arXiv:2305.04287v1 [cs.ET] 7 May 2023

基于欧几里得距离估计的量子 k 近邻算法

Enrico Zardini[1*]、Enrico Blanzieri[1,2] 和 Davide Pastorello[2,3]

[1]特伦托大学信息工程与计算机科学系，意大利特伦托省波沃市 Sommarive 路 9 号，邮编 38123。[2]特伦托基础物理与应用研究所，意大利特伦托省波沃市 Sommarive 路 14 号，邮编 38123。

[3]博洛尼亚大学数学系，意大利博洛尼亚市圣多纳托门广场 5 号，邮编 40126。

*通讯作者。电子邮箱：enrico.zardini@unitn.it；贡献作者：enrico.blanzieri@unitn.it；davide.pastorello3@unibo.it；

摘要

k近邻（***k***-NN）是一种基础的机器学习（ML）算法。过去几年中，人们提出了若干采用不同距离度量的量子版本。尽管欧几里得距离是机器学习中最广泛使用的距离度量之一，但在开发这些量子变体时，它并未得到太多关注。本文介绍了一种基于欧几里得距离的新型量子 ***k***-NN算法。具体而言，该算法采用所需量子比特数量较少的量子编码，以及不涉及预言机的简单量子电路，这些特点有利于实现该算法。除了数学公式和一些复杂度方面的观察外，本文还通过模拟进行了详细的实证评估。具体来说，结果表明，公式推导是正确的；测量次数受限时，算法性能会下降；在理想情况下，该算法与一些经典基线方法相比具有竞争力；并且增加测量次数可以提升其性能。

关键词：量子计算、量子机器学习、k 近邻、欧几里得距离

1 引言

量子机器学习（QML）是量子计算领域最新且最受欢迎的研究方向之一。尤其是，将量子计算应用于机器学习（ML）任务，能够提供一些有趣的解决方案；至少在理论层面上，这些方案具有相对于经典对应算法的量子优势。此外，量子机器学习似乎也是利用现有量子计算机原型来解决现实问题的有效途径。从这个意义上说，一种通用的“实用”方法是将量子算法作为更复杂学习方案的子程序来执行，在这些方案中，量子计算机作为协处理器融入混合架构。这种方法是另一条颇具吸引力的路径，有别于在量子硬件理想且通用这一（强）假设下，开发能够完整完成机器学习任务的量子算法。

过去十年中，已有多种有趣的量子机器学习算法从理论角度得到提出和表征；有时也经过了实证测试。值得关注的例子包括Rebentrost 等人提出的量子支持向量机 (2014)、Schuld 等人定义的基于距离的分类器 (2017)，以及 Abbas 等人 (2021讨论过其性能的量子神经网络。特别是，已有多种量子版本的 k-最近邻（k-NN）算法被提出（见第2节）。在机器学习中，k-NN 是一种非常简单且广泛使用的分类算法，它根据 k 个最近训练实例的标签，为未分类的数据实例分配一个标签。为此，必须选择数据表示空间中的一种合适参考距离。在经典领域，典型选择是汉明距离和欧几里得距离；而在量子领域（仅考虑量子 k-NN），欧几里得距离尚未得到充分关注。在本研究中，我们提出一种量子 k-最近邻算法，其欧几里得距离的计算基于一种新颖的量子编码，所需量子比特较少，且量子电路简单，因此实现尤其有利。与其他涉及欧几里得距离量子计算的算法（例如，通过 SWAP 测试）一样，假设可使用量子随机存取存储器（QRAM，Giovannetti 等人，2008）检索数据，即可获得相对于经典计算的指数级加速。否则，在时间复杂度方面并不存在真正的量子优势。从实践角度来看，本文通过分类准确率以及所找到最近邻的正确性（使用杰卡德指数评估）分析了所提出量子 k-NN 的性能。该算法使用 Qiskit 实现，并以三种不同的执行模式运行： 经典、状态向量和模拟。然而，所考虑实验所需的量子比特数量使得在真实量子计算机上进行实证评估无法实现。

本文的结构如下：第2 节介绍一些背景信息；第3 节介绍新的量子 k-最近邻算法，其基于欧几里得距离度量；第4 节讨论算法的实现；第5 节介绍实验评估及所得结果；第6 节总结全文。

2 背景

本节介绍量子机器学习的背景知识、文献中现有的量子 k-最近邻算法，以及欧几里得距离（平方）在量子机器学习领域中的应用。

2.1 量子机器学习

一般而言，机器学习是从所收集数据中提取信息的方法的自动化。如果数据分析技术是在传统数字计算机上实现的，我们称之为经典机器学习；如果采用量子计算机，则称之为量子机器学习。为开发新的量子机器学习方案所付出努力的一个普遍理由由Biamonte 等人(2017) 提出：由于（即使是小型的）量子系统也难以用经典计算机进行模拟，我们可以推测，（即使是小型的）量子处理器能够发现数据中的某些结构，而这些结构很难通过经典方式发现。因此，对于当今以及不久的将来可用的小规模量子机器而言，量子机器学习可能是实现非平凡应用的正确路径。另一方面，在普适性、大规模和容错等强假设下，可以构建若干优于其经典对应方法的量子机器学习算法。这对于理解量子计算的基础，以及展示量子计算机的实际潜力，都非常重要。然而，为了推动量子技术在近期的发展，在寻求新的量子机器学习方案时，考虑当前量子硬件的局限性是有益的。

从数学角度来看，开发可由量子机器执行的机器学习算法还有另一项相关动机，这源于量子力学与机器学习之间的形式类比：这两个领域都依赖于高维向量空间中的矩阵运算。在实践中，用于描述物理量子系统的希尔伯特空间可以用作数据表示的特征空间。在这一框架下，线性代数运算通过量子态的时间演化在物理上实现；例如，在量子计算的电路模型中，演化被描述为量子门（即酉算子）的作用。此外，将数据表示为量子态在空间资源方面也有优势，因为多量子比特系统的希尔伯特空间维度随量子比特数量呈指数增长。因此，少量量子比特朝目标态方向的受控演化，可能对应于对所考虑的特征空间执行复杂的线性代数运算。

4 基于欧几里得距离估计的量子 k 近邻

量子机器学习中的一个关键概念是量子编码，即将经典数据（例如符号列表）编码为量子态的任何一种过程。尤其是，在当前量子机器学习的发展阶段，将大量数据高效加载到量子架构中仍是一个重大瓶颈；事实上，运行若干知名量子机器学习算法所需的状态制备，只有在强假设——即量子随机存取存储器可用——成立时才能高效完成。具体而言，给定一个 n量子比特寄存器，令 $\{|i\rangle\}_{i=0、\,2^n-1}$ 为对应希尔伯特空间的一组固定正交归一基，我们称之为计算基。最简单的量子编码是基态编码，它将长度为 n 的比特串编码为构成计算基的状态。因此，使用 n 个量子比特即可编码 n 位经典信息，并由此带来一些有趣的量子优势，例如创建数据的叠加态，以及通过量子纠缠实现非经典关联。相比之下，从空间资源的角度来看，更高效的量子编码是振幅编码：它将由归一化复向量 $\mathbf{x} \in \mathbb{C}^{2^n}$ 表示的数据实例，编码到量子态相对于计算基的坐标（或振幅）中，即

$$|\psi\rangle = \sum_{i=0}^{2^n-1} x_i\,|i\rangle \qquad (n\text{-qubit state}).$$

振幅编码利用了量子存储器的指数级存储容量，但无法直接检索已存储的数据。事实上，振幅无法被观测，只能估计概率 $|x_i|^2$ 。本文采用的编码过程基于振幅编码。

最后，我们通过指出量子机器学习或许是发掘现有小型量子计算机有效应用的最有前景的途径，来结束本节导论。具体而言，也可以不再要求机器学习任务必须完全由量子计算完成，而采用混合方法：在这类方法中，量子协处理器能够高效解决更复杂学习方案中的特定子问题。此外，量子加速并非唯一值得追求的量子优势。在评估量子机器的学习性能时，预测准确率、表达能力、泛化能力以及避免训练平台的能力，也都是值得关注的评价指标。

2.2 量子 k 近邻

The k-最近邻（Fix 和 Hodges，1951年）是一种分类算法，包含三个步骤：计算与训练样本的距离；识别最近邻，即距离测试实例最近的样本；通过多数投票预测类别标签。人们已经提出了多种采用不同距离度量的量子变体，但它们的一个共同之处是利用

叠加态，以执行并行运算，例如同时计算与训练样本的距离（量子并行性）。

首先，Schuld 等人（2014）、Wi´sniewska 和 Sawerwain（2018）、Ruan 等人（2017）、Zhou 等人（2021）以及 Li 等人（2021）提出了采用汉明距离、因此要求特征为二进制的量子 k 近邻算法。具体而言，前两项研究通过酉操作将量子比特差值之和（通过受控非门得到的差值）编码到振幅中，从而计算汉明距离（这一思路最初由 Trugenberger 于 2002 年提出）。随后，通过直接测量来进行分类，而不显式地选取最近邻。相比之下，其他研究利用 Kaye（2004）提出的递增量子电路，以基态编码的方式获得距离值。此后，Ruan 等人（2017）通过或门和投影操作，选出距离低于给定阈值的数据并直接进行分类；Zhou 等人（2021）利用 D¨urr 的最小化算法（D¨urr 和 Høyer，1999）寻找 k 个最小距离值；而 Li 等人（2021）则应用一种受二分查找启发的新型量子搜索过程，以确定最小值。

对于非二元特征，广泛使用与向量夹角相关的距离度量，例如余弦距离。例如，Dang 等人（2018年）和 Wang 等人（2019年）已将此类量子 k-NN 变体应用于图像分类任务。具体而言，使用无需测量的 SWAP 测试（Buhrman 等人，2001）来计算距离，然后通过振幅估计算法（Brassard 等人，2002）将其数值传递到量子比特状态中。最后，利用 D¨urr 算法找出最近邻。Wiebe 等人（2015年）最先提出了这一流程，但当时仅用于寻找最近邻。相比之下，Afham 等人（2020年）和 Ma 等人（2021年）提出了一种概念上更简单的变体：通过迭代执行 SWAP 测试和测量，估计一个与训练实例的平方余弦相似度成正比的量。此外，该模型支持并行处理多个测试实例，如 Ma 等人（2021年）所示。事实上，Afham 等人最近还提出了另一种变体（Basheer 等人，2021年），但其流程与前述工作并无太大区别。具体而言，它涉及 SWAP 测试、量子模数转换算法（Mitarai 等人，2019年），以及 D¨urr 算法的一种变体。

非二元特征的其他有趣距离度量包括欧几里得距离、马氏距离和极距离。具体而言，第2.3节对欧几里得距离进行了深入讨论。关于其他距离度量，Gao 等人（2022年）提出了一种基于马氏距离的变体，而Feng 等人（2023）提出了一种基于极距离的量子 k-NN 算法，该算法通过可调参数结合了角度和模长信息。具体而言，马氏距离通过利用相位估计算法（Cleve 等人，1998），结合哈密顿量模拟（Rebentrost 等人，2018年）和受控旋转来计算；

