快速安全的分布式非负矩阵因式分解 [1]

摘要

非负矩阵分解（NMF）已成功应用于多个数据挖掘任务。最近，由于其在大型矩阵上的成本很高，对 NMF 加速的兴趣日益增加。另一方面，由于 NMF 普遍应用于可能涉及隐私数据（例如，医疗图像和记录）的图像和文本分析中，跨多个参与方（例如，医院）的 NMF 隐私问题值得关注。在本文中，我们研究了分布式 NMF 的加速和安全问题。首先，我们提出了一个分布式草图交替非负最小二乘（DSANLS）框架用于 NMF，该框架利用矩阵草图技术减少非负最小二乘子问题的大小，并具有收敛保证。对于第二个问题，我们展示了 DSANLS 经过修改后可以适应安全设置，但仅限于一个或有限次迭代。因此，我们提出了四种在同步和异步设置中均具有安全保证的高效分布式 NMF 方法。我们通过在几个真实数据集上进行广泛的实验，展示了我们提出的方法的优越性。

关键词：分布式非负矩阵分解；矩阵草图；隐私

1 引言

非负矩阵分解（NMF）是一种用于发现非负潜在因素和/或执行降维的技术。与一般的矩阵分解（MF）不同，NMF 限制两个输出矩阵因子必须是非负的。具体来说，NMF 的目标是将一个大型矩阵 $M \in \mathbb{R}^{m\times n}$ 分解为两个矩阵 $U \in \mathbb{R}^{m\times k}$ 和 $V \in \mathbb{R}^{n\times k}$ 的乘积，使得 $M \approx UV$。这里，$\mathbb{R}^{m\times n}$ 表示具有非负实数值的 $m \times n$ 矩阵集合，而 k 是一个用户指定的维度，通常 $k \ll m, n$。非负性在许多现实世界应用的特征空间中是固有的，NMF 的结果因子可以有自然的解释。因此，NMF 已经在包括文本挖掘 [1]、图像/视频处理 [2]、推荐 [3] 和社交网络分析 [4] 在内的多个领域中得到了广泛应用。

现代数据分析任务涉及的大数据矩阵规模和维度日益增长。例如，在十亿节点社交网络中的社区检测、在每帧约有 2700 万行的 4K 视频背景下的背景分离，以及在包含数百万词汇的词袋模型上的文本挖掘。预计数据量将在 “大数据” 时代增加，使得在整个 NMF 过程中将整个矩阵存储在主内存中变得不可能。因此，需要高性能和可扩展的分布式 NMF 算法。另一方面，近年来关于联合数据上的隐私保护数据挖掘的工作激增 [6]、[7]。与传统的隐私研究不同，后者强调保护单一机构中的个人信息，联合数据挖掘涉及多个参与方。每个参与方都有自己的保密数据集，所有参与方的数据联合起来用于在目标任务中取得更好的性能。由于 NMF 在图像和文本分析中的广泛应用，可能涉及跨多个参与方（例如医院）的隐私数据（例如医疗图像和记录）的使用，因此在联合数据上的 NMF 隐私问题值得关注。为了解决上述挑战，本文研究了分布式 NMF 的加速和安全问题。

首先，我们提出了一个分布式草图交替非负最小二乘（DSANLS）框架来加速 NMF。最先进的分布式 NMF 是 MPI-FAUN[8]，这是一个通用框架，它迭代地解决 U 和 V 的非负最小二乘（NLS）子问题。MPI-FAUN 背后的主要思想是利用 U 和 V 的行的局部更新计算的独立性，以最小化 NMF 算法中矩阵乘法操作的通信需求。与 MPI-FAUN 不同，我们的想法是通过减少 NMF 中每个 NLS 子问题的问题大小来加速分布式 NMF，从而降低整体计算成本。简而言之，我们通过使用矩阵草图技术来减小每个 NLS 子问题的大小：子问题中涉及的矩阵在每次迭代中都乘以一个特别设计的随机矩阵，大大减少了它们的维度。因此，每个子问题的计算成本显著降低。

