arXiv:2301.13079v1 [cs.DS] 30 Jan 2023

快速组合　用于最小最大相关聚类的算法

Sami Davies* Benjamin Moseley† Heather Newman†

2023年1月31日

摘要

我们提出了针对最小最大目标的快速相关聚类算法，可在完全图上提供常数因子近似。这些算法是该问题首批纯组合近似算法。我们在顶点集合上构造了一种新颖的半度量，称为相关性度量，它能指示我们的聚类算法哪些节点对应该属于同一个簇。本文通过实证表明，与以往工作相比，我们的算法仅以牺牲少量目标质量为代价，便显著提升了运行时间。此外，我们的算法可扩展至更大的网络，而已知算法在这些网络上实际上难以应用。

1 引言

在相关聚类中，输入为一个图 $G = (V,\ E)$ 其中每条边都标记为正标签 $(+)$ 或负标签 $(-)$。如果两个顶点由正边连接，则表示它们相似，并希望被聚类到一起。相反，负边表示顶点彼此不相似，希望被置于不同的簇中。我们作出一个常见假设，即G 是完全图；也就是说，每对顶点之间都有一条带标签的边。

如果一条边为负边，且连接同一簇中的两个顶点，或者为正边，且连接不同簇中的两个顶点，则称该边相对于某个聚类处于不一致 状态。请注意，对于三个或更多顶点，正负标签可能使得任何 聚类都存在一些不一致边，例如三个顶点之间的三条边中有两条正边和一条负边。

相关聚类的目标是找到一个聚类，使能够反映边不一致程度的目标函数最小化。最常考虑的目标是在顶点上最小化不一致的 ℓ_p-范数。在这里，每个顶点 u 都有若干条与其相邻的不一致边，数量为 $y(u)$，目标是最小化 $\sqrt[p]{\sum_{u\in V} y(u)^p}$。研究最为充分的目标是 $p = 1$，此时最小化不一致边的总数；在这种情况下，Pivot算法是一种常用的组合算法 [4]。当 $p = \infty$ 时，目标是最小化任意顶点上的最大不一致边数，这体现了聚类中一种良好的公平性概念。

我们的结果聚焦于 ℓ_∞-范数目标，我们将其称为最小最大 目标[1]。最小最大相关聚类最初受到社区检测应用的启发，这类应用不包含对立者，即社群中不存在与其社群在很大程度上不一致的成员[21]。此类问题出现在推荐系统、生物信息学和社会科学中 [10, 17,23, 21]。

*西北大学。获NSF计算创新奖学金支持。

†卡内基梅隆大学。Benjamin Moseley和Heather Newman部分获得了谷歌研究奖、Inform研究奖、aCarnegieBosch JuniorFacultyChair，以及NSF资助项目CCF-2121744和CCF-1845146的支持。[1]这也被称为极小极大或 ℓ_∞ 目标。

Milenkovic 和 Puleo（2016年）开创了对最小最大相关聚类及其他 ℓ_p-范数的研究。对于所有 $p \geqslant 1$，他们给出了 48- 近似算法。Charikar 、 Gupta 和 Schwartz （ 2017 年）将其改进为 7- 近似算法，Kalhan、Makarychev 和 Zhou（2019年）进一步将其改进为目前已知的最佳近似[2] 5。Milenkovic 和 Puleo（2016年）观察到，Pivot 对于 ℓ_∞ 范数会失效，尽管它对于ℓ_1 范数的期望近似比为 3-近似（参见附录 A 中的讨论）。关于相关聚类的更多内容请参见第 1.2 节，其中包括对最小最大等公平/局部目标的研究兴趣。

以往针对最小最大相关聚类的算法会对半定规划或线性规划解进行舍入 [21,8, 1, 16]。这些松弛规模庞大，至少需要 $|V|^3$ 个约束和 $|V|^2$ 个变量。因此，即使对于规模适中的图（例如，包含 300–500 个顶点的图），求解它们也不切实际。一个问题由此浮现：是否存在一种快速算法，能够解决最小最大相关聚类问题并提供强有力的理论保证？此外，一个既有助于回答这一问题、又本身颇具意义的方向是开发组合算法。一项关键挑战在于，若不使用线性规划或半定规划，就不清楚如何与最优解进行比较，因为目前尚无已知的、非显然的下界。

1.1 我们的贡献

本文为最小最大相关聚类提供了快速算法。我们的算法是首批具有理论保证的组合算法。使这些算法成为可能的关键技术洞见，是在顶点集合上引入一种半度量，我们称之为**相关性度量**。相关性度量可用于为问题的线性规划设置变量，从而使所得解满足可行性，并且可以证明其与 Min Max 目标下的不一致数最优值接近。因此，与以往工作类似，我们使用线性规划与最优解进行比较，但由于线性规划仅用于分析，无须实际求解。我们的相关性度量为相关聚类和线性规划带来了新的见解。

令 ω 表示 $O(n^\omega)$ 矩阵乘法在 $n \times n$矩阵上的运行时间常数，其中 $n = |V|$。

定理 1. 存在一种算法，可在 $O(n^\omega)$时间内对完全图上的最小最大相关聚类实现 *40*-近似。

当图的正边构成稀疏图时，运行时间可以显著缩短；这在实践中很常见。

推论 1. 假设 V 中的所有顶点的 $+$ 度数至多为 Δ，并且给定每个顶点的正邻居列表。存在一种算法，可在 $O(n\Delta^2$ 对数 $n)$时间内对完全图上的最小最大相关聚类实现 *40*-近似。

因此，我们有一种适用于稀疏图的近线性时间算法。这显著改进了先前的工作；据我们所知，即使对于稀疏图，其运行时间也不优于 $O(n^{2\omega})$ （见附录 A）。

接下来，我们将展示如何近似相关性度量以改善运行时间，即使图是稠密的，但近似因子会有所损失。

定理 2。 固定足够小的 $\varepsilon > 0$ 。对于某个仅取决于 ε 的常数 $c(\varepsilon)$ ，存在一种随机化算法，以概率 $1 - O(1/n)$ 在时间 $O(n^2$ 对数 $n/\varepsilon^2)$ 内，在完全图上对最小最大相关聚类实现 $c(\varepsilon)$- 近似。

最后，我们展示了我们的理论能够预测实践结果（参见第6节）。就目标值而言，我们的算法表现与 Kalhan、Makarychev 和 Zhou（2019）的算法相近，同时大幅缩短了运行时间，使其能够切实扩展到约包含 10,000 个顶点的图[3]。此外，我们的算法找到的簇通常具有实际意义，因为在真实世界和合成实例中，它们往往能够部分发现“真实”簇。

[2]请注意，所有这些结果仍然仅适用于完全图。[3]作为比较，Kalhan、Makarychev 和 Zhou 的算法在具有 $\approx$ 200 个顶点的图上运行时间超过 10 分钟。

1.2 相关工作

在相关聚类中，最常研究的目标是最小化不一致边的总数，这等价于最小化不一致向量的 ℓ_1 范数 $y \in \mathbb{R}^{|E|}_{\geqslant 0}$。需要注意的是，在边带权的情况下，可以研究最小化 ℓ_1 范数。该模型由 Bansal、Blum 和 Chawla（2004年）提出，其动机来自许多应用，例如图像分割、自然语言处理、基因表达模式聚类和区域选址规划 [24, 20, 7, 13]。Ailon、Charikar 和 Newman（2008年）提出的著名 Pivot算法在期望意义下达到了 3-近似。直到最近，在完全无权图上的 ℓ_1 相关聚类问题中，最佳近似算法仍是 Chawla、Makarychev、Schramm 和 Yaroslavtsev（2015年）给出的 $(2.06+\epsilon)$-近似；然而，Cohen-Addad、Lee 和 Newman（2022年）的工作打破了 2 这一阈值（他们实现了 $(1.994+\epsilon)$-近似），并成功利用了 Sherali-Adams层级的 $O(1/\epsilon^2)$ 轮。

最小最大相关聚类由 Puleo 和 Milenkovic（2016年）提出。他们证明，即使在完全图上，针对 Min Max目标的相关聚类也是 NP困难（见其论文附录 C）；在算法方面，他们为完全图和完全二部图提出了 48-近似算法。不久之后，Charikar、Gupta 和 Schwartz（2017年）为在相同图上最小化 ℓ_p 范数提出了 7-近似算法。对于 Min Max目标，即 $p=\infty$， 他们在一般加权图上提出了一个 $O(\sqrt{n})$-近似算法。Kalhān、Makarychev 和 Zhou（2019年）进一步推广了相关聚类局部目标方面的已知结果，提出了在一般加权图上最小化 ℓ_p范数的近似算法。这项工作还针对完全图和完全二部图上的 ℓ_p （包括$p=\infty$）目标，提出了一个 5-近似算法。

在此前所有研究相关聚类中任意 ℓ_p 范数目标的论文里，运行时间都依赖于求解一个至少包含 $\Omega(n^2)$ 个变量和 $\Omega(n^3)$ 个约束的线性规划 [21,8, 16]。我们看不出有什么明确的方法可以利用该线性规划的结构，确保求解时间低于 $O(n^{2\omega})$，即使对于稀疏图也是如此（参见附录 A 中的讨论）。

其他一些局部目标或体现某种公平性概念的目标也已得到研究 [3, 6, 2, 14, 15, 1]。例如，Ahmadi、Khuller 和 Saha（2019年）旨在寻找一种聚类，使得对于每个点，其到所属簇中其他点的平均距离不超过其到另一簇中点的平均距离。他们将半定规划（SDP）的解进行舍入，这一思路基于最小最大 k 均衡划分的相关研究。此外，Jafarov 等人（2021年）研究了 ℓ_p 目标，并进一步根据边是否相似对权重作出假设。

2 符号与预备知识

设 $G=(V,\ E)$ 是一个具有 n 个顶点和自环的完全图，其中每条边都带有正 $(+)$ 或负 $(-)$ 标签。令 E^+ 为 G 的正边集合，令 E^- 为负边集合。对于顶点 $u \in V$，定义 $N_u^- = \{v \in V \mid (u,\ v) \in E^-\}$ 为 u的负邻域，并定义 $N_u^+ = \{v \in V \mid (u,\ v) \in E^+\}$ 为 u的正邻域。

根据某个聚类，若 $e \in E^+$ 、u 和 v 处于不同的簇中，或者 $e \in E^-$ 、 u 和 v 处于同一个簇中，我们就称边 $e=(u,\ v) \in E$ 处于不一致状态。为便于计算，我们假设每个顶点都有一个正自环，即对所有 $v \in V$， $(v,\ v) \in E^+$。注意，这样做不会改变不一致边的集合；由于一个顶点总是与自身处于同一个簇中，自环永远不会造成不一致。以下事实对 G成立：

事实 1. 对于任意 $u \in V$,

$$n = |N_u^+| + |N_u^-|.$$

[4]稍微修改我们后面使用的邻域交集定义，可以避免自环。不过，直接假设存在正自环，记号和表述会更加简洁。

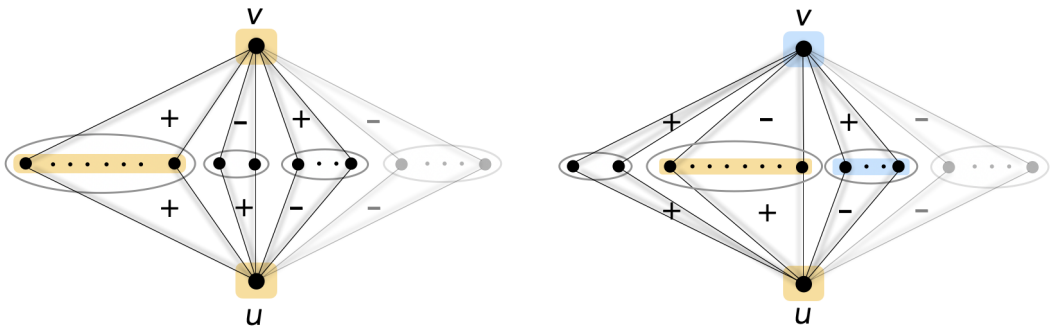

图 1：左： $|N_u^+ \cap N_v^+|$ 显著大于 $|(N_u^- \cap N_v^+) \cup (N_u^+ \cap N_v^-)|$。如果 v 和 u 位于同一个簇中，则对于所有 $w \in N_u^+ \cap N_v^+$，边 $(u、w)$ 和 $(w、v)$ 都不会形成不一致。**右：** $|(N_u^- \cap N_v^+) \cup (N_u^+ \cap N_v^-)|$ 显著大于 $|N_u^+ \cap N_v^+|$。如果 v 和 u 位于不同的簇中（黄色和蓝色），则对于所有 $w \in (N_u^- \cap N_v^+) \cup (N_u^+ \cap N_v^-)$，边 $(u、w)$ 和 $(w、v)$ 都不会形成不一致。

事实2. 令 $u、v \in V$，可能还包括 $u = v$。则

$$n = |N_u^+ \cap N_v^+| + |N_u^- \cap N_v^-| + |N_u^+ \cap N_v^-| + |N_u^- \cap N_v^+|.$$

事实 *2* 的证明。注意到 $N_u^+ = (N_u^+ \cap N_v^+) \cup (N_u^+ \cap N_v^-)$ 和 $N_u^- = (N_u^- \cap N_v^+) \cup (N_u^- \cap N_v^-)$。现在代入事实 1，即可得到事实 2。 □

令 $C = (C_1、\ldots\ldots、C_k)$ 表示将 V 划分为 k 个簇的一个划分。若 u 属于簇 C_i，则称 $C(u) = C_i$。令 $C(u)$ 表示与 u 不在同一个簇中的所有顶点的集合。对于给定的聚类 C，令 $y_C \in \mathbb{Z}_+^n$ 表示 C 的不一致向量，其索引为 V；对于 $u \in V$，坐标 $y_C(u)$ 表示与 u 关联且在 C 中属于不一致的边的数量。当上下文能够明确聚类时，我们将省略下标 C。

相关性度量 我们基于顶点的正邻域和负邻域引入一种新颖的半度量。我们将证明，相关性度量 d 满足三角不等式，从而确立其为半度量。

定义 1。 对于所有 $u、v \in V$，相对于相关性度量，u 与 v 之间的距离为

$$\begin{aligned} d_{uv} &= 1 - \frac{|N_u^+ \cap N_v^+|}{n - |N_u^- \cap N_v^-|} \\ &= 1 - \frac{|N_u^+ \cap N_v^+|}{|N_u^+ \cap N_v^+| + |N_u^+ \cap N_v^-| + |N_u^- \cap N_v^+|} \end{aligned} \tag{1}$$

其中，(1)由事实2推出。注意，由于 $u、v$ 具有正自环，因此 d_{uv} 是定义良好的，故 $0 \leqslant d_{uv} \leqslant 1$。

我们对 d_{uv} 的定义给出一些直观解释。见图 1。固定三个互不相同的顶点 $u, v, w \in V$。我们将考察 u 和 v 的聚类如何影响与它们关联的不一致。首先请注意，如果 $w \in N_u^- \cap N_v^-$，只要 w 被分配到与 u 和 v 都不同的簇中，边 $(u, w), (v, w)$ 就不会产生不一致，无论 u 和 v 是否属于同一个簇。也就是说，w 不应影响 u 和 v 是否属于同一个簇。如果在归一化时不减去 $|N_u^- \cap N_v^-|$，我们就无法正确识别完美聚类；而在这里，当 C 是一个完美聚类时，$d_{uv} = \mathbb{1}_{\{C(u) \neq C(v)\}}$。

另一方面，如果 $w \notin N_u^- \cap N_v^-$，那么 u 和 v 是否属于同一个簇，会直接影响 $(u、w)$ 或 $(v、w)$ 中的任意一个是否处于不一致状态。根据事实2的证明，受影响的边为 $(u、w)$ 和 $(v，w)$（对于 $w \in (N_u^+ \cap N_v^+) \cup (N_u^- \cap N_v^+) \cup (N_u^+ \cap N_v^-)$）。如果 u 和 v 属于不同的簇，那么对于 $w \in (N_u^- \cap N_v^+) \cup (N_u^+ \cap N_v^-)$，$(u、w)$ 都会，并且

$(v、\ w)$ 仍可能保持一致。然而，如果 $w \in N_u^+ \cap N_v^+$，那么 $(u、\ w)$ 和 $(v、\ w)$ 中至少有一个处于不一致状态，因此，与 u 和 v 相连的不一致（总数）至少为 $|N_u^+ \cap N_v^+|$。当 u 和 v 属于同一个簇时，也可以看到类似的情况，但此时与 u 和 v 相连的不一致至少有 $|(N_u^- \cap N_v^+) \cup (N_u^+ \cap N_v^-)|$ 个。总体而言，如果相对于 $|(N_u^- \cap N_v^+) \cup (N_u^+ \cap N_v^-)|$ 而言 $|N_u^+ \cap N_v^+|$ 较大，那么（粗略地说）如果将 u 和 v 分配到同一个簇，与它们相连的不一致会少于将其分配到不同簇的情况。相对于 $|(N_u^- \cap N_v^+) \cup (N_u^+ \cap N_v^-)|$ 而言，$|N_u^+ \cap N_v^+|$ 越大，根据（1），u 和 v 在相关性度量方面就越接近。

贯穿全文，定义 $\widehat{d}_{uv} = d_{uv}$ 若 $(u,\ v) \in E^+$ 且 $\widehat{d}_{uv} = 1 - d_{uv}$ 如果 $(u,\ v) \in E^-$.

