arXiv:2302.00566v3 [quant-ph] 8 Aug 2024

适用于NISQ的基于测量的量子聚类算法

SrushtiPatil[1,2], Shreya Banerjee[3,4*], Prasanta K. Panigrahi[3,5]

[1]*NNF*量子计算计划，*Neils Bphr*研究所，哥本哈根大学 哥本哈根 N DK-2200，丹麦。[2*]物理系，印度科学教育与研究学院蒂鲁帕蒂分校，安得拉邦，517619，印度。[3]量子科学与技术中心，Siksha 'O' Anusandhan大学，布巴内斯瓦尔，奥迪沙邦 751030，印度。[4]谢布鲁克大学量子研究所，魁北克，J1K2R1 加拿大。[5]物理科学系，印度科学教育与研究学院加尔各答分校，西孟加拉邦，741246，印度。

*通讯作者。电子邮箱：shreya93ban@gmail.com；投稿作者：srushtipatil926@gmail.com；director.cqst@soa.ac.in；

1 引言

量子霸权相较于经典计算机，目前是一个热门研究课题， [1], 其中，如何展示量子优势尤其具有挑战性。借助量子计算机，可以提升某些经典算法的性能。为实现加速，人们正在开发受纠缠和叠加原理等基本量子定律启发的量子算法，从而实现量子并行性。Feynman 于 1982 年首次认识到量子计算的潜力，并提出需要量子计算机来模拟复杂的量子系统，例如多体相互作用 [2]。展现出相较于经典算法的量子优势的知名量子算法包括 Deutsch-Jozsa算法 [3, 4], Bernstein-Vazirani算法 [5], Shor算法 [6], Grover算法 [7], 用于求解线性方程组的 Lloyd算法 [8], 等。当前，数据正呈指数级增长，通常称为大数据。数据分析，或更广义地说，

数据科学已成为一个活跃的研究领域，旨在揭示模式和结构。大数据在模式识别等多个领域引发了一项基本问题 [9, 10],图像处理 [11], 机器学习 [12], 等领域中的聚类问题 [13, 14]。迄今为止，已有多种用于聚类的经典和量子算法被提出，其资源复杂度各不相同。经典聚类算法包括 K 均值聚类 [15], 含噪应用的基于密度的空间聚类（DBSCAN） [16], 等。首个量子聚类算法由*Durr* 等人 [17]开发。此外，*Li* 等人 [18] 开发了一种基于量子博弈的聚类算法，并提出了另一种使用量子游走的量子算法 [19]。*Yu* 等人 [20] 针对多变量非线性问题提出了一种基于量子聚类的算法。文献中还提出了其他若干利用量子力学定律的算法 [21], ，尽管这些算法通常采用 Grover算法的变体，因此需要复杂度未知的量子黑盒 [7]。近期一些研究也探讨了量子方法如何提供替代性的聚类方法，例如概率量子聚类方法 [22],基于变分算法的量子启发式聚类 [23], 量子K均值聚类 [24, 25],基于无监督学习算法的量子谱聚类 [26], 适用于多维数据集的量子聚类算法 [27] 等。

在本研究中，我们提出了两种新颖的基于测量的量子聚类算法。就待聚类数据量和最远点之间的距离而言，这两种算法的时间复杂度均为多项式级。此外，它们有效地消除了对黑箱的需求。本文提出的第一种算法采用分裂式方法，我们称之为量子层次聚类算法（QHCA）。第二种算法是基于非锐测量的聚类算法（UMCA），它通过对数据集中的数据点进行非锐测量，并适当选择高斯窗的方差来实现聚类。对于这两种算法，我们发现其操作复杂度为 $O(D_{max}^2)$ 量级，其中 D_{max} 是数据集中任意两点之间的最大距离，此外还存在一个线性因子。本研究表明，在聚类结果保持较高准确率的情况下，可以将 D_{max} 适当地缩减至 $O(log_N)$。与其他量子聚类算法相比，我们发现所提出的方法具有相近的复杂度，同时更易于在含噪中等规模量子（NISQ）设备上实现。我们还在同心圆数据集上实现了这些算法；经典分裂式聚类方法无法处理该数据集，而我们的算法在准确率方面表现良好，且量子比特开销极低、测量复杂度较高。我们进一步表明，这些算法只需极少的量子比特和测量复杂度，因此适用于 NISQ 计算机。我们在包含 130 个城市的 Churrtiz 数据集上实现了 QHCA，并证明随着量子比特资源减少，该算法仍能良好运行。此外，我们还在包含 699 个数据点的带标签威斯康星州乳腺癌数据集上实现了 UMCA，发现仅使用 4 个量子比特以及关于量子比特数呈多项式级的测量次数，即可达到约 94% 的准确率。随后我们证明，在假设存在测量误差的情况下，UMCA 的表现优于无噪声测量，这使其特别适用于 NISQ 设备。

本文结构安排如下。在第2节，我们概述量子力学中的层次聚类和非锐测量。第3节介绍两种算法的工作流程。第4 节将介绍我们在多个基准数据集上实现这些算法的情况。最后，我们在第5节总结全文并展望未来工作。

2 前置知识

本节将简要回顾标准的聚类方法和非锐测量的基础知识。

2.1 经典聚类方法

聚类是指将具有相似特征的数据点分组。常用的聚类算法包括 K均值、DBSCAN、均值漂移聚类和层次聚类。不同算法对数据点进行聚类的方式各不相同。本文主要关注层次聚类，因为我们提出的量子算法之一采用了类似的方法。层次聚类是一种聚类分析方法，可以根据数据的相似性构建树状结构。层次聚类采用自底向上的方法时，称为凝聚式聚类：将所有数据点视为单个簇，再根据数据的相似性将其合并成更大的簇。该过程不断重复，直至形成一个完整的单一簇。使用层次聚类进行聚类的另一种方式是采用分裂式方法，即自顶向下进行。分裂式聚类通过递归地将较大的数据集 C 拆分成两个较小的子数据集，直到得到所需数量的簇。这种方法擅长识别大型簇，常见于数据挖掘、图像分割、决策等领域。通过这种方法得到的簇以层次二叉树的形式呈现，因此在许多现实问题中颇具吸引力，例如索引问题。分裂式聚类可用于将数据集拆分成更小的数据集，直至所有簇都只包含一个元素。对于包含 N 个对象的数据集，构建其层次结构需要 $N-1$步，每一步都有 $2^{N-1}-1$ 种将数据拆分为两个簇的可能方式。而在凝聚式方法中，将任意两个数据点聚类到一起会产生 $N(N-1)/2$种可能组合。凝聚聚类算法的时间复杂度为 $O(N^3)$，因为每次 $N-1$ 迭代都必须扫描 $N\times N$ 距离矩阵，以找出最小距离。使用优先队列可以将其降至 $O(N^2\log N)$。经过一些优化，时间复杂度还可降至 $O(N^2)$ [28]量级。采用穷举搜索的分裂式聚类，其时间复杂度为 $O(2^N)$。要合并（凝聚式）或拆分（分裂式）数据集，需要考量数据点之间的（不）相似度。对于任意两个数据点 $x\in C$ 和 $y\in C$，该度量由一个非负实值距离矩阵 $D_{xy}=d(x,\ y)$给出，其中每个数据点 $x\in C$ 具有 k 个属性，可表示为元组 $x=(x_1,\ x_2,\ \ldots\ldots,\ x_k)$。闵可夫斯基度量是一类用于量化此类不相似度的度量。对于