相比之下，极距离通过无需测量的 SWAP 测试和一对托福利门（其中一个为扩展托福利门）进行计算。随后，在这两项工作中，通过应用振幅估计算法（或其相干版本，该版本由Wiebe 等人，2015年）提出），将距离编码到量子比特状态中，并通过 D¨urr 算法（或基于该算法的算法，该算法由Miyamoto 等人，2019年提出）检索最近邻。最后，还值得一提的是，有人提出了一种基于量子排序子程序的量子 k-NN 算法，提出者为Quezada 等人（2022年）。该算法需要一个用于计算距离并将其编码到量子比特状态中的度量算子、一个用于识别已排序序列的预言机，以及 Grover 算法（Grover, 1996）；与其他工作一样，分类过程无需识别 k 最近邻即可直接完成。

2.3 量子欧几里得距离

欧几里得距离是机器学习中一种广为人知的距离度量。这里给出其平方版本的定义，因为这在后续章节中会很有用。具体而言，给定两个向量 u,v $\in \mathbb{R}^n$，它们之间的平方欧几里得距离 d^2(u,v) 定义为

$$d^2(\mathbf{u},\mathbf{v}) = \|\mathbf{u}-\mathbf{v}\|^2 = \|\mathbf{u}\|^2 - 2\langle \mathbf{u},\mathbf{v}\rangle + \|\mathbf{v}\|^2, \tag{1}$$

其中，$\langle$u, v$\rangle$ 是 u 与 v 之间的标量积。

所讨论的距离度量也已应用于量子机器学习领域。例如，Lloyd 等人 (2013) 提出了一种量子过程，用于估计一个数据点与某个簇的质心之间的平方欧几里得距离，即数据组中所包含元素的均值。具体而言，该算法依赖于 SWAP 测试，该测试应用于索引寄存器，且不要求输入向量具有单位范数。类似的过程被 Sarma 等人 (2020年) 用于提出一种混合 k均值聚类算法（其中质心通过经典方式计算），而 Getachew (2020年) 则将其用于 k中位数算法的混合版本。另一方面，Yu 等人 (2020年) 提出了三种量子算法，用于基于平方欧几里得距离估计数据集的三种相似度测量。具体而言，所有这些过程均不要求输入向量具有单位范数，利用量子干涉（由基变换产生），并使用振幅估计算法来确定相似度度量。最后，值得一提的是 Schuld 等人(2017) 设计的量子二元分类器。具体来说，该分类器电路由一个哈达玛门（量子干涉所必需）、一次条件测量以及最后一次测量组成。通过迭代上述过程，可以为每个类别估计一个与平方欧几里得距离相关的概率值。在这项最后的研究中，考虑的是具有单位范数的输入向量。

关于量子 k-NN 模型，据作者所知，文献中唯一一种基于欧几里得距离的变体由 Fastovets 等人 (2019年) 提出；该方法利用了 Lloyd

等人 (2013) 来估计成对距离值，并使用 D¨urr 的最小化算法来寻找 k 最近邻。其主要缺点在于，这两个步骤都需要进行多次迭代，因为二者都涉及最终测量。此外，D¨urr 的算法需要使用预言机，即一个黑盒函数。实际上，Wiebe 等人 (2015年) 提出的最近邻算法也采用欧几里得距离作为距离度量。然而，第 2.2 节所述的工作流程，实现起来相当复杂。最后，单连接的计算，即 Yu 等人 (2020年) 所考虑的集合相似度度量之一，可以被视为最近邻搜索的一种泛化形式。尽管如此，他们的量子算法使用了输入向量的倒数，因此从理论上讲，原始距离关系无法得到保留。

3 方法

本节介绍一种基于欧几里得距离度量的新型量子 k 近邻算法，并简要讨论该算法与经典对应算法相比的复杂度。

3.1 算法

在本文提出的量子 k 近邻算法中，通过一种新颖的编码和一个简单的量子电路，并行计算与所有训练实例的平方欧几里得距离相关的一个量；该电路无需受控 SWAP 门即可执行类 SWAP 测试过程。实际上，该算法利用量子干涉，将这些与距离相关的值编码到量子态的振幅中，再通过测量对其进行估计。值得强调的是，输入向量不进行单位范数归一化，因为这会造成显著的信息损失。此外，所需的量子比特数量较少，且不涉及预言机，因此该算法切实可行。下文将更详细、正式地介绍这一新算法的步骤。

3.1.1 数据预处理

考虑一个训练集 $\mathcal{U} = \{\mathrm{u}_0, ..., \mathrm{u}_{N-1}\}$，其中包含实值数据实例 $\mathrm{u}_j \in \mathbb{R}^d$，并令 $\mathcal{L} = \{l_0, ..., l_{N-1}\}$ 为相应标签的集合。此外，考虑一个测试实例 $\mathrm{u}' \in \mathbb{R}^d$，其标签未知。

该算法的预处理步骤包括将数据特征归一化到范围 $\left[-\frac{1}{2\sqrt{d}}, \frac{1}{2\sqrt{d}}\right]$。这样，所得向量的最大范数为 $1\ 2$，最大（平方）欧几里得距离则为 1。

3.1.2 初始态与编码

令 $\mathcal{V} = \{\mathrm{v}_0、...、\mathrm{v}_{N-1}\}$ 和 v' 分别表示经过上述预处理步骤后的训练集和测试实例。随后将量子电路初始化

为状态

$$|\psi\rangle = |0\rangle \otimes \left(\frac{1}{\sqrt{2}} (|0\rangle |\alpha\rangle + |1\rangle |\beta\rangle) \right), \tag{2}$$

其中

$$|\alpha\rangle = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} |j\rangle \sum_{i=0}^{F-1} x_{ji} |i\rangle ,$$

$$|\beta\rangle = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} |j\rangle \sum_{i=0}^{F-1} x'_{ji} |i\rangle .$$

此处， F 是一个正整数，其取值取决于所使用的编码，而 $\mathbf{x}_j = \{x_{ji}\}_{i=0,\ldots,F-1}$ 和 $\mathbf{x}'_j = \{x'_{ji}\}_{i=0,\ldots,F-1}$ 分别表示预处理后的训练数据和测试数据的量子编码形式。因此，所需的量子比特数为 $2 + \lceil \log_2 N \rceil + \lceil \log_2 F \rceil$。具体而言，本研究设计并测试了两种编码，其优点将在后续章节中讨论：扩展编码和平移编码。下面来看它们的定义。对于扩展编码， $F = 2d + 3$ 和

$$x_{ji} = \begin{cases} \frac{2}{\sqrt{3}} v_{ji} \\ \frac{2}{\sqrt{3}} v_{j(i-d)} \\ \frac{2}{\sqrt{3}} \|\mathbf{v}_j\| \\ 0 \\ \sqrt{1 - 4\|\mathbf{v}_j\|^2} \end{cases} \qquad x'_{ji} = \begin{cases} -\frac{2}{\sqrt{3}} v'_i & 0 \le i < d \\ -\frac{2}{\sqrt{3}} v'_{(i-d)} & d \le i < 2d \\ \frac{2}{\sqrt{3}} \|\mathbf{v}_j\| & i = 2d \\ \sqrt{1 - \frac{4}{3}(2\|\mathbf{v}'\|^2 + \|\mathbf{v}_j\|^2)} & i = 2d+1 \\ 0 & i = 2d+2, \end{cases}$$

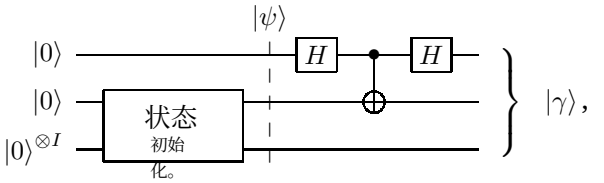
其中 v_{ji} 是第 i 个预处理后的训练实例的第 j 个特征，而v'_i 是预处理后的测试实例的第 i 个特征。对于平移编码， $F = 2d + 4$ 和

$$x_{ji} = \begin{cases} v_{ji} \\ v_{j(i-d)} \\ \|\mathbf{v}_j\| \\ \frac{1}{2} \\ 0 \\ \sqrt{\frac{3}{4} - 3\|\mathbf{v}_j\|^2} \end{cases} \qquad x'_{ji} = \begin{cases} -v'_i & 0 \le i < d \\ -v'_{(i-d)} & d \le i < 2d \\ \|\mathbf{v}_j\| & i = 2d \\ -\frac{1}{2} & i = 2d+1 \\ \sqrt{\frac{3}{4} - (2\|\mathbf{v}'\|^2 + \|\mathbf{v}_j\|^2)} & i = 2d+2 \\ 0 & i = 2d+3. \end{cases}$$

因此，这两种编码所需的量子比特数量相同。还值得强调的是，在这两种情况下，$\mathbf{x}j$ （因此 $|\alpha\rangle$）都独立于预处理后的测试实例 $\mathbf{v}'$，而 $\mathbf{x}'_j$ （因此$|\beta\rangle$）则取决于预处理后的训练集 $\mathcal{V}$。

3.1.3 Bell-H 操作与最终状态

完成初态制备后，执行一个记为 Bell-H 的操作。具体而言，Bell-H 对应于类 SWAP 测试过程，其中，通过受控非门（ CNOT 门）使最初制备为叠加态的目标状态（$|\alpha\rangle$ 和 $|\beta\rangle$）发生干涉。相应的量子电路（还包括初态制备，并以一条竖直虚线分隔）如下：



其中 $I = \lceil \log_2 N \rceil + \lceil \log_2 F \rceil$。实际上，Bell-H 量子电路由作用于第一个量子比特的哈达玛门、以第一个量子比特为控制端且以第二个量子比特为目标端的 CNOT 门，以及另一个作用于第一个量子比特的哈达玛门组成。因此，它与通常用于生成 Bell 态的标准 Bell 电路的区别在于多了一个后置哈达玛门。与标准 SWAP 测试相比，其一项显著优势是所需基本量子门的数量恒定（三个），与所涉状态的规模无关；其缺点是输入状态的制备更为复杂，尤其是在无法使用量子随机存取存储器的情况下。

Bell-H 操作后得到的输出态为

$$|\gamma\rangle = \frac{1}{2}\left(|0\rangle \otimes \left(\frac{1}{\sqrt{2}}(|0\rangle|\alpha\rangle + |0\rangle|\beta\rangle + |1\rangle|\beta\rangle + |1\rangle|\alpha\rangle)\right) + |1\rangle \otimes \left(\frac{1}{\sqrt{2}}(|0\rangle|\alpha\rangle - |0\rangle|\beta\rangle + |1\rangle|\beta\rangle - |1\rangle|\alpha\rangle)\right)\right), \quad (3)$$

测量第一个量子比特得到 1 的概率为 $\frac{1}{2}(1 - \langle\alpha|\beta\rangle)$（推导过程见附录 A.1）。通过提取索引寄存器上的求和（$|j\rangle$ 在 $|\alpha\rangle$ 和 $|\beta\rangle$中），并对第二个量子比特和特征寄存器进行偏迹（即舍弃）（$|i\rangle$ 在 $|\alpha\rangle$ 和 $|\beta\rangle$中），可以将最终状态写为

$$|\delta\rangle = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \left[\sqrt{1 - s(\mathbf{v}_j, \mathbf{v}')}\,|0\rangle + \sqrt{s(\mathbf{v}_j, \mathbf{v}')}\,|1\rangle\right]|j\rangle\,, \quad (4)$$