3 技术概述

3.1 线性规划松弛与 KMZ 算法

我们考虑该问题的标准线性规划松弛。不难看出，线性规划的约束在顶点上诱导出一个半度量空间。在 Kalhan、Makarychev 和 Zhou 的算法中，先求解线性规划以获得一个分数解，然后使用迭代舍入过程得到一个整数可行聚类。直观地说，线性规划半度量为解提供指导，其中彼此接近的顶点更有可能最终属于同一个簇。事实上，我们绝不会将距离足够远的顶点放入同一个簇中。

该线性规划包含变量 x_{uv}。在整数解中，$x_{uv} = 1$ 表示 u 和 v 属于不同的簇，而 $x_{uv} = 0$ 表示它们属于同一个簇。不一致向量为 y。

线性规划 3.1

$$\begin{aligned} & \min \max_{u \in V} y(u) \\ \text{s.t. } & y_u = \sum_{v \in N_u^+} x_{uv} + \sum_{v \in N_u^-} (1 - x_{uv}) && \forall u \in V \\ & x_{uv} \leqslant x_{uw} + x_{vw} && \forall u, v, w \in V \\ & 0 \leqslant x_{uv} \leqslant 1 && \forall u, v \in V. \end{aligned}$$

第一组约束确保不一致向量 y 根据 x变量进行设置。第二组约束对 x施加三角不等式。它们确保：如果 u 和w 属于同一个簇，且 v 和 w 属于同一个簇，那么 u 和 v 必须在一起。

我们将 Kalhan、Makarychev 和 Zhou 提出的算法称为 KMZ 算法。*KMZ 算法* 分为两个阶段：首先求解上述线性规划 3.1，然后在舍入算法 （算法 2）中对线性规划解进行舍入。我们在附录 A 中证明，舍入算法的运行时间为 $O(n^2)$，因此 KMZ 算法的运行时间主要取决于求解线性规划所需的时间；据我们所知，该时间的运行时间不优于$O(n^{2\omega})$。

3.2 组合算法

我们先概述定理 1 的证明思路；详情见第 4 节。我们的第一个算法分为两个步骤。首先，我们计算相关性度量 d_{uv} 对所有 $u, v \in V$ 的值，并通过设置 $x_{uv} = d_{uv}$ 得到线性规划 3.1 的一个手工构造解。然后，我们将d_{uv} 值作为输入，传入舍入算法（算法 2）。证明该过程能给出常数因子近似的分析分为三个关键步骤：

1. 相关性度量（或许出人意料地）满足三角不等式（见第 4.2 节），即 $x = d$ 是线性规划 3.1 的可行解。2. 将 $x = d$ 代入线性规划 3.1 得到的目标值，至多比一个最优整数解大 8 倍（见第 4.1 节）。这是定理 1 的核心，其难点在于分数成本中的不对称性，这迫使我们使用非局部收费论证。

3. Kalhan、Makarychev 和 Zhou（2019年）的舍入算法（算法 2）可用于将线性规划 3.1 的任意可行解舍入为整数解，同时目标值的损失至多为 5 倍。

我们的运行时间主要取决于对所有 $u, v \in V$ 计算 d_{uv} 所需的时间，该计算可在 $O(n^{\omega})$时间内完成。运行时间证明见第 4.3 节和附录 A。

对于稀疏图，无需计算所有顶点对 $u, v \in V$的相关性度量，只需在 $d_{uv} < 1$ 时计算 d_{uv}（此类顶点对至多有 $n\Delta^2$ 个）。其他顶点对的距离隐式地视为 1。随后，我们可以再次使用舍入算法（算法 2）。给定线性规划 3.1 的一个可行解，对于稀疏图，该舍入算法的运行时间为 $O(n\Delta^2 \log n)$。推论 1 的完整证明见第 4.3 节。

最后，在第 5 节中，我们将展示如何通过抽样估计 d_{uv}，从而得到一种算法：它以仍为常数的近似因子保证为代价，换取更快的运行时间，进而证明定理 2。为了计算这些估计值，我们从每个顶点的正邻域中抽取 $O(\log n)$ 个顶点。然后，我们利用这些样本来处理 u 和 v，以估计式 (1) 中的各项。然而，抽样能否奏效并非显而易见。首先，式 (1) 中并非每一项都能良好集中，因此我们需要利用d_{uv}的特殊结构。其次，初始估计值的分数成本无法得到控制。我们将证明，一个后处理阶段可以解决这一问题：在该阶段中，我们将一些估计值调低，将另一些调高。我们将后处理估计值作为舍入算法的输入。

4 相关性度量的性质

本节旨在证明定理 1，并建立相关性度量的结构性质。我们将 $\max_{u \in V}\{\sum_{v \in N_u^+} d_{uv} + \sum_{v \in N_u^-}(1 - d_{uv})\}$ 称为分数成本，即所设线性规划变量的目标值。第 4.1 节将界定分数成本。第 4.2 节证明相关性度量是半度量，因此满足线性规划 3.1。最后，我们在第 4.3 节推导定理 1。

4.1 界定分数成本

我们先给出几个有助于界定分数成本的命题 .

命题 1。 对于任意顶点 $u \in V$，令 $v \in N_u^+ \cap C(u)$。则

$$|N_u^+ \cap N_v^-| + |N_u^- \cap N_v^+| \leqslant y(u) + y(v).$$

命题 1 的证明。 取 $S = (N_u^+ \cap N_v^-) \cup (N_u^- \cap N_v^+)$。注意到 $N_u^+ \cap N_v^-$和 $N_u^- \cap N_v^+$不相交，因此 $|S| = |N_u^+ \cap N_v^-| + |N_u^- \cap N_v^+|$。固定 $w \in S$。由于 u和 v属于同一个簇，$(u、w)$和 $(v、w)$中恰有一个是不一致，并因此计入 $y(u)$或 $y(v)$。 □

用于证明定理 1 的主要技术结果是引理 1，不过为了使 ^引理的证明更加模块化，我们依赖断言 1 和断言 2。回顾一下，$d_{uv} = d_{uv}$ 如果 $(u, v) \in E^+$ 且$\sim d_{uv} = 1 - d_{uv}$如果 $(u, v) \in E^-$。

引理 1。 令 y 表示极小极大目标函数的最优解所对应的不一致向量。对于每个 $u \in V$ 以及 *OPT*= 最大值$_{z \in V}\, y(z)$,

$$\sum_{v \in V} \widehat{d}_{uv} \leqslant 8 \cdot \mathit{OPT}.$$

引理 1 的证明。 展开求和式，我们得到：

$$\sum_{v \in V} \widehat{d}_{uv} = \sum_{v \in N_u^+} d_{uv} + \sum_{v \in N_u^-} (1 - d_{uv})$$

$$= \underbrace{\sum_{v\in N_u^+} \frac{|N_u^+ \cap N_v^-| + |N_u^- \cap N_v^+|}{n - |N_u^- \cap N_v^-|}}_{S^+} + \underbrace{\sum_{v\in N_u^-} \frac{|N_u^+ \cap N_v^+|}{n - |N_u^- \cap N_v^-|}}_{S^-}$$

我们使用事实2重写第一个求和式。令第一个求和式表示为 S^+，第二个求和式表示为 S^-。要证明该引理，只需证明以下两个断言。

断言1。

$$S^+ \leqslant 3 \cdot OPT.$$

命题 2。

$$S^- \leqslant 5 \cdot OPT.$$

□

断言1的证明比命题 2的证明简洁得多。这部分是由于分数成本中的不对称性。

断言*1*的证明。注意，根据事实2，对于每个 $v \in V$，

$$|N_u^+| = |N_u^+ \cap N_v^-| + |N_u^+ \cap N_v^+| \leqslant n - |N_u^- \cap N_v^-|.$$

现在，我们可以通过按 u 和 v 是否属于同一个簇来划分总和，从而界定 S^+：

$$\begin{aligned}
S^+ &= \sum_{v\in N_u^+ \cap C(u)} \frac{|N_u^+ \cap N_v^-| + |N_u^- \cap N_v^+|}{n - |N_u^- \cap N_v^-|} + \sum_{v\in N_u^+ \cap \overline{C(u)}} d_{uv} \\
&\leqslant \frac{1}{|N_u^+|} \cdot \sum_{v\in N_u^+ \cap C(u)} (|N_u^+ \cap N_v^-| + |N_u^- \cap N_v^+|) + \sum_{v\in N_u^+ \cap \overline{C(u)}} 1 \\
&\leqslant \frac{1}{|N_u^+|} \cdot \sum_{v\in N_u^+ \cap C(u)} (y(u) + y(v)) + \sum_{v\in N_u^+ \cap \overline{C(u)}} 1 \qquad (2) \\
&\leqslant y(u) + \max_z y(z) + \sum_{v\in N_u^+ \cap \overline{C(u)}} 1 \qquad (3) \\
&\leqslant y(u) + \max_z y(z) + y(u) \qquad (4) \\
&\leqslant 3 \cdot \text{最优值}。
\end{aligned}$$

第 (2) 行由命题 1 得出，第 (3) 行由平均化论证得出（我们对 $|N_u^+ \cap C(u)|$ 项求和，然后除以 $|N_u^+|$），第 (4) 行则基于以下事实：如果 $(u,\ v) \in E^+$ 和 $u,\ v$ 属于不同的簇，那么 $(u,\ v)$ 是与 u 相邻的不一致。

□

命题 *2* 的证明。我们有

$$\begin{aligned}
S^- &= \sum_{v\in N_u^- \cap C(u)} (1 - d_{uv}) + \sum_{v\in N_u^- \cap \overline{C(u)}} \frac{|N_u^+ \cap N_v^+|}{n - |N_u^- \cap N_v^-|} \\
&\leqslant \sum_{v\in N_u^- \cap C(u)} 1 + \sum_{v\in N_u^- \cap \overline{C(u)}} \frac{|N_u^+ \cap N_v^+|}{n - |N_u^- \cap N_v^-|} \\
&\leqslant y(u) + \underbrace{\sum_{v\in N_u^- \cap \overline{C(u)}} \frac{|N_u^+ \cap N_v^+|}{n - |N_u^- \cap N_v^-|}}_{S},
\end{aligned}$$

其中，最后一个不等式源于以下事实：如果 $(u、\ v) \in E^-$ 和 $u、\ v$ 属于同一个簇，那么 $(u、\ v)$ 就是与 u相 incident 的一条不一致边。

接下来需要界定最后一个求和，将其记为 S。虽然我们可能会预期该论证会仿照断言1证明中对求和 $\sum_{v\in N_u^+\cap C(u)} d_{uv}$ 的界定方式，例如通过平均化论证，但这种方法行不通。总体而言，对于 $v\in N_u^+$，我们只能将求和 $\sum_{v\in N_u^+} d_{uv}$ 与和 u 及 v相 incident 的不一致边进行比较；但对于 $v\in N_u^-\cap C(u)$，我们必须将其与和除 u 和 v之外的顶点相 incident 的不一致边进行比较。直观地说，这是因为将 u 和 v 置于不同的簇中意味着，根据该聚类，$(u,\ v)$ 并不构成不一致，但任何顶点 $w\in N_u^+\cap N_v^+$必然与一条不一致边相 incident。此外，这种不一致可以计入另一个顶点——一个经过谨慎选择的 $v^*(w)\in C(w)$。

我们先将求和 S“翻转”。具体而言，我们希望将其视为对 $N_u^+\cap N_v^+$ 中的元素求和，并按 $n-|N_u^-\cap N_v^-|$ 进行缩放。

$$S=\sum_{v\in N_u^-\cap\overline{C(u)}}\frac{|N_u^+\cap N_v^+|}{n-|N_u^-\cap N_v^-|}=\sum_{w\in N_u^+}\sum_{\substack{v:\ v\in N_w^+,\\ v\in N_u^-\cap\overline{C(u)}}}\frac{1}{n-|N_u^-\cap N_v^-|}.$$

对每个 $w\in N_u^+$，定义：

$$V(w):=\{v\in V\mid v\in N_u^-\cap N_w^+\cap\overline{C(u)}\}$$

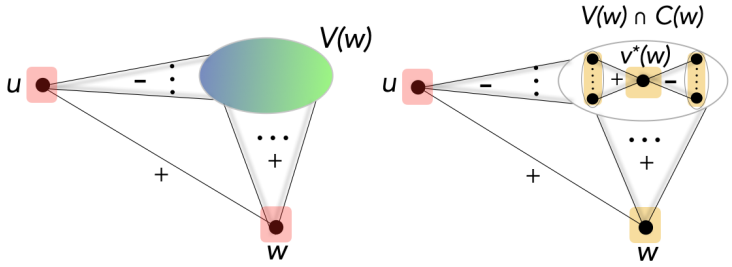

图 2：左：对求和 S_1 进行界定的示意图。这里，w 与 u（红色）属于同一簇，而 $V(w)$ 中的顶点属于其他不同的簇（蓝色/绿色）。**右：**对求和 $S_2(w)$ 进行界定的示意图，具体来说是第 (8) 行右侧的部分。黄色顶点属于同一簇。顶点 $v^*(w)$ 按照正文中的描述经过了精心选择。

注意，$V(u)=\varnothing$，因为 $N_u^-\cap N_u^+=\varnothing$，所以外层求和无需包含来自 u 的自环。因此我们有：

$$\begin{aligned}S&=\sum_{w\in N_u^+\setminus u}\ \sum_{v\in V(w)}\frac{1}{n-|N_u^-\cap N_v^-|}\\&=\underbrace{\sum_{\substack{w\in N_u^+\setminus u,\\ w\in C(u)}}\ \sum_{v\in V(w)}\frac{1}{n-|N_u^-\cap N_v^-|}}_{S_1}+\underbrace{\sum_{\substack{w\in N_u^+,\\ w\in\overline{C(u)}}}\ \sum_{v\in V(w)}\frac{1}{n-|N_u^-\cap N_v^-|}}_{S_2}\end{aligned}$$

令 S_1 表示左侧的和，令 S_2 表示右侧的和。首先，我们将界定 S_1（见图 2）：

$$S_1=\sum_{\substack{w\in N_u^+\setminus u,\\ w\in C(u)}}\ \sum_{v\in V(w)}\frac{1}{n-|N_u^-\cap N_v^-|}\leqslant\sum_{\substack{w\in N_u^+\setminus u,\\ w\in C(u)}}\ \sum_{v\in V(w)}\frac{1}{|N_u^+|}\tag{5}$$

$$=\sum_{\substack{w\in N_u^+\setminus u,\\ w\in C(u)}}\frac{|V(w)|}{|N_u^+|}\leqslant\sum_{\substack{w\in N_u^+\setminus u,\\ w\in C(u)}}\frac{y(w)}{|N_u^+|}\tag{6}$$

$$\leqslant \frac{\mathsf{OPT}}{|N_u^+|} \cdot |(N_u^+ \backslash u) \cap C(u)| \leqslant \mathsf{OPT}.$$