固定的 $p \geq 1$，对于任意两个数据点 x 和 y，其定义为：

$$L_p(x,y) = \left(\sum_{i=1}^{k} |x_i - y_i|^p \right)^{\frac{1}{p}} \tag{1}$$

对于 $p = 2$，它就变成了欧几里得度量。通常，欧几里得距离用作两个数据点之间的不相似性度量。

2.2 量子聚类方法

首个量子化聚类算法由 Durr 等人提出 [17], ，尽管该算法并非为解决聚类问题而开发。该算法计算了图问题的量子查询复杂度，并表明，在矩阵模型中，最小生成树的查询复杂度为 $O(N^{3/2})$ 阶；在数组模型中则为 $O(\sqrt{MN})$ 阶，其中 N 和 M 分别表示图中的顶点数和边数。计算出图的最小生成树后，可以通过移除给定图中最长的 $k-1$ 条边，将数据点划分为 k 个簇。已知矩阵模型的经典查询复杂度为 $O(N^2)$ 阶。此外，他们指出，基于最小生成树的聚类算法接近最优，即其他任何经典或量子算法都无法以优于 $O(N^{3/2})$ 的时间复杂度完成聚类。在基于最小生成树、分裂式聚类和 k-中位数的量子化聚类方法中，这些方法的速度都快于其经典对应方法 [29]。量子 k-中位数算法单次迭代的运行时间为 $O(\frac{1}{\sqrt{k}N^{3/2}})$，比经典方法快 $\sqrt{N/k}$ 倍。量子分裂聚类的运行时间为 $O(N\log N)$。在量子版本的 c-邻域图构建中，其中 c 为最近邻的数量，时间复杂度为 $O(dN\log N)$；其中 d 是数据点所在空间的维数，N 为数据点数量。然而，这些量子聚类算法使用了酉黑盒，较难实现；相比之下，本文提出的算法易于实现，且仅需一组固定的双量子比特门。我们在表1中对量子聚类算法的时间复杂度和可实现性进行了比较。

2.3 量子力学中的非锐测量

在量子力学中，测量过程以算符进行数学描述。对于给定的量子态 $|\psi\rangle$，测得与算符 $\hat{A}$ 相关的特定本征值 a 的概率为 $P(a) = |\langle\psi_a|\psi\rangle|^2$，其中 $|\psi_a\rangle$ 是 $\hat{A}$ 的本征态，其本征值为 a。量子力学的这种基本概率特性凸显了它与经典物理的显著差异：在经典物理中，测量通常不会改变被测系统的状态。量子测量由一组 $\{M_i\}$ 测量算符描述，其中索引 i 表示测量结果。假设与测量算符 M_i 相关的测量作用于量子态 $|\psi\rangle$，我们可以定义另一个算符 $E_i^{jk} = M_i^{jl}\, M_i^{lk}$，它满足 $\sum_i E_i = I.$。这组算符

算法	运算复杂度
量子 最小生成树 [17]	$O(N^{3/2})$
量子 K-中位数 [29]	$O\left(\frac{1}{\sqrt{kN^{3/2}}}\right)$
量子分裂聚类 [29]	$O(MN$ 对数 $N)$
量子 K-均值 [24]	$O(\sqrt{MNK})$
QHCA	$O(KN$(对数 $N)D_{max})$
UMCA	$O(KN$(对数 $N)\sqrt{D_{max}})$

表1。各种聚类算法的时间复杂度比较。 N、 M 和 K 分别表示数据点数量、单个数据点的维度以及所需簇的数量。

$\{E_i\}$ 称为 POVM（正算子值测度），而这些算符本身称为 POVM元素。一般而言，POVM元素彼此不必正交。算符集合 E_i 足以确定测量不同结果的概率。广义测量中的一个特殊情形是，测量算符彼此正交，这种测量称为投影测量或锐测量。这类算符的特征向量构成希尔伯特空间的一组正交归一基，测量结果对应于其中一个基态。这些测量称为投影测量或投影值（PV）测量，因为系统的初始状态 $|\psi\rangle$ 被投影到可观测量 A 的某个本征态 $|A_i\rangle$ 上。由于系统会完全坍缩到可观测量的某个本征态，从而破坏初始状态，因此这类测量称为锐测量。测量的锐性内在地存在于可观测量的数学结构中。然而在实践中，不可能对系统进行精确测量。对于经典系统，我们可以预测测量结果，因此测量是锐的；但在处理真实系统时，量子事件具有模糊性，测量也不精确，通常称为非锐。非锐测量由效应算子 表示，或简称为效应。任何量子非锐事件都可以定义为量子锐事件的加权平均。量子测量事件也可以看作一个特征谱介于 0 和 1 之间的自伴算子。在这一扩展框架下，我们定义正算子值测度（POVM），它由某些正自伴算子构成，其特征谱位于区间 [0,1]内。由此可将量子效应定义为一个自伴算子 E，其特征谱介于 0 和 1 [30]之间。效应是满足 $0 \leq E \leq I$的正有界自伴算子。举例来说，对于具有连续基集 $\{|x\rangle\}$ 且满足正交条件 $\langle x|x'\rangle = \delta_{xx'}$ 的无限维希尔伯特空间，可以将一个效应定义为：

$$E_y = \int_{-\infty}^{\infty} dx \frac{1}{(\sqrt{2\pi\Delta^2})} e^{\frac{-(y-x)^2}{(2\Delta^2)}} |x\rangle \langle x| , \tag{2}$$

其中， Δ 是该分布的标准差。它描述了对位置 y的一种不精确测量。所有效应算子都满足完备关系。算子 E_y 表示对点 y处位置的非锐测量。这些算子确实构成了一组效应算子，因为它们满足完备关系且均为正算子。