然而，应用矩阵草图带来了几个问题。首先，尽管每个子问题的大小显著减少，草图涉及的矩阵乘法带来了计算开销。其次，与单机设置不同，在分布式环境中数据分布在不同的节点上。在设计不佳的解决方案中，节点之间可能需要大量通信。特别是，每个节点只保留输入矩阵和生成的近似矩阵的部分，导致由于计算过程中的数据依赖性而产生的困难。此外，生成的随机矩阵应对所有节点相同，而在每次迭代中将随机矩阵广播到所有节点会带来严重的通信开销，可能成为分布式 NMF 的瓶颈。最后，在将每个原始子问题简化为草图随机新子问题后，尚不清楚算法是否仍然收敛，以及它是否收敛到原始 NMF 问题的稳定点。

我们的 DSANLS 解决了这些问题。首先，由于草图，额外的计算成本通过适当选择随机矩阵而降低。其次，用于草图的相同随机矩阵在每个节点上独立生成，因此在分布式 NMF 期间无需在节点之间传输它们。由于每个节点都有完整的随机矩阵，可以在适当的数据划分帮助下本地进行 NMF 迭代。因此，我们的矩阵草图方法不仅减少了计算开销，还减少了通信成本。此外，由于草图也移动了每个原始 NMF 子问题的最优解，我们提出了与理论保证它们收敛到原始子问题的稳定点的子问题求解器配对。

为了解决联合数据上安全分布式 NMF 问题，我们首先展示了经过修改的 DSANLS 可以适应这种安全设置，但仅限于一个或有限次迭代。因此，我们设计了新的在同步设置中的 Syn-SD 和 Syn-SSD 方法。它们后来被扩展到异步设置（即客户端/服务器）中的 Asyn-SD 和 Asyn-SSD。Syn-SSD 提高了 Syn-SD 的收敛速度，而没有增加太多的额外通信成本。它还通过草图减少了计算开销。所有提出的算法都是安全的，并有保证。安全分布式 NMF 问题本质上是困难的。所有参与方在过程中不应该能够推断出其他方的保密信息。据我们所知，我们是第一个研究联合数据上的 NMF。

总之，我们的贡献如下：

- DSANLS 是第一个利用矩阵草图减少每个 NLS 子问题的问题大小的分布式 NMF 算法，并且可以应用于密集和稀疏输入矩阵，并且有收敛保证。
- 我们提出了一个新颖的特别设计的子问题求解器（近端坐标下降），它帮助 DSANLS 更快地收敛。我们还讨论了使用投影梯度下降作为子问题求解器，表明它等同于原始（非草图）NLS 子问题上的随机梯度下降（SGD）。
- 对于安全分布式 NMF 问题，我们提出了在同步设置中高效的 Syn-SD 和 Syn-SSD 方法，并将它们扩展到异步设置。通过草图，它们的计算成本显著降低。它们是联合数据上第一个安全的分布式 NMF 方法。
- 我们使用几个（密集和稀疏的）真实数据集进行了广泛的实验，证明了我们的提议的有效性和可扩展性。

记号：对于矩阵 A，我们用 $A_{i:j}$ 来表示 A 的第 i 行第 j 列的元素。此外，i 或 j 可以省

Algorithm 1 Two-Block Coordinate Descent: Framework of Most NMF Algorithms

Input: M

Input: Iteration number T

initialize $U^0 \geq 0, V^0 \geq 0$

for $t = 0$ to $T - 1$ **do**

  $U^{t+1} \leftarrow \text{update}(M, U^t, V^t)$

  $V^{t+1} \leftarrow \text{update}(M, U^{t+1}, V^t)$

end for

return U^T and V^T

略来表示一列或一行，即 $A_{i:}$ 是 A 的第 i 行，$A_{:j}$ 是 A 的第 j 列。而且，i 或 j 可以被指标的子集所替代。例如，如果 $I \subset \{1, 2, \ldots, m\}$，$A_{I:}$ 表示由 I 中所有行构成的 A 的子矩阵，而 $A_{:J}$ 是由子集 $J \subset \{1, 2, \ldots, n\}$ 中所有列构成的 A 的子矩阵。