第 (5) 行中的不等式由事实 1 以及 $|N_u^- \cap N_v^-| \leqslant |N_u^-|$ 这一事实推出。第 (6) 行中的不等式由以下事实推出：如果 $w \in C(u) \backslash u$，那么 $V(w)$1) 中的每个成员都与 w 属于不同的簇，并且 2) 都与 w 之间有一条正边；因此，$|V(w)|$ 至多等于与 w 相连的不一致数量。

界定 S_2 更为复杂。对于 $w \in N_u^+ \cap C(u)$，定义

$$S_2(w) := \sum_{v \in V(w)} \frac{1}{n - |N_u^- \cap N_v^-|},$$

从而我们可以写成

$$S_2 = \sum_{\substack{w \in N_u^+, \\ w \in \overline{C(u)}}} S_2(w).$$

固定 $w \in N_u^+ \cap \overline{C(u)}$。我们将界定 $S_2(w)$（见图 2）。注意，u 和 w 都不在 $V(w)$中。

$$\begin{aligned} S_2(w) &= \sum_{\substack{v \in V(w), \\ v \in \overline{C(w)}}} \frac{1}{n - |N_u^- \cap N_v^-|} + \sum_{\substack{v \in V(w), \\ v \in C(w)}} \frac{1}{n - |N_u^- \cap N_v^-|} \\ &\leqslant \frac{|V(w) \cap \overline{C(w)}|}{|N_u^+|} + \sum_{\substack{v \in V(w), \\ v \in C(w)}} \frac{1}{n - |N_u^- \cap N_v^-|} \qquad (7) \\ &\leqslant \frac{y(w)}{|N_u^+|} + \sum_{\substack{v \in V(w), \\ v \in C(w)}} \frac{1}{n - |N_u^- \cap N_v^-|}. \qquad (8) \end{aligned}$$

在第 (7) 行中，我们再次使用了 $n - |N_u^- \cap N_v^-| \geqslant |N_u^+|$这一界；在第 (8) 行中，我们使用了这样的事实：从 w 到 $V(w) \cap C(w)$ 的边为正，因此存在不一致。

定义

$$v^*(w) = \operatorname*{arg\,max}_{v \in V(w) \cap C(w)} |N_u^- \cap N_v^-|$$

因此，对于所有 $v \in V(w) \cap C(w)$，$|N_u^- \cap N_v^-| \leqslant |N_u^- \cap N_{v^*(w)}^-|$。接着第（8）行，我们有：

$$\begin{aligned} S_2(w) &\leqslant \frac{y(w)}{|N_u^+|} + \sum_{\substack{v \in V(w), \\ v \in C(w), \\ v \in N_{v^*(w)}^+}} \frac{1}{n - |N_u^- \cap N_v^-|} + \sum_{\substack{v \in V(w), \\ v \in C(w), \\ v \in N_{v^*(w)}^-}} \frac{1}{n - |N_u^- \cap N_v^-|} \qquad (9) \\ &\leqslant \frac{y(w)}{|N_u^+|} + \sum_{\substack{v \in V(w), \\ v \in C(w), \\ v \in N_{v^*(w)}^+}} \frac{1}{n - |N_u^- \cap N_{v^*(w)}^-|} + \sum_{\substack{v \in V(w), \\ v \in C(w), \\ v \in N_{v^*(w)}^-}} \frac{1}{|N_u^+|} \qquad (10) \\ &\leqslant \frac{y(w)}{|N_u^+|} + \sum_{\substack{v \in V(w), \\ v \in C(w), \\ v \in N_{v^*(w)}^+}} \frac{1}{n - |N_u^- \cap N_{v^*(w)}^-|} + \frac{y(v^*(w))}{|N_u^+|} \qquad (11) \\ &= \frac{y(w)}{|N_u^+|} + \frac{|V(w) \cap C(w) \cap N_{v^*(w)}^+|}{n - |N_u^- \cap N_{v^*(w)}^-|} + \frac{y(v^*(w))}{|N_u^+|} \end{aligned}$$

$$
\begin{aligned}
&\leqslant \frac{y(w)}{|N_u^+|} + \frac{|N_u^- \cap N_{v^*(w)}^+|}{n - |N_u^- \cap N_{v^*(w)}^-|} + \frac{y(v^*(w))}{|N_u^+|} && (12)\\
&\leqslant \frac{y(w)}{|N_u^+|} + 1 + \frac{y(v^*(w))}{|N_u^+|} && (13)\\
&\leqslant 1 + \frac{2 \cdot \mathsf{OPT}}{|N_u^+|}
\end{aligned}
$$

第（9）行由以下事实得出： $v^*(w) \in N_{v^*(w)}^+$。第（10）行由 $v^*(w)$ 的定义以及之前使用的 $n - |N_u^- \cap N_v^-| \geqslant |N_u^+|$ 上界得出。第（11）行由以下事实得出： $v^*(w)$ 和 $N_{v^*(w)}^- \cap C(w)$ 属于同一个簇 $C(w)$，因此 $v^*(w)$ 与 $N_{v^*(w)}^- \cap C(w)$ 之间的边都是与 $v^*(w)$ 相 incident 的不一致边。第（12）行由以下事实得出： $V(w)$ 中的每个顶点都属于 N_u^-；第（13）行则由事实2得出。

最后，在得到 $S_2(w)$ 的界之后，我们可以对 S_2 进行界定，进而对 S 和 S^- 进行界定，从而完成命题 2 的证明。

$$
\begin{aligned}
S_2 = \sum_{\substack{w \in N_u^+, \\ w \in \overline{C(u)}}} S_2(w) &\leqslant \sum_{\substack{w \in N_u^+, \\ w \in \overline{C(u)}}} \left(\frac{2 \cdot \mathsf{OPT}}{|N_u^+|} + 1 \right) \leqslant 2 \cdot \mathsf{OPT} + \sum_{\substack{w \in N_u^+, \\ w \in \overline{C(u)}}} 1 \\
&\leqslant 2 \cdot \mathsf{OPT} + y(u) \leqslant 3 \cdot \mathsf{OPT}.
\end{aligned}
$$

最后，我们完成该断言的证明，因为

$$
S^- \leqslant \mathsf{OPT} + S = \mathsf{OPT} + S_1 + S_2 \leqslant \mathsf{OPT} + \mathsf{OPT} + 3 \cdot \mathsf{OPT} = 5 \cdot \mathsf{OPT}.
$$

□

我们指出，在界定 S 时，所采用的论证并非局部的。相反，我们利用了图的全局性质。我们认为，对最优值进行界定需要图的全局结构性质。

4.2 三角不等式

我们将证明，相关性度量满足三角不等式，即对于互不相同的 u、 v、 $w \in V$、 $d_{uv} + d_{vw} \geqslant d_{uw}$。

引理 2. 相关性度量满足三角不等式。

证明。 固定三个互不相同的顶点 u， v， $w \in V$。根据 d 的定义，我们看到 $d_{uv} + d_{vw} \geqslant d_{uw}$ 当且仅当

$$
1 - \frac{|N_u^+ \cap N_v^+|}{n - |N_u^- \cap N_v^-|} + 1 - \frac{|N_w^+ \cap N_v^+|}{n - |N_w^- \cap N_v^-|} \geqslant 1 - \frac{|N_u^+ \cap N_w^+|}{n - |N_u^- \cap N_w^-|},
$$

整理后等价于

$$
\frac{|N_u^+ \cap N_w^+|}{n - |N_u^- \cap N_w^-|} + 1 \geqslant \frac{|N_u^+ \cap N_v^+|}{n - |N_u^- \cap N_v^-|} + \frac{|N_w^+ \cap N_v^+|}{n - |N_w^- \cap N_v^-|}.
$$

为简便起见，令 $a_{uv} = n - |N_u^- \cap N_v^-|$，类似地定义 a_{uw} 和 a_{vw}。然后，我们可以将上述不等式两边同乘 $a_{uw} \cdot a_{uv} \cdot a_{vw}$，将其改写为

$$
a_{uv} a_{wv} \left(|N_u^+ \cap N_w^+| + n - |N_u^- \cap N_w^-| \right) \geqslant a_{uw} a_{vw} |N_u^+ \cap N_v^+| + a_{uw} a_{uv} |N_w^+ \cap N_v^+|. \qquad (14)
$$

利用事实2，我们可以改写公式14的左侧，并将其展开为全部三个顶点的交集：

$$\begin{aligned} a_{uv}a_{vw}\left(|N_u^+ \cap N_w^+| + a_{uw}\right) &= a_{uv}a_{vw}(|N_u^+ \cap N_w^+| + |N_u^+ \cap N_w^+| + |N_u^- \cap N_w^+| + |N_u^+ \cap N_w^-|) \\ &= a_{uv}a_{vw}(2|N_u^+ \cap N_v^+ \cap N_w^+| + 2|N_u^+ \cap N_v^- \cap N_w^+| \\ &\quad + |N_u^- \cap N_v^+ \cap N_w^+| + |N_u^- \cap N_v^- \cap N_w^+| \\ &\quad + |N_u^+ \cap N_v^+ \cap N_w^-| + |N_u^+ \cap N_v^- \cap N_w^-|). \end{aligned}$$

我们引入更多简写，以紧凑地表示 3 个邻域的交集。令变量 b 的下标表示我们要考察其正交集的顶点，因此可以将 $b_{uvw} = |N_u^+ \cap N_v^+ \cap N_w^+|$、$b_{uv} = |N_u^+ \cap N_v^+ \cap N_w^-|$、$b_u = |N_u^+ \cap N_v^- \cap N_w^-|$ 等记号更紧凑地写出。因此，公式 14 的左侧可以更简洁地表示为

$$a_{uv}a_{vw}\left(|N_u^+ \cap N_w^+| + a_{uw}\right) = a_{uv}a_{vw}\left(2b_{uvw} + 2b_{uw} + b_{uv} + b_{vw} + b_u + b_w\right). \tag{15}$$

接下来，我们用 b 变量表示因子 a_{uv}, a_{uw} 和 a_{vw}，其中令 $B = b_{uvw} + b_{uv} + b_{uw} + b_{vw}$：

$$\begin{aligned} a_{uv} &= |N_u^+ \cap N_v^+| + |N_u^+ \cap N_v^-| + |N_u^- \cap N_v^+| = b_{uv} + b_{uvw} + b_u + b_{uw} + b_v + b_{vw} \\ &= B + b_u + b_v \end{aligned}$$

类似地，$a_{uw} = B + b_u + b_w$ 和 $a_{vw} = B + b_v + b_w$. 将这些结果代入公式 15 后，公式 14 的完整左侧为

$$a_{uv}a_{vw}\left(|N_u^+ \cap N_w^+| + a_{uw}\right) = (B + b_u + b_v)(B + b_v + b_w)\left(B + b_{uvw} + b_{uw} + b_u + b_w\right). \tag{16}$$

现在我们继续改写公式 14 的右侧：

$$\begin{aligned} &a_{uw}a_{vw}|N_u^+ \cap N_v^+| + a_{uw}a_{uv}|N_w^+ \cap N_v^+| \\ &\quad = (B + b_u + b_w)(B + b_v + b_w)(b_{uvw} + b_{uv}) + (B + b_u + b_w)(B + b_u + b_v)(b_{uvw} + b_{vw}). \end{aligned} \tag{17}$$

利用公式 16 和 17，我们将公式 14 中的条件改写为

$$\begin{aligned} &(B + b_u + b_v)(B + b_v + b_w)\left(b_{uvw} + b_{uw} + B + b_u + b_w\right) \\ &\qquad \geqslant (B + b_u + b_w)(B + b_v + b_w)(b_{uvw} + b_{uv}) \\ &\qquad\quad + (B + b_u + b_w)(B + b_u + b_v)(b_{uvw} + b_{vw}), \end{aligned}$$

可以验证这一点为真，因为右侧的每一项也都出现在左侧。 □

4.3 完成定理 1 和推论 1 的证明

将所有内容结合起来，我们证明了第一个用于最小最大相关聚类的组合近似算法。

定理 *1* 的证明。 对于 Kalhan、Makarychev 和 Zhou 的舍入算法（算法 2）以相关性度量作为输入所输出的聚类 $\mathcal{C}$，令 $\mathsf{ALG}(u,v) = \mathbb{1}((u,v)$ 在 $\mathcal{C})$ 中不一致，且 $\mathsf{ALG}(u) = \sum_{v \in V} \mathsf{ALG}(u,v)$。我们看到

$$\mathsf{ALG}(u) = \sum_{v \in V} \mathsf{ALG}(u,v) \underset{*}{\leqslant} 5 \cdot \sum_{v \in V} \widehat{d}_{uv} \leqslant 40 \cdot \mathsf{OPT}. \tag{18}$$

第一个不等式成立，是因为相关性度量是线性规划 3.1 的可行解，因为所有 $0 \leqslant d_{uv} \leqslant 1$ 并且根据引理 2 d 满足三角不等式。第二个不等式根据引理 1 成立。

接下来，我们分析运行时间。我们的算法分为两个阶段：在阶段 (1) 中，它计算所有 $u, v \in V$ 的相关性度量，即 d_{uv}；在阶段 (2) 中，它使用以 d 为输入的舍入算法（算法 2）。我们的算法的阶段 (1) 需要 $O(n^\omega)$ 时间，其中 ω 是矩阵乘法常数。为说明这一点，注意如果 P 是邻接矩阵，则 P^2 统计每对顶点之间长度为 2 的路径数量，因此我们可以通过令 P 为正边的邻接矩阵，计算所有 u, v 的 $|N_u^+ \cap N_v^+|$。阶段 (2) 需要 $O(n^2)$ 时间（见附录 A）。 □

最后，我们将推导推论 1。对于稀疏图，无需为所有点对计算 d_{uv} $u, v \in V$；只需在 $d_{uv} < 1$ 时计算 d_{uv}（此类点对最多有 $n\Delta^2$ 个），并隐式处理其他点对。

推论 *1* 的证明。对于每个 $u \in V$，最多有 Δ^2 个顶点 v 满足 $|N_u^+ \cap N_v^+| > 0$，因此最多有 Δ^2 个顶点 v 满足 $d_{uv} < 1$。我们只需计算 d_{uv}，对于点对 $\{u, v\}$，当 v 位于 $N^2(u)$（相对于 u 的二跳邻域）时。对于固定的 u，计算 $N^2(u)$ 以及对每个 $v \in N^2(u)$ 计算 $|N_u^+ \cap N_v^+|$，可在 $O(\Delta^2)$ 时间内完成，因此对所有顶点 u 而言，总共需要 $O(n\Delta^2)$ 时间。

现在我们需要为每个 $v \in N^2(u)$（对于 $v \notin N^2(u)$、$d_{uv} = 1$）计算 d_{uv}。对于每个顶点对 $\{u, v\}$，这只需常数时间，因为

$$d_{uv} = 1 - |N_u^+ \cap N_v^+|/(n - |N_u^- \cap N_v^-|) = 1 - |N_u^+ \cap N_v^+|/\left(|N_u^+| + |N_v^+| - |N_u^+ \cap N_v^+|\right).$$