其中 $s(\mathbf{v}_j,\mathbf{v}')$ 是与 $\mathbf{v}_j$ 和 $\mathbf{v}'$ 之间的平方欧几里得距离相关的相似度度量；因此，距离越小，

$s(\mathbf{v}_j, \mathbf{v}')$ 值越高。具体而言，$s(\mathbf{v}_j, \mathbf{v}')$ 定义为

$$s(\mathbf{v}_j, \mathbf{v}') = P(qubit_1 = 1 \mid j) = \frac{1}{2}(1 - \langle \mathbf{x}_j, \mathbf{x}'_j \rangle), \tag{5}$$

其中，量子比特1是量子电路中的第一个量子比特，而 $\langle \mathbf{x}_j$、$\mathbf{x}'_j \rangle$ 的取值取决于所使用的编码方式，如表 1 所示（关于如何推导公式 4 的更详细说明见附录 A.2）。

表 1 两种编码的性质。请注意，$s(\mathbf{v}_j, \mathbf{v}')$ 的取值范围由预处理后的测试实例 $\mathbf{v}'$ 决定。

	扩展	平移
$\langle \mathbf{x}_j, \mathbf{x}'_j \rangle$ 值	$\frac{1}{3}(\|\mathbf{v}_j\|^2 - 2\langle \mathbf{v}_j, \mathbf{v}' \rangle)$	$\|\mathbf{v}_j\|^2 - 2\langle \mathbf{v}_j, \mathbf{v}' \rangle - \frac{1}{4}$
最小 $s(\mathbf{v}_j, \mathbf{v}')$ 范围	$[0.333, 0.5]$	$[0.5, 0.625]$
最大 $s(\mathbf{v}_j, \mathbf{v}')$ 范围	$[0, 0.666]$	$[0.25, 0.75]$

观察表1 的第一行并回顾公式（1），可以发现两个方面：$\langle \mathbf{x}_j, \mathbf{x}'_j \rangle$ 与两种编码下 $\mathbf{v}_j$ 和 $\mathbf{v}'$ 之间的欧几里得距离平方密切相关；项 $\|\mathbf{v}'\|^2$ 并未出现。然而，这并不是问题，因为对于所有训练实例而言，$\|\mathbf{v}'\|^2$ 的值都相同。表中包含的其他信息有助于理解每种编码的优势。具体而言，扩展编码最大化了 $s(\mathbf{v}_j, \mathbf{v}')$ 的可能取值范围，从而能够更好地表示相似度值，即降低表示结果对噪声存在的敏感性。相反，平移编码最大化了在第一个量子比特上测量得到 1 的概率，这种情况具有优势，原因将在下一节中变得更加清楚。最后，值得强调的是，$s(\mathbf{v}_j, \mathbf{v}')$ 的范围由 $\mathbf{v}'$决定；具体而言，最小范围对应于范数为零的测试实例，而最大范围对应于范数等于1 2 的测试实例（即可能的最大值）。

-

3.1.4 测量和距离估计

在 *Bell-H* 操作之后，对公式（4）中所示的量子比特状态，即量子电路中的第一个量子比特和索引寄存器 $|j\rangle$，进行测量。具体而言，首先测量第一个量子比特。因此，当获得 1（0）时，最近邻的索引将具有最高（最低）概率。如果先测量索引寄存器，则索引将被均匀采样。通过迭代执行量子电路和测量过程，将联合概率 $P(0, j)$ 和 $P(1, j)$ 估计为相对频率，从而进一步估计欧几里得距离。事实上，以下关系成立：

$$P(0, j) = \frac{1 + \langle \mathbf{x}_j, \mathbf{x}'_j \rangle}{2N} \implies \langle \mathbf{x}_j, \mathbf{x}'_j \rangle = 2N \times P(0, j) - 1, \tag{6}$$

$$P(1,j) = \frac{1 - \langle \mathbf{x}_j, \mathbf{x}'_j \rangle}{2N} \implies \langle \mathbf{x}_j, \mathbf{x}'_j \rangle = 1 - 2N \times P(1,j). \tag{7}$$

此外，对于扩展编码（见表1），

$$d(\mathbf{v}_j, \mathbf{v}') = \sqrt{\frac{3}{4} \langle \mathbf{x}_j, \mathbf{x}'_j \rangle + \|\mathbf{v}'\|^2}, \tag{8}$$

其中， $d(\mathbf{v}_j, \mathbf{v}')$ 是 $\mathbf{v}_j$ 与 $\mathbf{v}'$之间的欧几里得距离，而对于平移编码，

$$d(\mathbf{v}_j, \mathbf{v}') = \sqrt{\langle \mathbf{x}_j, \mathbf{x}'_j \rangle + \frac{1}{4} + \|\mathbf{v}'\|^2}. \tag{9}$$

无论使用哪种编码，都可以通过 $P(0,j)$ 或 $P(1,j)$ 估计欧几里得距离 $d(\mathbf{v}_j, \mathbf{v}')$。在本研究中，我们设计并测试了两种结合这两个联合概率信息的方法：平均值和差值。具体而言，平均距离估计是根据 $P(0,j)$ 估计出的欧几里得距离与根据$P(1,j)$ 估计出的欧几里得距离的平均值。而对于差异距离估计，标量积的值为

$$\langle \mathbf{x}_j, \mathbf{x}'_j \rangle = N \times (P(0,j) - P(1,j)),$$

根据所用编码，通过公式（8）或（9）即可得出欧几里得距离。

最后，值得指出两点：给定$P(0,j)$ 和 $P(1,j)$ 的估计值，对于两个索引 $j_1, j_2 \in \{0, ..., N-1\}$，相应的训练实例按照平均值和差值距离估计可能会有不同的排序；如果可以通过数学方法得出距离估计（即平方根内的参数非负），则平均值距离估计始终小于或等于相应的差异估计。有关这些观察结果的更多详情，请参阅附录B。

3.1.5 K 近邻与分类

估算出所有欧几里得距离 $d(\mathbf{v}_j, \mathbf{v}')$ 后，训练样本通常会根据这些距离进行排序。然后，确定 k 个最近邻，并通过对最近邻的标签进行多数投票来分类测试实例。

3.2 复杂度观察

就复杂度而言，基于欧几里得距离度量的量子 k-NN 算法与其经典对应算法之间的差异在于欧几里得距离的估计/计算。实际上，在这两种情况下，数据预处理、 k 最近邻的识别（通过距离排序操作完成）以及分类均以经典方式进行。

对于经典 k-NN 算法，欧几里得距离计算的复杂度为 $O(Nd)$。相比之下，在量子 k-NN 算法中，欧几里得距离并不被精确计算，而是通过估计获得。为此，需要一定次数的测量次数，即测量和迭代次数。具体而言，每次迭代都需要制备初始态、运行 Bell-H 量子电路，并测量量子比特的状态。初始态制备是一项相当复杂的操作。如果有量子随机存取存储器可用，并假设实数值 x_{ji} 和 x'_{ji} 以经典浮点数的形式存储在量子随机存取存储器中，则初始态可以以 $O(\log(NF))$ 的复杂度制备。实际上，状态 $|\alpha\rangle$ 和 $|\beta\rangle$ 可以从量子随机存取存储器中以复杂度$O(\log(NF))$检索。相比之下，如果没有量子随机存取存储器，则需要从 $|0\rangle^{\otimes(1+I)}$ 开始制备所需状态，而所需门的数量取决于量子处理器的架构。关于 Bell-H 量子电路，它由常数数量的基本量子门组成，因此其复杂度为 $O(1)$。测量步骤同样具有常数复杂度，因为测量是并行执行的。最后，给定状态计数，距离估计的计算代价为 $O(N)$。因此，如果有量子随机存取存储器可用，总复杂度为 $O(\text{shots} \times \log(NF) + N)$，这等于 $O(\text{shots} \times (\log N + \log d) + N)$。还应强调的是，$N$越高，估计欧几里得距离所需的测量次数就越多。总之，如果假设 shots 是固定值，则量子 k-NN 算法中欧几里得距离估计的复杂度为 $O(\log d + N)$，低于 $O(Nd)$。相反，如果假设 shots 以对数方式依赖于 N，则该复杂度为 $O(\log N \log d + N)$，仍低于 $O(Nd)$。

4 实现

本节介绍第 3 节中所述算法的实现。具体而言，基于欧几里得距离的量子 k-NN 已使用 Python 和 IBM 提供的开源软件开发工具包 Qiskit 实现（Anis 等人，2021）。该代码已公开发布于 https://github.com/ZarHenry96/euclidean-quantum-k-nn，支持多种执行模式，其中包括：

• 经典，它不涉及量子电路，而是运行经典 k-NN，并使用欧几里得距离度量，预处理步骤如第 3.1.1节所述；

• 状态向量，该过程处理量子电路的最终状态向量，实际上代表了具有无限次迭代的理想执行（在这种情况下，不执行测量）；

•模拟（代码中的局部模拟），通过从量子电路的最终概率分布中进行采样，以提供状态计数。

这些模式均未考虑噪声的存在。此外，图 1展示了一个示例电路（用于模拟模式）。

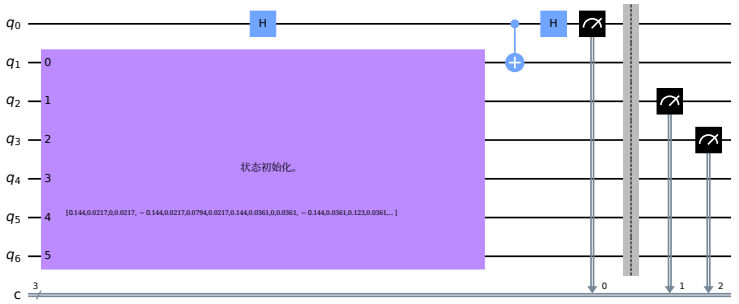


图 1 基于欧几里得距离的量子 k-NN 量子电路示例。具体而言， $N = 4$、 $d = 2$，且执行模式为 $simulation$ （$statevector$ 不包括最终测量）

一般而言，该算法的实现遵循第 3.1 节所述的内容，但有一些技术细节值得说明。关于预处理步骤，每个特征都按以下方式归一化：减去训练集中最大值与最小值的平均值；再除以特征范围（在训练集上计算；若该特征为常量，则设为 1）乘以 $\sqrt{d}$。此外，如果测试实例的某个特征在归一化后超出目标范围，则将其裁剪到超出范围的边界值。接下来，我们重点介绍涉及量子电路的执行模式，因为 经典执行模式的工作原理相当直接。具体而言，初始态 $|\psi\rangle$ （公式2）的制备方式是：将除第一个量子比特以外所有量子比特的振幅提供给 Qiskit 提供的初始化函数，而第一个量子比特默认处于状态 $|0\rangle$。对于未关联任何训练实例的索引（当 N 不是 2 的幂时会存在此类索引），计算距离值时会将其排除（因此，在联合概率估计中仍会保留这些索引）。此外，如果由于得到的状态计数分布导致公式（8）或（9）中平方根的被开方数小于零或大于一，则分别将距离值近似为零或一。最后，对于欧几里得距离相同的训练实例，按其在训练集中的索引升序排列（经典执行模式也遵循此规则）。