2 背景与相关工作

在本节中，我们首先介绍了非负矩阵分解（NMF）及其在分布式环境中的安全问题。然后，我们详细讨论了与本文相关的先前工作。

2.1 预备知识

2.1.1 NMF 算法

通常，NMF 可以定义为一个优化问题，如下所示：

$$\min_{U \in \mathbb{R}^{n \times k}, V \in \mathbb{R}_+^{n \times k}} \|M - UV^\top\|_F. \tag{1}$$

其中，$\|X\|_F = \sqrt{\sum_{ij} x_{ij}^2}$ 是矩阵 X 的 Frobenius 范数。

问题（1）难以直接求解，因为它是非凸的。因此，几乎所有的 NMF 算法都利用了两个块坐标下降方案（如算法 1 所示）：在优化其中一个因子 U 或 V 时，同时保持另一个因子固定。通过固定 V，我们可以通过解决非负最小二乘（NLS）子问题来优化 U：

$$\min_{U \in \mathbb{R}_+^{m \times k}} \|M - UV^T\|_F. \tag{2}$$

同样，如果我们固定 U，问题就变成了：

$$\min_{V \in \mathbb{R}_+^{n \times k}} \|M^T - VU^T\|_F. \tag{3}$$

在两个块坐标下降方案（精确或不精确）中，提出了不同的子问题求解器。第一个广泛使用的更新规则是乘法更新（MU），它基于主要化-最小化框架，其应用保证了目标函数的单调递减。另一种被广泛研究的方法是非负最小二乘（ANLS），它代表了一类方法，其中 U 和 V 的子问题按照算法 1 中描述的框架被精确求解。ANLS 被保证收敛到一个稳定点，并且在

实践中表现良好，特别是当使用活动集、投影梯度、拟牛顿或加速梯度方法作为子问题求解器时。因此，本文专注于 ANLS。

2.1.2 安全分布式 NMF

安全分布式 NMF 在实际应用中具有重要意义。假设两家医院 A 和 B 拥有相同表型的不同临床记录，M_1 和 M_2（即矩阵），为了法律或商业考虑，需要确保没有任何一家医院能直接向另一家医院透露个人记录。为了表型分类的目的，可以独立地应用 NMF 任务（即，$M_1 \approx U_1V_1$ 和 $M_2 \approx U_2V_2$）。然而，由于 M_1 和 M_2 在表型上具有相同的模式，可以采用连接矩阵 $M = [M_1, M_2]$ 作为 NMF 的输入，并通过共享相同项目（即表型）的潜在表示 U 来获得更好的用户（即患者）潜在表示 V_1 和 V_2：

$$M = [M_1 \quad M_2] \approx [U_1V_1^T \quad U_2V_2^T] = U[V_1^T \quad V_2^T] \tag{4}$$

在整个分解过程中，安全分布式 NMF 应保证参与方 A 只能访问 M_1、U 和 V_1，而参与方 B 只能访问 M_2、U 和 V_2。值得注意的是，上述两方的分布式 NMF 问题可以直接扩展到 N 方。在安全分布式 NMF 中，所有参与方对联合数据的要求实际上是所谓的 t-私密协议（在定义 1 中 $t = N - 1$），它源自安全函数评估。在本文中，我们将使用它来评估分布式 NMF 是否安全。

定义 1（t-私密协议）：所有 N 方都诚实地遵循协议，但它们也好奇地想根据自己的数据推断其他方的私有信息（即，诚实但好奇）。如果任何 t 方在协议结束时合谋，除了他们自己的输出外，他们什么也学不到，那么协议就是 t-私密的。