3 量子聚类算法

本节将介绍我们提出的基于测量的聚类算法。在第一个算法中，我们利用量子纠缠和并行性进行聚类。在第二个量子启发式算法中，则采用了非锐测量的概念。

3.1 量子层次聚类算法（QHCA）

相似度度量是构建聚类算法的关键因素。我们根据数据点的特征定义距离度量，以判断两个数据点的相似程度。我们考虑一个由 N 个元素组成的数据集，其中每个元素具有 d 个属性。任意两个数据点之间的不相似度由欧几里得距离度量给出，如公式1所示，其中 $p = 2$。距离度量相近的数据点会被聚到一起。下面介绍我们提出的算法，该算法根据数据点之间的距离对其进行聚类。

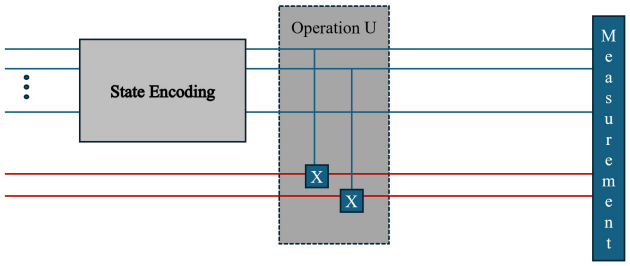


图1。QHCA 的电路表示。蓝色寄存器表示对距离进行编码的量子寄存器。红色寄存器是辅助寄存器，量子比特数为 $= log_2$（簇数）。

- 我们首先对数据集 C 中的各项运行经典排序算法， k_1 并选择彼此距离最远的两个数据点 k_2 ，也就是说， k_1 与 k_2之间的距离 D_{max} 是任意两对数据点之间的最大距离。在经典域中，这一过程通常需要进行 $O(N^2)$ 次比较，不过，利用[29]中提出的子程序，我们可以将这一成本大幅降至 $O(N)$ 的量级。

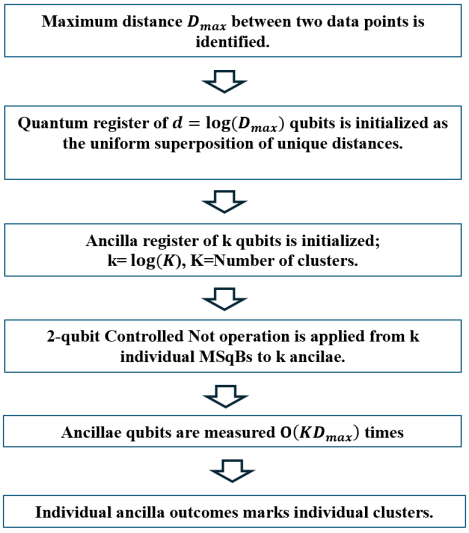


图2。QHCA 准备 K 个簇的流程图。

• 我们选择 k_1 作为原点，并根据各项最接近整数的二进制等价值，重新定义 N 各项与 k_1 之间的距离。这提供了一种将距离表示为 d–量子比特量子基向量态的方法，其中 $d = \lceil \log_2 D_{max} \rceil$。例如，如果点 k_1 与任意其他点 k_i之间的距离（其中 $1 \leq i \leq N$）由 $C_{k_1 i}$给出，我们将其表示为 $|D_{k_{1i}}\rangle$，其中 $D_{k_{1i}}$ 是十进制数 $\lfloor C_{k_1 i} \rceil$的二进制等价值，即最接近 $C_{k_1 i}$的整数。已知的最佳时间复杂度为 $O(\log\ N)$，空间复杂度为 $O(\log\ N)$。

• 现在，我们取一个 d 量子比特量子寄存器，其中 $d = O(\lceil \log_2 D_{max} \rceil)$，并将距离表示为该寄存器中所有表示各个距离的量子态 $|\psi\rangle$ 的均匀叠加态 $|D_{k_{1i}}\rangle$。我们强调，状态 $|\psi\rangle$ 中每个距离值只出现一次，即距离相近的数据点仅由一个基态表示。数学上，$|\psi\rangle$ 可写为：

$$|\psi\rangle = \frac{1}{\sqrt{N'}} \sum_{i=1}^{N'} |D_{k_{1i}}\rangle\,, \tag{3}$$

其中，N' 是寄存器中基向量的数量，且 $N' \leq N$。此外，量子比特数 d 的数量级为 $log_2 D_{max}$。如有必要，我们还允许对数据点之间的距离进行标量乘法，以在较低的 d 值与聚类准确率之间保持平衡。

制备量子态 $|\psi\rangle$ 有几种方法。制备该量子态最通用的方法是实现一个 Grover 预言机，其操作复杂度为

$O(\sqrt{\frac{\lceil D_{max} \rceil}{N'}})$ [7]。根据基态 N'的数量和性质，可以用显著更少的操作来制备该量子态 [31, 32]。

表示距离叠加态的量子态经过专门设计：当其寄存器中的最高有效量子比特（MSqB），即量子比特 d，处于状态 $|1\rangle$ 时，它表示距离原点 k_1 最远的数据点的距离。这些距离近似为 $\geq \frac{D_{max}}{2}$。相反，当量子比特 n 处于状态 $|0\rangle$ 时，它表示距 k_1 的距离，近似为 $\leq \frac{D_{max}}{2}$。同样，很容易看出，当量子比特 $n-j$ 处于状态 $|0\rangle$ 和 $|1\rangle$时，它分别近似表示距离 $\leq \frac{D_{max}}{2^j}$和 $\geq \frac{D_{max}}{2^j}$。

- 我们需要准备这些簇，使其与各自中心的距离不超过预先设定的值D_{min}；或者，我们也可以创建预先指定数量的簇，例如 K。在第一种情况下，我们使用 $m = \lceil \log_2 \frac{D_{max}}{D_{min}} \rceil$ 个辅助量子比特；在第二种情况下，使用 $m = \lceil \log_2 K \rceil$ 个辅助量子比特，并对寄存器 $n+m$ 应用操作U，如下所示：

$$U = \Pi_{i_n=0, i_m=0}^{m,m} C^{n-i_n} X_{i_m},$$

其中，i_n是距离寄存器中的第 $n-i_n$个最高有效量子比特，而 i_m是辅助寄存器中的第 i_m个最高有效量子比特。简单来说，该操作等价于对最高有效量子比特和辅助量子比特施加两个量子比特的受控非门。此部分的操作复杂度关于 K呈对数关系。