总之，值得强调的是，拉普拉斯平滑（Wilson，1927年）已应用于 模拟模式下联合概率的估计。实践中，给定状态 c 的计数 $|a\rangle|j\rangle$，其中 $a \in \{0,1\}$，概率 $P(a,\ j)$ 估计为

$$P(a,j) = \frac{c+p}{shots + 2Np},$$

其中， p 是添加的伪计数数量（每个状态各自添加），shots 是测量总次数。具体而言，伪计数仅累加

到重要索引的计数中，即实际与训练实例相关联的索引。

5 实证评估

本节介绍了所测试的方法、所用的数据集、采用的实验设置以及获得的结果。具体而言，实验在一台共享机器上运行，该机器配备 Intel Xeon Gold 6238R 处理器（运行频率为 2.20 GHz）和 125 GB 内存。

5.1 方法

量子 k近邻方法基于第 3 节介绍的欧几里得距离，已在不同的执行模式下，并采用不同的（编码，距离估计）配置进行了测试，相关信息见表 2。由于缺乏可免费使用且量子比特数量充足的量子设备，因此未在真实量子设备上进行运行。此外，为了进行比较，还考虑了一些经典基线方法（列于同一张表中）；具体而言，这些方法的结果数据取自 Zardini 等人(2022年)（更确切地说，取自 Zardini，2023年（b））。

表 2 测试的方法

基于欧几里得距离的量子 k-近邻算法		
执行模式	编码	距离估计
经典	-	-
状态向量	扩展，平移	平均值，差值
模拟	扩展，平移	平均值，差值

基线方法
基于余弦距离的k-近邻
随机森林（100 棵树）
采用 {高斯核、线性} 核的支持向量机

5.2 数据集

所有实验使用的数据集均取自 Zardini 等人 （2022年）的文章（更准确地说，取自 Zardini，2023年（a）），主要是为了便于将结果与基线方法进行比较。具体而言，这些数据集的属性列于表 3中。如上述文章所述，原始版本的数据集是根据数值特征等特定标准从 UCI机器学习库（Dua 和 Graff，2017）中选取的。随后，对这些数据集进行了预处理，以满足二元类别标签等具体要求。本文使用的数据集可与代码一同在 https://github.com/ZarHenry96/euclidean-quantum-k-nn获取，它们是

类别数量减少后、未经下采样得到的数据集。有关选择标准和预处理流程的更多信息，请参阅原文。

表 3 所用数据集的属性。请注意，数据集名称均为链接，指向这些数据集原始版本在 UCI 上的页面。

Name	类别	Size	特征
01_鸢尾花 setosa_versicolor	2	100	4
01_鸢尾花 setosa_virginica	2	100	4
01_鸢尾花 versicolor_virginica	2	100	4
02_输血（Yeh 等，2009年）	2	748	4
03_脊柱 2C_	2	310	6
04_种子 1_2_	2	140	7
05_大肠杆菌细胞质内膜 proteins	2	220	7
06_眼镜 1 2 _	2	146	9
07_乳腺组织 adi fadmasgla	2	71	9
08_乳腺癌_（Patrıcio 等，2018年）	2	116	9
09_英国和美国口音识别 _	2	210	12
10_叶片_11_9（Silva 等，2013）	2	30	14

5.3 实验设置

在所有实验中，均采用分层 k 折交叉验证作为验证技术。实际操作中，每个数据集被划分为 k 折，即若干子集。然后，$k-1$ 折构成训练集，剩余的一折作为测试集；这最后一步重复执行 k 次，以确保每个子集都恰好作为一次测试集。“分层”一词意味着，各折中类别之间的比例尽可能接近给定数据集中的比例。此外，所有实验均使用相同的随机种子生成各折；这样，所有方法都在相同的折上进行了评估。

量子 k-NN 实验所采用的参数值列于表 4。具体而言，对于所有执行模式，k折交叉验证（折）的折数均设为 5（这是机器学习中的常见取值），并考虑了四种不同的最近邻数量（k）。对于 模拟 模式，所有（编码、距离估计）配置均使用 1024 次测量（测量次数）进行测试，这是 Qiskit 提供的默认值；同时，也通过改变这一参数的取值对最佳配置进行了评估。此外，拉普拉斯平滑的伪计数数量任意设为 10，并使用不同的种子运行了 5 次，以获得统计证据。具体而言，每个测试实例的模拟随机种子均从一个“根”运行种子随机生成。最后值得指出的是，不同的 k 值是在不同的种子下评估的，而平均值 和 差值 距离估计是在相同的种子下评估的（也就是说，对于每个测试实例，两个距离估计均使用相同的状态计数计算）。

关于用于比较的基线方法（Zardini 等人，2022年），应用于输入数据特征的归一化过程是

表 4 量子 k-NN 实验的参数设置

通用参数	
折	5
k	3, 5, 7, 9

模拟参数	
测量次数	512[a], 1024, 2048[a], 4096[a], 8192[a]
伪计数	10
runs	5

[a]对于量子 k-NN，仅测试了在该测量次数下表现最佳的（*encoding, distance estimate*）配置。

标准的最小-最大归一化，其输出范围为 $[0, 1]$。折数、折的生成随机种子，以及经典k-NN（采用余弦距离）所考虑的 k 值，均与量子 k-NN 实验中使用的相同。最后，随机森林的运行次数为 5 次（这是一种随机方法）。

5.4 结果

结果以散点图和箱线图的形式呈现。具体而言，量子 k-NN 的评估指标包括分类准确率和所找到的最近邻的正确性；而对于基线方法，仅考虑分类准确率。更具体地说，给定一个折，准确率定义为

$$accuracy = \frac{number\ of\ correctly\ classified\ instances\ in\ the\ fold}{total\ number\ of\ instances\ in\ the\ fold};$$

在进行多次运行时，报告各次运行的平均值。至于所找到的最近邻的正确性，则采用杰卡德指数和平均杰卡德得分（Greene 等人，2014）进行评估。具体而言，给定一个测试实例，杰卡德指数（JI）定义为

$$Jaccard\ index\ (JI) = \frac{|\mathcal{S}_c \cap \mathcal{S}_f|}{|\mathcal{S}_c \cup \mathcal{S}_f|},$$

其中， $\mathcal{S}_c$ 是正确最近邻的集合（通过经典方法计算）， $\mathcal{S}_f$是找到的最近邻集合。由于每个测试实例都有一个杰卡德指数值，因此对每个折取其平均值，并报告该平均值在多次运行中的平均值。平均杰卡德（AJ）得分也采用相同的处理方式；给定一个测试实例，其定义为

$$Average\ Jaccard\ (AJ) = \frac{1}{k}\sum_{m=1}^{k} h(\mathcal{S}_{cm}, \mathcal{S}_{fm}),$$

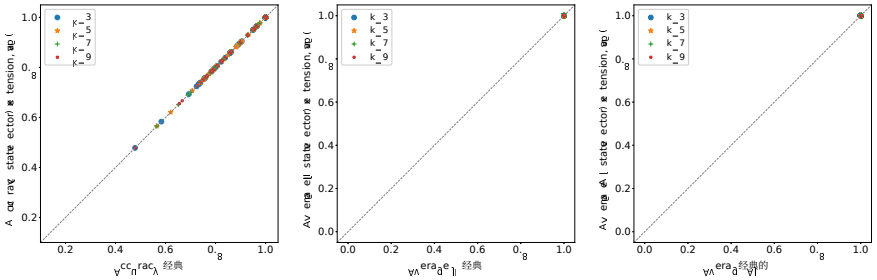
其中， k 是选取的最近邻数量，h 是计算杰卡德指数的函数， $\mathcal{S}_{cm}$ 是包含截至第 m 个最相似项的正确最近邻集合（ $\mathcal{S}_{fm}$同理）。最后，通过.....评估了所得结果的统计显著性。

威尔科克森符号秩检验（Wilcoxon，1945年），由于数据是配对的；在某些情况下（差值箱线图）还考虑了单样本 T 检验（Gosset，1908年）。

5.4.1 执行模式比较

我们首先来考虑 经典 和 状态向量 执行模式。如图 2所示，两种模式在准确率（图 2a）、杰卡德指数（图 2b）和平均 Jaccard 分数（图 2c）方面是等效的；威尔科克森符号秩检验（表 5）证实两者不存在统计差异。这里仅展示一种 状态向量 配置，即（扩展、平均值），但所有配置的结果都相同[1]。这证实了第 3 节中介绍的算法是正确的；事实上，在理想情况下，它能够达到与其经典对应算法相同的结果。还需注意，量子算法相较于其经典对应算法的优势在于执行时间。

接下来，我们将重点关注状态向量和模拟执行方式。具体而言，图3 比较了（3a）中的准确率（图3b）、杰卡德指数（图3c）以及平均杰卡德得分（图扩展，平均值）配置的表现。正如预期，测量次数有限（1024）会导致性能显著恶化，而状态向量



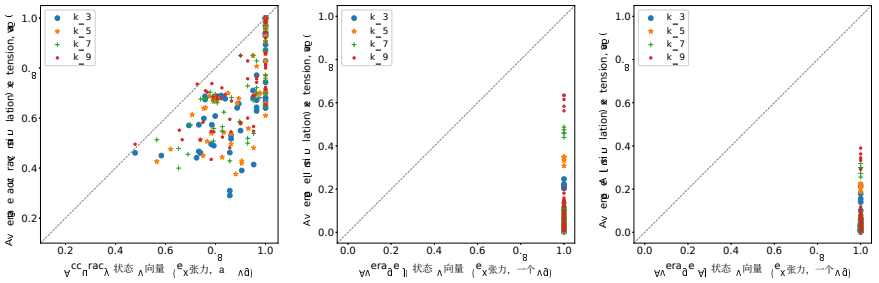
(a) (b) (c)

图 2 *classical* 和 *statevector* 执行模式在准确率 (a)、杰卡德指数 (b) 和平均杰卡德得分 (c) 方面的比较。*statevector* 所用的配置为 (*extension*, *avg*)，但所有配置的结果均相同。每个点对应一个数据集折。

表 5 应用于图 2 所示分布的威尔科克森符号秩检验（$\alpha = 0.05$）。表中报告的值是获得的 p 值

	k=3	**k=5**	**k=7**	**k=9**
图2a	1.000	1.000	1.000	1.000
图2b	1.000	1.000	1.000	1.000
图2c	1.000	1.000	1.000	1.000

[1]对于 *translation* 编码，在某个数据集的一个折中，由于距离值的数值近似，出现了两个最近邻互换的情况。因此，该折的平均杰卡德得分略低于 *classical*，但差异在统计学上并不显著（对于所有 k 值，p值 $=0.317$ ）。



(a) (b) (c)

图 3 *statevector* (*extension*, *avg*) 与 *simulation* (*extension*, *avg*) 在准确率 (a)、杰卡德指数 (b) 和平均杰卡德得分 (c) 方面的比较。 *simulation* 的测量次数为1024，每个点对应一个数据集折