既然我们已经为所有相关的顶点对计算出了 d，接下来需要说明舍入算法（算法 2）也可以在稀疏图上更快地运行。首先，为每个 $u \in V$ 初始化 $L_0(u)$。由于 $r = 1/5,$，我们只需检查对于 $v \in N^2(u)$ 是否有 $d_{uv} \leqslant 1/5$（否则为 $d_{uv} = 1$）。由于 $|N^2(u)| \leqslant \Delta^2$，初始化 $L_0(\cdot)$ 总共需要 $O(n\Delta^2)$ 时间。我们将在二叉堆中为每个 $u \in V_t$ 存储 $L_t(u)$；插入这些元素总共需要 $O(n\log\, n)$ 时间。在每个阶段 t，寻找使 $L_t(u)$ 最大的 u_t^* 需要 $O(1)$ 时间。然后，我们必须移除所有满足 $d_{u_t^* v} \leqslant 2/5$ 的 v。由于这些 v 必须属于 $N^2(u_t^*)$，且每次删除需要 $O(\log\, n)$ 时间，因此总共需要 $O(\Delta^2 \log n)$ 时间。对于 $u \in V_{t+1}$，我们还必须将 $L_t(u)$ 减小到 $L_{t+1}(u)$：对于阶段 t 期间移除的每个顶点 v，$N^2(v)$ 中至多有 Δ^2 个元素，因此 v 至多会导致 Δ^2 次对 $L_t(\cdot)$ 的更新。由于每个顶点只会被移除一次，关键更新操作总共贡献 $O(n\Delta^2 \log n)$ 时间。□

5 通过抽样实现更快的算法

本节将展示：无需为每一对 u，v 精确计算相关性度量，只需通过图的样本对其进行估计即可。这样，我们可以将我们的算法的运行时间从 $O(n^\omega)$ 提升至 $O(n^2 \log n)$。为此，我们将证明：(1) 这些估计值的分数成本以高概率与最优值相差一个常数倍（命题 9）；(2) 这些估计值以高概率满足近似三角不等式（命题 8）；(3) 将一个近似满足三角不等式的函数输入舍入算法（算法 2），足以使（$*$）中的值达到一个常数因子（引理 3）。这三个步骤共同推出定理 2。

我们将用 $\bar{d}_{uv}$ 表示对 d_{uv} 的初始估计值，并在第 5.1 节证明 $\bar{d}_{uv}$ 的性质。随后，我们将进行后处理阶段，在该阶段中，将一些 $\bar{d}_{uv}$ 向下舍入为 0，将另一些向上舍入为 1；虽然这一步并不显然，但它对于控制分数成本是必要的。我们将在第 5.2 节考察后处理阶段。最后，我们将用于舍入算法的输入记为 $\tilde{d}_{uv}$。

我们将反复使用以下关于随机变量之和的尾界。

定理 3（Chernoff-Hoeffding）。设 $X = X_1 + \cdots + X_m$，其中 $\{X_1, \ldots\ldots, X_m\}$ 是一组负相关的随机变量。定义 $\mu = \mathbb{E}[X]$。对于 $0 < \varepsilon < 1$，以下尾界成立：

$$\mathbb{P}(X \geqslant (1+\varepsilon)\mu) \leqslant e^{-\varepsilon^2 \mu/4}$$
$$\mathbb{P}(X \leqslant (1-\varepsilon)\mu) \leqslant e^{-\varepsilon^2 \mu/4}.$$

5.1 相关性度量的初始估计值

固定 $0 < \varepsilon < 1$。对于每个顶点 u，从 $|N_u^+|$ 中随机抽取 $\lceil C(\varepsilon) \cdot \log\ n \rceil$ 个顶点，其中 $C(\varepsilon) = 32/\varepsilon^2$。首先假设 $|N_u^+| \geqslant \lceil C(\varepsilon) \cdot \log\ n \rceil$，否则估计值是精确的。将该样本记为 S_u。

对于两个顶点 u，v，定义随机变量：

$$
\begin{aligned}
X_+^{(u,v)} &= \sum_{w \in N_u^+ \cap N_v^+} \mathbb{1}_{\{w \in S_u\}} \\
X_-^{(u,v)} &= \sum_{w \in N_u^+ \cap N_v^-} \mathbb{1}_{\{w \in S_u\}} \\
W^{(u,v)} &= \frac{|N_u^+|}{\lceil C(\varepsilon) \log n \rceil} \cdot X_+^{(u,v)} \\
Y^{(u,v)} &= \frac{|N_u^+|}{\lceil C(\varepsilon) \log n \rceil} \cdot X_-^{(u,v)}
\end{aligned}
$$

注意，上标是有序对。如果另一方面 $|N_u^+| < \lceil C(\varepsilon) \cdot \log\ n \rceil$，则设 $W^{(u,\ v)} = |N_u^+ \cap N_v^+|$ 和 $Y^{(u,\ v)} = |N_u^+ \cap N_v^-|$。

注意，$X^{(u,\ v)}_{+} + X^{(u,\ v)}_{-} = |S_u| = \lceil C(\varepsilon) \cdot \log n \rceil$。因此，

$$
Y^{(u,v)} = |N_u^+| - W^{(u,v)}. \tag{19}
$$

$Y^{(u,\ v)}$ 将作为 $|N_u^+ \cap N_v^-|$ 的估计值，而 $W^{(u,\ v)}$ 将作为 $|N_u^+ \cap N_v^+|$ 的估计值。将上标中 u 和 v 的顺序翻转，可得 $Y^{(v,\ u)}$ 是 $|N_u^- \cap N_v^+|$ 的一个估计值，而 $W^{(v,\ u)}$ 是 $|N_u^+ \cap N_v^+|$ 的第二个估计值。此外，注意到

$$
\mathbf{E}[X_+^{(u,v)}] = |N_u^+ \cap N_v^+| \cdot \frac{\lceil C(\varepsilon) \log n \rceil}{|N_u^+|} \tag{20}
$$

$$
\mathbf{E}[X_-^{(u,v)}] = |N_u^+ \cap N_v^-| \cdot \frac{\lceil C(\varepsilon) \log n \rceil}{|N_u^+|} \tag{21}
$$

以及当上标中 u、v 的顺序颠倒时的类似陈述。

令 u、v 标记为满足 $|N_v^+| \geqslant |N_u^+|$。 我们将 d_{uv} 的初始估计值（后处理之前）定义为

$$
\bar{d}_{uv} = \frac{Y^{(u,v)} + Y^{(v,u)}}{|N_u^+| + Y^{(v,u)}} = \frac{|N_u^+| - W^{(u,v)} + |N_v^+| - W^{(v,u)}}{|N_u^+| + |N_v^+| - W^{(v,u)}} \tag{22}
$$

其中，第二个等式由方程（19）成立。注意，由于 u 的正自环，$|N_u^+| \geqslant 1$，因此这是定义良好的。另外请注意，在分母中，我们使用 v 的样本对 $|N_u^+ \cap N_v^+|$ 的估计值 $W^{(v,\ u)}$，而不是 u 的。这在直觉上是合理的，因为 $|N_v^+| \geqslant |N_u^+|$，但这样做在技术上的必要性将在后文变得清晰。

事实3。 对于每个 u、$v \in V$、$W^{(u,\ v)}$、$Y^{(u,\ v)} \geqslant 0$。此外，$0 \leqslant \bar{d}_{uv} \leqslant 1$。

由于 $|N_u^+| = |N_u^+ \cap N_v^+| + |N_u^+ \cap N_v^-|$，两个加数中至少有一个始终不小于 $\frac{1}{2}|N_u^+|$。因此，我们可以在接下来的两个命题中证明，$|N_u^+ \cap N_v^+|$ 和 $|N_u^+ \cap N_v^-|$ 的估计值中至少有一个具有良好的集中性（命题2和命题3）。这种集中性确保我们能够分别从上界和下界用来自两侧的 d_{uv} 近似 d_{uv}（命题 4 和命题 5）。

命题 2。 设 $0 < \varepsilon < 1$。如果 $|N_u^+ \cap N_v^+| \geqslant \frac{1}{2}|N_u^+|$，那么以高概率 $W^{(u,\ v)} \geqslant (1-\varepsilon)|N_u^+ \cap N_v^+|$ 且 $W^{(u,\ v)} \leqslant (1+\varepsilon)|N_u^+ \cap N_v^+|$。同样地，如果 $|N_u^+ \cap N_v^+| \geqslant \frac{1}{2}|N_v^+|$，那么以高概率 $W^{(v,\ u)} \geqslant (1-\varepsilon)|N_u^+ \cap N_v^+|$ 且 $W^{(v,\ u)} \leqslant (1+\varepsilon)|N_u^+ \cap N_v^+|$。特别地，四个事件中的每一个以至多 $\frac{1}{n^{\varepsilon^2 C(\varepsilon)/8}}$ 的概率失败。

证明。 如果 $|N_u^+| \leqslant \lceil C(\varepsilon)\log\ n\rceil$，那么前两个界自动成立。因此，不妨设 $|N_u^+| \geqslant \lceil C(\varepsilon)\log\ n\rceil$。根据定理 3,

$$\begin{aligned}
\mathbf{P}\left(W^{(u,v)} \leqslant (1-\varepsilon)|N_u^+ \cap N_v^+|\right) &= \mathbf{P}\left(\frac{|N_u^+|}{\lceil C(\varepsilon)\log n\rceil}\cdot X_+^{(u,v)} \leqslant (1-\varepsilon)|N_u^+ \cap N_v^+|\right)\\
&= \mathbf{P}\left(X_+^{(u,v)} \leqslant (1-\varepsilon)|N_u^+ \cap N_v^+|\cdot\frac{\lceil C(\varepsilon)\log n\rceil}{|N_u^+|}\right)\\
&= \mathbf{P}\left(X_+^{(u,v)} \leqslant (1-\varepsilon)\mathbf{E}[X_u^+]\right)\\
&\leqslant e^{-\frac{\varepsilon^2}{4}\mathbf{E}[X_+^{(u,v)}]} \leqslant e^{-\frac{\varepsilon^2}{4}\cdot\lceil C(\varepsilon)\log n\rceil\cdot\frac{|N_u^+\cap N_v^+|}{|N_u^+|}} \leqslant e^{-\frac{\varepsilon^2}{8}\cdot C(\varepsilon)\log n},
\end{aligned}$$

其中，在最后两个不等式中我们使用了等式（20）和假设 $|N_u^+ \cap N_v^+| \geqslant \frac{1}{2}|N_u^+|$。类似地,

$$\mathbf{P}\left(W^{(u,v)} \geqslant (1+\varepsilon)|N_u^+ \cap N_v^+|\right) \leqslant e^{-\frac{\varepsilon^2}{8}\cdot C(\varepsilon)\log n}.$$

□

命题 3. 设 $0<\varepsilon<1$。如果 $|N_u^+ \cap N_v^-| \geqslant \frac{1}{2}|N_u^+|$，则以高概率 $Y^{(u,v)} \geqslant (1-\varepsilon)|N_u^+ \cap N_v^+|$且 $Y^{(u,v)} \leqslant (1+\varepsilon)|N_u^+ \cap N_v^+|$。同样，如果$|N_u^- \cap N_v^+| \geqslant \frac{1}{2}|N_v^+|$，则以高概率 $Y^{(v,u)} \geqslant (1-\varepsilon)|N_u^+ \cap N_v^+|$且 $Y^{(v,u)} \leqslant (1+\varepsilon)|N_u^+ \cap N_v^+|$。特别地，这四个事件中每个事件发生失败的概率至多为 $\frac{1}{n^{\varepsilon^2 C(\varepsilon)/8}}$。

证明。 证明与上文类似，但这里改为使用假设 $|N_u^+ \cap N_v^-| \geqslant (1/2)|N_u^+|$ 以及由式 (21) 得到的事实 $\mathbf{E}[X^{(u,v)}] = \frac{|N_u^+\cap N_v^-|}{|N_u^+|}\cdot\lceil C(\varepsilon)\log n\rceil$。 □

Prop命题 4。 对于任意 u、$v \in V$，以下结论以至少 $1-\frac{4}{n^{\varepsilon^2 C(\varepsilon)/8}}$ 的概率成立：

$$\bar{d}_{uv} \leqslant c_1(\varepsilon)\cdot d_{uv} + h_1(\varepsilon)$$

此外，由于 d_{uv}满足三角不等式，对于任意 u、v、w,

$$\bar{d}_{uv} \leqslant c_1(\varepsilon)\cdot(d_{uw}+d_{vw}) + h_1(\varepsilon)$$

其中 $c_1(\varepsilon)\to 1$ 和 $h_1(\varepsilon)\to 0$ 分别为 $\varepsilon\to 0$。特别地，取 $c_1(\varepsilon) = (1+\varepsilon)/(1-\varepsilon)$ 和 $h_1(\varepsilon) = 2\varepsilon/(1-\varepsilon)$。

证明。 不妨设 $|N_v^+| \geqslant |N_u^+|$。令 A_{uv}表示 d_{uv}的分子，令 B_{uv}表示分母，即:

$$\begin{aligned}
A_{uv} &= |N_u^+ \cap N_v^-| + |N_u^- \cap N_v^+|\\
B_{uv} &= |N_u^+ \cap N_v^-| + |N_u^- \cap N_v^+| + |N_u^+ \cap N_v^+|
\end{aligned}$$

我们根据 $|N_u^+ \cap N_v^+| \geqslant (1/2)|N_u^+|$和 $|N_u^+ \cap N_v^+| \geqslant (1/2)|N_v^+|$是否成立来分类。注意，可能的情况有三种，而不是四种，因为 $|N_u^+| \leqslant |N_v^+|$ 使得$|N_u^+ \cap N_v^+| < (1/2)|N_u^+|$和 $|N_u^+ \cap N_v^+| \geqslant (1/2)|N_v^+|$不可能同时成立。

情况 1. $|N_u^+ \cap N_v^+| \geqslant (1/2)|N_u^+|$和 $|N_u^+ \cap N_v^+| \geqslant (1/2)|N_v^+|$。

那么，以高概率

$$\bar{d}_{uv} = \frac{|N_u^+| - W^{(u,v)} + |N_v^+| - W^{(v,u)}}{|N_u^+| + |N_v^+| - W^{(v,u)}}$$

$$
\begin{aligned}
&\leqslant \frac{|N_u^+| - (1-\varepsilon)|N_u^+ \cap N_v^+| + |N_v^+| - (1-\varepsilon)|N_u^+ \cap N_v^+|}{|N_u^+| + |N_v^+| - (1+\varepsilon)|N_u^+ \cap N_v^+|} \\
&= \frac{|N_u^+ \cap N_v^-| + |N_u^- \cap N_v^+| + 2\varepsilon|N_u^+ \cap N_v^+|}{|N_u^+ \cap N_v^+| + |N_u^+ \cap N_v^-| + |N_u^- \cap N_v^+| - \varepsilon|N_u^+ \cap N_v^+|} \\
&= \frac{A_{uv} + 2\varepsilon(B_{uv} - A_{uv})}{B_{uv} - \varepsilon(B_{uv} - A_{uv})} \\
&\leqslant \frac{(1-2\varepsilon)A_{uv} + 2\varepsilon B_{uv}}{(1-\varepsilon)B_{uv}} = \frac{1-2\varepsilon}{1-\varepsilon} \cdot d_{uv} + \frac{2\varepsilon}{1-\varepsilon},
\end{aligned}
$$

其中，第一行用到了式（22），第二行用到了命题 2。

情形 2。 $|N_u^+ \cap N_v^-| \geqslant \frac{1}{2}|N_u^+|$ 且 $|N_u^- \cap N_v^+| \geqslant \frac{1}{2}|N_v^+|$。

然后以高概率成立。

$$
\begin{aligned}
\bar{d}_{uv} &= \frac{Y^{(u,v)} + Y^{(v,u)}}{|N_u^+| + Y^{(v,u)}} \\
&\leqslant \frac{(1+\varepsilon)|N_u^+ \cap N_v^-| + (1+\varepsilon)|N_u^- \cap N_v^+|}{|N_u^+| + (1-\varepsilon)|N_u^- \cap N_v^+|} \\
&\leqslant \frac{1+\varepsilon}{1-\varepsilon} \cdot d_{uv}
\end{aligned}
$$