请注意，单个矩阵转置操作可以将列连接矩阵转换为行连接矩阵。在本文中，我们不失一般性地只考虑按行连接的矩阵场景。

安全分布式 NMF 问题本质上是困难的。首先，参与方 A 需要与参与方 B 一起解决 NMF 问题以获得 U 和 V_1，同时，参与方 A 在整个过程中不应该能够推断出 V_2 或 M_2。这种安全要求使得它与传统的分布式 NMF 问题完全不同，后者的数据划分已经违反了安全要求。

2.2 相关工作

接下来，我们简要回顾了与本文相关的三个研究领域。

2.2.1 加速 NMF

并行 NMF 算法在文献中得到了很好的研究。然而，与单机设置中的并行设置不同，分布式设置中的数据共享和通信具有相当的成本。因此，我们需要专门的 NMF 算法来处理分布式环境中的大规模数据。这方面的第一个方法是基于 MU 算法的，主要关注稀疏矩阵，并应用精心的数据划分以最大化数据局部性和并行性。后来，CloudNMF，一个基于 MapReduce 的 NMF 算法，被实现并测试在大规模生物数据集上。另一个分布式 NMF 算法利用块更新进行本地聚合和并行。它还尽可能频繁地使用最近更新的数据，这比传统的并发对应物更有效。除了 MapReduce 实现之外，Spark 也因其在迭代算法中的优势而受到关注，例如使用 MLlib。最后，还有使用 X10 和 GPU 的实现。

最近和相关工作中最相关的是 MPI-FAUN，这是第一个使用 MPI 进行处理器间通信的 NMF 实现。MPI-FAUN 是灵活的，可以用于广泛的 NMF 算法，这些算法迭代求解 NLS 子问题，包括 MU、HALS 和 ANLS/BPP。MPI-FAUN 利用 U 和 V 的行的局部更新计算的独立性，以应用通信最优矩阵乘法。简而言之，完整矩阵 M 被分割到二维处理器网格上，不同节点上保留了 U 和 V 的多个副本，以减少 NMF 算法迭代期间节点之间的通信。

2.2.2 矩阵草图

矩阵草图是一种技术，之前已在数值线性代数、统计学和优化中使用。其基本思想如下。假设我们需要找到方程 $Ax = b$ 的解 x；$A \in \mathbb{R}^{m\times n}, b \in \mathbb{R}^m$。而不是直接求解这个方程，在矩阵草图的每次迭代中，生成一个随机矩阵 $S \in \mathbb{R}^{d\times m}(d \ll m)$，我们转而求解以下问题：$(SA)x = Sb$。显然，第一个方程的解也是第二个方程的解，但反之则不然。然而，现在问题的大小已从 $m \times n$ 减少到 $d \times n$。通过适当生成随机矩阵 S 和解决第二个方程的子问题的合适方法，可以保证我们通过迭代应用这种草图技术逐渐接近第一个方程的解。

据我们所知，只有一篇先前的工作将双重随机投影纳入 NMF 问题，在集中式环境中，与 SANLS（我们 DSANLS 算法的集中式版本）有类似的想法。然而，Wang 等人没有提供高效的子问题求解器，他们的方法在实际实验中不如非草图方法有效。此外，他们的工作中没有考虑数据稀疏性。此外，对于具有双重随机投影的 NMF，没有提供理论保证。简而言之，SANLS 与 [30] 不同，DSANLS 不仅仅是 [30] 的分布式版本。我们在本文中提出的方法在实践中是高效的，并且具有强大的理论保证。

2.2.3 联合数据上的安全矩阵计算

在联合数据挖掘中，各方合作在他们的未加密数据的联合上执行数据处理任务，而不会泄露他们的私有数据给其他参与者。文献中的大量工作研究了联合数据上的联邦矩阵计算算法，例如隐私保护梯度下降、特征向量计算、奇异值分解、k 均值聚类和谱聚类等。安全多方计算（MPC）被应用于保护参与方的隐私（例如，安全加法、安全乘法和安全点积）。这些安全 MPC 协议计算任意函数在 n 方之间，容忍多达 $t < \frac{1}{2}n$ 的损坏方，每比特成本为 $V(n)$。这些协议对于像安全 NMF 这样的特定任务来说太通用了。我们提出的协议不包含昂贵的 MPC 乘法协议，同时容忍多达 $n-1$ 个损坏的（静态的，诚实但好奇的）参与方。最近，Kim 等人提出了一种联邦方法，使用交替方向乘子法（ADMM）张量分解在多家医院之间学习表型；Feng 等人开发了一个隐私保护张量分解框架，用于在联合云设置中处理加密数据。