表6 对图 3 所示分布应用威尔科克森符号秩检验 ($\alpha = 0.05$)。表中报告的数值为所得的 p 值

	k=3	**k=5**	**k=7**	**k=9**
图 3a	1.624E-10	1.622E-10	5.140E-10	1.608E-09
图3b	1.626E-11	1.630E-11	1.629E-11	1.630E-11
图3c	1.629E-11	1.630E-11	1.630E-11	1.630E-11

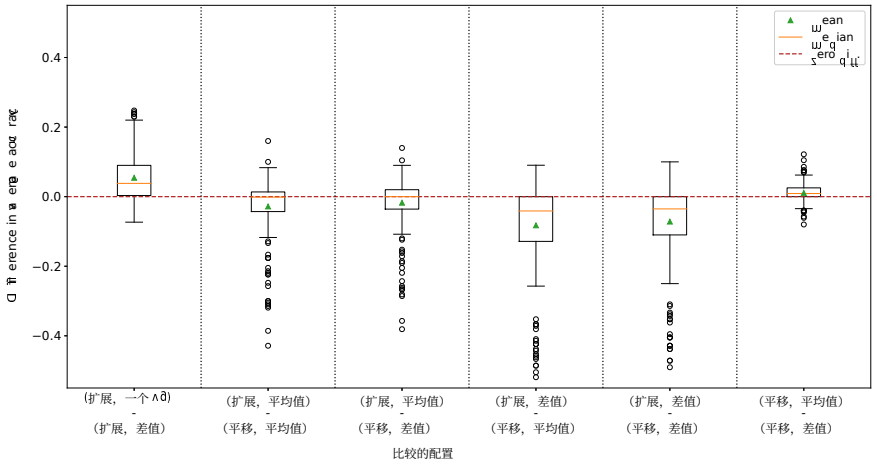
结果表明，在分类准确率和所找到的最近邻正确性方面，统计上均优于模拟方式，相关结果见表6。具体而言，杰卡德指数和平均 Jaccard 分数的性能下降更为明显。对于所有（编码，距离估计）配置，这些观察结果均成立；其他配置的类似图表及相关显著性表见附录C.1。

5.4.2 编码与距离估计对比

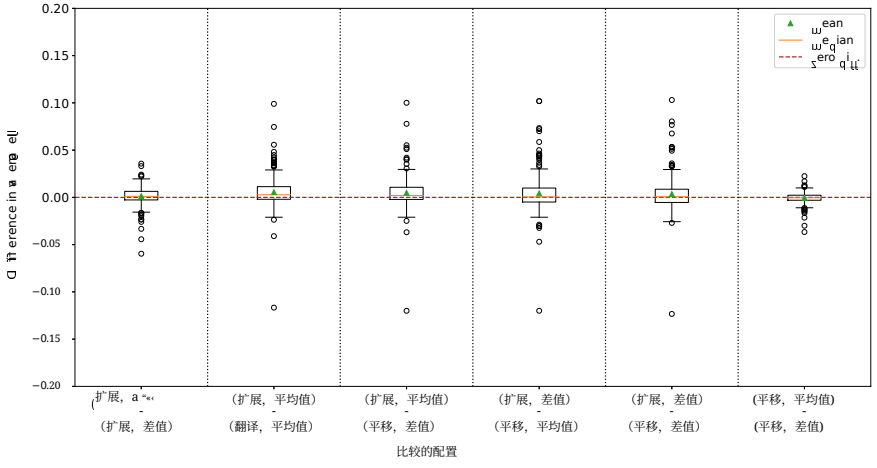
由于各种（编码、距离估计）配置在状态向量执行模式下得到的结果相同（欧几里得距离估计是精确的），因此这里只考虑模拟（测量次数为 1024）的情况。具体而言，我们通过差值箱线图比较各配置，其中每个数据点表示一个（数据集折，k 值）组合对应的差值。准确率和杰卡德指数的比较分别见图4a 和图4b，平均杰卡德得分的图因篇幅所限，见附录C.2 （图10）。

在分类准确率方面，取得最佳结果的配置是（平移，平均值），正如表7a 所证实的，该配置在统计上优于所有其他配置。总体而言，平移编码的准确率优于扩展编码；在编码相同的情况下，平均值距离估计优于差值距离估计。此外，除（扩展，平均值）与（平移，差值）的比较在中位数方面外，无论是中位数（Wilcoxon 符号秩检验）还是均值（单样本 T 检验），差异在统计上均显著。

令人惊讶的是，分类准确率最高的配置并非找到最佳最近邻的配置。事实上，在杰卡德指数方面取得最佳结果的配置是（扩展，平均值），如表 7 b 所示，与其他配置相比，差异几乎总是显著的。总体而言，扩展编码在杰卡德指数方面优于平移编码；而在编码相同的情况下，并没有明显占优的距离估计：使用平均值距离估计时，扩展编码表现更好；而差值距离估计则在平移编码下取得了更好的结果。差异



(a)



(b)

图4 在 *simulation* 执行模式下，(*encoding*, *distance estimate*) 配置在准确率 (a) 和杰卡德指数 (b) 方面的比较。测量次数为 1024，每个数据点对应一组 (*dataset fold*, *k value*) 配对的差值

表 7 应用于图4a (a) 和图4b (b) 所示分布的威尔科克森符号秩检验和单样本 T 检验。每一列对应一种不同的比较方式，第一个字母标识编码（E=*extension*、T=*translation*），第二个字母标识距离估计（A=*avg*、D=*diff*）。表中报告的数值为所获得的 p 值（$\alpha = 0.05$）

(a)

	EA-ED	EA-TA	EA-TD	ED-TA	ED-TD	TA-TD
Wilcoxon	9.378E-26	3.019E-05	0.069	1.771E-24	1.539E-21	1.084E-09
T 检验	1.236E-27	3.706E-07	0.001	1.404E-20	2.037E-18	2.492E-09

(b)

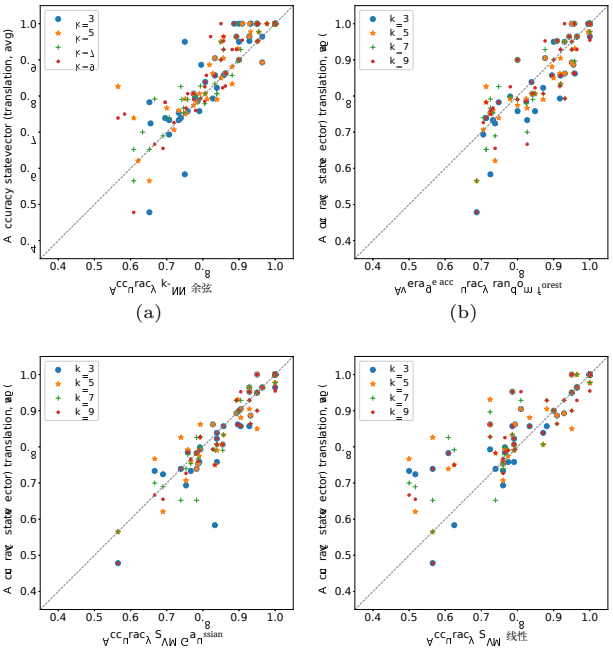
	EA-ED	EA-TA	EA-TD	ED-TA	ED-TD	TA-TD
Wilcoxon	0.003	8.888E-09	1.917E-07	0.010	0.043	0.104
T 检验	0.054	6.775E-07	1.102E-05	0.001	0.005	0.054

就中位数和均值而言，几乎所有差异都具有统计显著性；唯一的例外是均值上的（扩展，平均值）-（扩展，差值）比较，以及两项统计量上的（平移，平均值）-（平移，差值）比较。平均杰卡德得分（图10）呈现出类似趋势，其中（扩展，平均值）是表现最佳的配置。然而，在这种情况下，采用扩展编码和差值距离估计的表现最差，这意味着在该配置选出的 k 最近邻中，正确的邻居排在最后几位。此外，如表13所示，只有少数差异具有统计显著性。其中值得一提的是，（扩展，平均值）配置在中位数方面显著优于所有其他配置，而（扩展，差值）配置在均值方面也显著优于其他配置。

5.4.3 与基线方法的比较

选择了一些经典基线方法，以便在分类准确率方面进行比较。图5 展示了状态向量执行模式下（平移，平均值）配置的比较；实际上，正如前文所述，对于状态向量而言，所用配置并不重要。在实际的理想情况下，基于欧几里得距离度量的量子 k最近邻算法在统计上优于采用余弦距离度量的经典 k最近邻算法（图5a）和采用线性核的支持向量机（图5d），表8证实了这一点。相比之下，量子算法的表现不及随机森林（图5b）和采用高斯核的支持向量机（图5c），不过这些差异几乎都不具有统计显著性（只有与采用高斯核的支持向量机之间的差异，对于 $k = 3$而言具有显著性）。

针对（模拟执行模式下（平移，平均值）配置的类似对比图，也就是分类准确率表现最佳的配置，可参见附录C.3（图11）。具体而言，表 k中所列的所有基线方法在统计上均优于该执行模式下的量子模拟-NN，这一点也得到了表14的证实。



(a) (b) (c) (d)

图5 经典基线方法与 *statevector* 在准确率方面的比较。 *statevector* 所使用的配置为（*translation*， *avg*），但所有配置下的结果都相同。每个点对应一个数据集折

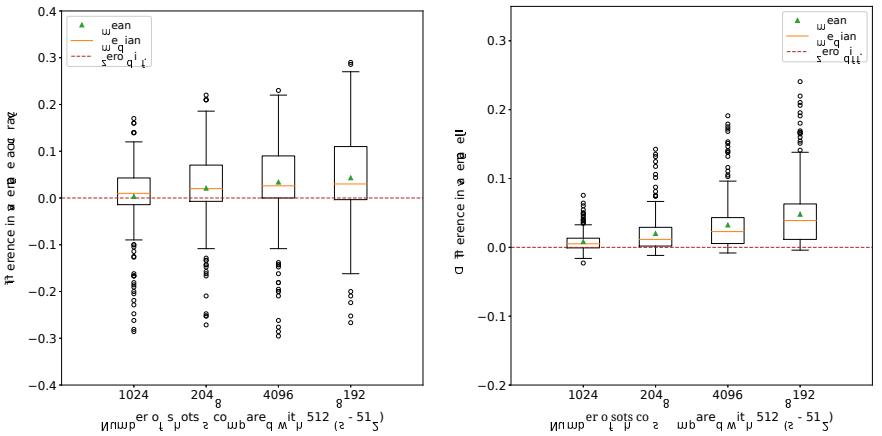
表8 对图5所示分布应用的威尔科克森符号秩检验（$\alpha = 0.05$）。表中报告的数值为所获得的 p 值

	k=3	**k=5**	**k=7**	**k=9**
图5a	0.003	0.001	6.502E-05	8.149E-05
图5b	0.074	0.165	0.095	0.258
图5c	0.046	0.407	0.103	0.062
图5d	0.045	0.005	0.012	0.007