- 如图1所示，同时测量两个寄存器后，可以轻松选择辅助寄存器的状态，并找出与之关联的簇。该过程会重复进行，直到获得所需数量的簇。为了通过测量刻画所有可能的簇，需要将整个过程重复 $O(\lceil \frac{D_{max}^2}{D_{min}} \rceil) O(KD_{max}))$次，因为某些簇中的数据点数量可能会少几个数量级。这意味着，所需测量次数关于量子比特数量呈指数增长，即 $O(2^{d+m})$，而关于最大距离和簇数量则呈线性增长。因此，QHCA 的总体时间复杂度为 $O(\lceil \frac{D_{max}^2}{D_{min}} \rceil N \log N) O(KD_{max} N \log N))$，而运算复杂度为 $O(k\sqrt{\frac{\lceil D_{max} \rceil}{N'}})$，结合了状态制备和聚类部分。

3.1.1 确定边界

每个簇的边界由表示距离所需的量子比特数以及进行聚类所需的量子比特数决定。如果我们考虑形成 k 个簇，且距离态向量的表示需要 d 个量子比特，那么每个簇 t 的上限和下限便可明确确定。假设 t^{th} 簇由其二进制等价值 $\{t_{k-1}...t_1 t_0\}$表示，而编码中心与该簇之间距离的比特串 d^t表示为 $\{d_{n-1}...d_1 d_0\}$，其中 $t_i,\ d_k \in \{0,1\}$ 且 $k-1 \leq i \leq 0,\ \ n-1 \leq k \leq 0$。

簇 t 的上限为

$$d_{max}^{t} = \sum_{j=0}^{k-1} 2^{j+k} \cdot t_j + \sum_{j=0}^{k-1} 2^{j} \tag{4}$$

下限由下式给出

$$d_{min}^{t} = \sum_{j=0}^{k-1} 2^{j+k} \cdot t_j. \tag{5}$$

该算法的示意图如图 2所示。

3.2 基于非锐测量的聚类算法（UMCA）

给定一组数据，我们找出任意两个点之间的最大距离，并从这两个点中任选一个，记为 k_1。随后，我们将每个不同的距离k_1 表示为一个量子态，该量子态由相应最近整数的二进制等价值编码。然后，我们将所有这些状态制备成等权叠加态。如第 2 节所述，可以为每个状态构造一组效应算子。每个效应算子的构造方式如下：若使用其中一个效应算子对制备好的叠加量子态进行非锐测量，例如 E_i，其中 $|i\rangle$ 是效应原点处的量子态（参见公式 2），则测量后，距离 $|i\rangle$ 较近的状态会自动聚为一簇。通过调节高斯分布的宽度，可以控制测量结果 [33],中状态 $|i\rangle$附近其他状态的概率，使非锐测量以较高概率“选中”那些更接近所选原点的状态。要将一个数据集划分为 K 个簇，需要选取 K 个原点，并对该数据集进行 K 次宽度固定的非锐测量。如果某个数据点重复出现在多个簇中，可以将其分配给其中任意一个簇。

在计算基中，表示所有距离所需的量子比特数为 $log_2(D_{max})$的量级，其中 D_{max} 是数据点之间的最大距离。我们将各个距离表示为计算基态，并对所有距离取等权叠加态，如式3所示，

$$|\psi\rangle = \frac{1}{\sqrt{N'}} \sum_{i=1}^{N'} |D_{k_{1i}}\rangle \,,$$

其中， N'是寄存器中存在的不同距离 的数量，且 $N' \leq N$。准备好该量子态后，可以通过效应算子，以最大距离为中心、预定义宽度，执行预定义次数的非锐测量。

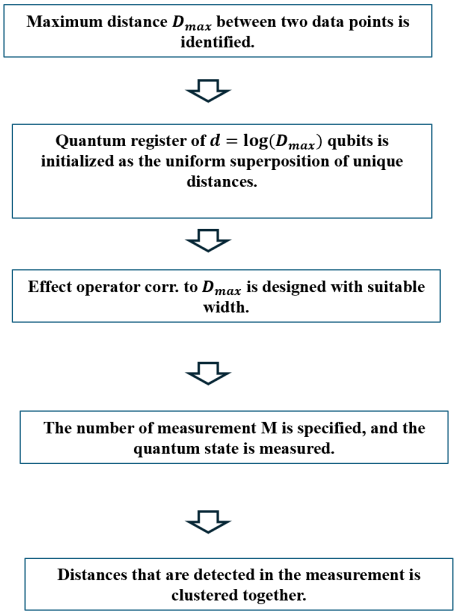


图3。基于非锐测量的聚类算法流程图。

对于这一特定情况，效应算子集合 $\{E_i\}$ 定义如下：

$$E_i = \sum_{j\in\{0\ldots00\},j\neq i}^{\{1\ldots11\}} \frac{1}{\sqrt{(2\pi\Delta^2)}} e^{-\frac{(i-j)^2}{2\Delta^2}} |j\rangle\langle j| , \tag{6}$$

其中， E_i 表示量子态 $|\psi\rangle$中对应第 i–个距离的效应算子， j 是第 j–个距离。该算子模拟高斯分布，其中 Δ 为高斯分布的标准差， i 为其中心。因此，每个效应算子都可以表示为一个对角矩阵，其元素由$\frac{1}{\sqrt{(2\pi\Delta^2)}}e^{-\frac{(i-j)^2}{2\Delta^2}}$ 给出。它也可以看作是正算子值测度 $|j\rangle\langle j|$的加权平均，权重为$\frac{1}{\sqrt{(2\pi\Delta^2)}}e^{-\frac{(i-j)^2}{2\Delta^2}}$ 。我们将效应算子保持在代表最大距离的量子态周围居中，这样便可以对距离原点最远的距离进行聚类。

下面给出基于非锐测量的聚类算法的工作流程。

- 与前一个算法类似，我们运行经典排序算法来找出相距最远的点，并任选其中一个作为原点。从该点出发，使用欧几里得

距离计算其他所有点的距离。随后，在一个包含 $d = log(D_{max})$ 个量子比特的量子寄存器中，创建所有不同距离对应量子态的均匀叠加态。

- 公式 6 所描述的效应算子对应于围绕某个特定距离的非锐测量，即效应算子的中心（i）可以选为其中一个距离。因此，可以有 2^d 个效应算子，每个都以一个距离为中心。然而，正如本文所示，有效聚类只需要一个效应算子。我们选择以最大距离为中心的效应算子，并将高斯分布的宽度设为 $\Delta = \frac{D_{max}}{k}$，其中 k 是所需的簇数。

- 接下来，我们围绕指定中心进行非锐测量，测量次数也预先指定。按照设计，距离原点较近的量子态会以更高的概率出现在测量结果中。测量次数取决于高斯分布的宽度以及簇的数量 k。

