏正交矩阵以及基于局部敏感哈希的有界随机扰动，用户端的计算耗时高于方案4。方案2使用梅森旋转算法生成伪随机数，需要维护较大的内部状态并执行额外的状态更新操作，导致用户端的计算时间增加。方案3采用稠密正交矩阵，由于矩阵中的非零元素增加，矩阵乘法计算复杂度提高，因此整体耗时进一步增加。方案5不利用局部敏感哈希对局部距离进行近似估计，而是基于全局距离信息计算扰动上界，该过程涉及对大量样本距离的统计与计算，计算代价显著增加，因此用户端的计算时间最多。

综上所述，基于局部敏感哈希的有界随机扰动是保障距离隐私安全的关键组成部分，缺失该机制将直接导致数据点间距离信息泄露；基于混沌系统的伪随机数生成与稀疏正交矩阵在保证安全性的同时，有效降低了用户的计算开销。上述组件在功能上相互补充，任一组件的移除或替换均会在安全性或效率方面造成性能下降。本文方案通过对上述组件的结合，实现了安全性与高效性的需求，验证了各组件的必要性与有效性。

6　结束语

本文围绕K-Means聚类的安全外包问题展开研究，旨在借助云计算的强大算力为计算资源受限的用户提供安全且高效的聚类分析服务。针对用户在云计算环境下的隐私保护需求，提出了一种基于稀疏矩阵变换和有界随机扰动的K-Means聚类外包方案。方案利用混沌系统产生伪随机数，通过Gram-Schmidt正交化构造稀疏正交矩阵，实现了明文数据的加密保护。为防止正交变换泄露数据间的距离信息，设计了基于局部敏感哈希的有界随机扰动方法，增强了聚类过程的数据安全性，节省了用户的计算和通信开销。通过理论分析证明了所提方案实现了准确性、安全性和高效性的设计目标。在4个真实数据集上的实验结果表明，本文方案在准确性和高效性方面表现良好，优于现有K-Means聚类外包方案。

参考文献：

[1] 宋冬冬, 王楠, 田树耀, 等. 基于聚类算法的车辆数据挖掘及可视化研究[J]. 计算机技术与发展, 2020, 30(10): 204-209.
Song D D, Wang N, Tian S Y, et al. Research on vehicle data mining and visualization based on clustering algorithm[J]. Computer Technology and Development, 2020, 30(10): 204-209.

[2] 孙灏铖, 刘力, 李凡长. 李群模糊C均值聚类图像分割算法[J]. 软件学报, 2024, 35(10): 4806-4825.
Sun H C, Liu L, Li F C. Lie group fuzzy C-means clustering algorithm for image segmentation[J]. Journal of Software, 2024, 35(10): 4806-4825.

[3] Jain M, Kaur G, Saxena V. A K-Means clustering and SVM based hybrid concept drift detection technique for network anomaly detection[J]. Expert Systems with Applications, 2022, 193: 116510.

[4] Nie F P, Li Z H, Wang R, et al. An effective and efficient algorithm for K-means clustering with new formulation[J]. IEEE Transactions on

Knowledge and Data Engineering, 2023, 35(4): 3433-3443.

[5] Hu H Z, Liu J X, Zhang X P, et al. An effective and adaptable K-means algorithm for big data cluster analysis[J]. Pattern Recognition, 2023, 139: 109404.

[6] Borlea I D, Precup R E, Borlea A B, et al. A unified form of fuzzy C-means and K-means algorithms and its partitional implementation[J]. Knowledge-Based Systems, 2021, 214: 106731.

[7] Liu X W. SimpleMKKM: simple multiple kernel K-means[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2023, 45(4): 5174-5186.

[8] Khoda Parast F, Sindhav C, Nikam S, et al. Cloud computing security: a survey of service-based models[J]. Computers & Security, 2022, 114: 102580.

[9] Zhang Z X, Zhang H L, Song X F, et al. Secure outsourcing evaluation for sparse decision trees[J]. IEEE Transactions on Dependable and Secure Computing, 2024, 21(6): 5228-5241.