5.4.4 测量次数分析

最后一项分析探讨了模拟执行模式下测量次数（测量）与性能之间的关系。具体而言，本次研究仅考虑表现最佳的量子 k-NN配置。由于量子 k-NN的主要目标是正确找到 k 最近邻，因此采用了（扩展，平均值）配置。事实上，如第5.4.2节所示，该配置取得了最佳杰卡德指数和平均杰卡德得分。结果以差值箱线图的形式呈现在图6 和图12中（后者见附录C.4），其中将512作为基线测量次数。

在实践中，增加测量次数通常能提升性能；对于杰卡德指数（图 6b）而言，这一趋势更加明显。



(a) (b)

图 6 在（*extension*, *avg*）配置下， *simulation* 执行模式中不同测量次数在准确率（a）和杰卡德指数（b）方面的比较。每个数据点对应一个（*dataset fold*, *k value*）对的差值。

表 9 对图 6a (a) 和图 6b (b) 所示分布应用威尔科克森符号秩检验和单样本 T 检验。表中报告的数值为所得 p 值 ($\alpha = 0.05$)

(a)

	1024-512	2048-512	4096-512	8192-512
Wilcoxon	0.001	3.131E-07	2.432E-11	6.830E-11
T 检验	0.473	5.648E-05	3.016E-08	7.904E-11

(b)

	1024-512	2048-512	4096-512	8192-512
Wilcoxon	1.151E-17	5.672E-33	4.761E-38	1.393E-40
T 检验	4.330E-17	5.712E-26	1.955E-30	1.561E-38

以及平均 Jaccard 分数（图 12）而言，尽管与准确率（图 6a）相比，其绝对值差异较小。此外，正如表 9 和表 15 （见附录 C.4）所示，几乎所有性能差异在中位数（威尔科克森符号秩检验）和均值（单样本 T 检验）方面都具有统计显著性；唯一的例外是 $1024-512$准确率均值方面的比较。最后值得强调的是，数据集越大，估算联合概率值所需的测量次数就越多。

6 结论

本文介绍了一种基于欧几里得距离度量的新型量子 k-NN 算法。具体而言，提出了两种新的输入编码方式。

已提出将数据编码为量子态振幅的不同方法，这些方法具有不同的特性且对量子比特的需求较低（这些编码方法不需要对输入数据进行单位范数归一化）。所采用的量子电路不涉及预言机，而是执行一种具有固定数量基本量子门的类 SWAP 测试过程；通过这种方式，可以并行计算与成对欧几里得距离相关的量。最后，在获得测量结果后（测量必须重复进行多次），介绍了两种不同的欧几里得距离值估算方法；最终的训练数据排序和分类均采用经典方法完成。

除了理论公式和一些复杂度观察之外，本文还提供了该算法的 Python 实现以及广泛的实证评估。首先，实验结果证实了该公式的正确性；状态向量执行模式（使用无限次测量次数进行理想执行）达到了与经典算法相同的性能。值得指出的是，该算法相较于经典对应算法的优势在于执行时间。正如预期，状态向量模式在分类准确率以及所找到的最近邻的正确性方面均优于模拟模式；后者的测量次数有限（在最近邻正确性方面，差异更为显著）。在测试的（编码，距离估计）配置中，（平移，平均值）在分类准确率方面取得了最佳结果，而（扩展，平均值）找到了最佳的最近邻。值得强调的是，在上述两种配置中，编码方式不同，而距离估计方式相同；不过，由于该算法的首要目标是找到正确的最近邻，因此（扩展，平均值）可视为整体最佳配置。就所考察的经典基线方法而言，其中一半在状态向量模式下取得了优于量子 k-NN 的结果，而模拟模式下的量子 k-NN 则始终表现较差。最后，对测量次数的分析证实，增加测量次数可以提升该算法在模拟模式下的性能。

未来的工作可以包括在不同的数据集上、以更多测量次数测试本文提出的模型，并在具有噪声特性的真实量子计算机上进行测试。

声明与声明内容

资助　本工作获得了 Q@TN 的支持。Q@TN 是特伦托大学、FBK-布鲁诺·凯斯勒基金会、INFN-国家核物理研究所和 CNR-国家研究委员会共同设立的联合实验室。此外，本工作还获得了 SERICS 项目（PE00000014）的部分支持。该项目隶属于由欧盟—下一代欧盟资助的意大利大学与研究部国家复苏与韧性计划。最后，作者衷心感谢意大利大学与研究部（MUR）。该部门依据“卓越系 2018—2022（第 232/2016 号法律）”倡议，为本研究提供了

实验中使用的计算资源。

Co利益冲突 作者声明不存在利益冲突。 t.

伦理审批 不适用。

参与知情同意 不适用。

发表知情同意 不适用。

数据和材料的可用性 支持本研究结果的数据（数据集、复现实验所需的信息以及收集到的结果数据）可在 Figshare 存储库中获取，https://doi.org/10.6084/m9.figshare.22598455.v1。此外，同一数据以及原始结果数据也可从 GitHub 存储库获取，https://github.com/ZarHenry96/euclidean-quantum-k-nn。

代码可用性 代码可在 GitHub 仓库中获取：https://github.com/ZarHenry96/euclidean-quantum-k-nn。

作者贡献 概念化：Enrico Zardini、Enrico Blanzieri、Davide Pastorello；数据整理：Enrico Zardini；形式分析：Enrico Zardini、Enrico Blanzieri、Davide Pastorello；调查：Enrico Zardini；方法：Enrico Zardini、Enrico Blanzieri、Davide Pastorello；软件：Enrico Zardini；监督：Enrico Blanzieri、Davide Pastorello；验证：Enrico Zardini；可视化：Enrico Zardini；初稿撰写：Enrico Zardini；审阅与编辑：Enrico Zardini、Enrico Blanzieri、Davide Pastorello。

附录 A 推导

与输出态相关的一些推导 $|\gamma\rangle$ （方程 3）见本附录。

A.1 测量第一个量子比特得到 1 的概率

测量状态 $|\gamma\rangle$ 中第一个量子比特得到 1 的概率为

$$
\begin{aligned}
P(1) &= \| |1\rangle \langle 1|\gamma\rangle \|^2 = \\
&= \left\| \frac{1}{2} |1\rangle \otimes \left(\frac{1}{\sqrt{2}} (|0\rangle |\alpha\rangle - |0\rangle |\beta\rangle + |1\rangle |\beta\rangle - |1\rangle |\alpha\rangle) \right) \right\|^2 = \\
&= \frac{1}{8} (\langle 0| \langle \alpha| - \langle 0| \langle \beta| + \langle 1| \langle \beta| - \langle 1| \langle \alpha|) \times \\
&\qquad \times (|0\rangle |\alpha\rangle - |0\rangle |\beta\rangle + |1\rangle |\beta\rangle - |1\rangle |\alpha\rangle) =
\end{aligned}
$$

$$= \frac{1}{8}(1 - \langle\alpha|\beta\rangle - \langle\beta|\alpha\rangle + 1 + 1 - \langle\beta|\alpha\rangle - \langle\alpha|\beta\rangle + 1) =$$
$$= \frac{1}{8}(4 - 2\langle\alpha|\beta\rangle - 2\langle\beta|\alpha\rangle) = \qquad (|\alpha\rangle \text{ and } |\beta\rangle \text{ have real coefficients})$$
$$= \frac{1}{8}(4 - 4\langle\alpha|\beta\rangle) =$$
$$= \frac{1}{2}(1 - \langle\alpha|\beta\rangle).$$

A.2 约化后的最终状态

给定 $|\gamma\rangle$，我们先将索引寄存器 $|j\rangle$ 内部的求和项$|\alpha\rangle$ 和 $|\beta\rangle$提取出来。这样，我们得到如下形式的状态

$$\frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} [...] |j\rangle ,$$

其中，$[...]$ 包括除属于索引寄存器的量子电路量子比特之外的所有量子电路量子比特。接下来，对量子电路中的第二个量子比特和特征寄存器 $|i\rangle$进行偏迹，即将它们舍弃；从数学角度来看，这相当于对描述该系统的密度算符关于这些量子比特计算偏迹。这样，我们便得到最终状态的一个约化版本，其中仅包含量子电路中的第一个量子比特和索引寄存器，可写为

$$\frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \left[\sqrt{P(0 \mid j)}\,|0\rangle + \sqrt{P(1 \mid j)}\,|1\rangle \right] |j\rangle .$$

最后，我们利用附录 A.1 中给出的推导。实际上，如公式（5）所示，P（$1 \mid j$）等于 $\frac{1}{2}(1-\langle \mathbf{x}j, \mathbf{x}'_j\rangle)$，因为索引寄存器上的求和（及其系数）已被提取出来。此外，根据全概率定律，P（$0 \mid j$）必定等于 $1 - P$（$1 \mid j$）。由此得到约化后的最终状态的定义$|\delta\rangle$，见公式（4）。

附录B 距离估计

在本附录中，我们提供有关平均值和差值距离估计的一些补充信息。具体而言，考虑两个预处理后的训练实例 $\mathbf{v}_{j_1}$ and $\mathbf{v}_{j_2}$，其值为 j_1、$j_2 \in \{0, ..., N-1\}$，以及一个预处理后的测试实例 $\mathbf{v}'$。设 $d_0(\mathbf{v}_{j_1}, \mathbf{v}')$ 和 $d_1(\mathbf{v}_{j_1}, \mathbf{v}')$ 分别为根据 $\mathbf{v}'$ 的联合概率 $P(0, j_1)$ 和 $P(1, j_1)$ 估算得到的欧几里得距离（$\mathbf{v}_{j_1}$ 的情况类似）。那么，以下关系成立：

$$avg = \frac{d_0(\mathbf{v}_{j_1}, \mathbf{v}') + d_1(\mathbf{v}_{j_1}, \mathbf{v}')}{2},$$

$$diff = \sqrt{\frac{d_0(\mathbf{v}_{j_1}, \mathbf{v}')^2 + d_1(\mathbf{v}_{j_1}, \mathbf{v}')^2}{2}}.$$

前者只是平均值距离估计的定义，而后者则可通过结合方程（8）（或9，取决于所选的编码）以及方程（6）和（7）轻松验证。

B.1 实例排序

根据平均距离估计和差异距离估计，$\mathbf{v}_{j_1}$ 和 $\mathbf{v}_{j_2}$ 的排序可能不同。具体来说，考虑以下情形：

$$\begin{aligned} d_0(\mathbf{v}_{j_1}, \mathbf{v}') &= 0.5 & \qquad d_0(\mathbf{v}_{j_2}, \mathbf{v}') &= 0.4 \\ d_1(\mathbf{v}_{j_1}, \mathbf{v}') &= 0.29 & \qquad d_1(\mathbf{v}_{j_2}, \mathbf{v}') &= 0.4\,. \end{aligned}$$

在这种情况下，$\mathbf{v}_{j_1}$ 和 $\mathbf{v}_{j_2}$ 的平均距离估计分别为 0.395 和 0.4，而差异距离估计分别为 0.409 和 0.4。因此，根据平均距离估计，$\mathbf{v}_{j_1}$ 比 $\mathbf{v}_{j_2}$ 更接近 $\mathbf{v}'$；而根据差异距离估计，$\mathbf{v}_{j_1}$ 则比它更远。