其中，第一行使用了（22），第二行使用了命题 3。

情形 3。 $|N_u^+ \cap N_v^+| \geqslant (1/2)|N_u^+|$ 但 $|N_u^+ \cap N_v^+| < (1/2)|N_v^+|$（即 $|N_u^- \cap N_v^+| > (1/2)|N_v^+|$）。然后以高概率成立。

$$
\begin{aligned}
\bar{d}_{uv} &= \frac{|N_u^+| - W^{(u,v)} + Y^{(v,u)}}{|N_u^+| + Y^{(v,u)}} \\
&\leqslant \frac{|N_u^+| - (1-\varepsilon)|N_u^+ \cap N_v^+| + (1+\varepsilon)|N_u^- \cap N_v^+|}{|N_u^+| + (1-\varepsilon)|N_u^- \cap N_v^+|} \\
&\leqslant \frac{|N_u^+ \cap N_v^-| + \varepsilon|N_u^+ \cap N_v^+| + (1+\varepsilon)|N_u^- \cap N_v^+|}{(1-\varepsilon)B_{uv}} \\
&\leqslant \frac{(1+\varepsilon)A_{uv}}{(1-\varepsilon)B_{uv}} + \frac{\varepsilon(B_{uv} - A_{uv})}{(1-\varepsilon)B_{uv}} \\
&\leqslant \frac{1+\varepsilon}{1-\varepsilon} \cdot d_{uv} + \frac{\varepsilon}{1-\varepsilon}.
\end{aligned}
$$

其中，第一行使用了（22），第二行使用了命题2和命题3。 □

命题 5。 对于任意 u、$v \in V$，以下结论以至少 $1 - \frac{4}{n^{\varepsilon^2 C(\varepsilon)/8}}$ 的概率成立：

$$
d_{uv} \leqslant c_2(\varepsilon) \cdot \bar{d}_{uv} + h_2(\varepsilon)
$$

其中 $c_2(\varepsilon) \to 1$ 和 $h_2(\varepsilon) \to 0$ 分别为 $\varepsilon \to 0$。特别地，取 $c_2(\varepsilon) = (1+\varepsilon)/(1-\varepsilon)$ 和 $h_2(\varepsilon) = 2\varepsilon/(1-\varepsilon)$。

证明。 证明过程与命题 4 相同，但使用 Chernoff 界的另一侧。

□

我们结合前两个命题，证明我们的抽样导出了近似三角不等式。

命题 6。 对于任意 u、 v、 $w \in V$，以下结论以至少 $1 - \frac{12}{n^{\varepsilon^2 C(\varepsilon)/8}}$ 的概率成立：

$$\bar{d}_{uv} \leqslant c_3(\varepsilon) \cdot (\bar{d}_{uw} + \bar{d}_{vw}) + h_3(\varepsilon)$$

其中 $c_3(\varepsilon) \to 1$ 和 $h_3(\varepsilon) \to 0$ 为 $\varepsilon \to 0$。特别地，取

$$c_3(\varepsilon) = \left(\frac{1+\varepsilon}{1-\varepsilon}\right)^2 \quad and \quad h_3(\varepsilon) = \left(\frac{2\varepsilon}{1-\varepsilon}\right)\left(1 + \frac{2(1+\varepsilon)}{1-\varepsilon}\right).$$

证明。 这直接由命题 4 和命题 5 得出。 □

然后，我们使用并合界证明，近似三角不等式以高概率对所有对全局成立。

命题 7。 固定 $\varepsilon > 0$。近似三角不等式 $\bar{d}_{uv} \leqslant c_3(\varepsilon) \cdot (\bar{d}_{uw} + \bar{d}_{vw}) + h_3(\varepsilon)$对所有三元组 u、 v、 w 同时 成立的概率至少为 $1 - O\left(\frac{1}{n}\right)$。

证明。 根据命题 6，任意三元组 u、 v、 w 不满足三角不等式的概率至多为 $\frac{12}{n^{\varepsilon^2 C(\varepsilon)/8}}$。 由于至多有 n^3 个三元组，因此至少有一个三元组不满足三角不等式的概率至多为 $\frac{12}{n^{\varepsilon^2 \frac{C(\varepsilon)}{8} - 3}} = \frac{12}{n}$，因为 $C(\varepsilon) = 32/\varepsilon^2$。 □

5.2 相关性度量的后处理估计值

现在我们定义最终估计值 $\tilde{d}_{uv}$:

$$\tilde{d}_{uv} = \begin{cases} 0 & \bar{d}_{uv} \leqslant 2h_3(\varepsilon) \text{ and } (u,v) \in E^+ \\ 1 & \bar{d}_{uv} \geqslant 1/(2c_3(\varepsilon)+1) \text{ and } (u,v) \in E^- \\ \bar{d}_{uv} & \text{otherwise} \end{cases}.$$

为了证明用 $\tilde{d}_{uv}$ 替代精确距离 d_{uv}时我们的算法是成功的，我们需要证明，以高概率：**(1)** $\tilde{d}_{uv}$ 满足近似三角不等式（命题 8）；**(2)** $\tilde{d}_{uv}$ 的分数成本可以与最优值进行比较（命题 9）。具体而言，将估计值舍入为 d ，可以用近似三角不等式中 d 所满足条件的轻微损失，换取对分数成本进行界定的能力。

命题 8。 固定 $\varepsilon > 0$。对于所有三元组 u、 v、 w，近似三角不等式同时成立的概率至少为 $1 - O(1/n)$:

$$\tilde{d}_{uv} \leqslant c_4(\varepsilon) \cdot \left(\tilde{d}_{uw} + \tilde{d}_{vw}\right) + h_4(\varepsilon),$$

其中 $h_4(\varepsilon) \to 0$ 和 $c_4(\varepsilon) \to 3$ 为 $\varepsilon \to 0$。特别地，取 $h_4(\varepsilon) = (4c_3(\varepsilon)+1)(2c_3(\varepsilon)+1)h_3(\varepsilon)$和 $c_4(\varepsilon) = (2c_3(\varepsilon)+1)c_3(\varepsilon)$。

命题 9. 固定 $0 < \varepsilon < 0.03$。对于 $D(\varepsilon) = 2c_3(\varepsilon)$，以下不等式以至少 $1 - O(1/n)$ 的概率成立：

$$\sum_{v \in N_u^+} \tilde{d}_{uv} + \sum_{v \in N_u^-} (1 - \tilde{d}_{uv}) \leqslant D(\varepsilon) \cdot \left(\sum_{v \in N_u^+} d_{uv} + \sum_{v \in N_u^-} (1 - d_{uv}) \right) \leqslant 8 \cdot D(\varepsilon) OPT.$$

命题 *8* 的证明。根据命题 7，

$$\bar{d}_{uv} \leqslant c_3(\varepsilon) \cdot (\bar{d}_{uw} + \bar{d}_{vw}) + h_3(\varepsilon) \tag{23}$$

以高概率对所有三元组 u、v、w 同时成立。我们需要证明，将中间估计 $d(\cdot)$ 替换为后处理估计值 $d(\cdot)$ 时，类似的不等式仍然成立。首先注意，只有当 $d_{uv} \geqslant 1/(2c_3(\varepsilon)+1)$时，我们才将 d_{uv} 向上取整为 1。因此在这种情况下

$$\widetilde{d}_{uv} = 1 = (1/(2c_3(\varepsilon)+1))\,(2c_3(\varepsilon)+1) \leqslant \bar{d}_{uv} \cdot (2c_3(\varepsilon)+1).$$

因此始终有

$$\widetilde{d}_{uv} \leqslant \bar{d}_{uv} \cdot (2c_3(\varepsilon)+1) \tag{24}$$

由于该不等式对 $d_{uv}=0$ 以及 $d_{uv}=d_{uv}$ 同样成立（利用 $d_{uv} \geqslant 0$，根据事实3），接下来注意到，只有在 $d_{uw} \leqslant 2h_3(\varepsilon)$时我们才会将 d_{uw} 向下取整为0。因此始终有

$$\bar{d}_{uw} - 2h_3(\varepsilon) \leqslant \widetilde{d}_{uw}. \tag{25}$$

这是因为：（1）如果将 d_{uw} 向上取整为 $d_{uw}=1$，则由于 $d_{uw} \leqslant 1$（事实3），该不等式成立；（2）如果 $d_{uw}=d_{uw}$，则该不等式自动成立；（3）如果将 d_{uw} 向下取整为 $d_{uw}=0$，那么 $d_{uw} \leqslant 2h_3(\varepsilon)$，因此该不等式再次成立。

将不等式（23）、（24）和（25）合并可得：

$$\begin{aligned}
\widetilde{d}_{uv} &\leqslant (2c_3(\varepsilon)+1)\bar{d}_{uv} \\
&\leqslant (2c_3(\varepsilon)+1)c_3(\varepsilon)(\bar{d}_{uw}+\bar{d}_{vw}) + (2c_3(\varepsilon)+1)h_3(\varepsilon) \\
&\leqslant (2c_3(\varepsilon)+1)c_3(\varepsilon)\left(\widetilde{d}_{uw}+\widetilde{d}_{vw}+4h_3(\varepsilon)\right) + (2c_3(\varepsilon)+1)h_3(\varepsilon) \\
&\leqslant c_4(\varepsilon)\left(\widetilde{d}_{uw}+\widetilde{d}_{vw}\right) + h_4(\varepsilon)
\end{aligned}$$

其中，$c_4(\varepsilon) = (2c_3(\varepsilon)+1)c_3(\varepsilon)$ 和 $h_4(\varepsilon) = (4c_3(\varepsilon)+1)(2c_3(\varepsilon)+1)h_3(\varepsilon)$。特别地，当 $\varepsilon \to 0$时，$c_4(\varepsilon) \to 3$和 $h_4(\varepsilon) \to 0$。

□

命题 *9* 的证明。只需证明该界逐点成立。例如，对于 $v \in N_u^+$，我们证明 $\widetilde{d}_{uv}$ 满足 $d_{uv} \leqslant D(\varepsilon)\cdot d_{uv}$；对于 $v \in N_u^-$，我们证明 $(1-d_{uv}) \leqslant D(\varepsilon)(1-d_{uv})$。

首先考虑 $v \in N_u^+$。如果 d_{uv} 向下舍入到 $d_{uv}=0$，则 $d_{uv} \leqslant D(\varepsilon)d_{uv}$ 自动成立。如果 d_{uv} 没有向下舍入，则 $d_{uv}=d_{uv} \geqslant 2h_3(\varepsilon)$。根据命题 4，以高概率，

$$2h_3(\varepsilon) \leqslant \widetilde{d}_{uv} = \bar{d}_{uv} \leqslant c_3(\varepsilon)d_{uv} + h_3(\varepsilon)$$

这里我们用到了 $c_1(\varepsilon) \leqslant c_3(\varepsilon)$ 和 $h_1(\varepsilon) \leqslant h_3(\varepsilon)$。因此 $d_{uv} \geqslant h_3(\varepsilon)/c_3(\varepsilon)$。进而，

$$\widetilde{d}_{uv} = \bar{d}_{uv} \leqslant c_3(\varepsilon)d_{uv} + h_3(\varepsilon) \leqslant c_3(\varepsilon)d_{uv} + c_3(\varepsilon)d_{uv} \leqslant D(\varepsilon)\cdot d_{uv}$$

如所需。

接下来考虑 $v \in N_u^-$的情况。这种情况会更复杂。如果 d_{uv} 向上舍入到 $d_{uv}=1$，则 $0 = 1-d_{uv} \leqslant D(\varepsilon)\cdot(1-d_{uv})$ 自动成立。否则，$d_{uv}=d_{uv} \leqslant 1/(2c_3(\varepsilon)+1)$。我们分两种情况讨论：

情况 1。 $h_3(\varepsilon) \leqslant d_{uv}$.

根据命题 5，以高概率。

$$c_3(\varepsilon)\cdot\bar{d}_{uv} + h_3(\varepsilon) \geqslant d_{uv}$$

其中我们使用了 $c_2(\varepsilon) \leqslant c_3(\varepsilon)$ 和 $h_2(\varepsilon) \leqslant h_3(\varepsilon)$。现在利用此情形的假设，

$$\bar{d}_{uv} \geqslant \frac{1}{c_3(\varepsilon)}\cdot d_{uv} - \frac{h_3(\varepsilon)}{c_3(\varepsilon)}$$

$$\bar{d}_{uv} \geqslant \frac{1}{c_3(\varepsilon)} \cdot d_{uv} - \frac{\bar{d}_{uv}}{c_3(\varepsilon)}$$
$$\left(1 + \frac{1}{c_3(\varepsilon)}\right) \cdot \bar{d}_{uv} \geqslant \frac{1}{c_3(\varepsilon)} \cdot d_{uv}$$
$$\bar{d}_{uv} \geqslant \frac{1}{c_3(\varepsilon)+1} \cdot d_{uv} \tag{26}$$

¯由于 $d_{uv} \leqslant 1/(2c_3(\varepsilon)+1)$, , 有

$$1 - \bar{d}_{uv} \leqslant 2\left(1 - (c_3(\varepsilon)+1)\bar{d}_{uv}\right).$$

现在，利用 (26),

$$1 - \widetilde{d}_{uv} = 1 - \bar{d}_{uv} \leqslant 2(1 - d_{uv}) \leqslant D(\varepsilon) \cdot (1 - d_{uv}).$$

¯**情况 2**。 $h_3(\varepsilon) \geqslant d_{uv}$。

与前一种情况一样，我们使用命题 5。结合本情况的假设可得，以高概率

$$d_{uv} \leqslant c_3(\varepsilon) \cdot \bar{d}_{uv} + h_3(\varepsilon)$$
$$d_{uv} \leqslant (c_3(\varepsilon)+1)h_3(\varepsilon)$$
$$1 - d_{uv} \geqslant 1 - (c_3(\varepsilon)+1)h_3(\varepsilon).$$

现在，由于 $c_3(\varepsilon) \to 1$ 和 $h_3(\varepsilon) \to 0$, 对于足够小的 ε （$\varepsilon < 0.03$ 即足够），有 $(c_3(\varepsilon)+1)h_3(\varepsilon) \leqslant 1/2$。进而，

$$1 - d_{uv} \geqslant 1/2$$
$$1 - d_{uv} \geqslant 1/2(1 - \bar{d}_{uv})$$

¯其中我们使用了 $d_{uv} \geqslant 0$ （事实3）。因此

$$1 - \widetilde{d}_{uv} = 1 - \bar{d}_{uv} \leqslant 2(1 - d_{uv}) \leqslant D(\varepsilon) \cdot (1 - d_{uv})$$

这就完成了该情况的证明，也完成了整个证明。 □

5.3 近似三角不等式足够

在本节中，我们将说明，与其向舍入算法（算法 2）输入一个半度量，不如输入一个近似为半度量的函数。若某个函数 d a (δ_1, δ_2)-半度量 满足除三角不等式之外的所有半度量条件，并满足 (δ_1, δ_2)-近似三角不等式，我们将其称为该函数:

定义 2。 函数 d： $V^2 \to \mathbb{R}_{\geqslant 0}$ 满足 (δ_1, δ_2)-近似三角不等式，当且仅当

$$d_{uv} \leqslant \delta_1(d_{uw} + d_{wv}) + \delta_2 \quad \forall u, v, w \in V.$$

算法 1。

输入： d 定义在 V 上的 (δ_1, δ_2)-半度量

输出： 聚类 $\mathcal{C}$。

1. 设 $V_0 = V$、 $r = r(\delta_1, \delta_2)$、 $t = 0$。
2. **当** $(V_t \neq \varnothing)$
   - Find $w_t = \arg\max_{w \in V_t} L_t(w) = \arg\max_{w \in V_t} \sum_{u \in \mathrm{Ball}(w,r) \cap V_t} r - d_{uw}$.
   - Create $C_t = \mathrm{Ball}(w_t, b \cdot r) \cap V_t$, for $b = b(\delta_1, \delta_2)$.
   - Set $V_{t+1} = V_t \backslash C_t$ and $t = t + 1$.
3. 返回 $\mathcal{C} = (C_0, \ldots, C_{t-1})$。

$\widehat{\ }$ $\widehat{\ }$回顾第3节中的内容：对于由算法 1 返回的聚类，如果 $d_{uv} = d_{uv}$ 当 $(u, v) \in E^+$， $d_{uv} = 1 - d_{uv}$ 当 $(u, v) \in E^-$，且 $\mathsf{ALG}(u, v) = \mathbb{1}((u, v)$ 在 $\mathcal{C})$ 中存在不一致，其中 $\mathcal{C}$。

令 d 满足 (δ_1, δ_2)-近似三角不等式，其中 $\delta_1 = \delta_1(\varepsilon) \widetilde{=} 3 + h_4(\varepsilon)$ 和 $\delta_2 = \delta_2(\varepsilon) = h_4(\varepsilon)$。（这是通过注意到 $c_4(\varepsilon) \leqslant 3 + h_4(\varepsilon)$得出的。因此， (δ_1, δ_2)-近似三角不等式弱于命题 8 中的不等式，但为了便于计算，我们使用前者。）

引理 3 （附录 B 中定理B.1 的类似结果 [16]）**.** 令 $r = r(\delta_1、\delta_2)$ 和 $b = b(\delta_1、\delta_2)$ 如附录 *B* 中所定义。[5] 令 $\mathcal{C}$ 为算法 *1* 返回的一个聚类。则，

$$\mathit{ALG}(u) = \sum_v \mathit{ALG}(u, v) \leqslant \frac{1}{r(\delta_1, \delta_2)} \sum_v \widehat{d}_{uv}$$

其中 $r(\delta_1、\delta_2) \to 1/121$。[6]

证明。 我们将遵循定理B.1 的证明。不过，参数 r 的变化以及 C_t 的构造方式（即，使用到 $b \cdot r$ 的半径，而非 $2r$）会导致各个情形在不同的位置划分。注意到 $b \cdot r < 1$。

$\widehat{\ }$定义收益$(u) = \sum_v d_{uv} - r \sum_v \mathsf{ALG}(u, v)$ 以及 $\Delta E_t = \{(u, v) \mid u \in C_t$ 或 $v \in C_t\}$. 该 n,

$$\mathrm{profit}_t(u, v) = \begin{cases} \widehat{d}_{uv} - r\mathsf{ALG}(u, v) & (u, v) \in \Delta E_t \\ 0 & o.w. \end{cases}$$

and

$$\mathrm{profit}_t(u) = \sum_{v \in V_t} \mathrm{profit}_t(u, v) = \sum_{(u,v) \in \Delta E_t, v \in V_t} \widehat{d}_{uv} - r \sum_{(u,v) \in \Delta E_t, v \in V_t} \mathsf{ALG}(u, v).$$

注意 由于 ΔE_t 彼此不相交，收益$(u) = \sum_t$收益$_t(u)$. 注意，如果 $u \notin V_t$，则收益$_t(u) = 0$.

