arXiv:2303.06316v1 [cs.LG] 11 Mar 2023

省下一颗神经元，就赚得一颗神经元：论二次网络的参数效率

Feng-Lei Fan，电气电子工程师学会会员，Hang-Cheng Dong，Zhongming Wu，Lecheng Ruan，Tieyong Zeng，Yiming Cui，Jing-Xiao Liao

摘要—受生物神经系统中神经元多样性的启发，许多研究提出设计新型人工神经元，并将神经元多样性引入人工神经网络。近期提出的二次神经元以二次运算取代传统神经元中的内积运算，已在许多重要任务中取得巨大成功。尽管二次神经元展现出令人振奋的结果，但仍有一个尚未解决的问题：二次网络的优越性能仅仅源于参数量增加，还是源于其内在表达能力？若不澄清这一问题，二次网络的性能就始终令人存疑。此外，解决这一问题就归结为寻找二次网络的杀手级应用。本文通过理论与实证研究表明，二次网络具有参数效率，从而证实其优越性能源于内在表达能力。这种内在表达能力源自二次神经元能够轻松表示非线性交互，而传统神经元则难以做到。理论上，我们从实空间和流形的角度推导了二次网络相较于传统网络的逼近效率。此外，从 Barron空间的角度，我们证明存在一个函数空间，其中的函数可由二次网络以维度无关的误差进行逼近，而传统网络的逼近误差则依赖于维度。实证方面，在合成数据、经典基准和现实世界应用上的实验结果表明，二次模型普遍具有参数效率，且效率提升幅度取决于任务。我们的代码已公开于 https://github.com/asdvfghg/quadratic efficiency。

索引词——神经元多样性、二次神经元、基于二次神经元的深度学习、效率

1 引言

大脑是迄今为止我们所知最聪明的系统。从历史上看，神经科学一直是人工网络研究的重要灵感来源，例如，人工神经元的建模 [1] 以及新认知机的发明 [2]，，后者是卷积模型的开创性工作。近期，人们提出了“NeuroAI”[3] 这一概念。尽管人工网络在受到神经科学启发后通常会走上自己的发展道路，但大量神经科学知识仍能有力地支撑深度学习的发展。由此可见，我们的大脑由数量众多、形态和功能各异的神经元组成 [4]。毫不夸张地说，神经元多样性是各种生物智能行为的重要因素。然而，在过去十年中，大多数深度学习研究都致力于通过设计新颖的架构来构建强大的网络模型，即快捷连接设计 [5], [6], [7]。这些成功的网络几乎完全建立在同一种神经元之上，这种神经元由内积和非线性激活组成，因此缺乏神经元多样性。为方便起见，我们将这类神经元称为常神经元，而由这些神经元构成的网络称为传统网络。我们感兴趣的是，既然人工网络是生物神经网络的微缩模型，那么赋予人工网络神经元多样性的作用和潜在收益就应当得到审慎考察。

受神经元多样性这一理念的启发，我们团队此前试制了一种称为二次神经元的新型神经元，它以简化的二次函数取代传统神经元中的内积。下文将基于二次神经元构建的网络称为二次网络。Fan等人证明，单个二次神经元能够执行异或逻辑运算，而传统神经元则不能 [8]。此外，样条理论 [9] 也展示了二次网络卓越的表达能力：以ReLU作为激活函数时，传统网络是分段线性函数，而二次网络是分段多项式。根据样条理论，分段多项式样条的逼近误差通常低于分段线性函数，因此二次网络具有更强的表达能力。与此同时，研究者提出了ReLinear算法，以解决二次网络的训练不稳定性问题 [9]。该算法将二次网络初始化为传统网络，并控制二次项的学习率。随后，二次网络被广泛用于解决现实问题，例如低剂量CT去噪 [10] 和轴承故障诊断[11]。这一系列关于二次神经元的研究旨在基于新型神经元发展深度学习，并验证将神经元多样性引入深度学习的潜力。尽管取得了相当不俗的成果，

• *Feng-Lei Fan*、*Zhong-Ming Wu* 和 *Tieyong Zeng* 任职于香港沙田香港中文大学数学系。
• *Jing-Xiao Liao* 和 *Hang-Cheng Dong* 任职于哈尔滨工业大学仪器科学与工程学院，地址为中国黑龙江省哈尔滨市，邮编 *150001*。
• *Yiming Cui* 任职于美国佛罗里达州盖恩斯维尔佛罗里达大学电气与计算机工程系，邮编 *32611*。
• *Lecheng Ruan* 任职于通用人工智能全国重点实验室、北京通用人工智能研究院（*BIGAI*），地址为中国北京，邮编 *100080*。

二次网络的参数效率问题仍未得到令人满意的解决。二次网络的优越性能究竟仅仅源于参数量增加，还是源于其内在的表达能力？这一问题对于二次网络以及其他类型的多项式网络的发展都至关重要。若不澄清这一点，二次网络的性能就始终令人存疑。此外，合理假设二次网络并非在所有任务上都比传统网络更具参数效率后，回答这一问题实际上就归结为寻找二次网络的杀手级应用，也就是说，找出最适合二次网络的任务，使其能够用少得多的参数实现更高的性能。我们早期的理论工作证明，存在一类函数，可以用多项式数量的二次神经元进行逼近，但若要达到相同的误差水平，则必须用指数数量的传统神经元进行逼近 [12]。然而，这一结果既不具普遍性，也不切实际，也就是说，它仅从理论上构造了一类非常特殊的函数，对实际中的参数效率问题并无启示。

鉴于参数效率的重要性，我们从理论和实证两个角度对其进行了系统而深入的阐述。我们证明了二次网络具有参数效率，从而证实二次网络的优越性能源于其内在表达能力，而非参数量的增加。这种内在表达能力源于二次神经元能够轻松表示非线性交互，而传统神经元则难以做到。理论方面，1）Yarotsky 早先构造了传统 ReLU 网络，用于逼近 Sobolev空间 [13]中的函数，并为网络的结构和参数建立了紧确的误差界。Chen 等人修改了这一分析方案，用于逼近低维流形 [14]上的函数。两种方案均达到最优误差率[15]。类似地，我们采用相同方案，分别构造二次ReLU网络来逼近Sobolov空间和流形上的函数。由于二次神经元中包含交互项，二次神经元无需堆叠参数就能轻松表示非线性交互，而传统神经元则难以表示这种非线性交互。因此，定理1和定理2表明，在逼近误差相同的情况下，所构造的二次网络所需参数量比相应的传统网络少一个数量级。2）从 Barron空间的角度来看，我们证明存在一个函数空间，其中的函数可以通过二次网络进行逼近，且不受维度灾难的影响；而传统网络的逼近误差则取决于维度（定理5）。这种差异是因为新增的二次运算扩展了 Barron空间，而非增加了更多参数。此外，反过来的情形并不成立，因为传统网络可以视为二次网络的退化情形。

通过对合成数据、经典基准和现实世界应用开展系统实验，我们从实证上验证了二次网络相较于其他具有竞争力的基线方法具有参数效率。实验还证实，在对象具有显著圆形特征的细胞和叶片分割任务中，二次网络具有很高的效率。因此，我们认为二次网络可以作为这类任务的优先模型。此外，尽管我们着重强调的是参数效率，但在某些任务中，二次模型也减少了浮点运算次数，从而体现出计算效率。总而言之，我们的贡献有两点：

- 通过对索博列夫空间、流形和 Barron空间上的函数进行逼近，我们证明了三个极具挑战性的定理。这些定理表明，二次网络在广义意义上应具有参数效率；二次网络的优越性能源于其内在的非线性表示能力，而非参数量的增加。

- 通过系统的实验，我们证实，在许多重要任务中，二次网络比传统网络更高效。此外，我们发现，对于圆形特征显著的任务，例如细胞和叶片分割任务，二次网络可能是优先选择。

2 相关工作

多项式神经元和二次神经元。20世纪70年代，I vakhnenko提出了所谓的数据处理组合法（GMDH） [16] ，该方法使用高阶多项式作为特征提取器。后来，Shin 等人提出了一种称为 π-σ 网络的高阶网络 [17]:

$$y_i = \sigma(\prod_j (\sum_k w_{kji} x_k + \theta_{ji})), \tag{1}$$

其中， y_i 是第 i 个输出， $\sigma(\cdot)$ 是非线性激活函数， w_{kji} 和 θ_{ji} 是第 j 个单元相对于输入 x_k的权重。π-σ 网络本质上是一种特殊的多项式网络。为提高多项式网络的参数效率，Milenkovic等人通过直接移除三次及更高阶项，提出了二阶神经网络 [18]。

近年来，高阶神经元被重新审视，并被引入全连接网络和卷积网络中。在 [19] 和 [20],中，通过张量分解和因子共享，大幅降低了高阶单元的复杂度。更多研究关注二次神经元，它们直接舍弃高阶项，并使用矩阵分解实现参数效率。我们在表 1 中总结了近期提出的二次神经元。可以看出，[21],[22], [23], [24] 中的神经元复杂度为 $\mathcal{O}(n^2)$，其中 n 是输入的维度，这对于构建深度网络而言仍然过高。此外，[25], [26], [27]中的神经元是我们提出的神经元 [8]的特例。

多项式网络的进展。在理论方面，一方面，研究者探讨了多项式网络的表达能力。借助代数几何，Kileel 等人利用代数簇的维度来刻画多项式网络的表达能力 [28]。Liao 等人证明，存在这样一个函数：由传统型神经元和二次神经元组成的异构网络，只需多项式数量的神经元即可近似该函数；而同质的传统型网络或

表1 最近提出的二次神经元概要。$\sigma(\cdot)$ 表示非线性激活函数。$\odot$ 表示 Hadamard 积。此处，$\mathbf{W} \in \mathbb{R}^{n\times n}$、$\mathbf{w}_i \in \mathbb{R}^{n\times 1}$，并且我们省略了这些神经元中的偏置项。

作者	公式
Zoumpourlis等人（2017年） [21], [22]	$\mathbf{y} = \sigma(\mathbf{x}^\top \mathbf{W}\mathbf{x} + \mathbf{w}^\top \mathbf{x})$
Jiang等(2019年) [23]	$\mathbf{y} = \sigma(\mathbf{x}^\top \mathbf{W}\mathbf{x})$
Mantini&Shah(2021年) [24]	
Goyal 等(2020年) [25]	$\mathbf{y} = \sigma(\mathbf{w}^\top(\mathbf{x} \odot \mathbf{x})$
Bu&Karpatne(2021年) [26]	$\mathbf{y} = \sigma((\mathbf{w}_1^\top \mathbf{x})(\mathbf{w}_2^\top \mathbf{x}))$
Xu 等（2022年） [27]	$\mathbf{y} = \sigma((\mathbf{w}_1^\top \mathbf{x})(\mathbf{w}_2^\top \mathbf{x}) + \mathbf{w}_3^\top \mathbf{x})$
Fan 等（2018年） [8]	$\mathbf{y} = \sigma((\mathbf{w}_1^\top \mathbf{x})(\mathbf{w}_2^\top \mathbf{x}) + \mathbf{w}_3^\top (\mathbf{x} \odot \mathbf{x}))$

二次网络则需要指数数量的神经元才能达到相同的误差水平 [29]。另一方面，研究者关注了多项式网络的鲁棒性验证。基于经典分支定界技术的现有传统型网络验证算法无法直接扩展到多项式网络。Rocamora 等人 [30] 利用多项式网络的二阶可微性，构建了一种适用于分支定界技术的下界方法，从而实现了对多项式网络鲁棒性的完备验证。

此外，越来越多的研究尝试将多项式网络应用于现实世界应用。Fan 等提出了一种用于低剂量 CT 去噪的二次自编码器 [10]。Nguyen等使用二次网络预测泡沫混凝土的抗压强度 [31]。Bu 等 [26] 探索了如何利用二次网络求解偏微分方程中的正问题与逆问题。Qi 和 Wang 展示了二次网络在高斯混合数据分类方面的优越性 [32]。Xu 等构建了一个具有算法加速功能的二次网络库（QuadraLib），以促进二次网络的应用研究 [27]。Liao 等将二次网络应用于一维振动信号的轴承故障诊断，并发现了二次神经元中固有的注意力机制（*qttention*） [11]。Chrysos等研究了多项式网络的架构，并在图像分类和生成方面取得了最先进的性能 [33]。然而，尽管二次模型在众多任务中取得了成功，究竟哪些应用最适合二次模型仍不清楚。

3 效率理论

3.1 预备知识

从数学上讲，给定 d 维输入 $\mathbf{x} \in \mathbb{R}^{d\times 1}$，传统型神经元先计算输入与神经元中权重向量$\mathbf{w}$ 的内积，然后应用非线性激活函数 $\sigma(\cdot)$:

$$y = \sigma(\mathbf{w}^\top \mathbf{x} + b). \tag{2}$$

相比之下，二次神经元用简化的二次函数取代内积:

$$y = \sigma((\mathbf{w}_1^\top \mathbf{x} + b_1)(\mathbf{w}_2^\top \mathbf{x} + b_2) + \mathbf{w}_3^\top (\mathbf{x} \odot \mathbf{x}) + b_3), \tag{3}$$

其中 $\mathbf{w}_1$,$\mathbf{w}_2, \mathbf{w}_3$ 是权重向量， b_1, b_2, b_3 是偏置，而 $\odot$是 Hadamard 积。除非另有说明，本文始终采用 ReLU 激活函数，该函数广泛用于神经网络。

3.2 效率：Sobolov S 上的函数 pace

定义 1 (索博列夫空间)**.** 设 $\mathcal{W}^{n,\infty}([0,1]^d)$ 为 *Sobolev* 空间，它由定义在 $[0,1]^d$ 上的函数构成，这些函数属于L^∞，且其弱导数的阶数不超过 n。函数 f 在 $\mathcal{W}^{n,\infty}([0,1]^d)$ 中的范数为

$$\|f\|_{\mathcal{W}^{n,\infty}([0,1]^d)} = \max_{\mathbf{n}:|\mathbf{n}|\le n} \operatorname{ess\,sup}_{\mathbf{x}\in[0,1]^d} \left|D^{\mathbf{n}} f(\mathbf{x})\right|, \tag{4}$$

其中 $\mathbf{n} = (n_1, n_2, \ldots, n_d) \in \{0, 1, \ldots, n\}^d$, $|\mathbf{n}| = n_1 + n_2 + \cdots + n_d \le n$，而 $D^{\mathbf{n}} f$ 是 n 阶弱导数。本质上，空间 $\mathcal{W}^{n,\infty}([0,1]^d)$ 是 $C^{n-1}([0,1]^d)$，其中函数截至 n 阶的导数均为利普希茨连续。

定义2 （函数空间）。 设 $F_{n,d}$ 是位于 $\mathcal{W}^{n,\infty}([0,1]^d)$ 中单位球内的一组函数，则有

$$F_{n,d} = \{f \in \mathcal{W}^{n,\infty}([0,1]^d) : \|f\|_{\mathcal{W}^{n,\infty}([0,1]^d)} \le 1\}. \tag{5}$$