[10] Anju J, Shreelekshmi R. A secure image outsourcing using privacy-preserved local color layout descriptor in cloud environment[J]. IEEE Transactions on Services Computing, 2024, 17(2): 378-391.

[11] Liu Z W, Hu C Q, Li R N, et al. A privacy-preserving outsourcing computing scheme based on secure trusted environment[J]. IEEE Transactions on Cloud Computing, 2023, 11(3): 2325-2336.

[12] Wu W, Liu J, Wang H M, et al. Secure and efficient outsourced K-means clustering using fully homomorphic encryption with ciphertext packing technique[J]. IEEE Transactions on Knowledge and Data Engineering, 2021, 33(10): 3424-3437.

[13] Ravi N, Scaglione A, Kadam S, et al. Differentially private K-means clustering applied to meter data analysis and synthesis[J]. IEEE Transactions on Smart Grid, 2022, 13(6): 4801-4814.

[14] Zhang P, Huang T, Sun X Q, et al. Privacy-preserving and outsourced multi-party K-means clustering based on multi-key fully homomorphic encryption[J]. IEEE Transactions on Dependable and Secure Computing, 2023, 20(3): 2348-2359.

[15] Li Y, Song X, Tu Y C, et al. GAPBAS: Genetic algorithm-based privacy budget allocation strategy in differential privacy K-means clustering algorithm[J]. Computers & Security, 2024, 139: 103697.

[16] 贾春福, 李瑞琪, 王雅飞. 基于同态加密的DBSCAN聚类隐私保护方案[J]. 通信学报, 2021, 42(2): 1-11.
Jia C F, Li R Q, Wang Y F. Privacy protection scheme of DBSCAN clustering based on homomorphic encryption[J]. Journal on Communications, 2021, 42(2): 1-11.

[17] Yang M M, Tjuawinata I, Lam K Y. K-means clustering with local d_χ-privacy for privacy-preserving data analysis[J]. IEEE Transactions on Information Forensics and Security, 2022, 17: 2524-2537.

[18] Zhang E, Li H M, Huang Y C, et al. Practical multi-party private collaborative k-means clustering[J]. Neurocomputing, 2022, 467: 256-265.

[19] Ni T J, Qiao M H, Chen Z L, et al. Utility-efficient differentially private K-means clustering based on cluster merging[J]. Neurocomputing, 2021, 424: 205-214.

[20] 王琛, 李佳润, 徐剑. 基于函数加密的密文卷积神经网络模型[J]. 通信学报, 2024, 45(3): 50-65.
Wang C, Li J R, Xu J. Convolutional neural network model over encrypted data based on functional encryption[J]. Journal on Communications, 2024, 45(3): 50-65.

[21] Liu D X, Bertino E, Yi X. Privacy of outsourced k-means clustering[C]// Proceedings of the 9th ACM Symposium on Information, Computer and Communications Security. New York: ACM Press, 2014: 123-134.

[22] Geelen R, Vercauteren F. Bootstrapping for BGV and BFV revisited[J]. Journal of Cryptology, 2023, 36(2): 12.

[23] Yuan J W, Tian Y F. Practical privacy-preserving MapReduce based K-means clustering over large-scale dataset[J]. IEEE Transactions on Cloud Computing, 2019, 7(2): 568-579.

[24] Ye J, Hu Z W, Zhang Z Q. General-purpose multi-user privacy-preserving outsourced k-means clustering[J]. Journal of Information Security and Applications, 2025, 89: 103976.

[25] Pang Q, Zhu J H, Möllering H, et al. BOLT: privacy-preserving, accurate and efficient inference for transformers[C]//Proceedings of the 2024 IEEE Symposium on Security and Privacy (SP). Piscataway: IEEE Press, 2024: 4753-4771.

[26] Tang Z G, Duan X F, Liang R H, et al. Efficient multi-party privacy preserving federated k-means based on homomorphic encryption[J]. Information Sciences, 2025, 717: 122335.

[27] Chen L, Zhang Z F, Wang X Q. Batched multi-hop multi-key FHE from ring-LWE with compact ciphertext extension[C]//Theory of Cryptography. Berlin: Springer, 2017: 597-627.