3 本文方法

3.1 本文方法概述

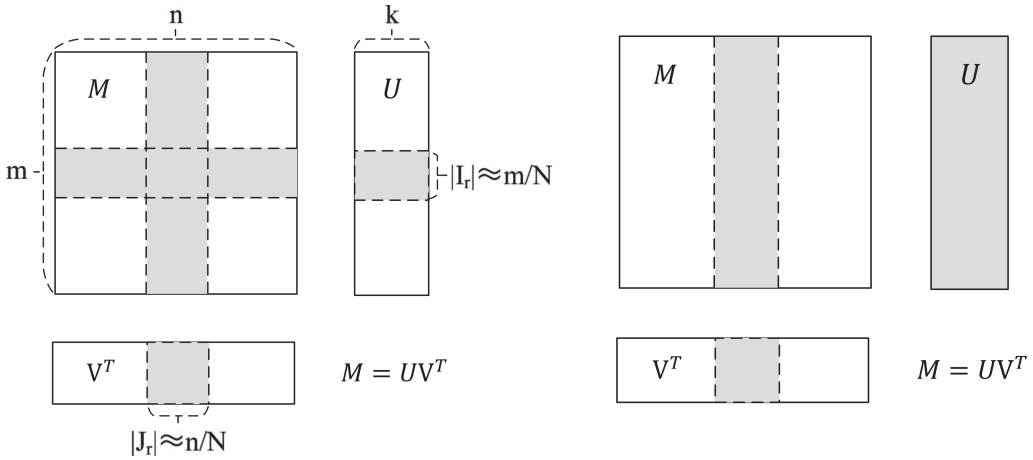
此部分对本文将要复现的工作进行概述。

3.1.1 DSANLS：分布式草图交替非负最小二乘概述

原文第节介绍了 DSANLS（分布式草图交替非负最小二乘）方法，旨在加速分布式环境中的非负矩阵分解（NMF）。DSANLS 遵循 ANLS（交替非负最小二乘）的框架，并通过引入

矩阵草图技术来减小每个 NLS（非负最小二乘）子问题的问题规模，从而降低整体计算成本。这种方法不仅减少了计算开销，还减少了通信成本，因为每个节点只需要处理矩阵的一个子集。方法示意如图 1(a)1所示。

DSANLS 的核心思想是在每次迭代中，通过随机矩阵对子问题中的矩阵进行草图化，从而降低问题的维度。这种方法减少了每个子问题的规模，同时保持了算法的收敛性。此外，DSANLS 不需要在节点间传输大型矩阵，而是在每个节点独立生成相同的随机矩阵，进一步降低了通信开销。



(a) DSANLS (b) Secure NMF

图 1. 将数据划分为 N 个节点，节点 r 的数据用阴影表示

3.1.2 安全分布式 NMF 概述

原文第 4 节探讨了如何在联合数据上进行安全的分布式 NMF，方法示意如图 1(b)1所示。在这种设置中，多个参与方拥有各自的保密数据集，并且他们希望在不泄露各自数据的情况下，共同利用这些数据进行 NMF 以获得更好的性能。

首先，我们指出 DSANLS 可以通过适当的修改来适应安全设置，但这种适应性仅限于有限次数的迭代。为了解决这个问题，我们设计了两种同步设置下的方法：Syn-SD 和 Syn-SSD。这两种方法通过在每个节点上独立更新矩阵 U 和 V 的副本，然后定期聚合这些副本来实现安全的 NMF。Syn-SSD 进一步通过在节点间交换矩阵的草图版本来减少通信成本，并提高算法的收敛速度。