注意，空间 $\mathcal{W}^{n,\infty}([0,1]^d)$ 足够一般，因为大多数现实学习任务中使用的函数都属于 $\mathcal{W}^{n,\infty}([0,1]^d)$。为简单起见，我们使用$F_{n,d}$ 中的函数，而非 $\mathcal{W}^{n,\infty}([0,1]^d)$，来验证二次网络的逼近效率。

引理1。 存在一个单隐藏层二次 *ReLU* 网络，可以实现映射 $\tilde{Q}: \mathbb{R}^2 \to \mathbb{R}$，并满足 $\tilde{Q}$: *i)* $Q(x,y) = xy$; *ii)* $Q(x,y)$ 含有2 个二次神经元，相应地需要4 个参数。

证明。 首先，给定输入 (x,y)，通过适当设置式（3）中的参数，可得

$$\begin{aligned} &\tilde{f}(x,y) \\ =&(w_{11}x + w_{12}y + b_1)(w_{21}x + w_{22}y + b_2) + w_{31}x^2 \\ &+ w_{32}y^2 + b_3 \\ =&xy, \end{aligned} \tag{6}$$

其中 $w_{11} = 1, w_{12} = 0, b_1 = 0; w_{21} = 0, w_{22} = 1, b_2 = 0; w_{31} = 0, w_{32} = 0, b_3 = 0$。然后，利用 ReLU 函数，将两个相位相反的并行二次神经元求和，即可构成乘积运算:

$$xy = \sigma(\tilde{f}(x,y)) + \sigma(-\tilde{f}(x,y)) = \tilde{Q}(x,y). \tag{7}$$

$\tilde{Q}(x, y)$ 是一个仅含2个神经元和4 个参数的单隐藏层二次网络。 □

定理1。 对于任意d, $n > 0$和 $\epsilon \in (0, 1)$，存在一个二次*ReLU*网络，使得

1. 二次*ReLU*网络可以近似来自$F_{n,\,d}$ 的任意函数，其误差界为 ϵ，这里的意义是 L^∞;

2. 这个二次*ReLU*网络至多有 $4d^n(N+1)^d(d+n-1)$ 个参数。

证明思路。首先，给定 $f \in F_{n,\,d}$，我们利用一个划分函数$\phi_{\mathbf{m}}$将输入空间划分为$(N+1)^d$ 个超网格，并对每个网格使用局部泰勒多项式 f_1 的和-乘积组合来近似 f。ˆ然后，我们构造一个二次 ReLU 网络 f 来精确表示ˆf_1。因此， f 也可以近似 f。最后，ˆ我们统计 f中使用的参数量。

证明。**步骤1：构造 f_1 以近似 f**。首先，我们需要以下函数将输入空间划分为超网格：

$$\phi_{\mathbf{m}}(\mathbf{x}) = \prod_{k=1}^{d} \psi\left(3N(x_k - \frac{m_k}{N})\right), \tag{8}$$

其中

$$\psi(x) = \begin{cases} 1, & |x| < 1 \\ 0, & |x| > 2 \\ 2 - |x|, & 1 \le |x| \le 2. \end{cases}$$

$\phi_{\mathbf{m}}(\mathbf{x})$ 的特征是：i) $\phi_{\mathbf{m}}(\mathbf{x})$ 的最大值为1：

$$\|\psi\|_\infty = 1 \text{ and } \|\phi_{\mathbf{m}}\|_\infty = 1, \forall \mathbf{m}; \tag{9}$$

ii) 定义域 $[0,1]^d$ 上的一组 $\phi_{\mathbf{m}}(\mathbf{x})$ 构成单位分解：

$$\sum_{\mathbf{m}} \phi_{\mathbf{m}}(\mathbf{x}) = 1, \mathbf{x} \in [0,1]^d,$$

其中$\mathbf{m} = (m_1, \ldots\ldots, m_d) \in \{0,1,\ldots,N\}^d$；iii) $\phi_{\mathbf{m}}(\mathbf{x})$ 是一个局部函数：

$$\text{supp } \phi_{\mathbf{m}} \subset \left\{\mathbf{x} : |x_k - \frac{m_k}{N}| < \frac{1}{N}, \forall k\right\}. \tag{10}$$

$\forall \mathbf{m} \in \{0,\ldots,N\}^d$，令 $P_{\mathbf{m}}(\mathbf{x})$ 为函数 f 在$\mathbf{x} = \mathbf{m}/N$处的 $(n-1)$ 次泰勒多项式，我们有

$$P_{\mathbf{m}}(\mathbf{x}) = \sum_{\mathbf{n}:|\mathbf{n}|<n} \frac{D^{\mathbf{n}} f}{\mathbf{n}!}\bigg|_{\mathbf{x}=\frac{\mathbf{m}}{N}} \left(\mathbf{x} - \frac{\mathbf{m}}{N}\right)^{\mathbf{n}}, \tag{11}$$

其中 $\mathbf{n}! = \prod_{k=1}^{d} n_k!$ 且 $\left(\mathbf{x} - \mathbf{m}/N\right)^{\mathbf{n}} = \prod_{k=1}^{d} \left(x - \frac{m}{N}\right)^n$。

定义 f_1 为

$$f_1 = \sum_{\mathbf{m}\in\{0,\ldots,N\}^d} \phi_{\mathbf{m}} P_{\mathbf{m}}. \tag{12}$$

由于 $\phi_{\mathbf{m}}$ 的截断效应，f_1 是 f 在$(N+1)^d$ 个超网格上的泰勒展开之和。只要 N 足够大，f_1 就可以任意精确地逼近 f。数学上，我们有

$$\begin{aligned} & |f(\mathbf{x}) - f_1(\mathbf{x})| \\ &= \left|\sum_{\mathbf{m}} \phi_{\mathbf{m}}(\mathbf{x}) \left(f(\mathbf{x}) - P_{\mathbf{m}}(\mathbf{x})\right)\right| \\ &\overset{(1)}{\le} \sum_{\mathbf{m}:|x_k - \frac{m_k}{N}| < \frac{1}{N} \forall k} |f(\mathbf{x}) - P_{\mathbf{m}}(\mathbf{x})| \\ &\overset{(2)}{\le} 2^d \max_{\mathbf{m}:|x_k - \frac{m_k}{N}| < \frac{1}{N} \forall k} |f(\mathbf{x}) - P_{\mathbf{m}}(\mathbf{x})| \\ &\overset{(3)}{\le} \frac{2^d d^n}{n!} \left(\frac{1}{N}\right)^n \max_{\mathbf{n}:|\mathbf{n}|=n} \operatorname{ess\,sup}_{\mathbf{x}\in[0,1]^d} |D^{\mathbf{n}} f(\mathbf{x})| \\ &\overset{(4)}{\le} \frac{2^d d^n}{n!} \left(\frac{1}{N}\right)^n. \end{aligned} \tag{13}$$

(1) 由公式 (9) 和 (10) 得出；(2) 由以下事实得出：任意$\mathbf{x}$ 至多属于 2^d 个函数 $\phi_{\mathbf{m}}$ 的支撑集；(3) 由泰勒余项的一个标准界得出；(4) 由 $\|f\|_{\mathcal{W}^{n,\infty}([0,1]^d)} \le 1$ 得出，这是我们的定义。因此，令

$$N = \left\lceil \left(\frac{n!}{2^d d^n} \frac{\epsilon}{2}\right)^{-1/n} \right\rceil, \tag{14}$$

其中 $\lceil\cdot\rceil$ 是向上取整函数，我们有 $\|f - f_1\|_\infty \le \frac{\epsilon}{2}$。

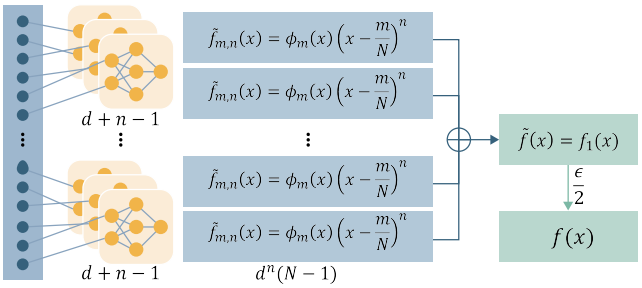

图 1. 使用二次网络逼近任意函数 f 的架构，其中$F_{n,d}$。

步骤 2：构造一个二次网络 f 以近似 f_1。令 $a_{\mathbf{m},\mathbf{n}} = \frac{D^{\mathbf{n}} f}{\mathbf{n}!}\Big|_{\mathbf{x}=\frac{\mathbf{m}}{N}} \le 1$，这是一个通用常数，我们可以将 $P_{\mathbf{m}}(\mathbf{x})$ 的表达式简化为

$$P_{\mathbf{m}}(\mathbf{x}) = \sum_{\mathbf{n}:|\mathbf{n}|<n} a_{\mathbf{m},\mathbf{n}} \left(\mathbf{x} - \frac{\mathbf{m}}{N}\right)^{\mathbf{n}}. \tag{15}$$

相应地，f_1 可简化为

$$f_1(\mathbf{x}) = \sum_{\mathbf{m}\in\{0,\ldots,N\}^d} \sum_{\mathbf{n}:|\mathbf{n}|<n} a_{\mathbf{m},\mathbf{n}} \phi_{\mathbf{m}}(\mathbf{x}) \left(\mathbf{x} - \frac{\mathbf{m}}{N}\right)^{\mathbf{n}}, \tag{16}$$

它是至多 $d^n(N+1)^d$ 个子函数的线性组合。每个子函数都是不超过$d+n-1$个一元线性因子的乘积：$\psi(3Nx_k - 3m_k)$ $k=1$、$\cdots$、d，以及至多 $n-1$ 个线性函数 $x_k - \frac{m_k}{N}$。

现在，我们来构造一个二次神经网络以表示 f_1。如图1所示，我们首先近似$\phi_{\mathbf{m}}(\mathbf{x})\left(\mathbf{x} - \mathbf{m}/N\right)^{\mathbf{n}}$，即不超过 $d+n-1$ 项的乘积。因此，乘法次数为 $d+n-2$。假设 Q 是根据引理1进行的乘法，我们可以通过迭代应用 $\tilde{Q}$来表示 $\phi_{\mathbf{m}}(\mathbf{x})\left(\mathbf{x} - \mathbf{m}/N\right)^{\mathbf{n}}$：

$$\begin{aligned} \tilde{f}_{\mathbf{m},\mathbf{n}}(\mathbf{x}) = \tilde{Q}\Big(& \psi\left(3Nx_1 - 3m_1\right), \\ & \tilde{Q}\left(\psi\left(3Nx_2 - 3m_2\right), \ldots, \tilde{Q}\left(x_k - \frac{m_k}{N}, \ldots\right)\ldots\right)\Big). \end{aligned} \tag{17}$$

因此，$f_{\mathbf{m},\mathbf{n}}$ 可以由一个参数量为$4(d+n-2)$的二次ReLU网络构造。此外，$\forall k$ 和$\mathbf{x} \in [0,1]^d$，根据引理1，使用 f_1 来近似 $\tilde{f}$ 不会产生误差，

$$\begin{aligned} & |\tilde{f}_{\mathbf{m},\mathbf{n}}(\mathbf{x}) - \phi_{\mathbf{m}}(\mathbf{x})\left(\mathbf{x} - \frac{\mathbf{m}}{N}\right)^{\mathbf{n}}| \\ &= |\tilde{Q}\left(\psi\left(3Nx_1 - 3m_1\right), \tilde{Q}\left(\psi\left(3Nx_2 - 3m_2\right), \tilde{Q}\left(\psi\left(3Nx_3 - 3m_3\right), \ldots\right)\right)\right) \\ &\quad - \psi\left(3Nx_1 - 3m_1\right)\psi\left(3Nx_2 - 3m_2\right)\psi\left(3Nx_3 - 3m_3\right)\ldots| \\ &= |\psi\left(3Nx_1 - 3m_1\right)\tilde{Q}\left(\psi\left(3Nx_2 - 3m_2\right), \tilde{Q}\left(\psi\left(3Nx_3 - 3m_3\right), \ldots\right)\right) \\ &\quad - \psi\left(3Nx_1 - 3m_1\right)\psi\left(3Nx_2 - 3m_2\right)\psi\left(3Nx_3 - 3m_3\right)\ldots| \\ &= |\psi\left(3Nx_1 - 3m_1\right)\psi\left(3Nx_2 - 3m_2\right)\psi\left(3Nx_3 - 3m_3\right), \ldots \\ &\quad - \psi\left(3Nx_1 - 3m_1\right)\psi\left(3Nx_2 - 3m_2\right)\psi\left(3Nx_3 - 3m_3\right)\ldots| \\ &= 0. \end{aligned} \tag{18}$$

其次，由于 $f_{\mathbf{m},\mathbf{n}}$ 只近似了 $f_1(\mathbf{x})$中的一项，现在将每个$f_{\mathbf{m},\mathbf{n}}$ 聚合起来，得到最终的网络：

$$\tilde{f} = \sum_{\mathbf{m}\in\{0,\ldots,N\}^d} \sum_{\mathbf{n}:|\mathbf{n}|<n} a_{\mathbf{m},\mathbf{n}} \tilde{f}_{\mathbf{m},\mathbf{n}}. \tag{19}$$

~ ~由于 $f_{\mathbf{m},\ \mathbf{n}}$ 由局部项 $\phi_{\mathbf{m}}(\mathbf{x})$组成，$f_{\mathbf{m},\ \mathbf{n}}$ 也是局部的。因此，使用 f 近似 f_1 时，误差为零:

$$\begin{aligned}&\left|\tilde{f}(\mathbf{x})-f_1(\mathbf{x})\right|\\ \overset{(1)}{=}&\left|\sum_{\mathbf{m}\in\{0,\ldots,N\}^d}\sum_{\mathbf{n}:|\mathbf{n}|<n}a_{\mathbf{m},\mathbf{n}}\left(\tilde{f}_{\mathbf{m},\mathbf{n}}(\mathbf{x})-\phi_{\mathbf{m}}(\mathbf{x})\left(\mathbf{x}-\frac{\mathbf{m}}{N}\right)^{\mathbf{n}}\right)\right|\\ \overset{(2)}{=}&\left|\sum_{\mathbf{m}:\mathbf{x}\in\operatorname{Supp}\phi_{\mathbf{m}}}\sum_{\mathbf{n}:|\mathbf{n}|<n}a_{\mathbf{m},\mathbf{n}}\left(\tilde{f}_{\mathbf{m},\mathbf{n}}(\mathbf{x})-\phi_{\mathbf{m}}(\mathbf{x})\left(\mathbf{x}-\frac{\mathbf{m}}{N}\right)^{\mathbf{n}}\right)\right|\\ =&0.\end{aligned}\tag{20}$$

在上式中，(1) 根据式（16）得出，(2) 则~是由于$f_{\mathbf{m},\ \mathbf{n}}$的局部性。

根据**步骤1**，我们有 $\|f-f_1\|_\infty\leq\frac{\epsilon}{2}$。因此，如果选择 $N=\left\lceil\left(\frac{n!}{2^d d^n}\frac{\epsilon}{2}\right)^{-1/n}\right\rceil$，

$$\|\tilde{f}-f\|_\infty\leq\left\|\tilde{f}-f_1\right\|_\infty+\|f_1-f\|_\infty\leq 0+\frac{\epsilon}{2}\leq\epsilon.\tag{21}$$