[28] Sakellariou G, Gounaris A. Homomorphically encrypted k-means on cloud-hosted servers with low client-side load[J]. Computing, 2019, 101(12): 1813-1836.

[29] Yang H M, Liang S P, Luo X Z, et al. PIPC: privacy- and integrity-preserving clustering analysis for load profiling in smart grids[J]. IEEE Internet of Things Journal, 2022, 9(13): 10851-10861.

[30] Zhou H C, Wornell G. Efficient homomorphic encryption on integer vectors and its applications[C]//Proceedings of the 2014 Information Theory and Applications Workshop (ITA). Piscataway: IEEE Press, 2014: 1-9.

[31] Halevi S, Shoup V. Bootstrapping for HElib[J]. Journal of Cryptology, 2021, 34(1): 7.

[32] Jiang B B. Multi-key FHE without ciphertext-expansion in two-server model[J]. Frontiers of Computer Science, 2021, 16(1): 161809.

[33] Tian C L, Yu J, Zhang H L, et al. Novel secure outsourcing of modular inversion for arbitrary and variable modulus[J]. IEEE Transactions on Services Computing, 2022, 15(1): 241-253.

[34] Zhang L Y, Liu Y S, Wang C, et al. Improved known-plaintext attack to permutation-only multimedia ciphers[J]. Information Sciences, 2018, 430: 228-239.

[35] Li Z Y, Xie D, Liu S Q, et al. Known-plaintext attacks to thumbnail-preservation encryption using Pix2pix generative adversarial network[C]// Proceedings of the ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). Piscataway: IEEE Press, 2025: 1-5.

[36] 义理林, 柯俊翔. 混沌保密光通信研究进展[J]. 通信学报, 2020, 41(3): 168-181.
Yi L L, Ke J X. Research progress of chaotic secure optical communication[J]. Journal on Communications, 2020, 41(3): 168-181.

[37] Kocarev L, Lian S G. Chaos-based cryptography: theory, algorithms and applications[M]. Berlin: Springer, 2011.

[38] Knuth D E. The art of computer programming[M]. Massachusetts: Addison-Wesley , 2001.

[39] Aeberhard S, Coomans D, Vel O D. Comparative analysis of statistical

pattern recognition methods in high dimensional settings[J]. Pattern Recognition, 1994, 27(8): 1065-1077.
[40] Lakshmi B J, Madhuri K B, Shashi M. An efficient algorithm for density based subspace clustering with dynamic parameter setting[J]. International Journal of Information Technology and Computer Science, 2017, 9(6): 27-33.
[41] Tian Y, Fu M Y, Wu F. Steel plates fault diagnosis on the basis of support vector machines[J]. Neurocomputing, 2015, 151: 296-303.
[42] Zdrodowska M, Dardzińska A, Kasperczuk A. Using data mining tools in wall-following robot navigation data set[C]//Proceedings of the 2020 International Conference Mechatronic Systems and Materials (MSM). Piscataway: IEEE Press, 2020: 1-5.
[43] Matsumoto M, Nishimura T. Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator[J]. ACM Transactions on Modeling and Computer Simulation，1998, 8(1): 3-30.

[作者简介]

赵韦（1996– ），女，山东潍坊人，哈尔滨工程大学博士生，主要研究方向为隐私计算、云计算安全、安全外包计算等。

谭静文（1996– ），女，黑龙江哈尔滨人，哈尔滨工程大学博士生，主要研究方向为流量识别、网站指纹攻击与防御。

王焕然（1988– ），男，黑龙江哈尔滨人，博士，哈尔滨工程大学讲师、硕士生导师，主要研究方向为社交网络、隐私保护、表示学习。

韩帅（1991– ），女，黑龙江哈尔滨人，博士，哈尔滨工程大学讲师、硕士生导师，主要研究方向为数据安全、数据管理、查询处理。

杨武（1974– ），男，黑龙江哈尔滨人，博士，哈尔滨工程大学教授、博士生导师，主要研究方向为无线传感器网络、点对点网络、信息安全。

赖明珠（1976– ），女，黑龙江哈尔滨人，博士，海南师范大学副教授、硕士生导师，主要研究方向为模式识别、图像处理、信息隐藏。