引理 4 （ [16]中的引理 B.2 的类似结论）。 *</style}*对于每个 $u \in V_t$，收益$_t(u) \geqslant 0$。因此，收益$(u) \geqslant 0$，从而得到引理 *3*。

取 $c_1 = c_1(\delta_1, \delta_2)$ 以及 $c_2 = c_2(\delta_1, \delta_2)$，其定义见附录 B。注意 $c_2 \cdot r < 1$ 和 $c_1 < c_2$，因此引理 4 的证明中的各个情形是合理的。

证明。 令 $w = w_t$ 为 L_t的最大化者。

情形 3。*($d_{uw} \in [0, c_1 \cdot r] \cup [c_2 \cdot r, 1]$)*

在这种情况下，只需证明收益$_t(u, v) \geqslant 0$ 对于 $v \in V_t$ 和 $(u, v) \in \Delta E_t$成立即可。由断言 3 以及引理 5 和引理 6 可得。

情形 4。*($d_{uw} \in (c_1 \cdot r, c_2 \cdot r)$)*

根据引理 7 和引理 8，收益$_t(u) \geqslant L_t(w) - L_t(u) \geqslant 0$，因为 w 是 L_t的最大化者。

□

断言 3 （ [16] 中断言 B.3 的类似表述）。 令 $u, v \in V_t$ 且 $(u, v) \in E^-$。则收益$_t(u, v) \geqslant 0$。

[5]对我们而言， $\delta_1(\varepsilon) \to 3$ 和 $\delta_2(\varepsilon) \to 0$ 是 $\varepsilon \to 0$。当 δ_1 趋近于任意常数时，只要 $\delta_2(\varepsilon) \to 0$ 为 $\varepsilon \to 0$，也可以得到类似的结果。[6]如果 $\delta_1(\varepsilon) \to 1$，那么 $r(\delta_1, \delta_2) \to 1/5$ 当 $\varepsilon \to 0$时成立，从而恢复满足精确三角不等式时成立的舍入结果。

证明。注意，若 $\mathsf{ALG}(u,\ v)=0$ 或 $(u,\ v)\notin\Delta E_t$，则该断言自动成立。因此，我们考虑 $\mathsf{ALG}(u,\ v)=1$ 且 $(u,\ v)\in\Delta E_t$的情形。在这种情况下，u 和 v 必定都属于 C_t，因为 $(u,\ v)\in E^-\cap\Delta E_t$。根据 C_t的定义，$d_{uw}\leqslant b\cdot r$ 和 $d_{vw}\leqslant b\cdot r$。由近似三角不等式，$d_{uv}\leqslant 2\delta_1 b\cdot r+\delta_2$。这给出

$$\mathrm{profit}_t(u,v)=\widehat{d}_{uv}-r\cdot\mathsf{ALG}(u,v)=1-d_{uv}-r\geqslant 1-2\delta_1 b\cdot r-\delta_2-r\geqslant 0,$$

其中，最后一个不等式由常数的选取可得。 □

引理5（引理B.4的类似结果，见 [16]）。设 $u\in V_t$和 $d_{uw}\in[0,c_1\cdot r]$。则收益$_t(u,v)\geqslant 0$ 对所有 $v\in V_t$ 和 $(u,v)\in\Delta E_t$均成立。

证明。由于断言 3，我们可以假设 $(u,\ v)\in E^+$。由于 $d_{uw}\leqslant c_1\cdot r$ 和 $c_1\leqslant b$，我们有$u\in$球$(w,\ b\cdot r)$，因此 $u\in C_t$。所以 $(u,\ v)\in E^+$是不一致，当且仅当 $v\notin C_t$，即当且仅当 $d_{vw}>b\cdot r$。根据我们对常数的选取

$$\begin{aligned}\mathrm{profit}_t(u,v)&\geqslant\widehat{d}_{uv}-r\mathsf{ALG}(u,v)\geqslant d_{uv}-r\geqslant\frac{1}{\delta_1}d_{vw}-d_{uw}-r-\delta_2/\delta_1\\&\geqslant\frac{1}{\delta_1}b\cdot r-c_1\cdot r-r-\delta_2/\delta_1\geqslant 0.\end{aligned}$$

□

引理6（引理B.5的类似结果，见 [16]）。设 $u\in V_t$和 $d_{uw}\in[c_2\cdot r,1]$。则收益$_t(u,v)\geqslant 0$ 对所有 $v\in V_t$ 和 $(u,v)\in\Delta E_t$均成立。

证明。根据断言 3，我们可以假设 $(u,\ v)\in E^+$。由于 $d_{uw}\geqslant c_2\cdot r$ 和 $c_2>b$，我们有$u\notin$球$(w,\ b\cdot r)\cap V_t=C_t$。进而，$(u,\ v)\in E^+$是不一致，当且仅当 $v\in C_t$（注意$(u,\ v)\in\Delta E_t$根据假设），即 $d_{vw}\leqslant b\cdot r$。根据常数的选取，这意味着

$$\mathrm{profit}_t(u,v)=d_{uv}-r\geqslant\frac{1}{\delta_1}d_{uw}-d_{vw}-r-\delta_2/\delta_1\geqslant\frac{1}{\delta_1}c_2\cdot r-b\cdot r-r-\delta_2/\delta_1\geqslant 0.$$

□

断言 4（[16]中的断言B.6的类似表述）。令 $u\in V_t,\ d_{uw}\in(c_1\cdot r,\ c_2\cdot r]$，以及 $v\in$球$(w,\ r)\cap V_t$。那么收益$_t(u,\ v)\geqslant r-d_{vw}$。

证明。

情形 5。$d_{uw}\in(c_1\cdot r,\ b\cdot r]$。

在此情形下，$u\in C_t$，因为 $d_{uw}\leqslant b\cdot r$。此外，由于 $v\in$球$(w,\ r)\cap V_t$ 且 $b\geqslant 1$，我们有$v\in$球$(w,\ b\cdot r)\cap V_t$，因此 $v\in C_t$ 也成立。因此，若 $(u,\ v)\in E^+,\ ALG(u,\ v)=0$，并根据常数的选取

$$\begin{aligned}\mathrm{profit}_t(u,v)=\widehat{d}_{uv}-r\mathsf{ALG}(u,v)=d_{uv}-0&\geqslant\frac{1}{\delta_1}d_{uw}-d_{vw}-\delta_2/\delta_1\\&\geqslant\frac{1}{\delta_1}c_1\cdot r-d_{vw}-\delta_2/\delta_1=r-d_{vw}.\end{aligned}$$

另一方面，如果 $(u,\ v)\in E^-$，那么

$$\begin{aligned}\mathrm{profit}_t(u,v)=\widehat{d}(u,v)-r\cdot\mathsf{ALG}(u,v)&\geqslant 1-d_{uv}-r\geqslant 1-\delta_1 d_{uw}-\delta_1 d_{vw}-r-\delta_2\\&=1-\delta_1 d_{uw}-r-(\delta_1-1)d_{vw}-d_{vw}-\delta_2\\&=1-(\delta_1 b+\delta_1)r-d_{vw}-\delta_2\geqslant r-d_{vw}\end{aligned}$$

案例 6。$d_{uw}\in(b\cdot r,\ c_2\cdot r]$。

如果 $(u,\ v) \in E^+$，那么

$$\begin{aligned}\text{profit}_t(u,v) = \widehat{d}_{uv} - r\mathsf{ALG}(u,v) &\geqslant d_{uv} - r \geqslant \frac{1}{\delta_1} d_{uw} - d_{vw} - r - \delta_2/\delta_1 \\ &= (\frac{1}{\delta_1} b - 1)r - d_{vw} - \delta_2/\delta_1 \geqslant r - d_{vw}.\end{aligned}$$

若 $(u,\ v) \in E^-$，则

$$\begin{aligned}\text{profit}_t(u,v) = \widehat{d}_{uv} - r\mathsf{ALG}(u,v) &\geqslant 1 - d_{uv} - r \geqslant 1 - \delta_1 d_{uw} - \delta_1 d_{vw} - r - \delta_2 \\ &\geqslant 1 - (\delta_1 c_2 + \delta_1)r - d_{vw} - \delta_2 \geqslant r - d_{vw},\end{aligned}$$

其中我们用到了 $\delta_1 \geqslant 3$ 蕴含 $1 - \delta_1 < 0$。 □

引理 7 （ [16]中的引理 B.7 的类似结果）。 若 $d_{uw} \in (c_1 \cdot r,\ c_2 \cdot r]$，则 $P_{high}(u) \geqslant L_t(w)$。

证明。 根据断言 4，收益$_t(u,\ v) \geqslant r - d_{vw}$对所有 $v \in$球$(w,\ r) \cap V_t$均成立。因此

$$P_{high}(u) = \sum_{v \in \text{Ball}(w,r) \cap V_t} \text{profit}_t(u,v) \geqslant \sum_{v \in \text{Ball}(w,r) \cap V_t} r - d_{vw} = L_t(w).$$

□

断言 5 （ [16]中的断言 B.8 的类似结果）。 设 $u,\ v \in V_t$。则收益$_t(u,\ v) \geqslant \min(d_{uv} - r,\ 0)$.

证明。 若 $(u,\ v) \in E^-$，则收益$_t(u,\ v) \geqslant 0$，根据断言 3。若 $(u,\ v) \notin \Delta E_t$，则收益$_t(u,\ v) = 0$，根据定义。因此我们可以假设 $(u,\ v) \in E^+ \cap \Delta E_t$。在这种情况下，

$$\text{profit}_t(u,v) = \widehat{d}_{uv} - r\mathsf{ALG}(u,v) \geqslant d_{uv} - r \geqslant \min(d_{uv} - r, 0).$$

□

引理 8 （ [16]中引理 B.9 的类似结论）。 设 $u \in V_t$。则 $P_{low}(u) \geqslant -L_t(u)$，其中 $P_{low}(u) = \sum_{v \in V_t \setminus Ball(w,\ r)}$收益$_t(u,\ v)$。

证明。

$$\begin{aligned}P_{low}(u) = \sum_{v \in V_t \setminus \text{Ball}(w,r)} \text{profit}_t(u,v) &\geqslant \sum_{v \in V_t \setminus \text{Ball}(w,r)} \min(d_{uv} - r, 0) \\ \geqslant \sum_{v \in V_t} \min(d_{uv} - r, 0) &= \sum_{v \in \text{Ball}(u,r) \cap V_t} d_{uv} - r = -L_t(u)\end{aligned}$$

其中在第二行中我们使用了断言 5，在第三行中我们使用了所有项均为非正这一事实，并且在第四行中我们使用了：如果 $v \notin$球$(u,\ r)$，则 $\min(d_{uv} - r,\ 0) = 0$。 □

□

5.4 完成定理 2 的证明

本节将前面各节的分析联系起来，以证明采样可用于改进我们针对最小最大相关聚类的组合算法的运行时间，同时仍能获得常数因子近似。

定理 *2* 的证明。 我们针对 ℓ_∞ 范数的相关聚类采样算法如下。按照第 5.2 节所述，通过采样计算 d 。将 d 作为算法 2 的输入。令 $\mathsf{ALG}(u, v) = \mathbb{1}((u, v)$ 在 $\mathcal{C})$ 中存在不一致，且 $\mathsf{ALG}(u) = \sum_{v \in V} \mathsf{ALG}(u, v)$。若 $v \in N_u^+$，则定义 $d_{uv} = d_{uv}$；若 $u \in N_v^-$，则定义 $d_{uv} = 1 - d_{uv}$。根据第 (18) 行，我们看到，以概率 $1 - O(1/n)$,

$$\mathsf{ALG}(u) = \sum_{v \in V} \mathsf{ALG}(u, v) \underset{*}{\leqslant} \frac{1}{r(\varepsilon)} \cdot \sum_{v \in V} \widehat{\widetilde{d}}_{uv} \underset{**}{\leqslant} \frac{1}{r(\varepsilon)} \cdot D(\varepsilon) \cdot \mathsf{OPT}. \tag{27}$$

其中 $D(\varepsilon) = 2 \cdot \left(\frac{1+\varepsilon}{1-\varepsilon}\right)^2$ 和 $r(\varepsilon) = r(\delta_1, \delta_2)$（回顾一下，$\delta_1 = \delta_1(\epsilon)$ 和 $\delta_2 = \delta_2(\epsilon)$），如附录 B 中所定义。不等式（$*$）由命题 8 和引理 3 得出，而不等式（$**$）由命题 9 得出。

接下来，我们分析运行时间。如前所述，共有两个阶段：（1）计算估计值 $\widetilde{d}_{uv}$；（2）使用以 d 为输入的舍入算法（算法 2）。阶段（1）需要 $O(n^2 \log n/\varepsilon^2)$ 时间。首先，我们为每个顶点 u 计算样本 S_u，由于我们从 N_u^+ 中抽取 $O(\log n/\varepsilon^2)$ 个顶点，因此这需要 $O(n \log n/\varepsilon^2)$ 时间。然后，对于 $O(n^2)$ 对中的每一对 u，v，我们计算 $W^{(u,\,v)}$ 和 $Y^{(u,\,v)}$，这需要 $O(\log n/\varepsilon^2)$ 时间。因此，对于所有成对顶点，按照（22）计算 $\bar{d}_{uv}$ 需要 $O(n^2 \log n/\varepsilon^2)$ 时间。最后，通过舍入从 $\bar{d}$ 获得 $\widetilde{d}$ 需要 $O(n^2)$ 时间。根据附录 A 中的讨论，阶段（2）需要 $O(n^2)$ 时间。两个阶段总共需要 $O(n^2 \log n/\varepsilon^2)$, 的运行时间，定理 2 的证明完毕。

□

6 实验

在本 节中，我们介绍支持理论结果的实验。[7] 我们展示 e:

