# 数据感知定制激活函数可降低神经网络误差

Fuchang Gao[1] 和 Boyu Zhang[2]

[1] 爱达荷大学数学与统计科学系，875 Perimeter Drive，MS 1103，美国爱达荷州莫斯科市，邮编 83844-1103，fuchang@uidaho.edu，个人主页：https://www.webpages.uidaho.edu/ fuchang/[2] 爱达荷大学建模、协作与创新研究所，875 Perimeter Dr，MS 1122，美国爱达荷州莫斯科市，邮编 83844-1122

arXiv:2301.06635v1 [cs.LG] 16 Jan 2023

**摘要。** 激活函数在神经网络中发挥着关键作用，但目前的现成神经网络很少关注所用激活函数的具体选择。本文表明，数据感知的激活函数定制能够显著降低神经网络误差。我们首先从线性代数角度简要解释激活函数在神经网络中的作用；随后结合 Diaconis-Shahshahani 逼近定理，提出一组优良激活函数的判定标准。作为案例研究，我们考察目标函数部分可交换的回归任务，$i.e.\ f(u,\ v,\ w) = f(v,\ u,\ w)$，其中 $u,\ v \in \mathbb{R}^d$ 和 $w \in \mathbb{R}^k$，并证明对于此类目标函数，至少在一层中使用偶激活函数，即可保证预测值保留部分可交换性，从而实现最佳性能。由于偶激活函数在实践中鲜少使用，我们依据上述标准设计了“seagull”偶激活函数 $\log(1 + x^2)$。在二十多个维数为 9–25、具有不同局部光滑性、曲率和可交换程度的实例上进行实证测试后发现，只需在一个已优化的神经网络中简单替换为“seagull”激活函数，就能使误差降低一个数量级。这种改进在将激活函数替换应用于可交换变量首次相连的层时最为显著。尽管低维数据的改进幅度最大，但在 CIFAR10 图像分类数据集上的实验表明，即使对于高维情形，使用“seagull”也能降低误差。这些结果共同凸显了定制激活函数作为一种提升神经网络性能的通用方法所具有的潜力。

**关键词：** 激活函数、神经网络、部分可交换性、定制

## 1 引言

过去十年间，深度神经网络在高维数据分析方面取得了显著成功，尤其是在计算机视觉和自然语言处理领域。近年来，这些成功激发了人们对将深度神经网络应用于从实验数据中进行科学发现的浓厚兴趣，包括在各种问题场景中识别隐藏的物理原理和控制方程 [1,10,17,19]。

科学原理的发现既可以是概念性和定性的，也可以是精确且定量的，但人们的一个共同愿望是发现目标变量与输入变量之间可解释且定量的关系。要实现这一宏大目标，必须将现有科学知识融入神经网络设计。尽管人们已致力于提升深度神经网络的可解释性，但它们仍在很大程度上被视为“黑箱”。这种“黑箱”特性有时很方便，因为用户无需完全理解其内部运作方式，就能创建实用的模型并获得结果。然而，这些模型往往难以在不同应用场景之间实现良好泛化，导致用户难以针对感兴趣的新应用重新设计模型。

在现实世界应用中，数据集往往附带额外信息，而这些信息对领域专家非常重要，例如变量之间潜在的约束，或关于哪些变量（测量值）比其他变量更可靠的元数据。由于计算成本、抽样偏差或模型本身的局限等各种原因，从数据中从头推断这些额外信息可能很困难

。例如，对于基于图像进行目标分类，垂直轴反射不变性是一项必不可少的知识，但仅凭有限的训练数据很难学到这一信息。

与此同时，将这类知识融入模型，有望提升模型在当前特定数据集上的表现。目前，通常通过数据增强或在损失函数中引入额外项来实现这一点。本文强调，依据底层数据结构对神经网络进行解析式架构设计，并定制激活函数，相较于数据增强或定制损失函数，是一种更高效的方法。

许多研究者都认同这一观点。例如， [22] 研究了如何调整深度神经网络结构，以实现另一种对称性。其方法提升了模型的泛化能力，这并不令人意外。另一方面，通过有目的地设计神经网络架构及其组件，并在数据上测试其有效性，我们或许可以让机器学习模型引导我们发现数据中隐藏的新科学规律。

本文首先从线性代数的角度解释神经网络的内部工作机制。这种简单而直观的解释有助于我们更好地理解激活函数在神经网络中的作用。随后，我们结合 Diaconis-Shahshahani 逼近定理，讨论定制激活函数的重要性及其准则。作为案例研究，我们考察目标函数 $f$ 部分可交换这一特殊但常见的情形，即 $f(u,\ v,\ w) = f(v,\ u,\ w)$ 对于$u,\ v \in \mathbb{R}^d$，且 $w \in \mathbb{R}^k$。我们证明，对于这类目标函数，至少在一层中使用偶激活函数，可以保证预测值保留部分可交换性，从而实现最佳性能。由于偶激活函数在实践中很少使用，我们根据良好激活函数的准则设计了一种“海鸥”激活函数 $\log(1+x^2)$，并在二十多个维数为 9–64 的回归问题上进行了实证测试，结果显示性能显著提升。无论原始激活函数、损失函数、优化器如何，也无论数据中是否存在潜在噪声，都观察到了性能提升；不过，当替代激活函数应用于可交换变量首次相连接的那一层时，提升最为明显。虽然低维数据上的提升最大，但我们在 CIFAR10 图像分类问题上的实验表明，即使对于可交换程度较低的高维情形，提升仍然可能十分显著。