然后，我们还提出了异步设置下的 Asyn-SD 和 Asyn-SSD 方法，以适应工作负载不平衡的情况。这些异步方法允许节点在没有全局同步屏障的情况下独立地与服务器交换信息，从而提高了算法的效率和可扩展性。

这些方法在保护参与方数据隐私的同时，实现了分布式 NMF 的加速和安全。通过在多个真实数据集上的广泛实验，我们证明了这些方法的有效性和优越性。

4 复现细节

4.1 与已有开源代码对比

所有代码均参考 https://github.com/qianyuqiu79/DSANLS/中的开源代码。

4.2 实验环境搭建

实验环境使用课题组部署的集群，29 个活跃节点的 Linux 集群，2.27TB 内存，总共 792 个核。

根据开源代码中的说明,复现这个工作的代码的先决条件是实现需要消息传递接口(MPI)以及英特尔数学内核库（MKL)，并且在编译之前确保正确设置环境变量 “MKLROOT”，并且 MPI 可以被编译器找到。

4.3 使用说明

基本使用方法

./distNMF <input file name> k [optional arguments]

Optional arguments 可以是:

- -i< 迭代次数 >
- -s< 行草图比率 >< 列草图比率 > 草图比率是介于 0 - 1 之间的浮点数
- -o< 详细的频率 >
- -m< 草图法的选择 >，选择包括：0 - 子采样, 1 - 高斯, 2 - 均衡
- -g<> 使用投影梯度下降来求解子问题（默认求解器是正则化坐标下降）
- -t<alpha> <beta>，其中 alpha 和 beta 是决定步长变化率的参数, 对于坐标下降，$\mu = \beta * iter^{\alpha}$, 对于梯度下降，$\eta = \alpha / \ 1.0 + iter * \beta$
- -u< 约束 >，U 和 V 的初始化随机数的上界

输入矩阵格式：输入矩阵是一个二进制文件，存储在根节点上（程序会自动将数据分发给其他节点)。

浮点数精度: 浮点数的精度（单精度或双精度）可以通过更改 “common.h” 中的宏 “DOUBLE_PRECISION” 来设置。注意，输入文件中浮点数的精度必须与此一致。

4.4 创新点

这篇论文的创新点主要包括以下几个方面：

- 分布式草图交替非负最小二乘（DSANLS）框架：提出了 DSANLS 框架，这是一个分布式 NMF 算法，它通过矩阵草图技术减少每个非负最小二乘子问题的大小，从而降低计算和通信成本，并保证算法的收敛性。
- 矩阵草图技术的应用：在 NMF 中创新性地应用矩阵草图技术，减少了子问题的规模，同时保持了算法的收敛性，这对于处理大规模数据集尤为重要。
- 安全分布式 NMF 方法：设计了在同步和异步设置下均具有安全保证的高效分布式 NMF 方法，这些方法是首次在联合数据上实现 NMF 的同时保护数据隐私。

- 隐私保护：在多个参与方的数据联合使用中，提出了保护隐私的方法，确保在 NMF 过程中不会泄露任何个体信息，这对于涉及敏感数据的应用场景（如医疗数据）尤为重要。
- 实验验证：在多个真实数据集上进行了广泛的实验，验证了所提出方法的有效性和可扩展性，展示了其在实际应用中的优越性能。
- 理论分析：提供了 DSANLS 算法的理论分析，包括计算和通信复杂度分析以及收敛性分析，为算法的设计和应用提供了理论支持。
- 适应不同工作负载的算法设计：针对均匀和不平衡工作负载，分别设计了同步和异步的分布式 NMF 算法，提高了算法在不同场景下的适用性和效率。