- 定理 1 的保证能够预测我们的算法在真实世界数据集和合成数据集上的性能。我们的算法的解质量与 KMZ 算法相近。
- 相关性度量的分数成本与线性规划 3.1 的目标值相近。
- 我们的算法具有 可扩展性：我们可以处理大规模图（最多包含 $\approx 10{,}000$ 个顶点），而 KMZ 算法由于求解（规模庞大的）线性规划这一瓶颈，仅适用于最多包含 ≈ 300 个顶点的图。
- 我们算法找到的大型簇可能具有实际意义，因为该算法能够部分发现真实世界和合成实例中的“真实情况”簇。

我们的实验聚焦于精确算法。我们观察到，精确算法在实际应用中足够快。

真实数据集描述。我们从斯坦福大型网络数据集合集获取了表示社交网络的数据集 [19, 22, 18]。[8] 具体而言，我们使用了*ego-Facebook*数据集，其中包含 10 个图，这些图是 Facebook 社交网络的子图。每个子图（即自我网络）代表特定用户的好友列表及其中的连接关系。我们将其转换为完全符号图：用户之间存在连接时表示为正边，不存在连接时表示为负边。（有关这些图的统计信息，请参见附录C.2中的表4和表5。）每个自我网络都附有“真实情况”圈；每个圈都是用户标记为一个社群的一组顶点。需要注意的是，这些圈不一定构成好友列表的划分，因为它们可能彼此重叠，也可能未涵盖整个好友列表。

对于每个 Facebook 自我网络，我们都使用矩阵乘法实现运行了我们的精确算法。对于其中五个规模足够小的自我网络，我们还求解了

[7]我们的实验代码可在 https://github.com/hanewman/MinMax-Correlation-Clustering- [8] https://snap.stanford.edu/data/ego-Facebook.html 找到。

线性规划，以界定我们算法的近似比。所用的线性规划求解器是 Gurobi。对于后面提到的数据集，我们还运行了 KMZ 算法，作为额外的比较方法。设 r_1 为 $L_t(\cdot)$ 中的半径，设 r_2 为算法 2 中用于截取 C_t 的半径。尽管定理 1 对 $r_1 = 1/5$ 和 $r_2 = 2/5$, 成立，但在实践中，这些半径可能会使目标值接近稀疏图中的最大正度。通过更保守地设置超参数，我们能够取得甚至优于定理 1 所保证的结果。我们进行了参数扫描（附录 C.2 中的图 3），发现 $r_1 = r_2 = 0.7$ 适用于我们的算法，而 $r_1 = r_2 = 0.4$ 适用于 KMZ 算法，因此报告了使用这些参数得到的结果。各数据集的最佳半径另见附录 C.3。最后，我们对所有数据集运行了 Pivot 算法（见附录 C.1），以作进一步比较 [4]。参见表 1、表 2 和图 3。

近似质量。 对于表1中的五个小型数据集，我们的算法的代价最多是线性规划代价的2倍，因此最多是最优值的两倍。就目标值而言，我们的算法与KMZ 算法表现相近。此外，相关性度量的分数成本与线性规划的代价始终相差约2倍。最后，我们的算法的目标值通常略低于其分数成本。对于大型数据集（表2），由于运行线性规划的代价过高，我们无法运行线性规划；由于线性规划无法扩展，我们没有最优值的下界，因此无法界定近似比。对于这些数据集，我们将算法与Pivot进行比较，结果表明我们的算法以显著优势胜过Pivot。

	分数成本	线性规划目标	我们的目标值	KMZ目标	Pivot 目标值
FB 348	74.37	39.13	72	89	85.03
FB 414	35.53	19.66	34	38	50.73
FB 686	58.59	30.48	47	69	65.72
FB 698	22.31	10.64	20	18	23.51
FB 3980	14.31	7.34	12	13	16.36

表1： 我们算法与KMZ算法在五个小型 Facebook 数据集上的分数值和目标值比较。“分数成本”列记录了相关性度量的分数成本。此外，我们还运行了Pivot算法；记录的数值是经过500次试验取平均后的目标值。

	分数成本	我们的目标值	Pivot 目标值	我们的运行时间	#顶点
FB 0	64.02	49	71.78	0.20	333
FB 107	181.49	152	216.65	1.76	1034
FB 1684	103.99	93	130.71	0.98	786
FB 1912	227.74	220	259.01	0.93	747
FB 3437	98.36	107	99.1	0.47	534

表2： 各列与表1相同。此处每个数据集的规模都会使线性规划的运行时间变得难以接受。运行时间以秒为单位列出。

运行时间和可扩展性。我们的算法的运行时间显著快于 KMZ 算法（表3）。例如，在仅包含 224 个顶点的 FB 348 上，KMZ 算法耗时超过 30 分钟，而我们的算法仅耗时十分之一秒 [9]。

[9]即使提前终止线性规划，例如在原问题与对偶问题之间的差距达到 1 时终止，这仍然至少需要 10 分钟。不过需要注意的是，这样做会影响输出的距离。

事实上，我们可以快速处理非常大规模的图；在一个拥有 12,008 个节点的社交网络上[10]，我们的算法运行时间仅略超过 4 分钟。有关可扩展性的更多详情，请参见附录 C.5。

与真实簇的比较。我们还分析了我们的算法找到的簇是否发现了用户标注的真实圈子。对于每个数据集，我们只考虑大小至少为10的"大型"簇，因为小簇（包括若干单例聚类）意义较小。对于每个大型簇，我们找出其中包含该簇顶点数量最多的真实圈子。结果绘于附录C.2的图4中。我们发现，对于数据集 FB 348、414、686、1684和1912，几乎每个大型簇都几乎完全包含在其最匹配的真实圈子中（即每个大型簇中有80%至100%的顶点位于其最匹配的真实圈子内）。对于其他数据集，没有那么多证据表明算法发现了真实圈子。

	我们的运行时间	KMZ运行时间	#顶点
FB 348	0.10	1847.99	224
FB 414	0.06	207.92	150
FB 686	0.06	337.9	168
FB 698	0.02	3.42	61
FB 3980	0.01	2.03	52

表3： 我们的方法与 KMZ 算法在五个小型 Facebook 数据集上的运行时间（以秒为单位）比较。

合成数据集。 我们出于两个原因考虑了合成数据集。第一，在许多现实世界数据集上运行线性规划和 KMZ 算法的代价过高。第二，这是为了进一步测试我们的算法是否能够发现真实簇。我们取了一个包含 100 个顶点和 10 个大小为 10 的正团的图（因此该图存在完美聚类），并引入了 20 个级别的噪声。在每个级别 i，我们随机将 $45i$ 条边翻转为相反的符号。然后，我们像之前一样使用 $r_1 = r_2 = 0.7$ 应用了我们的算法。我们发现，对于最多 495 次翻转（$i = 11$），大小为 10 的原始聚类几乎完全由我们的算法保留（在某些情况下，最多有三个顶点脱离并形成单例聚类）。我们还发现，在所考虑的所有噪声级别中，我们找到的几乎每个簇都至少有 88% 包含于某个真实簇中（仅有 6 个簇例外）。有关这些实验的更多图表，请参见附录 C.4。

7 结论

我们提出了一种更快、完全组合的 $O(1)$近似算法，用于最小最大相关聚类。我们根据顶点的 $+$ 和 $-$ 邻域的交集大小，为问题的线性规划构造了一个分数解，并证明 Kalhan、Makarychev 和 Zhou 提出的线性规划舍入算法（见 [16]）适用于我们手工构造的分数解。单就这一点而言，这个结果就令人惊讶！它也引出了一个值得未来研究的问题：给定一个线性规划，我们何时可以利用底层实例结构中可观测的组合性质，构造出经证明质量良好的线性规划分数解？这一通用框架有望显著改进其他问题的运行时间。

另一个未来方向是为 ℓ_1 范数寻找一种手工构造的分数解，以获得用于经典相关聚类的其他组合算法。此外，尽管我们展示了相关性度量在 ℓ_∞ 范数下的成功，但它也可能作为其他 ℓ_p 范数的替代分数解发挥作用。尽管这些方向本身在理论上颇具意义

[10] https://snap.stanford.edu/data/feather-lastfm-social.html

，但寻找其他 ℓ_p 范数的快速算法也具有实际意义，因为 $p \in (1, \infty)$ 在局部公平性（$p = \infty$）和全局最优性（$p = 1$）这两个相互竞争的目标之间起到插值作用。

参考文献

[1] S. Ahmadi、S. Khuller 和 B. Saha。《通过多割实现极大极小相关聚类》。载于：整数规划与组合优化国际会议。施普林格。2019年，第 13–26 页。[2] S. Ahmadi 等。《公平相关聚类》。载于：*arXiv* 预印本 *arXiv:2002.03508*（2020年）。[3] S. Ahmadian 等。《公平相关聚类》。载于：人工智能与统计国际会议。机器学习研究论文集。2020年，第 4195–4205 页。[4] N. Ailon、M. Charikar 和 A. Newman。《聚合不一致信息：排序与聚类》。载于：*ACM*学报（*JACM*）55.5（2008年），第 1–27 页。[5] N. Bansal、A. Blum 和 S. Chawla。《相关聚类》。载于：机器学习56.1（2004年），第 89–113 页。[6] M. Bateni 等。《可扩展且改进的个体公平聚类算法》。载于：可信与负责任机器学习研讨会，*NeurIPS 2022*。[7] A. Ben-Dor 和 Z. Yakhini。《聚类基因表达模式》。载于：第三届计算分子生物学年度国际会议论文集。1999年，第 33–42 页。[8] M. Charikar、N. Gupta 和 R. Schwartz。《图割与聚类中的局部保证》。载于：整数规划与组合优化国际会议——第 *19* 届会议，*IPCO 2017*，加拿大安大略省滑铁卢，*2017*年*6*月*26*日至*28*日，论文集。F. Eisenbrand 和 J. K¨onemann 编。第 10328 卷。计算机科学讲义。施普林格，2017年，第 136–147 页。 DOI：10.1007/978-3-319- 59250-3_12。 URL：https://doi.org/10.1007/978-3-319-59250-3_12。[9] S. Chawla 等。《完全图和完全k部图上相关聚类的近最优线性规划舍入算法》。载于：第四十七届 *ACM* 计算理论研讨会论文集。2015年，第 219–228 页。[10] Y. Cheng 和 G. M. Church。《表达数据的双聚类》。载于：智能系统与分子生物学国际会议。第 8 卷。2000年。2000年，第 93–103 页。[11] M. B. Cohen、Y. T. Lee 和 Z. Song。《在当前矩阵乘法时间内求解线性规划》。载于：*ACM*学报（*JACM*）68.1（2021年），第 1–39 页。[12] V. Cohen-Addad、E. Lee 和 A. Newman。《使用 Sherali-Adams 的相关聚类》。载于：计算机科学基础研讨会（*FOCS*）。（2022年）。[13] E. D. Demaine 和 N. Immorlica。《部分信息下的相关聚类》。载于：近似、随机化与组合优化：算法与技术。施普林格，2003年，第 1–13 页。

[14] Z. Friggstad 和 R. Mousavi。《具有全局和局部保证的公平相关聚类》。载于：算法与数据结构研讨会。施普林格。2021年，第 414–427 页。[15] J. Jafarov 等。《具有非对称分类错误的局部相关聚类》。载于：国际机器学习会议。机器学习研究论文集。2021年，第 4677–4686 页。[16] S. Kalhan、K. Makarychev 和 T. Zhou。《具有局部目标的相关聚类》。载于：神经信息处理系统进展第 *32* 卷：*2019*年神经信息处理系统年度会议，*NeurIPS 2019*，*2019*年*12*月*8*日至*14*日，加拿大不列颠哥伦比亚省温哥华。H. M. Wallach 等编。2019年，第 9341–9350 页。 URL: https://proceedings.neurips.cc/paper/2019/hash/785ca71d2c85e3f3774baaf438c5c6eb-Abstract.html。[17] H.-P. Kriegel、P. Kr¨oger 和 A. Zimek。《高维数据聚类：子空间聚类、基于模式的聚类和相关聚类综述》。载于：*ACM*知识发现与数据挖掘汇刊（*TKDD*）3.1（2009年），第 1–58 页。[18] J. Leskovec、J. Kleinberg 和 C. Faloutsos。《图演化：稠密化与直径收缩》。载于：*ACM*知识发现与数据挖掘汇刊（*TKDD*）1.1（2007年），第 2–es 页。[19] J. Leskovec 和 J. Mcauley。《学习发现自我网络中的社交圈》。载于：神经信息处理系统进展。F. Pereira 等编。第 25 卷。Curran Associates, Inc.，2012年。 URL: https : / / proceedings . neurips.cc/paper/2012/file/7a614fd06c325499f1680b9896beedeb- Paper. pdf。[20] A. McCallum 和 B. Wellner。《身份不确定性的条件模型及其在名词共指中的应用》。载于：神经信息处理系统进展17（2004年）。[21] G. J. Puleo 和 O. Milenkovic。《具有局部有界误差的相关聚类与双聚类》。载于：第 *33* 届国际机器学习会议论文集，*ICML 2016*，美国纽约市，*2016*年*6*月*19*日至*24*日。M. Balcan 和 K. Q. Weinberger 编。第 48 卷。JMLR 研讨会与会议论文集。JMLR.org，2016年，第 869–877 页。 URL: http :/ / proceedings . mlr . press/v48/puleo16.html。[22] B. Rozemberczki 和 R. Sarkar。《图上的特征函数：志趣相投者，从统计描述符到参数模型》。载于：第 *29* 届 *ACM* 国际信息与知识管理会议论文集（*CIKM '20*）。ACM。2020年，第 1325–1334 页。[23] P. Symeonidis 等。《基于恒定且一致的值的最近双聚类协同过滤》。载于：信息检索11.1（2008年），第 51–75 页。[24] A. Wirth。《相关聚类》。载于：机器学习与数据挖掘百科全书。C. Sammut 和 G. I. Webb 编。施普林格，2017年，第 280–284 页。 DOI: 10.1007/978-1-4899-7687-1_176。 URL: https://doi.org/10.1007/978-1-4899-7687-1_176。

一种舍入算法及其运行时间

现在详细介绍 Kalhan、Makarychev 和 Zhou 的舍入算法，我们将借助[16]。对于顶点 $u \in V$，半径为 ρ、以 u 为中心、相对于 x 在 V 上的球定义为球$(u, \rho) = \{v \in V \mid x_{uv} \leqslant \rho\}$。该算法以迭代方式进行，每次迭代中都会对顶点进行聚类。在第 t 步，未聚类顶点的集合记为 $V_t \subseteq V$。从V_t中选取一个特殊顶点作为簇中心，具体而言，选择使下列值最大化的顶点

$$L_t(u) = \sum_{v \in \mathsf{Ball}(u,r) \cap V_t} r - x_{uv},$$

这表示球$(u, r) \cap V_t$ 中的顶点朝中心聚集的程度。

算法 2. *[舍入算法]*

输入： x 上的半度量 V。**输**

出： 聚类 $\mathcal{C}$。

1. 令 $V_0 = V$, $r = 1/5$, $t = 0$。
2. **当** $(V_t \neq \varnothing)$
   - Find $u_t^* = \arg\max_{u \in V_t} L_t(u) = \arg\max_{u \in V_t} \sum_{v \in \mathrm{Ball}(u,r) \cap V_t} r - x_{uv}$.
   - Create $C_t = \mathrm{Ball}(u_t^*, 2r) \cap V_t$.
   - Set $V_{t+1} = V_t \backslash C_t$ and $t = t + 1$.
3. 返回 $\mathcal{C} = (C_0, \ldots, C_{t-1})$。