复杂度：

量子寄存器的制备复杂度与前一种算法相似，即经典预处理需要 $O(N\log N)$，而状态制备需要 $O\sqrt{\frac{D_{max}}{N'}}$。我们在第 4 节中证明，即使仅进行多项式次数（关于量子比特数 d）的测量，该方法也能达到较高的准确率。要在量子硬件上执行这种测量，除了按指定方式设置测量装置外，还可以在计算基态中对所有不同距离的叠加态进行投影测量，然后为它们赋予适当的权重。聚类依据（非锐）测量后量子态的相应振幅进行。

4 算法实现

在本节中，我们将演示两种算法在三个标准数据集上的实现。首先，我们在同心圆数据集以及城市的 Churriz 数据集上实现量子聚类算法。我们使用状态矢量模拟器和 QASM 模拟器，在同心圆数据集上实现 QHCA。

4.1 在同心圆数据集上实现 QHCA

我们使用 *sklearn* 软件包生成了一个包含 400 个样本的数据集，这些样本分布在两个以 (0,0) 为圆心的同心圆上，如图 6所示，并在该数据集上实现了我们的算法。数据位于二维欧几里得平面中。我们首先计算从原点到最远数据点的最大距离 D_{max}，并将其乘以 10 进行缩放。要将所有不同距离编码为均匀叠加态，需要 $n = log_2(D_{max})$ 个量子比特。我们还发现，尽管数据集包含 400 个点，但经过预处理后，只有 12 个不同的距离需要编码。将这些距离编码到量子态后，我们将辅助寄存器中的量子比特数设为 1，因为此处需要得到 2 个簇。针对该数据集的 QHCA 量子电路见图 4

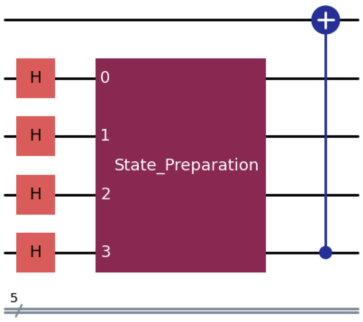


图 4。用于对同心圆数据集进行聚类的电路。

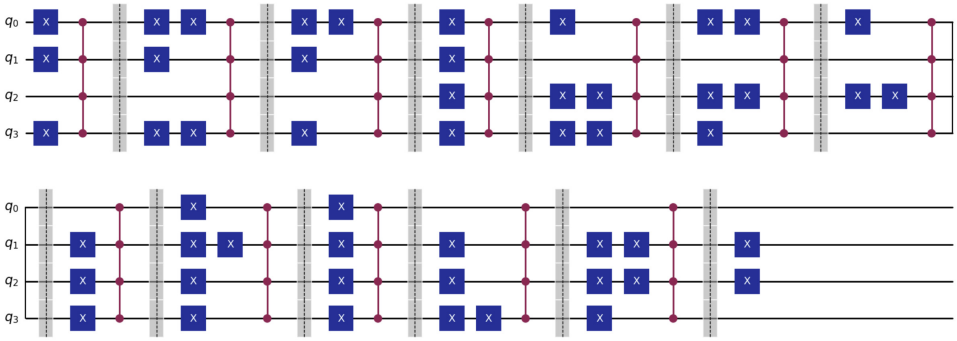


图 5。图 4中的“状态制备”模块。

和图 5。这里我们使用格罗弗预言机制备所有距离的均匀叠加态。由于初始量子态中只有 12 个距离，格罗弗预言机的最优操作为$O(\sqrt{\frac{12}{16}})$ $i.e.,$ ≈ 1。聚类时，我们只实现了一次该预言机。图 4展示了通过 QHCA 对图 6 所示的同心圆数据集进行聚类的电路。图5 展示了聚类前用于制备距离均匀叠加态的格罗弗预言机。将寄存器制备为不同距离的均匀叠加态后，我们从最高有效量子比特向辅助量子比特施加受控非门，并测量整个寄存器，最终得到各个簇。整个过程至少需要重复 2^{4+1}次，即 32 次。为了展示量子聚类算法相较于经典算法的优势，我们还使用经典层次聚类算法对同一数据集进行了聚类。结果如图 6所示。经典算法未能正确地对数据进行聚类，而量子算法则实现了准确聚类。此外，我们给出了电路分别进行 10000 次和 32 次测量时的结果。从结果可以看出，在这两种情况下，量子算法的表现都优于经典算法；不过，将电路测量 10000 次所得的结果略优于测量 32 次的结果。

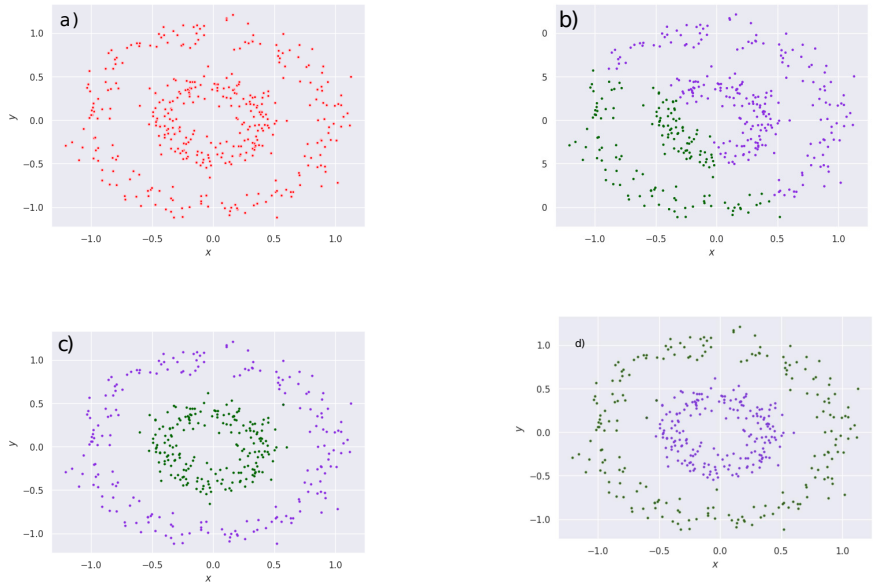


图6。（a）使用 sklearn 生成的 400 个点的数据集，噪声比为 0.1，单位为 $D_{max} = 12.48$ 。b）使用传统经典分裂聚类算法对该数据集进行分类。c）使用 10000 次测量，通过 QHCA算法对该数据集进行分类。d）使用 QHCA 并进行 32 次测量，对该数据集进行分类。不同颜色代表不同的簇。