B.2 幅值

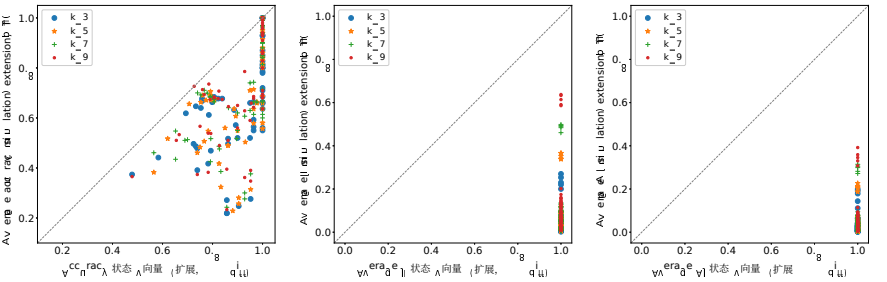
假设 $d_0(\mathbf{v}_{j_1}, \mathbf{v}')$ 和 $d_1(\mathbf{v}_{j_1}, \mathbf{v}')$ 在数学上可计算，即公式（8）（或根据所选编码方式而定的公式（9））中平方根内的参数均为非负数。那么，平均距离估计始终小于或等于相应的差异距离估计。事实上，当且仅当 $(d_0(\mathbf{v}_{j_1}, \mathbf{v}') - d_1(\mathbf{v}_{j_1}, \mathbf{v}'))^2 < 0$ 时，情况才会相反，而这是不可能的。

如果由于所得状态计数分布而不满足关于平方根参数的假设，那么平均值 距离估计可能会高于相应的差值 估计。事实上，在此处提供的实现中，负值的平方根被近似为零，正如第 4节所述。

附录C 其他图表

本附录提供了更多结果图和相关统计显著性表格。

C.1 执行模式比较

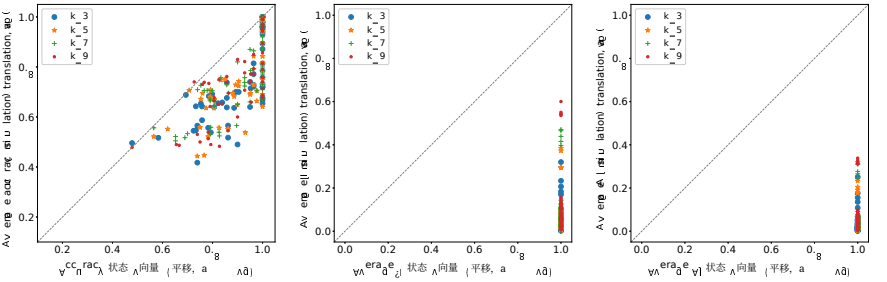


(a) (b) (c)

图 7 从准确率 (a)、杰卡德指数 (b) 和平均杰卡德得分 (c) 三个方面比较 *statevector*(*extension*, *diff*) 和 *simulation*(*extension*, *diff*)。 *simulation* 的测量次数为 1024，每个点对应一个数据集折。

表 10 对图 7 所示分布应用的威尔科克森符号秩检验（$\alpha = 0.05$）。表中报告的数值是获得的 p 值

	k=3	k=5	k=7	k=9
图 7a	1.106E-10	1.106E-10	7.530E-11	1.719E-10
图 7b	1.628E-11	1.629E-11	1.630E-11	1.630E-11
图 7c	1.630E-11	1.630E-11	1.630E-11	1.630E-11



(a) (b) (c)

图8 *statevector*(*translation*, *avg*) 与 *simulation*(*translation*, *avg*) 在准确率 (a)、杰卡德指数 (b) 和平均杰卡德得分 (c) 方面的比较。 *simulation* 的测量次数为 1024，每个点对应一个数据集折

表 11 对图8所示分布应用的威尔科克森符号秩检验 ($\alpha = 0.05$)。表中报告的数值为所得的 p 值

	k=3	k=5	k=7	k=9
图8a	1.234E-10	3.492E-10	3.492E-10	6.149E-10
图8b	1.628E-11	1.630E-11	1.630E-11	1.630E-11
图8c	1.630E-11	1.630E-11	1.630E-11	1.630E-11

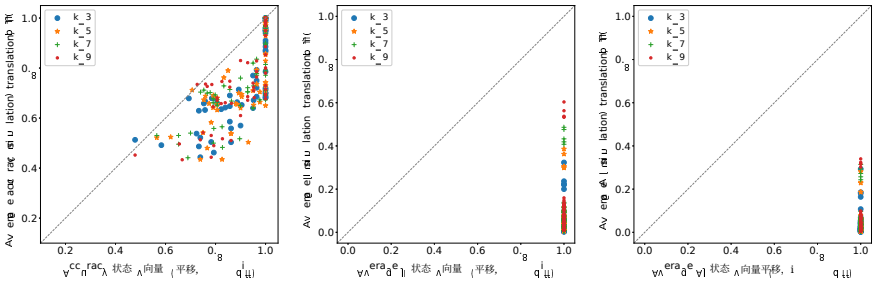


图 9 *statevector*(*translation*, *diff*) 与 *simulation*(*translation*, *diff*) 在准确率 (a)、杰卡德指数 (b) 和平均杰卡德得分 (c) 方面的比较。*simulation* 的测量次数为 1024，每个点对应一个数据集折

表 12 对所示分布应用威尔科克森符号秩检验 ($\alpha = 0.05$) in 图 9。表中报告的数值为得到的 p 值

	k=3	k=5	k=7	k=9
图9a	1.379E-10	1.712E-10	7.516E-11	2.523E-10
图9b	1.627E-11	1.629E-11	1.630E-11	1.630E-11
图9c	1.629E-11	1.630E-11	1.630E-11	1.630E-11

C.2 编码与距离估计比较

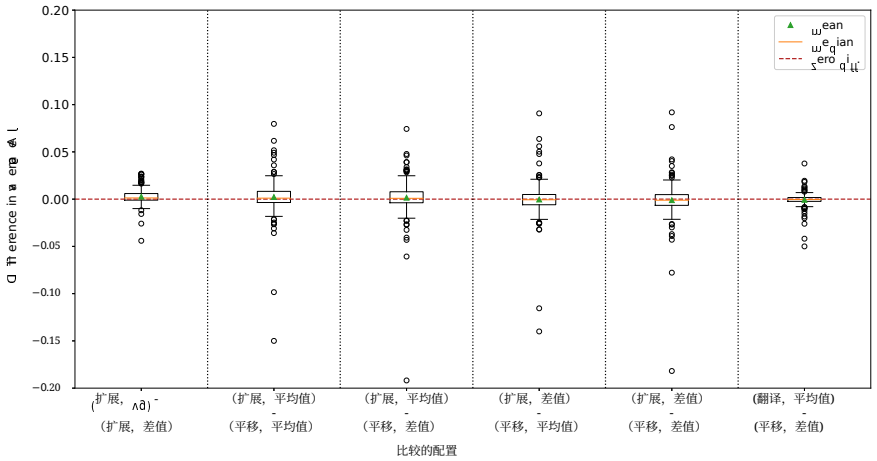


图 10： *simulation* 执行模式下各 (*encoding*, *distance estimate*) 配置的平均杰卡德得分比较。测量次数为 1024，每个数据点对应一组 (*dataset fold*, *k value*) 配对的差值

表 13 对图 10 所示的分布进行威尔科克森符号秩检验和单样本 T 检验。每列对应一项不同的比较，第一个字母表示编码（E=*extension*、T=*translation*），第二个字母表示距离估计（A=*avg*、D=*diff*）。表中报告的数值为所得 p 值（$\alpha = 0.05$）

	EA-ED	EA-TA	EA-TD	ED-TA	ED-TD	TA-TD	
Wilcoxon	1.556E-07	0.002	0.008	0.315	0.111	0.043	
T 检验	6.693E-07	0.057	0.202	0.735	0.374	0.123	

C.3 与基线方法的比较

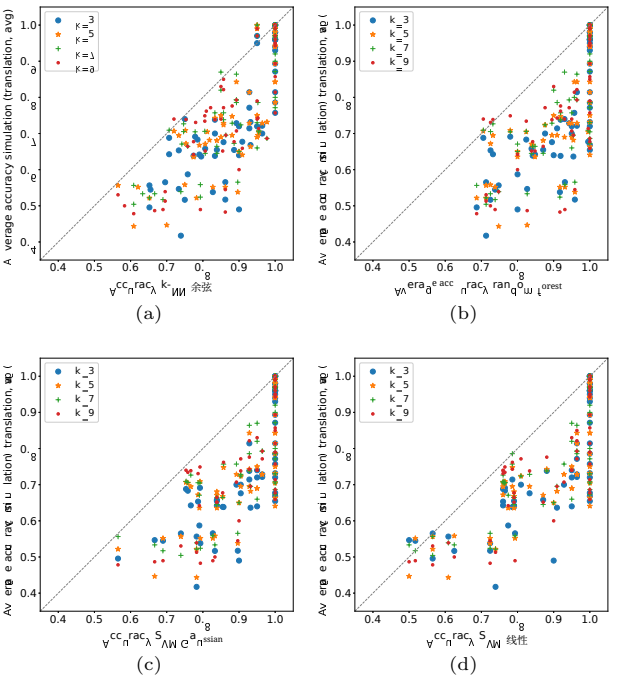


图 11 经典基线方法与 *simulation* (*translation*, *avg*) 在准确率方面的比较。*simulation* 的测量次数为 1024，每个点对应一个数据集折

表 14 对图 11 所示分布进行威尔科克森符号秩检验（$\alpha = 0.05$）得到的结果。表中报告的数值为所得 p 值。

	k=3	**k=5**	**k=7**	**k=9**
图11a	2.147E-10	5.660E-10	8.949E-10	1.210E-09
图11b	1.105E-10	2.521E-10	3.705E-10	6.632E-10
图 11c	1.105E-10	2.380E-10	3.494E-10	3.500E-10
图 11d	3.172E-10	3.976E-10	1.299E-09	3.502E-10

C.4 测量次数分析

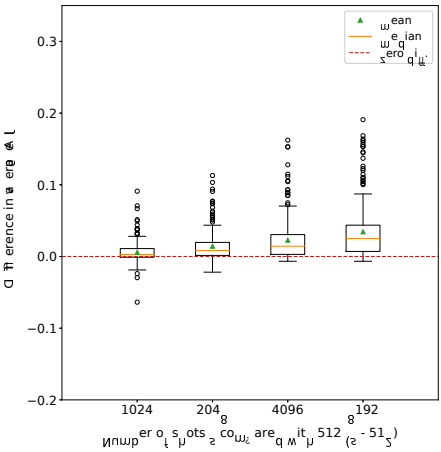


图12 在（*extension*， *avg*）配置下， *simulation* 执行模式中不同测量次数的平均杰卡德得分比较。每个数据点对应一组（*dataset fold*， *k value*）的差值

表 15 对图 12 所示分布进行的威尔科克森符号秩检验和单样本 T 检验。表中报告的值为所获得的 p 值（$\alpha = 0.05$）

	1024-512	2048-512	4096-512	8192-512
Wilcoxon	1.977E-11	8.804E-30	2.327E-37	1.030E-39
T 检验	3.037E-09	6.719E-23	1.862E-27	3.622E-33