令 $\mathsf{LP}(u, v)$ 表示边 (u, v) 在其目标值中的代价，因此可以设置 $x_{uv} = \mathsf{LP}(u, v)$，如果 $(u, v) \in E^+$；并设置 $x_{uv} = 1 - \mathsf{LP}(u, v)$，如果 $(u, v) \in E^-$。对于上述算法的输出，令 $\mathsf{ALG}(u, v) = \mathbb{1}((u, v)$ 为不一致的边 $)$，且 $\mathsf{ALG}(u) = \sum_{v \in V} \mathsf{ALG}(u, v)$。Kalhan、Makarychev 和 Zhou 证明了

$$\mathsf{ALG}(u) = \sum_{v \in V} \mathsf{ALG}(u, v) \underset{*}{\leqslant} 5 \cdot \sum_{v \in V} \mathsf{LP}(u, v) \leqslant 5y(u), \tag{28}$$

这为任意 ℓ_p 范数的完全图带来了一个 5 近似算法。他们成果中的技术工作在于证明公式 28 中以 * 标记的不等式。（第二个不等式源于线性规划是一个松弛。）

运行时间 我们首先说明舍入算法的运行时间为 $O(n^2)$，并且可以通过以下步骤获得：

- 对于每个 $u \in V$，预先计算 $L_t(u) = \sum_{v \in Ball(u,\ r)} r - d_{uv}$。这需要 $O(n^2)$ 时间。
- 在迭代 t 中，需要 $O(n)$ 时间来找出所有未聚类顶点上的 $L_t(u)$ 的最大值，即在 $u \in V_t$ 上找出最大值；创建该簇需要 $O(n)$ 时间。所有迭代累计需要 $O(n^2)$ 时间。
- 移除一个顶点时，还必须移除其对剩余顶点 u 的 $L_t(u)$ 贡献。每移除一个顶点需要 $O(n)$ 时间，因为它可能属于 $O(n)$ 个图球。此外，每个顶点只会被移除一次。因此，更新 L_t 值总共需要 $O(n^2)$ 时间。

总的来说，完整 KMZ 算法的运行时间主要取决于求解线性规划以获得半度量 x 所需的时间。该线性规划包含 $O(n^2)$ 个变量和 $O(n^3)$ 个约束，适用于 $n = |V|$。鉴于当前求解线性规划的最佳算法，仅求解线性规划就显然需要不低于 $O(n^{2\omega})$ 的运行时间 [11]。我们将论证

为什么这是一个合理的理论基准，可用于比较运行时间。在稀疏网络中，即由 $+$ 条边构成的图是稀疏的情况下，人们可能会想知道，是否可以通过固定正邻域交集为空的顶点对所对应的线性规划变量的值，来减少求解器必须求解的变量数量。然而，很容易构造出一些小型示例，表明这样做会违反三角不等式；而且，目前完全不清楚是否甚至能够保证某种近似三角不等式。最后，也许可以通过求解线性规划，更快地获得一个有严格近似最优保证的线性规划解，但这类研究将相当复杂，并且需要采用完全不同的方法。

B 第5.3节中常数的选取

令 $\varepsilon > 0$ 足够小， $\delta_1 = 3 + h_4(\varepsilon)$，并且 $\delta_2 = h_4(\varepsilon)$，其中 $h_4(\varepsilon)$ 如命题 8 中所述。注意， $h_4(\varepsilon) \to 0$ 与 $\varepsilon \to 0$一样。取

$$
\begin{aligned}
r &= \frac{1 - \delta_2 - \delta_1\delta_2 - \delta_1^3\delta_2 - \delta_1^2\delta_2}{\delta_1^2 + \delta_1^3(\delta_1 + 1) + \delta_1 + 1} \\
c_1 &= \delta_1 + \frac{\delta_2}{r} \\
b &= (c_1 + 1)\delta_1 + \frac{\delta_2}{r} \\
c_2 &= \delta_1(b + 1) + \frac{\delta_2}{r}.
\end{aligned}
$$

可以验证，以下不等式满足第5.3节中的需要。

$$
\begin{aligned}
&b \geqslant 1 && c_2 \cdot r < 1 \\
&c_1 \leqslant b < c_2 && 1 - 2\delta_1 b \cdot r - \delta_2 - r \geqslant 0 \\
&\frac{1}{\delta_1} b \cdot r - c_1 \cdot r - r - \delta_2/\delta_1 \geqslant 0 && \frac{1}{\delta_1} c_2 \cdot r - b \cdot r - r - \delta_2/\delta_1 \geqslant 0 \\
&\frac{1}{\delta_1} c_1 \cdot r - \delta_2/\delta_1 \geqslant r && 1 - (\delta_1 b + \delta_1)r - \delta_2 \geqslant r \\
&(\frac{1}{\delta_1} b - 1)r - \delta_2/\delta_1 \geqslant r && 1 - (\delta_1 c_2 + \delta_1)r - \delta_2 \geqslant r
\end{aligned}
$$

C 补充实验信息

C.1 Pivot算法说明

Ailon、Charikar 和 Newman 提出的 Pivot算法是一种随机化算法，对于完全图 [4] 上的经典相关聚类（即 ℓ_1 范数），其期望近似比为 3-近似。对于 Min Max目标，Pivot 的表现可能较差 [21]。Pivot算法如下：对顶点进行均匀随机排序。按此顺序遍历顶点。访问某个顶点时，检查它是否已被标记为已聚类。如果是，则访问下一个顶点。否则，将当前顶点称为枢轴，并新建一个簇，其中包含该枢轴以及所有尚未聚类且属于枢轴的正邻域的顶点。将这些顶点标记为已聚类。

C.2 第6节的补充图表

本节包含第6节所讨论实验的补充图表。

	FB 348	FB 414	FB 686	FB 698	FB 3980
#顶点	224	150	168	61	52
#边	6384	3386	3312	540	292
最大正度	100	58	78	30	19

表4： 表1中五个小型 Facebook 数据集的图统计信息。

	FB 0	FB 107	FB 1684	FB 1912	FB 3437
#顶点	333	1034	786	747	534
#边	5038	53498	28048	60050	9626
最大正度	78	254	137	294	108

表5： 表2中五个大型 Facebook 数据集的图统计信息。

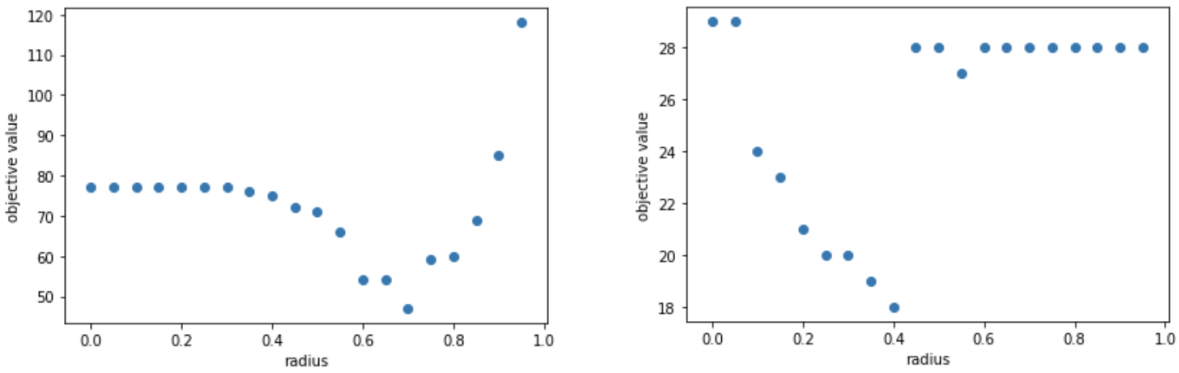

图3： 展示当我们扫描公共半径 $r_1 = r_2$ 时，我们的算法（左）和 KMZ 算法（右）的目标值变化情况的图，分别对应数据集 FB 686（左）和 FB 698（右）。其他数据集的图相似，因此在表1和表2中，我们分别为我们的算法和 KMZ 算法使用了 0.7 和 0.4 的半径。

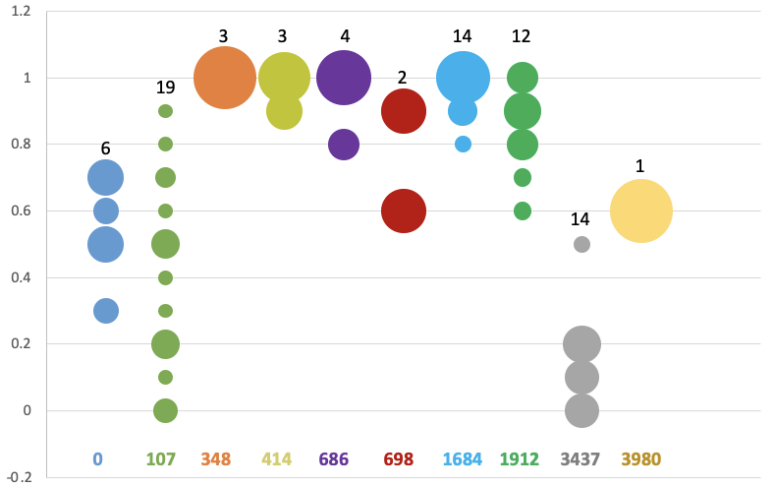

图4： 对于每个数据集（下方），其上方的气泡表示我们算法发现的大型簇（大小为 $\geqslant 10$）在多大程度上包含于真实簇中。每个数据集的大型簇数量标注在列顶端。每个气泡代表相应数据集中的一定数量的大型簇；气泡大小与其代表的大型簇数量成正比。对于每个大型簇，我们找出其最佳真实簇，即包含该大型簇中顶点数量最多的真实簇。 y 轴表示大型簇包含在其最佳真实簇中的比例。例如，对于数据集 FB 698，假设某个大型簇中有 90%（含）至 100%（不含）的顶点包含在其最佳真实簇中，那么它由垂直位置在 0.9 处的红色圆圈表示。若某个大型簇中有 60% 至 70% 的顶点包含在其最佳真实簇中，则由垂直位置在 0.6 处的红色圆圈表示。由于这两个红色圆圈大小相同，因此一半的大型簇属于这两类中的每一类。请注意，对于数据集 FB 348、414、686、1684 和 1912，在 y 轴位置为 1 处有气泡；这些气泡表示完全（即 100%）包含在真实簇中的大型簇。尤其是对于数据集 FB 348， *every* 大型簇 100% 包含在一个真实簇中。

C.3 Facebook 数据集上的双参数扫描结果

为便于讨论第6节和附录C.2中的内容，我们通过强制采用公共半径 $r = r_1 = r_2$，将我们的算法和 KMZ 算法应用于 Facebook 数据集。对于我们的算法，我们使用了 $r = 0.7$；对于 KMZ 算法，我们在 所有数据集上使用了 $r = 0.4$ 。在本节中，我们给出更具针对性的分析：对于每个数据集和每种算法，我们分别为该数据集找到最佳的 r_1 和 r_2（不要求 $r_1 = r_2$）。结果列于表6和表7中。尽管这些结果与第6节和附录C.2中报告的结果相似，但为完整起见，我们仍将其列出。

	FB 348	FB 414	FB 686	FB 698	FB 3980
分数成本	74.37	35.53	58.59	22.31	14.31
线性规划目标	39.13	19.66	30.48	10.64	7.34
我们的目标	71	31	43	18	12
KMZ目标	69	28	47	17	13
Pivot 目标值	85.03	50.73	65.72	23.51	16.36
(r_1, r_2)	(0.45, 0.7)	(0.1, 0.65)	(0.3, 0.75)	(0.8, 0.8)	(0.7, 0.7)
KMZ (r_1, r_2)	(0.2, 0.45)	(0.05, 0.6)	(0.3, 0.6)	(0.05, 0.5)	(0.3, 0.6)

表6： 五个小型 Facebook 数据集的实验结果。这些对 (r_1, r_2) 和 KMZ (r_1, r_2) 指的是双参数扫描中的一对最优半径（而不是像第5.3节和附录C.2中那样设置 $r_1 = r_2$），并用于每种算法的舍入阶段。

	FB 0	FB 107	FB 1684	FB 1912	FB 3437
分数成本	64.02	181.49	103.99	227.74	98.36
我们的目标值	49	134	93	187	77
Pivot 目标值	71.78	216.65	130.71	259.01	99.1
(r_1, r_2)	(0.7, 0.7)	(0.65, 0.65)	(0.7, 0.7)	(0.65, 0.65)	(0.3, 0.7)

表7： 五个较大型 Facebook 数据集的实验结果。如表6所示，(r_1, r_2) 指双参数扫描中的最优半径。

C.4 合成数据：含噪声的完美聚类

下面展示第6节中讨论的合成实验的各种图表。

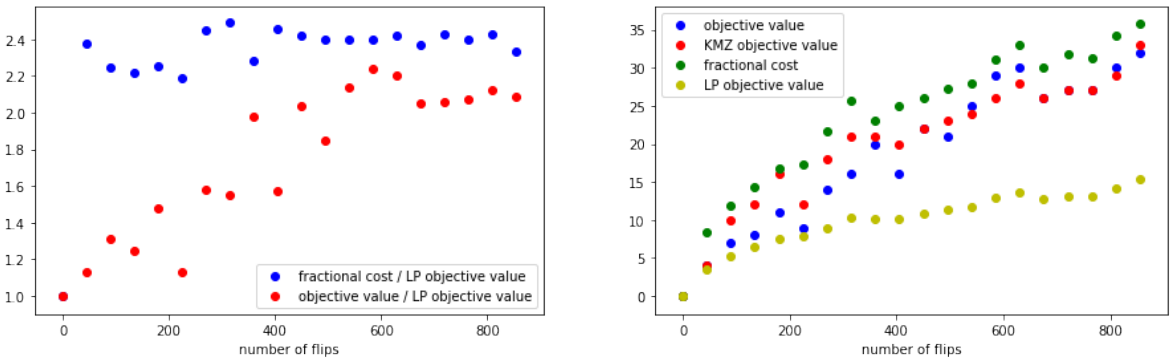

图5： 在上述图中，“分数成本”指我们相关性度量的分数成本。“目标值”指我们算法的目标值，而“KMZ目标值”指 KMZ 算法的目标值。对于我们的算法，这些目标值对应于 $r_1 = r_2 = 0.7$；对于 KMZ 算法，则对应于$r_1 = r_2 = 0.4$。

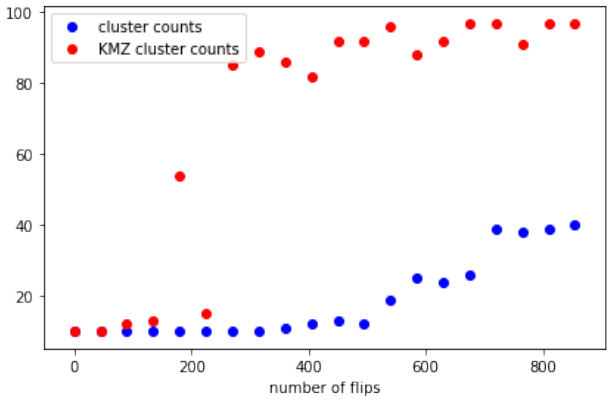

图 6　　： “簇数量”一词指我们算法输出的簇数量，　　d
“KMZ 簇 数量”指 KMZ 算法输出的数量。

C.5 我们算法的可扩展性

为了展示我们算法的可扩展性，我们在三个大型数据集上运行了我们的精确算法（使用矩阵乘法实现），每个数据集约包含 10,000 个顶点。第一个数据集记为 LastFM [11]，是亚洲音乐服务 LastFM 的用户社交网络，其中顶点表示用户，正边表示相互关注关系 [22]。另外两个数据集记为 ca-HepTh 和 ca-HepPh [12]，是 arXiv 上高能物理作者的协作网络，其中顶点表示作者，正边表示合作者 [18]。有关我们的算法在这三个数据集上的运行时间，请参见表8。我们观察到，该算法在每个数据集上的运行时间约为两到四分钟。

	我们的运行时间	#顶点	#边
LastFM	102.74	7624	27806
ca-HepTh	165.42	9877	51971
ca-HepPh	250.91	12008	237010

表8： 我们的精确算法在三个大型数据集上的运行时间（以秒为单位）及其规模。

[11] https://snap.stanford.edu/data/feather-lastfm-social.html [12] https://snap.stanford.edu/data/ca-HepTh.html, https://snap.stanford.edu/data/ca-HepPh.html