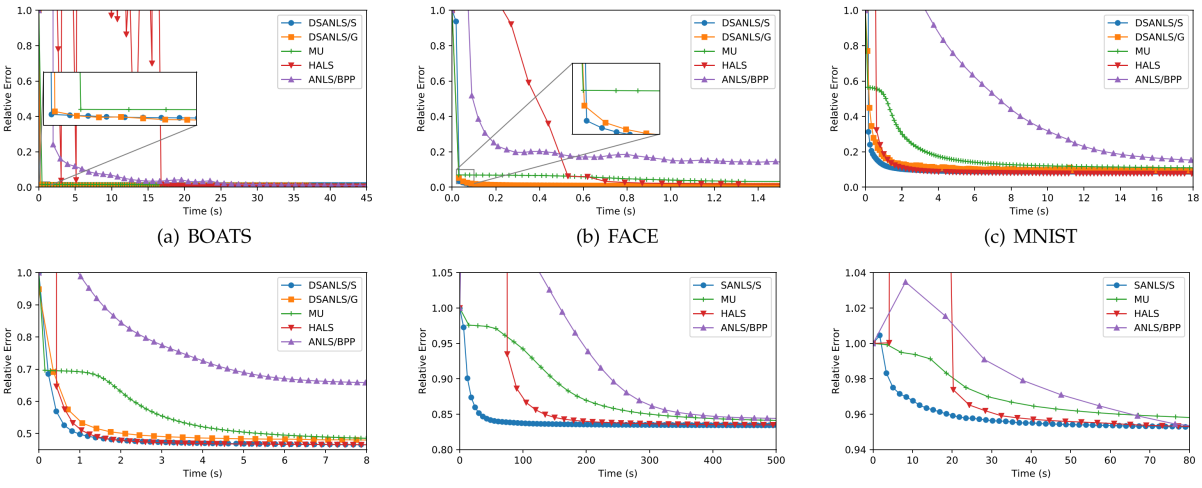
这些创新点表明，这篇论文不仅在技术层面提出了新的方法来加速和保护分布式 NMF，而且在理论和实践层面都进行了深入的研究和验证。

5 实验结果分析

实验使用几个（密集和稀疏的）真实数据集进行评估，这些数据集用于不同的 NMF 任务，包括视频分析、图像处理、文本挖掘和社区检测。默认情况下，我们使用 10 个节点，并将分解秩 k 设置为 100。由于在大型数据集 RCV1 和 DBLP 上使用高斯随机矩阵速度太慢，我们只为它们使用子采样随机矩阵。

5.1 加速一般 NMF 的评估

5.1.1 性能比较



(a) BOATS (b) FACE (c) MNIST

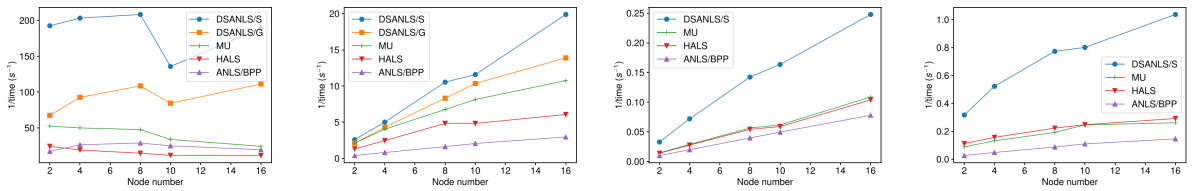
(d) GISETTE (e) RCV1 (f) DBLP

图 2. 一般分布 NMF 随时间的相对误差

由于每次迭代的时间被我们提出的 DSANLS 显著减少，与 MPI-FAUN 相比，在图 2 中，我们展示了 DSANLS 和 MPI-FAUN 实现的 MU、HALS 和 ANLS/BPP 在 6 个真实公共数据集上的相对误差随时间变化的情况。观察到 DSANLS/S 在所有 6 个数据集中表现最佳，尽管 DSANLS/G 具有更快的每次迭代收敛速率。MU 相对收敛速度慢，并且通常具有较差的收敛结果；另一方面，HALS 可能在早期轮次中振荡，但收敛相当快，并且得到一个很好的解。