采用穷举搜索的分裂式聚类的时间复杂度为 $O(2^N)$，而凝聚聚类算法的时间复杂度为 $O(N^3)$，因为每次迭代都必须扫描距离矩阵以找出最小距离。所提出的量子聚类算法的时间复杂度为 $O(\lceil \frac{D_{max}^2}{D_{min}} \rceil N \log N)$ 量级，取值适当的 D_{max} 后，相较于其经典对应算法具有优势。

4.2 在 Churriz数据集上实现 QHCA

我们使用包含130个城市的Churritz数据集，根据距离运行算法，将城市划分为4个簇。数据可在 [34]中找到。首先，计算任意两个城市之间的最大距离 D_{max} ，结果为938。842千米。此外，我们取相距最远的两个城市之一作为原点，并据此对所有其他城市进行聚类。该数据集在欧几里得 $x - y$ 平面中的分布如图7（左）所示。邻近城市的聚类结果如图7 （右）所示。对该数据集进行聚类需要10个量子比特（用于编码距离）+2（用于聚类）。

我们进一步缩小距离，并检查 QHCA 在该数据集上的性能。我们通过将距离除以不断增大的 2 的幂来缩小距离。

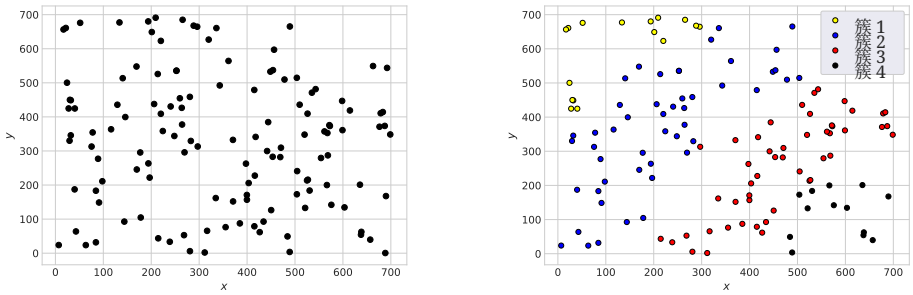


图7。Churritz 数据集包含 130 座城市， $D_{max} = 938.842$ 公里。每个点表示欧几里得平面中城市的位置（左），以及 QHCA 实施后相邻城市形成的簇（右）。

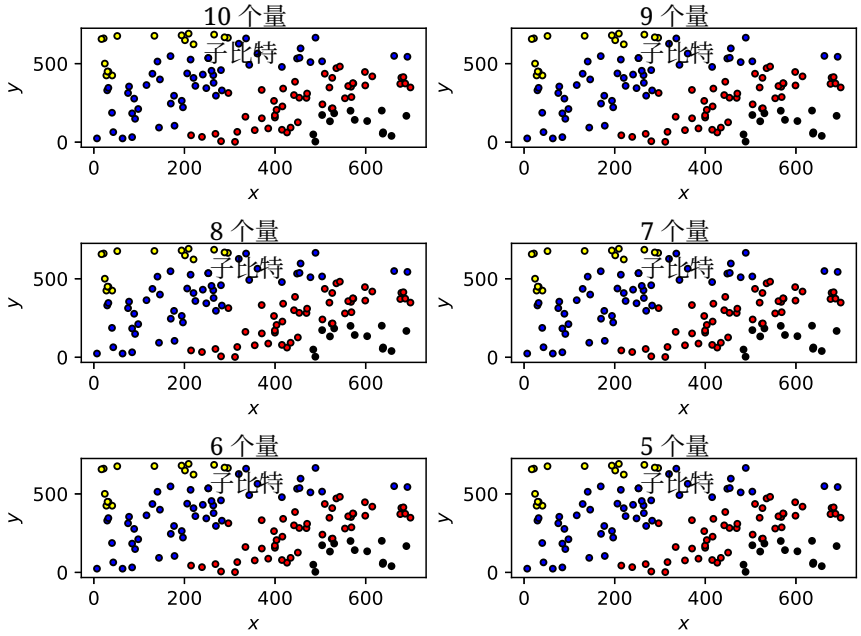


图8。缩小城市间距离后，Churritz数据集的聚类结果。聚类结果保持不变，但所需的量子资源减少了。

递归地使用缩小后的距离对数据集进行聚类，并采用 QHCA。从图8可以看出，这并不会影响数据的聚类结果。使用 10 个量子比特、考虑最大距离 938。842 km 的聚类结果，与使用 5 个量子比特、考虑 29。33 km 的聚类结果相似。

4.3 在威斯康星州癌症数据集上实现基于非锐测量的算法

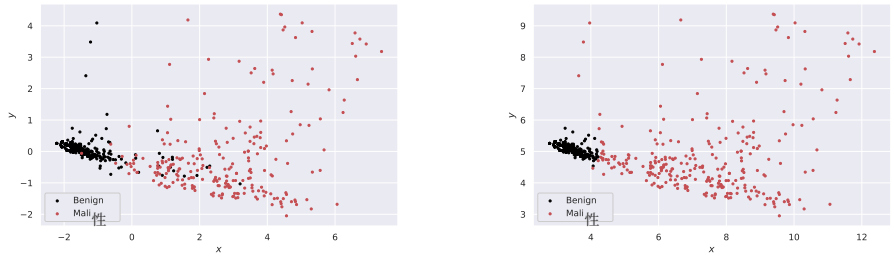


图9。特征降维后威斯康星乳腺癌数据集的原始分类（左），以及使用基于非锐测量的聚类算法进行分类的结果（右），采用4个量子比特和 $O(4^3)$次测量。准确率为93.99%。

威斯康星乳腺癌数据集是一个带标签数据集，包含699条记录和9种不同属性，例如肿块厚度、细胞大小、细胞形状等。为了将数据集聚为恶性或良性类别，并便于可视化，我们进行主成分分析（PCA），将特征降至2个维度。图9 左侧的图片展示了数据的原始分类。此外，我们使用基于非锐测量的聚类算法对数据集进行分类。点之间的最大距离为10，只需4个量子比特即可对数据集进行分类。我们构造不同距离的均匀叠加态。对于这个特定案例，该量子态为：

$$|\psi_{wbc}\rangle = \sum_{i=|0000\rangle}^{|1010\rangle} |i\rangle, \tag{7}$$

这意味着，对于这一特定情况，可以使用 $O(log_2 10)$门复杂度和电路深度 [31]来设计该量子态。

我们选择基态 1010 作为高斯效应算子的中心，因为它代表最大距离。由于我们将进行二分类，因此将高斯的宽度设为 $\frac{10}{2} = 5$。最后，我们选择在量子态 $|\psi_{wbc}\rangle$ 上应用的测量次数 M，以实现有效聚类。根据该协议，在测量中检测到的状态会被聚为一簇。对于此协议，我们将 M 严格限制为关于 $d = 4$的多项式。我们将 M 从 d^1 变化到 d^{10}，发现当 $M = d^3$时，该协议能够将数据集分为良性类和恶性类，准确率达到 93。99%。使用这些参数对数据集进行分类的结果如图 9右侧图所示。