~直接计算可得，f 总计消耗不超过$4d^n(N+1)^d(d+n-2)$ 个参数。

□

注1。 表2根据 [13]中的逼近方案，对比了传统型网络与二次ReLU网络的参数量。可以看出，二次构造比传统型构造高效得多，节省的参数量达一个数量级。如此显著节省的关键在于，二次构造能够自然地表达乘法运算。因此，我们的分析也可以推广到一般的多项式网络，以证明其参数效率。

表2 基于 [13] 中的逼近方案，传统型和二次 ReLU 网络逼近索博列夫空间中函数所用参数量的比较。

网络	运算	参数量
传统型 [13]	乘积	$\mathcal{O}$(对数(1/ϵ))
	总计	$d^n(N+1)^d(d+1)n\mathcal{O}(\log(1/\epsilon))$
二次（我们的方法）	乘积	4
	总计	$4d^n(N+1)^d(d+n-2)$

3.3 效率：流形上的函数

关于深度学习似乎不会遭受维数灾难，一种观点是，现实世界中的许多高维数据都沿着环境高维空间中的一个低维潜在流形分布 [34]。根据这一观点，还必须确认二次网络在流形上而非欧几里得空间中逼近函数的效率。这里，我们表明，二次网络也能通过利用二次神经元可以非常自然地表达乘积运算这一特点，更高效地逼近低维流形上的函数。在讨论主定理之前，我们先介绍与流形相关的概念。

定义 3（图）。 流形 $\mathcal{M}$ 上的一个图（U，ψ）是一个有序对，其中 $U\subset\mathcal{M}$ 是一个开集，且 $\psi:\ U\to\mathbb{R}^d$。此外，ψ 和 ψ^{-1} 都是连续的。

开集 U 本质上是$\mathcal{M}$的一个局部邻域，而 ψ 则在 U上定义了一个坐标系。可能存在两个彼此重叠的图。我们称两个图（U_1，ψ_1）和（U_2，ψ_2）相容，当且仅当 $\psi_1\circ\psi_2^{-1}$: ψ_2（$U_1\cap U_2$）$\to\psi_1$（$U_1\cap U_2$）以及 $\psi_2\circ\psi_1^{-1}$: ψ_1（$U_1\cap U_2$）$\to\psi_2$（$U_1\cap U_2$）都连续。一个图不足以描述整个流形。因此，让我们定义一个图册。

定义 4 (C^k 图册)。 *A* C^k-图册是一个集合 $\{(U_i,\ \psi_i)\}$，其中任意两个坐标图都是 C^k 相容且 $\cup_{i=1}^{C_\mathcal{M}}U_i=\mathcal{M}$。

定义 5 (光滑流形)。 光滑流形是带有 C^∞ 图册的流形。

定义 6. 若存在常数$B>0$，使得对于任意$\mathbf{x}\in\mathcal{M}$，都有 $|\mathbf{x}_i|\leq B, i=1,\cdots,d$，则称流形有界。

命题 1 (C^∞ 单位分解。 设 $\{U_i\}_{i=1}^{C_\mathcal{M}}$ 覆盖光滑流形 $\mathcal{M}$，其中 $C_\mathcal{M}$ 为坐标图的数量，则存在一个单位分解 $\{\rho_i\}_{i=1}^{C_\mathcal{M}}$ 满足

$$\sum_{i=1}^{C_\mathcal{M}}\rho_i=1,\tag{22}$$

其中，ρ_i 的支撑集包含于 U_i中。

定义 7 (C^n 流形上的函数。 设 $\mathcal{M}$为光滑流形。若对于 $\mathcal{M}$ 的任意坐标图(U,ψ)，复合函数$h\circ\psi^{-1}$: $\psi(U)\to\mathbb{R}$ 连续可微至 n阶，则称函数 h: $\mathcal{M}\to\mathbb{R}$ 为 C^n。

定义 8 (H¨older 空间 $H^{n,\alpha}$)。 函数 f: $\mathcal{M}\mapsto\mathbb{R}$ 属于 $H^{¨}$*older* 空间 $H^{n,\alpha}$，其中 n 为正整数且 $\alpha\in(0,1]$满足 $f\in C^{n-1}$，并且对于任意坐标图(U,ϕ) 以及 $|\mathbf{n}|=n$，有

$$\begin{aligned}&\left|D^{\mathbf{n}}\left(f\circ\phi^{-1}\right)|_{\phi(\mathbf{x}_1)}-D^{\mathbf{n}}f\circ\phi^{-1}|_{\phi(\mathbf{x}_2)}\right|\\ &\leq\|\phi\left(\mathbf{x}_1\right)-\phi\left(\mathbf{x}_2\right)\|_2^\alpha,\quad\forall\mathbf{x}_1,\mathbf{x}_2\in U\end{aligned}\tag{23}$$

定义 9 (流形的可达半径，[35]的定义 2.1）。将流形 $\mathcal{M}$ 的中轴记为 $\mathrm{Med}(\mathcal{M})$，它是 $\mathcal{M}$的子集，而$\mathrm{Med}(\mathcal{M})$ 中的点在 $\mathcal{M}$上至少有两个邻点。将距离函数记为 $\mathcal{M}$: distan~ee($\mathbf{z}$ $\mathcal{M}$) = $\{\in\mathbb{R}^D\mid\exists\neq\in\mathcal{M}\|-\|=\|-\|=d\ \mathcal{M}\}$ $\inf_{\mathbf{p}\in\mathcal{M}}\|$ $\mathbf{p}\,\mathbf{z}\|$。于是，$\mathrm{Med}(\mathcal{M})\,\mathbf{z}\,\mathbf{p}\,\mathbf{q}$，$\mathbf{p}\,\mathbf{z}\mathbf{q}\,\mathbf{z}(\mathbf{z})$。$\mathcal{M}$ 的可达半径 $\tau>0$ 定义为

$$\tau=\inf_{\mathbf{p}\in\mathcal{M}}\operatorname{distance}(\mathbf{p},\mathrm{Med}(\mathcal{M}))=\inf_{\mathbf{z}\in\mathrm{Med}(\mathcal{M})}\operatorname{distance}(\mathbf{z},\mathcal{M})\tag{24}$$

从几何上看，点$\mathbf{x}\in\mathcal{M}$ 的可达半径 τ 可以解释为密切圆的半径，且该半径至少为τ。较大的可达半径意味着流形的变化不剧烈。在我们的构造中，可达半径控制着为 $\mathcal{M}$选择图册的过程。

命题 2 (引理 2，出自 [14])。 对于 $i=1,\ \cdots,\ C_\mathcal{M}$，函数 h_i 属于 $H^{n,\ \beta}$ 且

$$\begin{aligned}&|D^{\mathbf{n}}(h_i\circ\psi_i^{-1})|_{\psi_i(\mathbf{x}_1)}-D^{\mathbf{n}}(h_i\circ\psi_i^{-1})|_{\psi_i(\mathbf{x}_2)}|\\ &\leq L_i||\psi_i(\mathbf{x}_1)-\psi_i(\mathbf{x}_2)||_2^\beta,\ \forall\,\mathbf{x}_1,\mathbf{x}_2\in U_i,\end{aligned}\tag{25}$$

其中 $L_i=2^{n+1}2\sqrt{d}\mu_i\lambda_i(2r)^{1-\beta}$ 是一个霍尔德系数，取决于维度 d 和通用常数 μ_i，λ_i，r。

命题3 (引理3，出自 [14])。对于任意 $i \in\{1,2,\cdots,C_{\mathcal{M}}\}$ 和 $\mathcal{K}_i=\{\mathbf{x}\in\mathcal{M}: r^2-\Delta\le||\mathbf{x}-\mathbf{c}_i||_2^2\le r^2\}$，存在一个常数 c，其值取决于 h_i 和 ψ_i，使得

$$\max_{\mathbf{x}\in\mathcal{K}_i}|h_i(\mathbf{x})|\le\frac{c(\pi+1)}{r(1-r/\tau)}\Delta. \tag{26}$$

定理2。给定一个具有正可达半径的有界流形 $\mathcal{M}$ 和一个函数 $h:\mathcal{M}\mapsto\mathbb{R}\in H^{n,\beta}$，对于任意 $\epsilon>0$，都存在一个网络 $\hat{h}$，使得

$$\sup_{\mathbf{x}\in\mathcal{M}}|h(\mathbf{x})-\hat{h}(\mathbf{x})|<\epsilon, \tag{27}$$

并且 $\hat{h}$ 至多有 $C_{\mathcal{M}}(c_1'\epsilon^{-\frac{d}{n+\beta}})$ 个参数，其中 $C_{\mathcal{M}}$ 是流形上的分区数，c_1 是一个通用常数。

证明概要：我们的策略是分治法。首先，我们为所关注的流形构造一个图册 $\{(U_i,\psi_i)\}_{i=1}^{C_{\mathcal{M}}}$。然后，基于这样的图册，将属于 H¨older 空间的函数 h 分区，并使用 ψ_i^{-1}映射到欧几里得空间。之后，我们使用泰勒多项式 $\bar{h}_i$ 以任意小的误差对其进行逼近，而二次 ˆ ¯网络 f_i 能够无误差地逼近 h_i。接下来，需要将函数ˆ h_i 反投影回原始ˆ ¯流形。因此，我们定义 $h_i=h_i\circ\psi_i$ 和一个界定ˆ函数 $\mathbb{1}_{\Delta}$。最终的逼近器 h 是 h_i 与 $\mathbb{1}_{\Delta}$之间若干ˆ乘积的和。最后，我们计算总误差和参数量。图2展示了我们的策略及相关中间函数。

证明。**步骤1：构造一个图册**。设 $\mathbb{R}^D$ 中的开欧几里得球为 $\mathcal{B}(\mathbf{c}\ r)$，其中$\mathbf{c}$ 是球心，r 是半径。由于 $\mathcal{M}$ 是紧致的，我们可以找到有限个点 $\{\mathbf{c}_i\}_{i=1}^{C_{\mathcal{M}}}$并令 $r<\tau/2$，τ是 $\mathcal{M}$的可达半径，以确保 $\mathcal{M}\subset\cup_i\mathcal{B}(\mathbf{c}_i,r)$。我们选取 $\{(U_i,\psi_i)\}_{i=1}^{C_{\mathcal{M}}}$作为图册，其中 $U_i=\mathcal{M}\cap\mathcal{B}(\mathbf{c}_i,r)$。$C_{\mathcal{M}}$ 有上界

$$C_{\mathcal{M}}\le\left\lceil\frac{SA(\mathcal{M})}{r^d}T_d\right\rceil, \tag{28}$$

其中 $SA(\mathcal{M})$ 是 $\mathcal{M}$ 的表面积，T_d 是 $U_i,i=1,\cdots,C_{\mathcal{M}}$的最大厚度。

在每个图上，我们将 ψ_i 定义为一种投影，通过平移、缩放和旋转，将流形的局部邻域映射到欧几里得空间 $[0,1]^d$ 中。从数学上来说，$\psi_i(\mathbf{x})$ 可表示为

$$\psi_i(\mathbf{x})=b_i(\mathbf{V}_i(\mathbf{x}-\mathbf{c}_i)+\mathbf{s}_i), \tag{29}$$

其中 $\psi_i(\mathbf{x})\in[0,1]^d$，$b_i\in(0,1]$ 是缩放因子，$\mathbf{V}_i$ 是旋转矩阵，$\mathbf{s}_i$ 是平移向量。$\psi_i(\mathbf{x})$ 可以由单隐藏层二次网络直接实现。

步骤 2：划分与估计。我们将 h 划分为 $\{h_i\}_{i=1}^{C_{\mathcal{M}}}$ 使得

$$h(\mathbf{x})=\sum_{i=1}^{C_{\mathcal{M}}}h(\mathbf{x})\cdot\rho_i(\mathbf{x})=\sum_{i=1}^{C_{\mathcal{M}}}h_i(\mathbf{x}). \tag{30}$$

二次网络可以任意精确地近似 $h_i\circ\psi_i^{-1}(\mathbf{x})$。这是因为 h_i 属于 H¨older 类，我们可以采用类似的泰勒展开思路。

¯这里，令 $h_i(\mathbf{x})$ 为一个满足与前一小节相同条件的泰勒多项式。我们有

$$\bar{h}_i(\mathbf{x})=\sum_{\mathbf{m}\in\{1,\ldots,N\}^d}\phi_{\mathbf{m}}(\mathbf{x})P_{\mathbf{m}}(\mathbf{x}), \tag{31}$$

其中，$\phi_{\mathbf{m}}$ 在式（8）中定义，$P_{\mathbf{m}}(\mathbf{x})$ 是函数 $h_i\circ\psi_i^{-1}(\mathbf{x})$ 在 $\mathbf{m}\ N$ 处的 n 次泰勒多项式，如式（11）所示。因此，$h_i\circ\psi_i^{-1}(\mathbf{x})$ ¯与 $h_i(\mathbf{x})$ 的逼近误差为

$$\begin{aligned}
&\sup_{\mathbf{x}\in[0,1]}|h_i\circ\psi_i^{-1}(\mathbf{x})-\bar{h}_i(\mathbf{x})|\\
&=\sup_{\mathbf{x}\in[0,1]}|\sum_{\mathbf{m}}\phi_{\mathbf{m}}h_i\circ\psi_i^{-1}(\mathbf{x})-\sum_{\mathbf{m}}\phi_{\mathbf{m}}P_{\mathbf{m}}(\mathbf{x})|\\
&\le\sup_{\mathbf{x}\in[0,1]}\sum_{\mathbf{m}}\phi_{\mathbf{m}}|(h_i\circ\psi_i^{-1}(\mathbf{x})-P_{\mathbf{m}}(\mathbf{x}))|\\
&\le\sup_{\mathbf{x}\in[0,1]}\sum_{\mathbf{m}:\left|x_k-\frac{m_k}{N}\right|<\frac{1}{N}\forall k}|(h_i\circ\psi_i^{-1}(\mathbf{x})-P_{\mathbf{m}}(\mathbf{x}))|\\
&\le\sup_{\mathbf{x}\in[0,1]}2^d\max_{\mathbf{m}:\left|x_k-\frac{m_k}{N}\right|<\frac{1}{N}\forall k}|h_i\circ\psi_i^{-1}(\mathbf{x})-P_{\mathbf{m}}(\mathbf{x})|\\
&\overset{(1)}{\le}\frac{2^dd^n}{n!}\Big(\frac{1}{N}\Big)^n\max_{\mathbf{n}=n}\Big|D^{\mathbf{n}}(h_i\circ\psi_i^{-1})|_{\frac{\mathbf{m}}{N}}-D^{\mathbf{n}}(h_i\circ\psi_i^{-1})|_{\boldsymbol{y}}\Big|\\
&\overset{(2)}{\le}\frac{2^dd^n}{n!}\Big(\frac{1}{N}\Big)^n2^{n+1}\sqrt{d}\mu_i\lambda_i(2r)^{1-\beta}||\frac{\mathbf{m}}{N}-\mathbf{x}||_2^{\beta}\\
&\le\frac{2^dd^n}{n!}\Big(\frac{1}{N}\Big)^n2^{n+1}\sqrt{d}\mu_i\lambda_i(2r)^{1-\beta}d^{2/\beta}\Big(\frac{1}{N}\Big)^{\beta},
\end{aligned} \tag{32}$$