令人惊讶的是，尽管 ANLS/BPP 被认为是最先进的 NMF 算法，但它在所有 6 个数据集中并不表现良好。正如我们将看到的，这是由于其每次迭代的高成本。

5.1.2 可扩展性比较

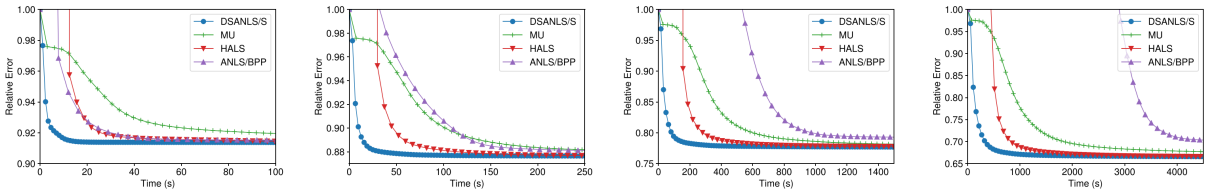


(a) FACE (b) MNIST (c) RCV1 (d) DBLP

图 3. 对于一般分布的 NMF，每次迭代时间的倒数作为簇大小的函数

我们改变了集群中使用的节点数量，从 2 到 16，并记录了每个算法 100 次迭代的平均时间。图 3 显示了每次迭代时间的倒数作为使用的节点数量的函数。所有算法都表现出良好的可扩展性，对于所有数据集（几乎是一条直线），除了 FACE（即图 3a）。FACE 是最小的数据集，其列数为 300，而 k 默认设置为 100。当 n/N 小于 k 时，复杂度由 k 主导，因此，增加节点数量不会降低计算成本，但可能会增加通信开销。总的来说，我们可以观察到 DSANLS/子采样与所有其他算法相比具有最低的每次迭代成本，而 DSANLS/高斯与 MU 和 HALS 具有类似的成本。ANLS/BPP 具有最高的每次迭代成本，解释了图 2 中 ANLS/BPP 的不良性能。

5.1.3 改变 k 值的性能比较



(a) k=20 (b) k=50 (c) k=200 (d) k=500

图 4. 一般分布 NMF 随时间的相对误差，变化 k 值

尽管调整分解秩 k 不在本文范围内，我们比较了 DSANLS 与 MPI-FAUN 在 RCV1 上改变 k 值从 20 到 500 的性能。从图 4（$k = 100$）观察到 DSANLS 在所有 k 值上都优于最先进的算法。自然，所有算法的相对误差随着 k 的增加而减少，但它们也需要更长的时间来收敛。

6 总结与展望

本研究针对分布式非负矩阵分解（NMF）的加速问题展开深入探讨，复现了一种新型分布式 NMF 算法——DSANLS。该算法在传统 ANLS（交替非负最小二乘）框架的基础上，创新性地引入矩阵草图（Matrix Sketching）技术，通过降维处理有效减少了每个 NLS 子问题的规模，从而显著提升了计算效率。为验证算法的实际性能，我们在多个真实数据集上进行

了系统性实验，实验结果充分证明了 DSANLS 算法在处理高维矩阵数据时具有良好的可扩展性和计算效率。

然而，本研究仍存在一些局限性。由于技术限制，我们未能成功部署 scureNMF 安全框架，这使得分布式 NMF 算法的安全性问题未能得到充分验证和解决。展望未来，我们计划对针对分布式 NMF 的安全性问题展开深入研究探索以及 DSANLS 算法在更多实际应用场景中的潜力

通过本研究复现，我不仅学习了一种为大规模矩阵分解提供了高效的解决方案，也为我学习分布式机器学习算法的优化与安全保护提供了新的思路。

参考文献

[1] Yuqiu Qian, Conghui Tan, Danhao Ding, Hui Li, and Nikos Mamoulis. Fast and secure distributed nonnegative matrix factorization. *IEEE Transactions on Knowledge and Data Engineering*, 34(2):653–666, 2020.