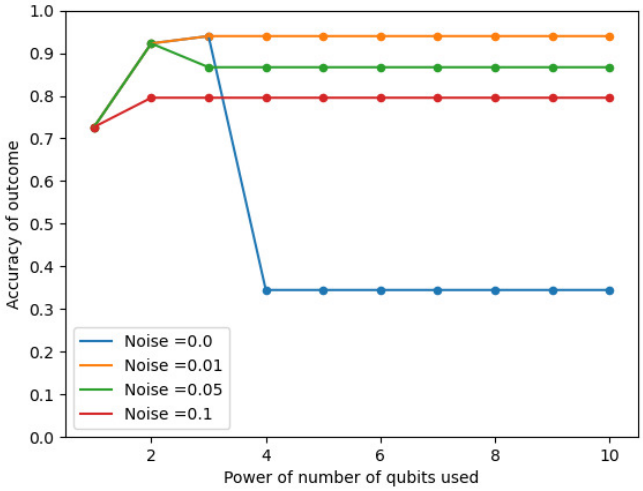


图 10。量子非锐测量聚类协议在威斯康星乳腺癌数据集上采用不同测量设置时的准确率。X 轴表示分类极限中量子比特数量的多项式次幂，Y 轴表示协议的准确率。蓝色曲线表示无噪声测量。曲线的不同颜色表示不同的测量设置，具体取决于测量中假设的噪声水平。

我们进一步在图 10中展示了该协议在不同测量和聚类设置下的准确率变化。图 10 中的蓝色曲线表示一种理想情形：测量中检测到的所有状态都被聚为一簇。如图所示，随着测量次数增加，该协议的准确率起初会提高；当量子态 $|\psi_{wbc}\rangle$ 被测量 4^3 次时，准确率达到最大值。然而，测量次数进一步增加后，准确率便会下降，并最终稳定在 34%。这是设计所预期的结果，因为效应算子的设计目标是以某种概率（尽管很小）测量所有状态。我们提出了一种应对这一情况的方法：不再将测量结果中出现的所有状态都纳入聚类，而只选择出现概率高于某个阈值的状态。对于这个问题，我们将该阈值设为 $m_\epsilon = \epsilon \times d^k$，其中 ϵ 是一个介于 0 和 1 之间的小数。我们在图 ϵ 10中给出了 ϵ 的三个不同取值对应的结果，并将其标记为“噪声”，因为量m_ϵ 与量子硬件测量设置中噪声所导致的测量误差类似。如图 10所示，设置聚类概率阈值的算法在 $\epsilon = 0.01$时效果最佳；这相当于以量子比特数的任意多项式次幂测量该状态，而且通常优于“理想”情形。我们报告，当 $\epsilon \in [0.01, 0.1]$时，准确率保持

如图所示，位于红色和黄色曲线之间10，对于 $\epsilon \geq 0.1$，准确率会下降，因为在这种情况下，聚类限值 m_ϵ 增加过多，聚在一起的基态数量也更少。

即使考虑 $m_\epsilon = 0.0$，也很容易看出，该算法所需的量子比特数量非常少，甚至少于 log(N)（其中 N 为数据集中的数据点数量）；其门操作复杂度为多项式级，且测量次数关于量子比特数量也是多项式级。我们还可以看到，在假设存在读出误差的情况下，该算法的表现优于其理想情况下的对应算法。因此，该算法非常适用于含噪中等规模量子（NISQ）计算机。

5 结论

在本研究中，我们提出了两种适用于 NISQ 的无监督数据聚类算法。QHCA 的时间复杂度为 $O(KD_{max}N\log N)$，优于经典层次聚类算法（划分式算法为 $O(2^N)$，凝聚式算法为 $O(N^3)$）。这两种算法的量子部分均与数据点数量无关。我们证明，即使测量次数很少，QHCA 在线性不可分数据集上的表现也优于经典划分式聚类。我们还进一步证明，对于 Churritz 城市数据集，该算法仅用 7 个量子比特即可完成聚类。第二种算法是制备与测量算法，利用非锐测量将相似的数据点聚在一起。我们展示了该算法能够以较高准确率成功分类威斯康星州癌症数据集，并进一步证明，完成这一任务仅需多项式次数的测量。我们还证明，在假设存在测量噪声的情况下，该算法的表现优于理想情形。本研究中，我们将效应算子的宽度设为常数，并通过穷举搜索找到最优测量限值。未来，我们计划对这些参数进行优化，以获得更好的结果。此外，在真实量子设备上实现这些算法超出了本文的研究范围；不过，我们希望在未来考察真实 NISQ 设备中的噪声与非锐测量之间的相关性，并研究其对聚类准确率的影响。

参考文献

[1] Arute, F., Arya, K., Babbush, R.,Bacon, D., Bardin, J.C., Barends, R., Biswas,R.,Boixo, S., Brandao,F.G.,Buell, D.A., 等：使用可编程超导处理器实现量子霸权。《自然》 **574**(7779), 505–510 (2019)https://doi.org/10.1038/s41586-019-1666-5[2] Feynman, R.P., 等：用计算机模拟物理。理论物理国际期刊**21**(6/7) (2018)[3] Deutsch, D., Jozsa, R.：通过量子计算快速解决问题。数学、物理与工程科学学报 **439**(1907), 553–558 (1992) https://doi.org/10.1098/rspa.1992.0167