其中，（1）由泰勒余项的界以及$\boldsymbol{y}$ 是$\mathbf{m}\ N$ 与$\mathbf{x}$的线性插值可得；（2）由命题 2 得出。类似地，¯令

$$N\ge\Big(\frac{2^dd^n}{\delta n!}2^{n+1}\sqrt{d}\mu_i\lambda_i(2r)^{1-\beta}d^{2/\beta}\Big)^{\frac{1}{n+\beta}}, \tag{33}$$

得到

$$\frac{2^dd^n}{n!}\Big(\frac{1}{N}\Big)^n2^{n+1}\sqrt{d}\mu_i\lambda_i(2r)^{1-\beta}d^{2/\beta}\Big(\frac{1}{N}\Big)^{\beta}\le\delta. \tag{34}$$

¯然后，由于 $h_i(x)$ 是通过欧几里得空间中的泰勒多项式ˆ构造的，二次网络 f_i 可以无误差地逼近它，正如我们在上一小节中已经证明的那样。因此，h_i ˆ与 f_i 之间的逼近误差也不超过 δ。若在式（33）中选择 N，则二次网络至多需要 $c_1\delta^{-\frac{d}{n+\beta}}$ 个参数来逼近 $\bar{h}_i(x)$，其中 $c_1=4d^n(N+1)^d(d+n-2)$。

步骤 3：反投影与界定。函数 h_i是一个逼近器，用于逼近支撑集为 $[0,1]^d$的 $h_i\circ\psi_i^{-1}$。¯ ¯为了逼近支撑在流形第 i个图上的目标函数 h_i，我们需要将 h_i 反投影到流形中。为此，我们将 h_i 与ˆ ˆ ¯ ˆψ_i 复合，得到 h_i：$h_i=h_i\circ\psi_i$。h_i 只能在 U_i的定义域内很好地逼近 h_i。因此，我们需要将ˆh_i 界定在 U_i的区域内。

在数学上，我们将截断函数定义为半梯形函数 $\mathbb{1}_{\Delta}(s)$ 与径向函数 $s_i(\mathbf{x})$的复合。这是因为每个 U_i 都包含在 $\mathcal{B}(\mathbf{c}\ r)$中。对于 $i=1,\ 2,\ \cdots,\ C_{\mathcal{M}}$，径向函数为

$$s_i(\mathbf{x})=||\mathbf{x}-c_i||_2^2, \tag{35}$$

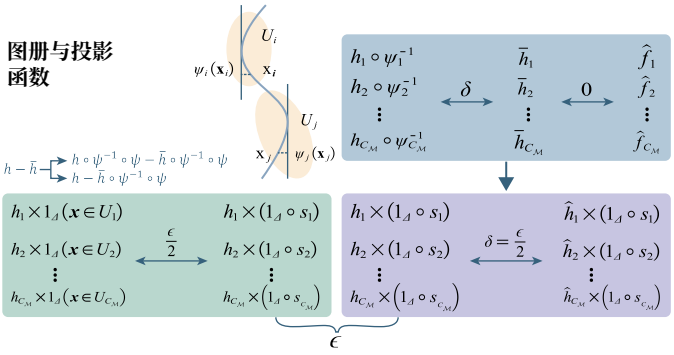

图2. 使用二次网络逼近流形上的函数 h 的概览。

而半梯形函数 $\mathbb{1}_\Delta(s)$ 为

$$\mathbb{1}_\Delta(s)=\begin{cases}1, & s<r^2-\Delta\\ -\frac{1}{\Delta}(s-r^2), & r^2-\Delta\le s\le r^2\\ 0, & s>r^2,\end{cases}\tag{36}$$

其中，Δ 根据总误差 ϵ 选取。

步骤 4：总误差。为了实现截断，我们ˆ直接将 h_i 与 $\mathbb{1}_\Delta\circ s_i$ 相乘。最终的逼近器为 $\hat{h}=\sum_{i=1}^{C_\mathcal{M}}\hat{}h_i\times(\mathbb{1}_\Delta\circ s_i)$。现在我们可以估计总误差了

$$\begin{aligned}&\sup_{\mathbf{x}\in\mathcal{M}}|h(\mathbf{x})-\hat{h}(\mathbf{x})|\\ =&\sup_{\mathbf{x}\in\mathcal{M}}|\sum_{i=1}^{C_\mathcal{M}}(h_i(\mathbf{x})\times\mathbb{1}(\mathbf{x}\in U_i)-\hat{h}_i(\mathbf{x})\times(\mathbb{1}_\Delta\circ s_i))|\\ \le&\sup_{\mathbf{x}\in\mathcal{M}}\sum_{i=1}^{C_\mathcal{M}}|h_i(\mathbf{x})\times\mathbb{1}(\mathbf{x}\in U_i)-\hat{h}_i(\mathbf{x})\times(\mathbb{1}_\Delta\circ s_i)|\\ \le&\sup_{\mathbf{x}\in\mathcal{M}}\sum_{i=1}^{C_\mathcal{M}}\underbrace{|h_i(\mathbf{x})\times\mathbb{1}(\mathbf{x}\in U_i)-h_i(\mathbf{x})\times(\mathbb{1}_\Delta\circ s_i)|}_{A_{i,1}}\\ &+\sup_{\mathbf{x}\in\mathcal{M}}\sum_{i=1}^{C_\mathcal{M}}\underbrace{|h_i(\mathbf{x})\times(\mathbb{1}_\Delta\circ s_i)-\hat{h}_i(\mathbf{x})\times(\mathbb{1}_\Delta\circ s_i)|}_{A_{i,2}}\\ =&C_\mathcal{M}\Big(\frac{c(\pi+1)}{r(1-r/\tau)}\Delta+\delta\Big),\end{aligned}\tag{37}$$

其中，$A_{i,1}$ 满足命题3，而 $A_{i,2}$ 由公式 (32)-(34) 得出。接下来

$$|h_i(\mathbf{x})-\hat{h}_i(\mathbf{x})|=|h_i\circ\psi^{-1}\circ\psi(\mathbf{x})-\bar{h}_i\circ\psi(\mathbf{x})|\le\delta.\tag{38}$$

最后，我们选取 $\delta=\frac{\epsilon}{2C_\mathcal{M}}$，使得 $\sum_{i=1}^{C_\mathcal{M}}A_{i,2}\le\frac{\epsilon}{2}$。类似地，我们选择 $\Delta=\frac{r(1-r/\tau)2c}{\pi+1}\frac{\epsilon}{2C_\mathcal{M}}$，使得 $C_\mathcal{M}c(\pi+1)\frac{r(1-r/\tau)}{c}\Delta\le\frac{\epsilon}{2}$。

接下来，我们来计算所得网络中的参数量。i) 对于泰勒多项式逼近器，我们在**步骤 2** 中计算得出，用于逼近二次网络的参数量不超过 $C_\mathcal{M}c_1\epsilon^{-\frac{d}{n+\beta}}$，以逼近 $h(\mathbf{x})$，其中 $N\ge\left(\frac{2^{d+1}C_\mathcal{M}d^n!2^{n+1}\sqrt{d}\mu_i\lambda_i(2r)^{1-\beta}d^{2/\beta}}{\epsilon n}\right)^{n+\beta}$。ii) 对于反投影函数 $\mathbb{1}_\Delta$，可通过单层二次网络实现。对径向函数 s_i 的逼近可由一个含 $d^{3.75}/\epsilon$ 个神经元的单隐层二次网络实现 [12]。因此，计算 $\mathbb{1}_\Delta(s)$ 所需的并行次二次网络总数为 $C_\mathcal{M}$ 层，含有 $C_\mathcal{M}d^{3.75}/\epsilon$ 个参数。最后，二次ReLU网络中的参数总数为 $C_\mathcal{M}(c_1\epsilon^{-\frac{d}{n+\beta}}+d^{3.75}/\epsilon)\approx C_\mathcal{M}(c_1'\epsilon^{-\frac{d}{n+\beta}})$。

□

注释 2（最优误差）。表3比较了基于方案 [14]，在逼近 H¨older 空间中的函数时，传统型网络与二次 ReLU 网络所使用的参数量。可以看出，二次构造使参数量呈指数级减少。有人可能会认为，在第3.2节和第3.3节中，我们只是展示了二次网络在表示具有特定结构的函数时效率高得多。然而，还有其他表示函数的方案，例如使用傅里叶变换；目前尚不清楚二次网络在这些方案下是否仍能实现参数效率。这里我们认为，尽管存在其他类型的逼近方案，我们所采用的表示方案在实现最低误差率方面是最优的。因此，我们的理论推导仍有充分依据，因为在逼近某个空间中的函数时，通常会优先考虑具有最优误差率的逼近方案。

表3 基于逼近方案 [14]，传统型和二次ReLU网络逼近 H¨older 空间中的函数时所用参数量的比较$H^{n,\beta}$ $c_1'<c'$

网络	运算	#参数
传统型 [14]	乘积	$\mathcal{O}(\ln(1/\epsilon))$
	总计	$c'C_\mathcal{M}(\epsilon^{-\frac{d}{n+\beta}}\log(1/\epsilon)+d\log(1/\epsilon))$
二次（我们的方法）	乘积	18
	总计	$c_1'C_\mathcal{M}\epsilon^{-\frac{d}{n+\beta}}$

3.4 效率：Barron空间分析

从 Barron 空间的角度出发，我们通过证明二次网络能够避免维数灾难，阐明了其具有很高的参数效率。在神经网络逼近理论中，要达到相同的误差水平，所构造网络结构的复杂度通常会随函数维度的增加而增加 [13],[36], [37]。这种现象称为维数灾难。然而，Barron 空间中的函数可以用单隐层网络表示，其误差率不依赖维度，而是由目标函数的某种范数控制。常规神经元的 Barron 空间，记作 $\mathcal{B}_2^{(1)}$，是在 [38], [39]中推导得出的。本文通过推导二次神经元 Barron 空间（记作 $\mathcal{B}_2^{(2)}$），证明二次神经元也具有这种维度无关性。接着，由于 $\mathcal{B}_2^{(2)}$ 由 $\mathcal{B}_2^{(1)}$ 构成，使用二次神经元逼近 $\mathcal{B}_2^{(2)}\setminus\mathcal{B}_2^{(1)}$ 中的函数不依赖维度，而使用传统神经元则依赖维度，由此表明二次神经元在避免维数灾难方面具有参数效率。

定义 10 (常规神经元的 Barron 空间， $\mathcal{B}_2^{(1)}$ [38])。该空间由满足以下表示形式的函数 f： $X \mapsto \mathbb{R}$ 组成：

$$f(\mathbf{x}) = \int_{\Omega} a\sigma\left(\boldsymbol{b}^T\mathbf{x} + c\right)\rho(da, d\boldsymbol{b}, dc), \quad \mathbf{x} \in X, \tag{39}$$

其中，$\Omega = \mathbb{R}^1 \times \mathbb{R}^d \times \mathbb{R}^1$, ρ 是在$(\Omega\ \Sigma_\Omega)$ 上的概率分布，Σ_Ω 是Ω上的 *Borel* σ-代数，而 $\sigma(x) = \max\{x, 0\}$ 是 *ReLU* 激活函数。我们将式(39) 所示表示的 *Barron* 范数定义为

$$\|f\|_{\mathcal{B}_2^{(1)}} = \inf_{\rho}\left(\mathbb{E}_\rho\left[|a|^2\left(\|\boldsymbol{b}\|_1 + |c|\right)^2\right]\right)^{1/2}. \tag{40}$$

下确界取遍所有满足式(39) 对所有$\mathbf{x} \in X$均成立的 ρ。*Barron*空间 $\mathcal{B}_2^{(1)}$ 由式(39)定义的、具有有限 *Barron* 范数的连续函数构成。

注意，当 $m \to \infty$时，式（39）可以类比为由 m 个神经元组成的两层神经网络：

$$f_m(\mathbf{x}; \Theta) := \frac{1}{m}\sum_{j=1}^{m} a_j\sigma\left(\boldsymbol{b}_j^T\mathbf{x} + c_j\right), \tag{41}$$

其中， $\Theta = (a_1, \boldsymbol{b}_1, c_1, \ldots, a_m, \boldsymbol{b}_m, c_m)$ 是所有参数的集合。这直观地说明了为什么使用传统神经元近似式（39）中的函数具有很高的效率 [38]。此外，Barron空间具有相当广泛的包容性。根据 [39], 中的命题1，对于给定函数，只要其傅里叶分布具有可积密度且一阶矩有限，该函数就属于Barron空间。因此，Barron空间已经能够满足大多数现实任务的需求。由于 Barron空间的包容性，我们基于Barron空间得到的参数效率结果具有普遍意义。

命题4 (定理1，见 [38])。对于任意 $f \in \mathcal{B}_2^{(1)}$ 和$m > 0$，存在一个双层神经网络 $f_m(\mathbf{x}; \Theta) = \frac{1}{m}\sum_{k=1}^{m} a_k\sigma\left(\boldsymbol{b}_k^T + c_k\right)$ 使得

$$\|f(\cdot) - f_m(\cdot; \Theta)\|^2 \leq \frac{3\|f\|_{\mathcal{B}_2^{(1)}}^2}{m}. \tag{42}$$

逆逼近定理也可以得到证明。为陈述逆逼近定理，定义如下：

$$\begin{aligned}\mathcal{N}_Q^{(1)} = \{&\frac{1}{m}\sum_{k=1}^{m} a_k\sigma\left(\boldsymbol{b}_k^T\mathbf{x} + c_k\right): \\ &\frac{1}{m}\sum_{k=1}^{m}|a_k|\left(\|\boldsymbol{b}_k\|_1 + |c_k|\right) \leq Q, m \in \mathbb{N}^+\}.\end{aligned} \tag{43}$$

命题5 (定理2，见 [38])。设 f^* 是定义在 X上的连续函数。假设存在常数 Q和函数序列 $(f_m) \subset \mathcal{N}_{(1)Q}$，使得对所有$\mathbf{x} \in X$，都有 $f_m(\mathbf{x}) \to f^*(\mathbf{x})$。则存在一个概率分布 ρ^* 定义在$(\Omega\ \Sigma_\Omega)$上，使得

$$f^*(\mathbf{x}) = \int a\sigma\left(\boldsymbol{b}^T\mathbf{x} + c\right)\rho^*(da, d\boldsymbol{b}, dc), \tag{44}$$