参考文献

Abbas A、Sutter D、Zoufal C 等（2021）量子神经网络的力量。《自然计算科学》，1：403–409。https://doi.org/10.1038/s43588-021-00084-1，网址 https://www.nature.com/articles/s43588-021-00084-1#citeas

Afham A、BasheerA、Goyal SK（2020年）量子 k 近邻机器学习算法。网址 https://arxiv.org/abs/2003.09187v1

Anis MS、Abraham H、AduOffei 等（2021）Qiskit：一个用于量子计算的开源框架。https://doi.org/10.5281/zenodo.2573505

Basheer A、Afham A、Goyal SK（2021）量子 *k*-最近邻算法。网址 https://arxiv.org/abs/2003.09187

Biamonte J、Wittek P、Pancotti N 等（2017）量子机器学习。《Nature》549：195–202。 https://doi.org/10.1038/nature23474，网址 https://doi.org/10.1038/nature23474

Brassard G、Hoyer P、Mosca M 等（2002）量子振幅放大与估计。《Contemporary Mathematics》305：53–74

Buhrman H、Cleve R、Watrous J 等（2001）量子指纹识别。物理评论快报 87:167,902。https://doi.org/10.1103/PhysRevLett.87.167902, URLhttps://link.aps.org/doi/10.1103/PhysRevLett.87.167902

Cleve R, Ekert A, Macchiavello C, 等（1998）重新审视量子算法。《伦敦皇家学会会刊 A 辑：数学、物理与工程科学》454(1969):339–354。https://doi.org/10.1098/rspa.1998.0164, URLhttp://dx.doi.org/10.1098/rspa.1998.0164Dang Y, Jiang N, Hu H, 等（2018）基于量子 K 近邻算法的图像分类。《量子信息处理》17(9):239。https://doi.org/10.1007/s11128-018-2004-9, URLhttps://doi.org/10.1007/s11128-018-2004-9Dua D, Graff C（2017）UCI机器学习库。URLhttp://archive.ics.uci.edu/mlD¨urr C, Høyer P（1999）一种寻找最小值的量子算法。URL https://arxiv.org/abs/quant-ph/9607014Fastovets DV, Bogdanov YI, Bantysh BI, 等（2019）量子计算理论中的机器学习方法。载于：2018年微纳电子国际会议，国际光学与光子学学会，第 11022 卷。SPIE，俄罗斯兹韦尼哥罗德，第 752–761 页，URLhttps://doi.org/10.1117/12.2522427Feng C, Zhao B, Zhou X, 等（2023）一种基于极距离的增强型量子 K 近邻分类算法。《熵》25(1)。https://doi.org/10.3390/e25010127, URLhttps://www.mdpi.com/1099-4300/25/1/127Fix E, Hodges JL（1951年）《判别分析，非参数判别：一致性性质》。技术报告第 4 号，美国空军航空医学学校，兰道夫机场。Gao LZ, Lu CY, Guo GD, 等（2022年）基于马哈拉诺比斯距离的量子 K 近邻分类算法。《物理学前沿》10。https://doi.org/10.3389/fphy.2022.1047466, U R L https://www.frontiersin.org/articles/10.3389/fphy.2022.1047466Getachew AT（2020年）使用并行欧几里得距离估算器的量子 K 中位数算法。https://doi.org/10.48550/ARXIV.2012.11139, URLhttps://arxiv.org/abs/2012.11139

Giovannetti V, Lloyd S, Maccone L（2008）量子随机存取存储器。《物理评论快报》100:160,501。https://doi.org/10.1103/PhysRevLett.100.160501，URLhttps://link.aps.org/doi/10.1103/PhysRevLett.100.160501

Gosset WS（1908年）均值的可能误差。Biometrika 6(1):1–25。网址 http://dx.doi.org/10.2307/2331554，最初以笔名“Student”发表。

Greene D、O’Callaghan D、Cunningham P（2014）主题模型有多少个主题？主题模型的稳定性分析。载于：Calders T、Esposito F、H¨ullermeier E 等（编）《数据库中的机器学习与知识发现》。Springer Berlin Heidelberg，柏林，海德堡，第498–513页

GroverLK（1996）一种用于数据库搜索的快速量子力学算法。载于：第二十八届 ACM 计算理论年度研讨会论文集。计算机协会，纽约，美国，STOC ’96，第212–219页，https://doi.org/10.1145/237814.237866，网址 https://doi.org/10.1145/237814.237866

Kaye P（2004）使用一个辅助比特的可逆加法电路及其在量子计算中的应用。https://doi.org/10.48550/ARXIV.QUANT-PH/0408173，网址 https://arxiv.org/abs/quant-ph/0408173Li J、Lin S、Yu K 等（2021）基于汉明距离的量子 K-最近邻分类算法。《量子信息处理》21（1）：18。https://doi.org/10.1007/s11128-021-03361-0，网址 https://doi.org/10.1007/s11128-021-03361-0Lloyd S、Mohseni M、Rebentrost P（2013）用于监督式和非监督式机器学习的量子算法。https://doi.org/10.48550/ARXIV.1307.0411，网址 https://arxiv.org/abs/1307.0411Ma Yz、Song Hf、Zhang J（2021）基于类别张量网络状态的量子 K-最近邻分类算法。《国际理论物理学杂志》60（3）：1164–1174。https://doi.org/10.1007/s10773-021-04742-y，网址 https://doi.org/10.1007/s10773-021-04742-yMitarai K、Kitagawa M、Fujii K（2019）量子模拟-数字转换。《物理评论 A》99：012,301。https://doi.org/10.1103/PhysRevA.99.012301，网址 https://link.aps.org/doi/10.1103/PhysRevA.99.012301Miyamoto K、Iwamura M、Kise K（2019）寻找 k-最小值的量子算法。https://doi.org/10.48550/ARXIV.1907.03315，网址

https://arxiv.org/abs/1907.03315

Patrıcio M、Pereira J、Cris´ostomo J 等（2018）利用抵抗素、葡萄糖、年龄和身体质量指数预测乳腺癌的存在。《BMC Cancer》18（1）：29。
https://doi.org/10.1186/s12885-017-3877-1，网址
https://doi.org/10.1186/s12885-017-3877-1

Quezada LF、Sun GH、Dong SH（2022年）基于量子排序算法的 k 近邻分类器的量子版本。《物理学年鉴》534(5):2100,449。
https://doi.org/https://doi.org/10.1002/andp.202100449，网址
https://onlinelibrary.wiley.com/doi/abs/10.1002/andp.202100449

Rebentrost P、Mohseni M、Lloyd S（2014）用于大数据分类的量子支持向量机。《物理评论快报》113:130,503。https://doi.org/10.1103/PhysRevLett.113.130503，网址https://link.aps.org/doi/10.1103/PhysRevLett.113.130503

Rebentrost P、Steffens A、Marvian I 等（2018年）非稀疏低秩矩阵的量子奇异值分解。《物理评论 A》97:012,327。https://doi.org/10.1103/PhysRevA.97.012327，网址https://link.aps.org/doi/10.1103/PhysRevA.97.012327

Ruan Y, Xue X, Liu H 等（2017）基于汉明距离度量的 K 近邻分类量子算法。《国际理论物理学杂志》56(11):3496–3507。
https://doi.org/10.1007/s10773-017-3514-4，
URLhttps://doi.org/10.1007/s10773-017-3514-4

Sarma A、Chatterjee R、Gili K 等（2020年）超导处理器上的量子无监督与监督学习。《量子信息与计算》20(7–8):541–552。https://doi.org/10.26421/QIC20.7-8-1

Schuld M、Sinayskiy I、Petruccione F（2014）用于模式分类的量子计算。载于：Pham DN、Park SB（编）《PRICAI 2014：人工智能趋势》。Springer International Publishing，Cham，第208–220页

Schuld M, Fingerhuth M, Petruccione F（2017）利用量子干涉电路实现基于距离的分类器。Europhysics Letters 119（6）：60,002。
https://doi.org/10.1209/0295-5075/119/60002，网址
https://dx.doi.org/10.1209/0295-5075/119/60002

Silva PFB、Mar¸cal ARS、da Silva RMA（2013）叶片区分特征评估。Springer《计算机科学讲义》7950：197–204

TrugenbergerCA（2002）量子模式识别。《量子信息处理》1(6):471–493。
https://doi.org/10.1023/A:1024022632303，
URLhttps://doi.org/10.1023/A:1024022632303

Wang Y, Wang R, Li D 等（2019年）使用量子 K 近邻算法改进手写数字识别。《国际理论物理学杂志》58(7):2331–2340。
https://doi.org/10.1007/s10773-019-04124-5,
URLhttps://doi.org/10.1007/s10773-019-04124-5

Wiebe N、Kapoor A、Svore KM（2015年）用于监督和非监督学习的近邻方法量子算法。《量子信息与计算》15(3–4)：316–356。
https://doi.org/10.26421/QIC15.3-4-7

Wilcoxon F（1945年）通过排序方法进行个体比较。《生物统计学通报》1(6)：80–83。网址 http://www.jstor.org/stable/3001968

Wilson EB（1927年）概率推断、继承定律与统计推断。《美国统计学会杂志》22(158)：209–212。 https://doi.org/10.1080/01621459.1927.10502953，网址 https://www.tandfonline.com/doi/abs/10.1080/01621459.1927.10502953

Wi´sniewska J、Sawerwain M（2018年）通过量子 kNN 和 SVM 方法识别二元厄米矩阵的模式。《越南计算机科学杂志》5(3):197–204。
https://doi.org/10.1007/s40595-018-0115-y，网址
https://doi.org/10.1007/s40595-018-0115-yYeh IC、Yang KJ、Ting TM（2009年）使用伯努利序列对 RFM 模型进行知识发现。《专家系统与应用》36(3):5866–5871。https://doi.org/10.1016/j.eswa.2008.07.018，网址 https://doi.org/10.1016/j.eswa.2008.07.018Yu K、Guo GD、Li J 等（2020年）基于欧几里得距离的相似度测量量子算法。《国际理论物理学杂志》59(10):3134–3144。https://doi.org/10.1007/s10773-020-04567-1，网址 https://doi.org/10.1007/s10773-020-04567-1Zardini E（2023年（a））量子机器学习流水线数据集。
https://doi.org/10.6084/m9.figshare.22333102.v1，网址
https://figshare.com/articles/dataset/QMLPipeline Datasets/22333102Zardini E（2023年（b））量子机器学习流水线原始结果。Figsharehttps://doi.org/10.6084/m9.figshare.22333147.v1，网址
https://figshare.com/articles/dataset/QMLPipeline Raw Results/22333147Zardini E、Blanzieri E、Pastorello D（2022年）量子机器学习流水线的实现与实证评估：用于局部分类。
https://doi.org/10.48550/ARXIV.2205.05333，网址

https://arxiv.org/abs/2205.05333

Zhou NR、Liu XX、Chen YL 等（2021年）基于 K-L 变换的量子 K 近邻图像分类算法。《国际理论物理学杂志》60(3):1209–1224。https://doi.org/10.1007/s10773-021-04747-7，网址 https://doi.org/10.1007/s10773-021-04747-7