[4] Collins, D., Kim, K., Holton, W.: Deutsch-Jozsa算法作为量子计算测试。物理评论 A 58(3), 1633 (1998) https://doi.org/10.1103/ PhysRevA.58.R1633[5] Bernstein, E., Vazirani, U.: 量子复杂性理论。载于：计算理论年度 ACM 研讨会论文集，STOC '93，第 11–20 页。美国计算机协会，纽约，美国 (1993)。https://doi.org/10.1145/167088.167097 。https://doi.org/10.1145/167088.167097[6] Shor, P.W.: 量子计算机上的素因数分解与离散对数的多项式时间算法。SIAM 计算杂志 26(5), 1484–1509 (1997) https://doi.org/10.1137/S0097539795293172[7] Grover, L.K.: 用于数据库搜索的快速量子力学算法。载于：计算理论年度 ACM 研讨会论文集，STOC '96，第 212–219 页。美国计算机协会，纽约，美国 (1996)。https://doi.org/10.1145/237814.237866 。https://doi.org/10.1145/237814.237866[8]Harrow, A.W., Hassidim, A., Lloyd, S.: 线性方程组的量子算法。物理评论快报 103, 150502 (2009) https://doi.org/10.1103/ PhysRevLett.103.150502[9]Horn, D., Gottlieb, A.: 基于量子力学、用于模式识别问题中数据聚类的算法。物理评论快报 88, 018702 (2001) https://doi.org/10.1103/PhysRevLett.88.018702[10] Diday, E., Simon, J.C.: 载于：Fu, K.S. (编) 聚类分析，第 47–94 页。Springer，柏林，海德堡 (1976)。https://doi.org/10.1007/978-3-642-96303-2 3 。https: // doi.org/10.1007/978-3-642-96303-2 3[11] Mandal, A., Banerjee, S., Panigrahi, P.K.: 簇上的量子图像表示。载于：2021 年 IEEE 量子计算与工程国际会议 (QCE)，第 89–99 页 (2021)。https://doi.org/10.1109/QCE52317.2021. 00025[12] Ahuja, R., Chug, A., Gupta, S., Ahuja, P., Kohli, S.: 载于：Yang, X.-S., He, X.- S. (编) 机器学习的分类与聚类算法及其应用，第 225–248 页。Springer，Cham (2020)。https://doi.org/10.1007/ 978-3-030-28553-1 11 。https://doi.org/10.1007/978-3-030-28553-1 11[13] Everitt, B.S., Landau, S., Leese, M., Stahl, D.: 聚类分析：Wiley 概率与统计丛书。英国西萨塞克斯奇切斯特 Southern Gate：约翰 · 威立父子公司 (2011)[14] Kaufman, L., Rousseeuw, P.J.: 在数据中寻找群组：聚类分析导论。约翰·威立父子公司，美国 (2009)。https://doi.org/10.1002/ 9780470316801

[15] Hartigan, J.A., Wong, M.A.：算法 AS 136：一种 K 均值聚类算法。《应用统计》28(1)，100–108（1979）https://doi.org/10.2307/2346830[16]Ester, M., Kriegel, H.-P., Sander, J., Xu, X.：一种用于在含噪声的大型空间数据库中发现簇的基于密度的算法。载于：第二届国际知识发现与数据挖掘会议论文集。KDD'96，第226–231页。AAAI出版社，???（1996）。https://www.bibsonomy.org/bibtex/289e04610c1b5f2fa147398826b502a2d/nosebrain[17] D¨urr, C., Heiligman, M., HOyer, P., Mhalla, M.：若干图问题的量子查询复杂度。SIAM 计算杂志 35(6)，1310–1328（2006）[18] Li, Q., He, Y., Jiang, J.-p.：一种基于量子博弈的新型聚类算法。《物理学杂志 A：数学与理论》42(44)，445303（2009）[19] Li, Q., He, Y., Jiang, J.-p.：一种基于量子游走的混合经典-量子聚类算法。《量子信息处理》10，13–26（2011）[20] Yu, Y., Qian, F., Liu, H.：基于量子聚类的加权线性规划支持向量回归，用于多变量非线性问题。《软计算》14(9)，921–929（2010）[21] A¨ımeur, E., Brassard, G., Gambs, S.：量子聚类算法。载于：第24届国际机器学习会议论文集。ICML'07，第1–8页。美国计算机协会，纽约，NY，美国（2007）。https: //doi.org/10.1145/1273496.1273497 .
https://doi.org/10.1145/1273496.1273497[22] Casa˜na-Eslava, R.V., Lisboa, P.J.G., Ortega-Martorell, S., Jarman, I.H., Mart´ın-Guerrero, J.D.：概率量子聚类。《知识型系统》194，105567（2020）
https://doi.org/10.1016/j.knosys.2020.105567[23] Bermejo, P., Or´us, R.：变分量子聚类与量子启发式聚类。《科学报告》13(1)，13284（2023）
https://doi.org/10.1038/s41598-023-39771-6[24] Khan, S.U., Awan, A.J., Vall-Llosera, G.：在含噪声中等规模量子计算机上进行 K 均值聚类。CoRR abs/1909.12183（2019）1909.12183[25] Kavitha, S., Kaulgud, N.：用于利用量子电路方法检测心脏病的量子 K 均值聚类方法。《软计算》27(18)，13255–13268（2023）https://doi.org/10.1007/s00500-022-07200-x[26] Li, Q., Huang, Y., Jin, S., Hou, X., Wang, X.：用于无监督学习的量子谱聚类算法。《中国科学：信息科学》65(10)，200504（2022）https://doi.org/10.1007/s11432-022-3492-x[27] Gopalakrishnan, D., Dellantonio, L., Pilato, A.D., Redjeb, W., Pantaleo, F., Mosca, M.：qLUE：一种用于多维数据集的量子聚类算法

数据集（2024）。https://arxiv.org/abs/2407.00357

[28] Day, W.H., Edelsbrunner, H.：凝聚层次聚类方法的高效算法。J. Classif. **1**(1), 7–24（1984）[29]A¨ımeur, E., Brassard, G., Gambs, S.：无监督学习的量子加速。Mach. Learn.**90**(2)，261–287（2013）
https://doi.org/10.1007/s10994-012-5316-5[30] Foulis, D.J., Bennett, M.K.：效应代数与非锐量子逻辑。Found. Phys. **24**(10)，1331–1352（1994）[31] Shukla, A., Vedula, P.：一种用于制备均匀量子叠加态的高效量子算法。Quantum Information Processing **23**(2)，38（2024）
https://doi.org/10.1007/s11128-024-04258-4[32] Mozafari, F., Riener,H., Soeken, M., De Micheli, G.：制备均匀量子态的高效布尔方法。IEEE Transactions on Quantum Engineering**2**，1–12（2021）
https://doi.org/10.1109/TQE.2021.3101663[33] Barui, A., Pal, M., Panigrahi, P.K.：利用非锐测量处理阈值量子图像的新方法。Quantum Information Processing **23**(3)，76（2024）https://doi.org/10.1007/s11128-024-04282-4[34]Reinelt, G.： {TSPLIB}：来自各种来源、具有各种类型的TSP（及相关问题）样例实例库。
http://comopt.ifi.uni-heidelberg.de/software/TSPLIB95/ （2014）