对所有 $\mathbf{x} \in X$。此外，我们有 $f^* \in \mathcal{B}_2^{(1)}$ 与

$$\|f^*\|_{\mathcal{B}_2^{(1)}} \leq Q. \tag{45}$$

命题 4 和 5 表明，Barron空间 $\mathcal{B}_2^{(1)}$ 是专为采用传统神经元的两层网络定义的函数空间，该网络满足直接逼近定理和逆逼近定理。使用两层传统网络逼近 $\mathcal{B}_2^{(1)}$ 中的函数不会遭遇维数灾难。同理，我们也可以推导出两层二次网络的 Barron 空间。现在先来定义二次神经元 Barron 空间：

定义11 (二次神经元 Barron 空间， $\mathcal{B}_2^{(2)}$)。该空间由满足以下表示形式的函数 g： $X \mapsto \mathbb{R}$ 构成：

$$\begin{aligned}&g(\mathbf{x}) = \\ &\int_{\Omega} a\sigma((\boldsymbol{w}_1^\top\mathbf{x} + b_1)(\boldsymbol{w}_2^\top\mathbf{x} + b_2) + \boldsymbol{w}_3^\top(\mathbf{x} \odot \mathbf{x}) + c)\rho(da, d\boldsymbol{w}, d\boldsymbol{b}),\end{aligned} \tag{46}$$

其中 $\mathbf{x} \in X$、$\Omega = \mathbb{R}^1 \times \mathbb{R}^{3d} \times \mathbb{R}^3$、$\boldsymbol{w} = (\boldsymbol{w}_1, \boldsymbol{w}_2, \boldsymbol{w}_3)$、$\boldsymbol{b} = (b_1, b_2, b_3)$， ρ是定义在$(\Omega\ \Sigma_\Omega)$上的概率分布，其中Σ_Ω 是Ω上的一个 *Borel* σ代数，并且 $\sigma(x) = \max\{x, 0\}$ 是 *ReLU* 激活函数。我们将表示式 *Eq.*(46) 的 *Barron* 范数定义如下：

$$\begin{aligned}&\|g\|_{\mathcal{B}_2^{(2)}} = \\ &\inf_{\rho}\left(\mathbb{E}_\rho\left[|a|^2((\|\boldsymbol{w}\|_1 + |b_1|)(\|\boldsymbol{w}_2\|_1 + |b_2|) + \|\boldsymbol{w}_3\|_1 + |b_3|)^2\right]\right)^{1/2},\end{aligned} \tag{47}$$

下确界取遍所有满足 *Eq.* (46) 对所有$\mathbf{x} \in X$均成立的 ρ。*Barron*空间 $\mathcal{B}_2^{(2)}$ 是由 *Eq.*(46) 定义且具有有限 *Barron* 范数的连续函数集合。

现在，我们证明双层二次神经网络的直接逼近定理。

定理 3 (二次神经元直接定理)**.** 对于任意$g \in \mathcal{B}_2^{(2)}$ 和 $m > 0$，存在一个双层神经网络

$$\begin{aligned}&g_m(\mathbf{x}; \Theta) \\ =&\frac{1}{m}\sum_{k=1}^{m} a_k\sigma((\boldsymbol{w}_{1k}^\top\mathbf{x} + b_{1k})(\boldsymbol{w}_{2k}^\top\mathbf{x} + b_{2k}) + \boldsymbol{w}_{3k}^\top(\mathbf{x} \odot \mathbf{x}) + b_{3k}),\end{aligned} \tag{48}$$

其中 Θ 表示神经网络中的一组参数$\{(a_k, \boldsymbol{w}_k, \boldsymbol{b}_k)\ k \in [m]\}$，满足

$$\|g(\cdot) - g_m(\cdot; \Theta)\|^2 \leq \frac{2\|g\|_{\mathcal{B}_2^{(2)}}^2}{m}. \tag{49}$$

证明。 设 ρ 为一个概率分布，满足

$$g(\mathbf{x}) = \mathbb{E}_\rho[a\sigma((\mathbf{w}_1^\top\mathbf{x} + b_1)(\mathbf{w}_2^\top\mathbf{x} + b_2) + \mathbf{w}_3^\top(\mathbf{x} \odot \mathbf{x}) + b_3)] \tag{50}$$

and

$$\begin{aligned}&\mathbb{E}_\rho\left[|a|^2((\|\mathbf{w}\|_1 + |b_1|)(\|\mathbf{w}_2\|_1 + |b_2|) + \|\mathbf{w}_3\|_1 + |b_3|)^2\right] \\ \leq&(1 + \varepsilon)\|g\|_{\mathcal{B}_2}^2,\end{aligned} \tag{51}$$

其中 ε 将在后面确定。令 $\phi(\mathbf{x}; \Theta) = a\sigma((\mathbf{w}_1\mathbf{x} + b_1)(\mathbf{w}_2^\top\mathbf{x} + b_2) + \mathbf{w}_3^\top(\mathbf{x} \odot \mathbf{x}) + b_3)$，其中 $\Theta = (a, \mathbf{w}, \mathbf{b}) \sim \rho$。于是我们有 $\mathbb{E}_{\Theta\sim\rho}[\phi(\mathbf{x}; \Theta)] = g(\mathbf{x})$。令 $\Theta = \{\Theta_j\}_{j=1}^m$ 为从 $\rho(\cdot)$ 中抽取的独立同分布随机变量，并考虑 $\phi(\mathbf{x}; \Theta)$ 的经验平均：

$$\hat{g}_m(\mathbf{x}; \Theta) = \frac{1}{m}\sum_{j=1}^{m}\phi\left(\mathbf{x}; \theta_j\right). \tag{52}$$

现在我们来计算逼近误差 $\mathcal{E}(\Theta)$ $=$ $\mathbb{E}_{\mathbf{x}}\left[\left|g_m(\Theta)-g(\mathbf{x})\right|^2\right]$ 。关于 Θ 的期望为

$$
\begin{aligned}
&\mathbb{E}_{\Theta}[\mathcal{E}(\Theta)] = \mathbb{E}_{\Theta}\mathbb{E}_{\mathbf{x}}\left|\hat{g}_m(\mathbf{x};\Theta)-f(\mathbf{x})\right|^2 \\
&= \mathbb{E}_{\mathbf{x}}\mathbb{E}_{\Theta}\left|\frac{1}{m}\sum_{j=1}^{m}\phi\left(\mathbf{x};\Theta_j\right)-g(\mathbf{x})\right|^2 \\
&= \frac{1}{m^2}\mathbb{E}_{\mathbf{x}}\sum_{j,k=1}^{m}\mathbb{E}_{\Theta_j,\Theta_k}\left[\left(\phi\left(\mathbf{x};\Theta_j\right)-f(\mathbf{x})\right)\left(\phi\left(\mathbf{x};\Theta_k\right)-g(\mathbf{x})\right)\right] \\
&\leq \frac{1}{m^2}\sum_{j=1}^{m}\mathbb{E}_{\mathbf{x}}\mathbb{E}_{\Theta_j}\left[\left(\phi\left(\mathbf{x};\Theta_j\right)-g(\mathbf{x})\right)^2\right] \\
&\leq \frac{1}{m}\mathbb{E}_{\mathbf{x}}\mathbb{E}_{\Theta\sim\rho}\left[\phi^2(\mathbf{x};\Theta)\right] \\
&\leq \frac{(1+\varepsilon)\|g\|_{\mathcal{B}_2}^2}{m}.
\end{aligned}
\tag{53}
$$

定义事件 $E = \left\{\mathcal{E}(\Theta) < \frac{2\|g\|_{\mathcal{B}_2}^2}{m}\right\}$。根据马尔可夫不等式，我们有

$$
\mathbb{P}\{E\} = 1-\mathbb{P}\{E_c\} \geq 1-\frac{\mathbb{E}_{\Theta}[\mathcal{E}(\Theta)]}{2\|g\|_{\mathcal{B}_2}^2/m} \geq \frac{1-\varepsilon}{2} > 0. \tag{54}
$$

设 $\epsilon < 1$，则 $\mathbb{P}\{E\} > 0$。因此，任取 Θ in$E$，由此定义的网络 Θ 即可完成证明。 □

现在，我们证明两层二次网络的逆逼近定理。首先，为方便起见，我们约定当 m 趋于无穷时，单隐层二次网络的输出函数仍然有界。

$$
\begin{aligned}
&\mathcal{N}_Q^{(2)} \\
=&\{\frac{1}{m}\sum_{k=1}^{m}a_k\sigma((\mathbf{w}_{1k}^{\top}\mathbf{x}+b_{1k})(\mathbf{w}_{2k}^{\top}\mathbf{x}+b_{2k})+\mathbf{w}_{3k}^{\top}(\mathbf{x}\odot\mathbf{x})+b_{3k}): \\
&\frac{1}{m}\sum_{k=1}^{m}|a_k|\left(\left(\|\mathbf{w}_{1k}\|_1+|b_{1k}|\right)\left(\|\mathbf{w}_{2k}\|_1+|b_{2k}|\right)+\|\mathbf{w}_{3k}\|_1+|b_{3k}|\right) \\
&\leq Q, m\in\mathbb{N}^+\}.
\end{aligned}
\tag{55}
$$

定理4。设 g^* 是 X 上的连续函数。假设存在常数 Q 和函数序列 $(g_m) \subset \mathcal{N}_Q^{(2)}$，使得对所有 $\mathbf{x} \in X$，均有 $g_m(\mathbf{x}) \to g^*(\mathbf{x})$。那么存在定义在 (Ω, Σ_Ω) 上的概率分布 ρ^*，使得

$$
\begin{aligned}
&g^*(\mathbf{x}) \\
=&\int_{\Omega}a\sigma((\boldsymbol{w}_1^{\top}\mathbf{x}+b_1)(\boldsymbol{w}_2^{\top}\mathbf{x}+b_2)+\boldsymbol{w}_3^{\top}(\mathbf{x}\odot\mathbf{x})+b_3)\rho(da,d\boldsymbol{w},d\boldsymbol{b}),
\end{aligned}
\tag{56}
$$

对所有 $\mathbf{x} \in X$。此外，我们有 $g^* \in \mathcal{B}_2^{(2)}$ 其中

$$
\|g^*\|_{\mathcal{B}_2^{(2)}} \leq Q. \tag{57}
$$

Proof. Let $\Theta_m = \left\{\left(a_k^{(m)}, \mathbf{w}_{1k}^{(m)}, \mathbf{w}_{2k}^{(m)}, \mathbf{w}_{3k}^{(m)}, \mathbf{b}_k^{(m)}\right)\right\}_{k=1}^{m}$ be parameters in g_m, and let $A = \sum_{k=1}^{m}\left|a_k^{(m)}\right|$ and $\alpha_k = \frac{\left|a_k^{(m)}\right|}{A}$. 然后我们可以定义一个概率测度：

$$
\begin{aligned}
&\rho_m \\
&=\sum_{k=1}^{m}\alpha_k\delta\left(a-\frac{\operatorname{sign}\left(a_k^{(m)}\right)A}{m}\right)\delta\left(\mathbf{w}-\mathbf{w}_k^{(m)}\right)\delta\left(\boldsymbol{b}-\boldsymbol{b}_k^{(m)}\right),
\end{aligned}
\tag{58}
$$

满足

$$
g_m\left(\mathbf{x};\Theta_m\right) = \int a\sigma\left(\boldsymbol{b}^T\mathbf{x}+c\right)\rho_m(da,d\mathbf{w},d\mathbf{b}). \tag{59}
$$

我们可以从以下集合中构造 $(a,\mathbf{w},\mathbf{b})$

$$
\begin{aligned}
&\{(a,\mathbf{w},\mathbf{b}): |a_k| \leq Q, \\
&\left(\|\mathbf{w}_{1k}\|_1+|b_{1k}|\right)\left(\|\mathbf{w}_{2k}\|_1+|b_{2k}|\right)+\|\mathbf{w}_{3k}\|_1+|b_{3k}| \leq 1\}
\end{aligned}
\tag{60}
$$

使得对所有 m，支撑集 $(\rho_m) \subset \mathcal{N}_Q^{(2)}$。由于 $\mathcal{N}_{(2)Q}$ 是紧的，序列 (ρ_m) 是紧的，因为 $m \to \infty$。根据普罗霍罗夫定理 [40]，存在一个子序列 (ρ_{m_t})，它弱收敛于概率测度 ρ^*。此外，由于 支撑集 $(\rho_m) \subset \mathcal{N}_{(2)Q}$，我们还有 支撑集 $(\rho^*) \subset \mathcal{N}_Q^{(2)}$。因此，我们有

$$
\|g^*\|_{\mathcal{B}} \leq Q. \tag{61}
$$

此外，由于 ρ^* 是 ρ_{m_k}的弱极限，我们有

$$
\begin{aligned}
&g^*(\mathbf{x}) \\
&= \lim_{k\to\infty}\int_{\Omega}a\sigma((\mathbf{w}_1^{\top}\mathbf{x}+b_1)(\mathbf{w}_2^{\top}\mathbf{x}+b_2)+\mathbf{w}_3^{\top}(\mathbf{x}\odot\mathbf{x})+c)d\rho_{m_t} \\
&= \int a\sigma((\mathbf{w}_1^{\top}\mathbf{x}+b_1)(\mathbf{w}_2^{\top}\mathbf{x}+b_2)+\mathbf{w}_3^{\top}(\mathbf{x}\odot\mathbf{x})+c)d\rho^*(da,d\mathbf{w},d\mathbf{b}).
\end{aligned}
\tag{62}
$$

□

定理5。对于任意 $g \in \mathcal{B}_2^{(2)} \setminus \mathcal{B}_1^{(2)}$，使用两层二次网络逼近 g 可以避免维数灾难，而使用两层传统网络逼近 g 则无法避免。

证明。结合命题 4 和命题 5，$\mathcal{B}_2^{(2)}$ 是两层传统网络在不遭受维数灾难的情况下能够逼近的最小空间。结合定理 3 和定理 4，$\mathcal{B}_2^{(2)}$ 是两层二次网络在不遭受维数灾难的情况下能够逼近的最小空间，由此即可完成证明。

□

备注 3。从索博列夫空间、流形和 Barron空间等角度开展的理论研究令人鼓舞。我们认为，二次网络的参数效率应具有普遍性，因为我们建立的理论推导没有不切实际的假设，而且所研究的函数空间也具有包容性。然而，我们认为，实际能获得多大效率提升确实取决于具体任务。一方面，我们的理论结果具有构造性。由于神经网络的优化是一个高度非凸问题，二次网络未必会像我们的构造所示那样学到映射。另一方面，我们的理论推导与逼近能力的刻画有关。在现实任务中，也应考虑网络的泛化能力。例如，如果没有精心整理的大数据，就可能出现过拟合问题，导致二次网络无法实现预期的效率。

4 合成实验

为了说明我们的工作，我们构造了两个具有启发性的合成示例（高维同心超球面和高斯混合数据），以展示二次网络的效率。在这两项实验中，参数效率的提升源于二次神经元中的非线性二次项能够便捷地表示非线性决策边界。高维同心超球面实验表明，使用二次网络不受维数灾难影响，这印证了我们的 Barron空间分析。高斯混合数据实验则表明，二次网络的效率适用于广泛的任务，因为高斯混合模型（高斯混合模型）是描述数据分布的一种应用极广的模型，而数据分类是机器学习的基础任务之一。请注意，先前的工作 [32] 已经使用二次网络对二维空间中的高斯混合数据进行分类。在这里，我们将这项工作推广到高维空间，并着重探讨效率。

4.1 同心超球面

我们在 d维空间中生成两个同心超球面。每个超球面包含 2,000 个样本，并受到均值为0 、标准差为0.03的高斯噪声轻微扰动。两个超球面的半径分别为1和0.7。我们假设这两个超球面分别来自两个类别。总计4,000 个样本经过随机打乱后，按照$0.8:0.2$的比例划分为训练集和测试集。我们分别采用单隐层 ReLU 传统型网络和二次网络对同心超球面进行分类。对于这两种网络，学习率设为0.01，批量大小设为64，训练轮数设为50。我们采用 Adam 作为优化器。实验在配备 RTX 3080 Ti 12GB 图形处理器和 Intel i9 10900k 中央处理器的设备上进行。

表 4 同心超球面上的分类结果，以及分类所用二次神经元和传统神经元的数量。

维度	3	10	20	100	200
二次神经元数量	**1**	**1**	**1**	**1**	**1**
准确率（%）	**100.00**	**100.00**	**100.00**	**100.00**	**100.00**
传统神经元数量	8	40	150	350	700
准确率（%）	**100.00**	**100.00**	99.75	88.25	82.75

首先，同心超球面的分类结果如表 4 所示，其中二次网络相较于传统型网络的效率得到了充分体现。

无论维度如何，一个二次神经元总能完美分类同心超球面，因为二次神经元中的幂次项能够自然形成超球面边界。相比之下，传统网络即使使用数百个神经元，也无法在高维空间中实现完全分类。这是由于维数灾难，即在高维空间中，需要指数数量的传统神经元才能生成封闭边界，将内层超球面包围起来。接下来，如图3所示，单个二次神经元的决策边界是平滑的圆形，而单神经元的传统网络则无法做到。虽然由六个神经元构成的传统网络可以胜任这项任务，但其决策边界是分段线性函数，不适合这些圆环。

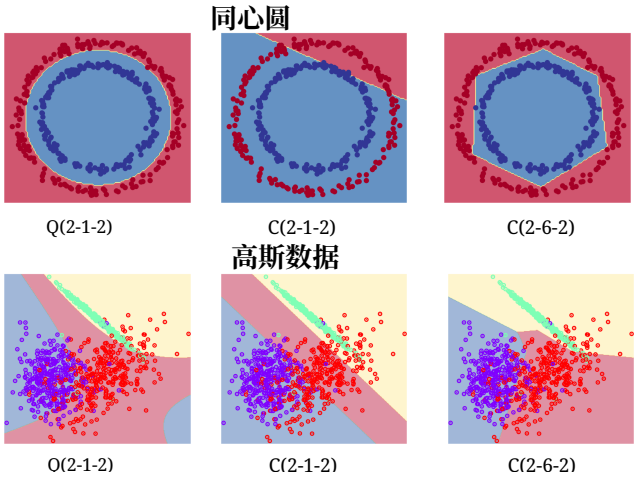

图 3. 二次神经网络和传统型神经网络在圆形数据和高斯混合数据分类任务中的决策边界。Q(2-x-2) 是一个含有 x 个神经元的单隐藏层二次网络，而 C(2-x-2) 是一个含有 x 个神经元的单隐藏层传统网络。

4.2 高斯混合数据

我们利用 Python 的 Scikit-learn 库中的函数 $make\ classification()$– [41] 生成多类别高斯混合数据。每个簇的高斯分布均值为0 ，标准差为1。总计 5000 个样本，在二维空间中分为 3 类以便可视化，在更高维空间中分为 10 类以模拟复杂情形。我们采用全连接单隐藏层 ReLU 神经网络对高斯混合数据进行分类。学习率设为 0.01，批量大小设为 64，训练轮数设为 200。对于二次网络，我们采用 ReLinear训练策略 [9], ，其中二次项的学习率设为 1×10^{-4}。我们采用 Adam 作为优化器。实验在配备 RTX 3080 Ti 12GB 图形处理器和 Intel i9 10900k 中央处理器的设备上进行。

表5 高维高斯混合数据上的分类结果。

结构	#参数量	#浮点运算次数	准确率（%）
C(20-150-10)	4.66K	4.50K	79.60
C(20-150-100-10)	19.26K	19.00K	79.20
Q(20-30-10)	2.82K	2.70K	**82.00**
C(500-90-10)	46.00K	45.90K	63.80
C(500-120-10)	61.33K	61.20K	66.60
Q(500-30-10)	46.02K	45.90K	**67.40**

首先，如图3所示，使用二次神经元可以自然地形成平滑的非线性决策边界，其区分能力比边界呈分段线性的传统神经元更精确。传统网络需要更多神经元才能达到同等精度。此外，分类结果见表5。在神经元数量相同的情况下，二次网络始终优于传统网络，即，二次网络的准确率

在对20维高斯数据进行分类时，二次网络的准确率比传统网络高7.2%。对于500维数据，二次网络的准确率高出10%。即使使用更多神经元，传统网络的性能仍逊于二次网络。鉴于高斯混合假设被广泛认为在机器学习中行之有效，我们的结果表明，二次网络在广泛的应用中可以节省参数。此外，这种节省得益于二次网络中的二次运算能够直接表示平滑的非线性决策边界。

5 公共基准实验

5.1 点云分割

近年来，点云因其广泛的应用而备受关注，例如自动驾驶 [42] 以及作为三维空间中的基础对象表示。在此，我们评估二次网络的参数效率是否同样适用于点云相关任务。我们选用 S3DIS [43] 进行测试，它是广泛采用的点云语义分割基准。所提出的二次网络可直接替换经典模型：DGCNN [44] 和 PointNet [45]。为确保比较公平，我们只将线性聚合替换为二次聚合，不改变其他机制。训练所用的所有超参数均与 PointNet 和 DGCNN 相同。对于 S3DIS，按照先前工作 [44], [45],[46], [47], , 批量大小设为 32。我们使用 4 块 Tesla V100 GPU 训练模型 100 个训练轮数。按照标准做法，首先对原始输入点进行网格采样，生成4，096 个点。除非另有说明，我们使用缩放和抖动作为数据增强。对于 ScanNet，我们训练 100 个训练轮数，权重衰减为 0.1，批量大小为 32。通过采样将输入点数量设为8，192 。除随机抖动外，数据增强方式与 S3DIS 相同。我们使用平均准确率（mAcc）和交并比（mIOU）作为评估指标。

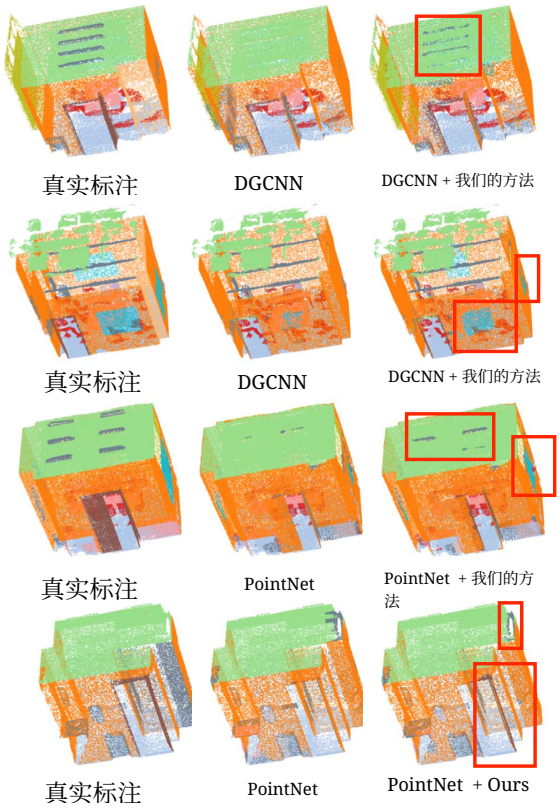

图4. 配备二次神经元的模型，其分割准确率优于原始模型。

表6 S3DIS上的点云分割结果及相应的模型大小。

模型	# 参数量	mAcc	mIOU
PointNet（原始）	3.53M	53.1	45.0
PointNet（加宽）	3.95M	53.4	45.2
PointNet（更深）	3.87M	54.9	46.1
PointNet+我们的方法	3.81M	**56.5**	**48.1**
DGCNN（原始）	0.98M	57.0	49.3
DGCNN（更宽）	1.53M	57.5	49.6
DGCNN（更深）	1.45M	59.1	49.9
DGCNN+我们的方法	1.42M	**61.9**	**50.6**

S3DIS 上的点云分割结果及对应的模型大小如表6所示。我们通过增加特征通道（“更宽”）或堆叠更多层（“更深”），来增加 PointNet [45]和 DGCNN [44] 的参数量。结果表明，二次网络的性能优于更宽、更深的 PointNet 和 DGCNN，尽管后两者的参数更多。更值得注意的是，二次网络的性能大幅领先于其对应模型，例如，比更深的 DGCNN 的 mACC 高 2.8%，比更宽的 PointNet。图4展示了不同模型的点云分割结果。可以看出，采用二次神经元的模型呈现出更清晰的分割效果。由于二次网络的参数量较少，我们认为，性能提升源于二次网络中的非线性聚合，它有助于提取有用的非线性特征。

5.2 图像识别

ImageNet 上的图像识别 [48] 是评估模型的黄金标准。这里我们也将二次网络应用于图像识别任务，以验证其参数效率。ImageNet-1K [48] 基准包含来自 1K 个类别的 128 万张训练图像和 5 万张验证图像。我们使用 MMCLS [49] 作为基础实现环境。我们采用 AdamW优化器，训练 100 轮，使用余弦衰减学习率调度器，并进行 20 轮线性预热。批量大小设为 1024，初始学习率设为 0.001，权重衰减设为 0.05。为进行公平比较，我们对原始 ResNet-50 [50] 和我们的模型使用相同的超参数。二次ResNet模型通过在原始 ResNet-50 的第四阶段移除六层，并替换为一个3层二次神经网络（QNN）得到。

表7 ImageNet 上的分类准确率。

模型	参数数量	Top 1（%）	Top 5（%）
ResNet-50	25M	75.3	92.4
QResNet-47	24M	**75.4**	**92.5**

比较结果见表7，其中 ResNet-50 的性能数据来自 mmclassification 软件包的官方报告 [49]。可以看出，参数量更少的二次网络，其 Top-1 和 Top-5准确率略高于传统网络。由于性能提升和参数节省幅度都较为有限，本实验中二次网络的参数效率也较为有限。我们认为，二次网络通常具有参数效率，但实际能够获得多大的参数效率提升取决于具体任务。

6 现实世界应用

6.1 轴承故障诊断效率

确保风力涡轮机和航空发动机等旋转机械的可靠性，是工业领域的一项关键课题，有助于避免巨大的经济损失。据统计 [51], ，轴承故障占电磁驱动系统故障的比例高达 70%。因此，识别轴承故障的来源至关重要。在此，我们使用二次卷积神经网络（QCNN）解决轴承故障诊断问题。该问题可归结为将故障分为不同类别。与图像分类不同，轴承故障诊断模型处理的是由加速度传感器测得的时序振动信号。因此，这些模型属于一维卷积神经网络。

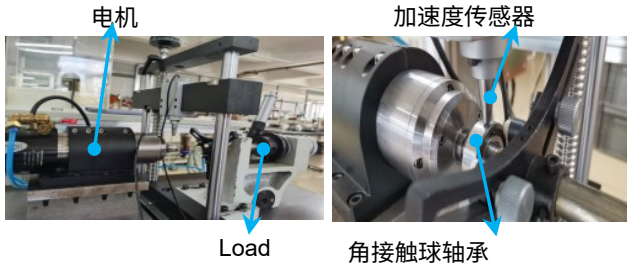

图5. 角接触球轴承试验台。

数据集准备。 我们的合作者收集了角接触球轴承数据集。所选轴承为 HC7003，这是一种常见的高速主轴轴承。轴承信号采集自哈尔滨工业大学航空轴承技术与装备工信部重点实验室。我们采用激光刻蚀，在轴承外圈（OR）、内圈（IR）和滚珠上注入故障，设置 1 至 3 个等级（轻微、中度、严重）。轴承故障均为尺寸相同但深度不同的裂纹。裂纹越深，故障越严重。因此，轴承信号被归为十类 {“健康”“滚珠裂纹（轻微）”“滚珠裂纹（中度）”“滚珠裂纹（严重）”“外圈裂纹（轻微）”“外圈裂纹（中度）”“外圈裂纹（严重）”“内圈（轻微）”“内圈（中度）”“内圈（严重）”}。图5展示了试验台。在试验中，

将电机转速设为恒定值（1800 r/min），并使用 NI USB-6002 以 12kHz 的采样率采集振动信号。我们记录了 47s 的轴承振动（ 561,152 每类 561,152 个点）。图 6 展示了数据集中十个类别的时域原始信号。

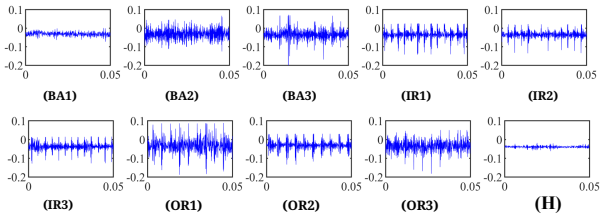

图 6. 数据集中十个类别对应的原始信号。

实验设置。我们使用最先进水平的一维卷积神经网络模型进行比较，例如，DCA-BiGRU [52], AResNet [53], RNN- WDCNN [54], MA1DCNN [55], 和 WDCNN [56]。除 WDCNN 外，这些模型都在卷积神经网络主干中加入了额外模块，例如注意力模块（AResNet、MA1DCNN）和循环卷积网络（RNN- WDCNN）。二次卷积神经网络与 WDCNN 采用相同的结构，但在卷积层中引入了二次神经元。为充分利用数据，每个信号都被切分为 2,048 个点作为输入。数据集按 0.5: 0.25: 0.25 的比例划分为训练集、验证集和测试集。所有方法的超参数设置如下：学习率为 0.3；批量大小为 64；训练轮数为 200。此外，我们采用 Adam 作为优化器。对于二次卷积神经网络，ReLinear策略中二次项的学习率设为 0.1。所有方法均运行十次以计算均值和方差，实验在配备 12GB 显存的 RTX 3080 Ti 图形处理器上进行。

表8 角接触球轴承数据集上不同模型的规模及其平均准确率（%）。其中，GRU 是门控循环单元，RNN 是并行循环神经网络。

模型	通道数	参数量	浮点运算次数	准确率(%)
DCA-BiGRU	[4,2,3]+2*门控循环单元	150.02K	1220.01M	97.42±1.39
AResNet	[3*128,256,2*512]	2650.11K	260.40M	96.14±2.10
RNN-WDCNN	[2*16,2*32,6*64]+循环神经网络	554.10K	54.30M	91.48±3.32
MA1DCNN	[2*32,2*64,2*128]	324.68K	299.70M	93.12±3.10
WDCNN	[16,32,4*64]	66.79K	1.61M	97.60±1.46
QCNN	[6*16]	16.50K	1.39M	**98.80±0.85**

实验结果。 表8列出了不同模型的规模和性能。首先，由于采用了注意力模块和循环神经网络等额外模块，DCA-BiGRU、AResNet、RNN-WDCNN 和 MA1DCNN 的参数量较大，但它们似乎并未显著提升性能。其次，在所有模型中，二次卷积神经网络的准确率最高，方差最低。与 WDCNN 相比，尽管二次卷积神经网络的参数量少 5.5 倍，浮点运算次数仅为其 0.18 倍，但表现更好。相较于其他最先进的竞争模型，二次卷积神经网络更加紧致，浮点运算次数也少得多。因此，我们总结认为，二次卷积神经网络在轴承故障诊断任务中具有较高的参数效率，其出色性能并非源于参数量增加。

6.2 汽车数据集上的效率

汽车经销商需要高分辨率、标准化、多角度的汽车照片，向客户展示其产品。然—

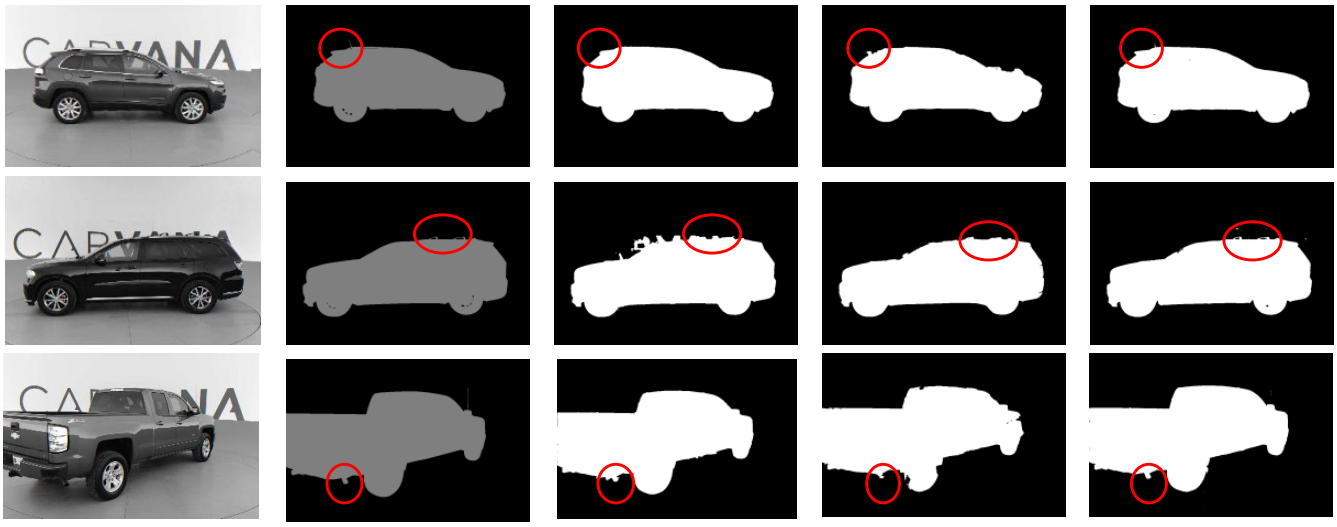

图 7. 三种对比方法的汽车分割结果。

而，复杂的拍摄环境往往使人难以一次拍摄就获得理想的照片。因此，需要专业的图像编辑人员对照片进行后期处理，以提升图像质量。为此，自动且准确的汽车分割是一个重要的中间步骤，有助于减少人工劳动、提高效率。优秀的汽车分割算法能够显著提升最终产品的质量，并最终加快销售。本文评估二次网络在汽车分割任务中是否具有参数效率。

实验设置。 本任务的数据集取自 Carvana图像掩码挑战赛 [57]。为方便起见，我们将该数据集称为 Carvana 数据集。在线汽车公司 Carvana 收集了 318 辆不同汽车的图像，每辆车都有 16 张高分辨率的标准旋转图像，分辨率为 1,918 × 1,080；但强烈的反光和相似颜色的背景使这项任务颇具挑战性。

该挑战赛于 2018年 完成，冠军团队采用的方案是使用基于 VGG 主干网络的基础 U-Net。因此，我们通过将 U-Net 中的神经元替换为二次神经元来构建 QUNet 模型，并将其与该任务中的最先进模型进行比较：UNet++ [58] 和 R2UNet[59]。由于原始图像过大，我们采用一些表现优异团队使用的预处理方法，将每张图像调整为 960×640。数据集按 0.8: 0.1: 0.1 的比例划分为训练集、验证集和测试集。为确保比较公平，所有方法采用相同的超参数。具体而言，学习率为 1×10^{-5}，并采用余弦退火调度 [60]；批量大小为 2；训练轮数为 5。此外，我们使用 RMSProp 作为基线方法的优化器，并使用 Adam 训练 QUNet。具体来说，对于 QUNet，我们采用 ReLinear策略，将二次项的学习率设为 10^{-4}。我们还采用梯度裁剪范数方法，将最大范数设为 0.01，以限制权重过度增长。

实验结果。表 9 总结了不同模型的规模以及在 Carvana 数据集上的浮点运算次数和分割性能。性能指标为平均准确率（mACC）和平均交并比（mIOU）。首先，所有基线方法的表现都令人满意，因为这项任务并不特别困难。图像中只有一个目标，汽车与背景之间的区别十分明显。其次，我们的 QUNet 通过减少所有阶段中的模块数量，降低了参数量和浮点运算次数。尽管如此，QUNet 的表现仍优于其他模型，也就是说，平均交并比甚至提升了 2.4% 以上。最后，如图 7 所示，UNet++ 未能重建完整的边缘，而 R2UNet 也无法精确保留边缘，例如汽车轮廓发生了扭曲。相比之下，QUNet 的分割结果接近真实掩码。由于二次网络在处理汽车分割任务时具有参数效率，因此二次网络在该任务中的优势源于其内在的非线性表征，而非参数量的增加。

表 9 不同模型的规模与浮点运算次数，以及它们在 Carvana 数据集上的分割性能。

方法	UNet 模块	参数数量	浮点运算次数	平均准确率（%）	平均交并比(%)
U-Net++	[32,64,128,256,512]	9.16M	0.33T	99.01	95.62
R2UNet	[64,128,256,512,1024]	39.09M	1.43T	99.08	95.41
QUNet	[32,64,128,256]	5.78M	0.26T	99.59	98.08

6.3 细胞数据集上的效率

细胞分割是显微图像分析领域的一项关键任务，旨在识别并定位图像中细胞的边界 [61]。表现良好的细胞分割算法能够促进众多下游生物医学应用，也是基于细胞图像的分析的基石技术[62]。近年来，深度学习模型取得了显著进展，尤其是 UNet 架构，推动这一领域获得了巨大进步

在这一领域。本文评估二次网络在汽车分割任务中是否具有参数效率。

实验设置。 2018年数据科学竞赛（DSB2018）数据集包含670张在各种条件下采集的分割后的细胞核图像。目标是从图像中找出细胞核，以辅助事后医学分析。这些细胞核来自各种细胞类型 [63]。与 Carvana 数据集中的汽车分割不同，DSB2018 的图像包含大量小型细胞核，这是典型的小目标分割问题，需要模型精细地识别每个细胞核。所有图像都被压缩至 256×256 ，以便直接输入神经网络。

在本实验中，我们选择 ResNet34UNet [64],U-Net ++ [58]、ChannelUNet [65], AttentionUNet、R2UNet [59], 和全卷积网络 [66] 作为基线方法，并相应地设计了 QUNet，以二次神经元替换 U-Net 中的神经元。DSB2018 数据集按 0.8: 0.1: 0.1 的比例划分为训练集、验证集和测试集。为保证比较公平，所有方法采用相同的超参数。学习率、批量大小和训练轮数分别设为 10^{-3}、8 和 20。我们使用 Adam 作为所有方法的优化器。具体而言，对于 QUNet，我们采用 ReLinear策略 [9],，其中二次项的学习率设为 1×10^{-4}。我们还采用梯度裁剪范数方法，并将最大范数值设为 0.01，以限制权重过度增长。

实验结果。 我们在表 10 中总结了分割结果，以及各模型的参数量和浮点运算次数。首先，与其他模型相比，QUNet 的表现更优。它的平均准确率和平均交并比最高，参数量最少，这充分证明了其参数效率。其次，为了直观地比较不同算法的表现，图 8 展示了分割掩码。在较大的圆圈中，出现了一个多余的掩码。ChannelUNet 和 AttentionUNet 将这些斑点误判为细胞并进行了标记。R2UNet 未能分割出完整的细胞。相比之下，QUNet 对这组细胞的分割效果良好。此外，在较小的圆圈中，有两个细胞距离很近，因此 Res34UNet、UNet++、ChannelUNet、AttentionUNet 和 FCN32s 的掩码看起来发生了混叠。只有 R2UNet 和 QUNet 呈现出清晰可辨的边界。我们认为，二次神经元在近似圆形
区域，因此，二次网络更适合分割细胞等具有圆形特征的对象。这是内在表示的结果，而非参数堆叠的结果。

表 10 DSB2018 数据集上的分割结果、模型参数量和浮点运算次数。

模型	UNet 模块	参数数量	浮点运算次数	平均准确率（%）	平均交并比（%）
ResNet34UNet	[64,128,256,512]	21.66M	6.11G	96.68	80.23
U-Net++	[32,64,128,256,512]	9.16M	34.76G	96.47	80.36
ChannelUNet	[64,128,256,512,1024]	49.15M	76.32G	96.84	82.66
AttentionUNet	[64,128,256,512,1024]	34.88M	66.68G	96.90	81.72
R2UNet	[64,128,256,512,1024]	39.09M	0.15T	94.74	72.17
FCN32s	[32,64,128,256,512]	18.64M	22.5G	96.39	79.37
QUNet	[32,64,128,256]	5.78M	27.73G	97.15	82.16

6.4 农业图像数据集上的效率

实验设置。CVPPP叶片分割数据集 [67] 包含标注完善的叶片图像，可用于

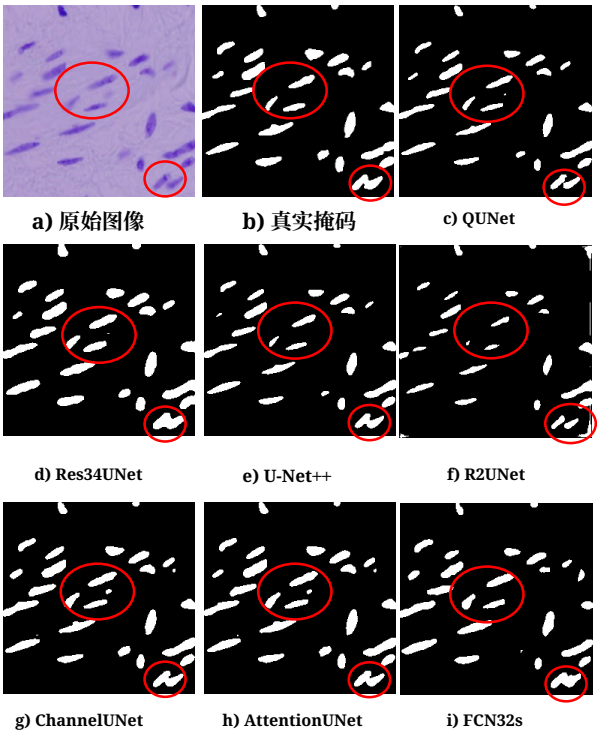

图 8. 七种对比方法的细胞分割图像。

表 11 叶片分割

模型	参数数量	mIOU	F1	召回率	准确率	精确率
Unet	25.81M	80.58%	88.13%	94.45%	99.32%	84.76%
QUnet	24.45M	83.40%	90.08%	89.65%	99.48%	92.72%
PSPNet	21.44M	76.54%	85.43%	94.31%	99.27%	81.06%
QPSPNet	21.40M	82.11%	89.18 %	89.44%	99.50%	89.88%
U-Net++	27.73M	77.32%	86.10%	93.56%	99.22%	83.28%
QUnet++	26.09M	83.27%	90.34%	93.48%	99.35%	88.87%

对种植田中的图像分割任务质量进行基准评测。具体而言，其中两个数据集展示了拟南芥的不同基因型，另一个则展示了不同处理条件下的烟草（Nicoticana tobacum）。在训练过程中，输入图像会在 X 至 X 的范围内随机调整尺寸，然后裁剪为 X × X 的大小，再输入网络。在推理阶段，输入图像会直接输入网络，不会像训练过程中那样使用任何数据增强流程。关于训练/测试集划分，将包含 (512 + 2497) 张图像的 A1 和 A4 文件夹合并后用于训练，而 A2 和 A3 中的 (125 + 109) 张图像用于测试。所有训练超参数均与 U-Net [64],PSPNet [68],和 U-Net++ [58]相同。

实验结果。我们在表11中总结了分割结果和模型规模。引入二次神经元后，平均交并比、F1分数、召回率和精确率等评估指标均可大幅提升。图9展示了不同模型的分割结果。从图9可以看出，二次模型对叶片茎部附近的分割图有所改善。

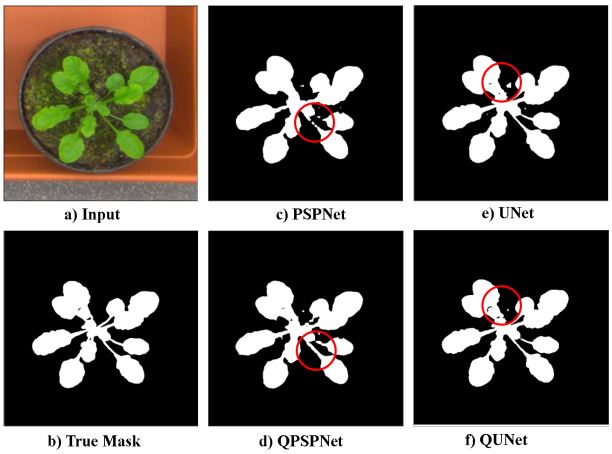

图9. 四种对比方法的叶片分割图像。

7 讨论与结论

我们在表12中总结了二次网络在不同实验中的参数效率。针对各种任务开展的实验表明，二次网络具有参数效率，但与传统型网络一样，二次网络也并非万能。在处理同心超球面、细胞和叶片等圆形数据时，与传统型网络相比，二次网络展现出更高的参数效率。我们相信，这类任务是二次网络充分发挥潜力的理想战场和关键应用。然而，将其应用于ImageNet时，参数效率表现一般。由于没有任何一种神经元类型能够适用于所有情况，我们认为，在深度学习中引入神经元多样性前景广阔。选择神经元时应考虑任务的性质。

表12 任务及其实现的参数效率。

任务	参数效率
同心超球面	★★★★★
高斯混合数据	★★★★
ImageNet	★
点云分割	★★★
轴承故障诊断	★★★★
细胞数据	★★★★
农业数据	★★★★★

本文从理论和实验两方面充分验证了二次网络的效率，证明二次网络的优异表现并非源于参数量增加，而是源于其内在表达能力。我们推导了二次网络对实空间和流形上函数的逼近方法，由此证明，ReLU二次网络比传统型ReLU网络更高效地逼近Sobolev空间单位球中的一类函数。类似结果也推广到了流形上。此外，我们还从Barron空间的角度展示了二次网络的参数效率。进一步地，我们通过合成实验、基准测试和现实世界应用，实证验证了二次网络的参数效率。我们注意到，二次网络擅长处理圆形特征。未来，更多研究应聚焦于针对特定任务设计合适的神经元类型，以最大限度地提升模型性能，同时也应关注二次网络的计算效率。

参考文献

[1] W. S. McCulloch 和 W. Pitts，《神经活动中内在观念的逻辑演算》，《数学生物物理学通报》，第 5 卷，第 4 期，第 115–133 页，1943年。[2] K. Fukushima 和 S. Miyake，《新认知机：一种用于视觉模式识别机制的自组织神经网络模型》，载于《神经网络中的竞争与合作》，第 267–285 页，施普林格，1982年。[3] A. Zador、B. Richards、B. Ölveczky、S. Escola、Y. Bengio、K. Boahen、M. Botvinick、D. Chklovskii、A. Churchland、C. Clopath 等，《迈向下一代人工智能：推动神经人工智能革命》，arXiv预印本 arXiv:2210.08340，2022年。[4] H. Peng、P. Xie、L. Liu、X. Kuang、Y. Wang、L. Qu、H. Gong、S. Jiang、A. Li、Z. Ruan 等，《分子定义细胞类型中单个神经元的形态多样性》，《自然》，第 598 卷，第 7879 期，第 174–181 页，2021年。[5] K. He、X. Zhang、S. Ren 和 J. Sun，《用于图像识别的深度残差学习》，载于计算机视觉与模式识别会议论文集，第 770–778 页，2016年。[6] G. Huang、Z. Liu、L. Van Der Maaten 和 K. Q. Weinberger，《密集连接卷积网络》，载于计算机视觉与模式识别会议论文集，第 4700–4708 页，2017年。[7] F. Fan、D. Wang、H. Guo、Q. Zhu、P. Yan、G. Wang 和 H. Yu，《关于人工神经网络的稀疏快捷连接拓扑》，IEEE人工智能汇刊，2021年。[8] F. Fan、W. Cong 和 G. Wang，《一种新型机器学习神经元》，《国际生物医学工程数值方法期刊》，第 34 卷，第 2 期，文章编号 e2920，2018年。[9] F.-L. Fan、M. Li、F. Wang、R. Lai 和 G. Wang，《二次网络的表达能力与可训练性》，arXiv预印本 arXiv:2110.06081，2021年。[10] F. Fan、H. Shan、M. K. Kalra、R. Singh、G. Qian、M. Getzin、Y. Teng、J. Hahn 和 G. Wang，《用于低剂量CT去噪的二次自编码器（q-ae）》，IEEE医学成像汇刊，第 39 卷，第 6 期，第 2035–2050 页，2019年。[11] J.-X. Liao、H.-C. Dong、Z.-Q. Sun、J. Sun、S. Zhang 和 F.-L. Fan，《用于有效且可解释轴承故障诊断的嵌入注意力的二次网络（Qttention）》，arXiv预印本 arXiv:2206.00390，2022年。[12] F. Fan、J. Xiong 和 G. Wang，《二次深度网络的通用逼近》，《神经网络》，第 124 卷，第 392 页，2020年，第 383 页。[13] D. Yarotsky，《深度ReLU网络逼近的误差界》，《神经网络》，第 94 卷，第 103–114 页，2017年。[14] M. Chen、H. Jiang、W. Liao 和 T. Zhao，《低维流形上函数的深度ReLU网络高效逼近》，NeurIPS，第 32 卷，第 8174–8184 页，2019年。[15] R. A. DeVore、R. Howard 和 C. Micchelli，《最优非线性逼近》，《数学手稿》，第 63 卷，第 4 期，第 469–478 页，1989年。[16] A. G. Ivakhnenko，《复杂系统的多项式理论》，IEEE系统、人和控制论汇刊，第 4 期，第 378 页，1971年，第 364 页。[17] Y. Shin 和 J. Ghosh，《π-σ网络：一种用于模式分类与函数逼近的高效高阶神经网络》，载于 IJCNN-91-西雅图国际神经网络联合会议，第 1 卷，第 13–18 页，电气电子工程师学会，1991年。[18] S. Milenkovic、Z. Obradovic 和 V. Litovski，《二阶神经网络中基于退火的动态学习》，载于国际神经网络会议（ICNN'96）论文集，第 1 卷，第 458–463 页，电气电子工程师学会，1996年。[19] G. Chrysos、S. Moschoglou、G. Bouritsas、J. Deng、Y. Panagakis 和 S. P. Zafeiriou，《深度多项式神经网络》，IEEE模式分析与机器智能汇刊，2021年。[20] G. Liu 和 J. Wang，《树突网络：用于分类、回归和系统辨识的白盒模块》，IEEE控制论汇刊，2021年。[21] G. Zoumpourlis、A. Doumanoglou、N. Vretos 和 P. Daras，《用于基于CNN学习的非线性卷积滤波器》，载于IEEE国际计算机视觉会议论文集，第 4761–4769 页，2017年。

[22] P. Micikevicius、S. Narang、J. Alben、G. Diamos、E. Elsen、D. Garcia、B. Ginsburg、M. Houston、O. Kuchaiev、G. Venkatesh 等，《混合精度训练》，arXiv预印本 arXiv:1710.03740，2017年。[23] Y. Jiang、F. Yang、H. Zhu、D. Zhou 和 X. Zeng，《非线性卷积神经网络：通过二次卷积改进卷积神经网络》，《神经计算与应用》，第 32 卷，第 12 期，第 8507–8516 页，2020年。[24] P. Mantini 和 S. K. Shah，《卷积二次神经网络》，载于2020年第25届国际模式识别会议（ICPR），第 9819–9826 页，电气电子工程师学会，2021年。[25] M. Goyal、R. Goyal 和 B. Lall，《通过归一化激活改进多项式神经网络》，载于2020年 IJCNN，第 1–8 页，电气电子工程师学会，2020年。[26] J. Bu 和 A. Karpatne，《二次残差网络：用于求解涉及偏微分方程的物理学正问题与逆问题的新型神经网络》，载于2021年 SIAM国际数据挖掘会议（SDM）论文集，第 675–683 页，SIAM，2021年。[27] Z. Xu、F. Yu、J. Xiong 和 X. Chen，《QuadraLib：用于架构优化和设计探索的高性能二次神经网络库》，《机器学习与系统论文集》，第 4 卷，第 503–514 页，2022年。[28] J. Kileel、M. Trager 和 J. Bruna，《关于深度多项式神经网络的表达能力》，《神经信息处理系统进展》，第 32 卷，2019年。[29] J.-X. Liao、B.-J. Hou、H.-C. Dong、H. Zhang、J. Ma、J. Sun、S. Zhang 和 F.-L. Fan，《由二次神经元赋能的异构自编码器》，arXiv预印本 arXiv:2204.01707，2022年。[30] E. A. Rocamora、M. F. Sahin、F. Liu、G. G. Chrysos 和 V. Cevher，《多项式网络的可靠且完备的验证》，arXiv预印本 arXiv:2209.07235，2022年。[31] T. Nguyen、A. Kashani、T. Ngo 和 S. Bordas，《用于预测泡沫混凝土强度的高阶神经元深度神经网络》，《计算机辅助土木与基础设施工程》，第 34 卷，第 4 期，第 316–332 页，2019年。[32] T. Qi 和 G. Wang，《二次神经网络在高斯混合数据分类方面优于传统型神经网络》，《工业、生物医学与艺术可视化计算》，第 5 卷，文章 11，2022年，第 1 页。[33] G. G. Chrysos、M. Georgopoulos、J. Deng、J. Kossaifi、Y. Panagakis 和 A. Anandkumar，《用多项式神经网络增强深度分类器》，载于欧洲计算机视觉会议，第 692–716 页，施普林格，2022年。[34] L. W. Tu，《流形》，载于《流形导论》，第 47–83 页，施普林格，2011年。[35] E. Aamari、J. Kim、F. Chazal、B. Michel、A. Rinaldo 和 L. Wasserman，《估计流形的可达半径》，《电子统计学期刊》，第 13 卷，第 1 期，第 1359–1399 页，2019年。[36] Z. Shen、H. Yang 和 S. Zhang，《由神经元数量刻画的深度网络逼近》，arXiv预印本 arXiv:1906.05497，2019年。[37] J. Lu、Z. Shen、H. Yang 和 S. Zhang，《光滑函数的深度网络逼近》，《SIAM数学分析期刊》，第 53 卷，第 5 期，第 5465–5506 页，2021年。[38] C. Ma、L. Wu 等，《Barron空间及神经网络模型的流诱导函数空间》，《构造逼近》，第 55 卷，第 1 期，第 369–406 页，2022年。[39] A. R. Barron，《S形函数叠加的通用逼近界》，《IEEE信息论汇刊》，第 39 卷，第 3 期，第 930–945 页，1993年。[40] Y. V. Prokhorov，《随机过程的收敛与概率论中的极限定理》，《概率论及其应用》，第 1 卷，第 2 期，第 157–214 页，1956年。[41] F. Pedregosa、G. Varoquaux、A. Gramfort、V. Michel、B. Thirion、O. Grisel、M. Blondel、P. Prettenhofer、R. Weiss、V. Dubourg 等，《Scikit-learn：Python机器学习》，《机器学习研究期刊》，第 12 卷，10月号，第 2825–2830 页，2011年。[42] Y. Cui、R. Chen、W. Chu、L. Chen、D. Tian、Y. Li 和 D. Cao，《自动驾驶中的图像与点云融合深度学习：综述》，《IEEE智能交通系统汇刊》，第 23 卷，第 2 期，第 722–739 页，2021年。[43] I. Armeni、O. Sener、A. R. Zamir、H. Jiang、I. Brilakis、M. Fischer 和 S. Savarese，《大规模室内空间的三维语义解析》，载于计算机视觉与模式识别会议，第 1534–1543 页，2016年。[44] Y. Wang、Y. Sun、Z. Liu、S. E. Sarma、M. M. Bronstein 和 J. M. Solomon，《用于点云学习的动态图卷积神经网络》，《ACM图形学汇刊》，2019年。

[45] C. R. Qi、H. Su、K. Mo 和 L. J. Guibas，“Pointnet：用于三维分类与分割的点集深度学习”，载于计算机视觉与模式识别会议，页码652–660，2017年。[46] Z. Zhang、B.-S. Hua 和 S.-K. Yeung，“Shellnet：利用同心壳层统计量实现高效点云卷积神经网络”，载于国际计算机视觉会议（ICCV），2019年。[47] C. R. Qi、L. Yi、H. Su 和 L. J. Guibas，“Pointnet++：度量空间中点集的深度层次特征学习”，NeurIPS，第30卷，2017年。[48] J. Deng、W. Dong、R. Socher、L.-J. Li、K. Li 和 L. Fei-Fei，“Imagenet：大规模层次图像数据库”，载于2009年计算机视觉与模式识别会议，页码248–255，电气电子工程师学会，2009年。[49] M. Contributors，“OpenMMLab的图像分类工具箱与基准测试。”
https://github.com/open-mmlab/ mmclassification，2020年。[50] K. He、X. Zhang、S. Ren 和 J. Sun，“用于图像识别的深度残差学习”，2015年。[51] A. H. Bonnett 和 C. Yung，“效率提升与可靠性提升”，IEEE工业应用杂志，第14卷，第1期，页码29–36，2008年。[52] X. Zhang、C. He、Y. Lu、B. Chen、L. Zhu 和 L. Zhang，“基于注意力机制的小样本故障诊断”，Measurement，第187卷，文章编号110242，2022年。[53] X. Zhong、F. Wang 和 H. Ban，“为核电厂旋转机械故障诊断开发即插即用的抗噪模块”，[54] A. Shenfield 和 M. Howarth，“一种用于检测和识别滚动轴承故障的新型深度学习模型”，Sensors，第20卷，第18期，文章编号5112，2020年。[55] H. Wang、Z. Liu、D. Peng 和 Y. Qin，“基于多注意力一维卷积神经网络的轮对轴承故障诊断判别特征理解与学习”，IEEE工业信息学汇刊，第16卷，第9期，页码5735–5745，2019年。[56] W. Zhang、G. Peng、C. Li、Y. Chen 和 Z. Zhang，“一种基于原始振动信号的新型深度学习故障诊断模型，具有良好的抗噪与域适应能力”，Sensors，第17卷，第2期，文章编号425，2017年。[57] Carvana，“Carvana图像掩码挑战赛。”
https://www.kaggle.com/competitions/carvana-image-masking-challenge/overview。[58] Z. Zhou、M. M. Rahman Siddiquee、N. Tajbakhsh 和 J. Liang，“U-Net++：一种用于医学图像分割的嵌套式U-Net架构”，载于《医学图像分析中的深度学习与临床决策支持多模态学习》，页码3–11，施普林格，2018年。[59] M. Z. Alom、M. Hasan、C. Yakopcic、T. M. Taha 和 V. K. Asari，“基于U-Net（R2U-Net）的循环残差卷积神经网络，用于医学图像分割”，arXiv预印本 arXiv:1802.06955，2018年。[60] I. Loshchilov 和 F. Hutter，“随机梯度下降热重启：带热重启的随机梯度下降”，arXiv预印本 arXiv:1608.03983，2016年。[61] R. Lewis，“细胞分割分析权威指南”，2021年。
https://blog.biodock.ai/ definitive-guide-to-cell-segmentation-analysis/。[62] F. Lux 和 P. Matula，“结合标记控制分水岭与深度学习进行细胞分割”，arXiv预印本 arXiv:2004.01607，2020年。[63] DSB2018，“2018年数据科学竞赛。”
https://www.kaggle.com/competitions/data-science-bowl-2018/overview。[64] O. Ronneberger、P. Fischer 和 T. Brox，“U-Net：用于生物医学图像分割的卷积网络”，载于国际医学图像计算与计算机辅助干预会议，页码234–241，施普林格，2015年。[65] Y. Chen、K. Wang、X. Liao、Y. Qian、Q. Wang、Z. Yuan 和 P.-A. Heng，“Channel-UNet：用于肝脏和肿瘤分割的空间通道卷积神经网络”，Frontiers in Genetics，第10卷，文章编号1110，2019年。[66] W. Sun 和 R. Wang，“结合数字表面模型的超高分辨率遥感图像语义分割全卷积网络”，IEEE地球科学与遥感快报，第15卷，第3期，页码474–478，2018年。[67] H. Scharr、M. Minervini、A. Fischbach 和 S. A. Tsaftaris，“带注释的莲座状植物图像数据集”，载于欧洲计算机视觉会议。苏黎世¨，瑞士，页码6–12，2014年。[68] H. Zhao、J. Shi、X. Qi、X. Wang 和 J. Jia，“金字塔场景解析网络”，载于IEEE计算机视觉与模式识别会议论文集，页码2881–2890，2017年。