## 2 激活函数作用的线性代数解释

对于具有 $k$个隐藏层的深度神经网络，其近似可以写为:

$$f(x_1, x_2, \ldots, x_d) \approx \sum_{i=1}^{m_k} g\left(\phi_i^k(x_1, x_2, \ldots, x_d)\right) w_i^k + b^k,$$

其中 $\phi_i^k$由 $\phi_i^1(x_1,\ x_2,\ \ldots\ldots,\ x_d) = x_1 w_{1i}^1 + x_2 w_{2i}^1 + \cdots + x_d w_{di}^1 + b_i^1$递归定义，a nd

$$\phi_i^r(x_1, x_2, \ldots, x_d) = \sum_{j=1}^{m_r} g\left(\phi_j^{r-1}(x_1, x_2, \ldots, x_d)\right) w_{ji}^r + b_i^r, \quad r = 2, 3, \ldots, k.$$

众所周知，至少有一个隐藏层的神经网络可以近似任意多元连续函数（[6] [8] [9]）。虽然证明并不困难，但通常较为抽象。例如， [6] 中的证明使用了 Hahn-Banach 定理。为了更好地理解神经网络的内部工作机制，我们在此使用线性代数来解释其近似机制。为清晰起见，我们只考虑具有一个隐藏层的神经网络，并假设误差用均方误差衡量。

对于每个输入数据 $\mathbf{x}^j = (x_{j1},\ x_{j2},\ \ldots\ldots,\ x_{jd})$，其输出

$$\sum_{i=1}^{m} g\left(x_{j1} w_{1i} + x_{j2} w_{2i} + \cdots + x_{jd} w_{di} + b_i\right) \alpha_i + \beta$$

旨在逼近目标值 $y_j$。如果我们记

$$X = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \ldots & \vdots \\ x_{N1} & x_{N2} & \cdots & x_{Nd} \end{pmatrix}, \quad W = \begin{pmatrix} w_{11} & w_{12} & \cdots & w_{1m} \\ w_{21} & w_{22} & \cdots & w_{2m} \\ \vdots & \vdots & \ldots & \vdots \\ w_{d1} & w_{d2} & \cdots & w_{dm} \end{pmatrix}, \tag{1}$$

并记 $y=(y_1,y_2,\ldots,y_N)^T$, $b=(b_1,b_2,\ldots,b_m)$, $\alpha=(\alpha_1,\alpha_2,\ldots,\alpha_m)^T$，并令 $\mathbb{1}$ 为一个 $N$ 维列向量，其各元素均为 1，则该问题可表述为求解以下问题的最小二乘解

$$g(XW+\mathbb{1}b)\alpha+\mathbb{1}\beta=y,$$

其中，$g(U)$表示将函数 $g$应用于 $U$的每个条目。

当 $g(XW+\mathbb{1}b)$固定时，存在一个解析解

$$\alpha=\left(g(XW+\mathbb{1}b)-\overline{g(XW+\mathbb{1}b)}\right)^{+}y, \tag{2}$$

$$\beta=\overline{y}-\overline{g(XW+\mathbb{1}b)}\alpha, \tag{3}$$

其中，$U^{+}$表示矩形矩阵 $U$的摩尔–彭若斯广义逆，$M$表示矩阵 $M$的行均值。因此，这个问题的主要目标是找到 $W$和 $b$，使得 $g(XW+\mathbb{1}b)$的仿射包能够尽可能好地逼近 $N$维向量 $y$。

注意，矩阵 $XW+\mathbb{1}b$ 的秩至多为 $d+1$。激活函数的作用是生成一个秩高得多的矩阵 $g(XW+\mathbb{1}b)$，以便其列向量的仿射组合能够逼近 $N$维向量 $y$。并非每个函数 $g$都有这样的潜力。例如，如果 $g(t)$是一个次数为 $p$的多项式，那么无论$W$的大小如何，矩阵 $g(XW+\mathbb{1}b)$的秩都至多为 $\binom{d+p}{p}$。幸运的是，以下定理指出，多项式基本上是唯一受到这一限制的函数。

**定理1**。假设 $g(t)$ 在某个开区间 $I\subset R$ 内不是多项式，且该区间内存在一点使得 $g$ 具有无穷阶可微性。对于任意互不相同的点 $x_i$, $1\le i\le N$ *in* $R^d$，且满足$N\gg d$，以及任意 $d<m\le N$，存在一个 $d\times m$ 矩阵 $W$和一个 $m$维行向量 $b$，使得$g(XW+\mathbb{1}b)$ 的秩为 $m$，其中 $X$ 是一个 $N\times d$ 矩阵，其第 $i$ 行为向量 $x_i$。若 $g(t)=\max\{t,0\}$，同样的结论也成立。另一方面，若 $g$是次数为 $p$的多项式，则无论 $W$的大小如何，$g(XW+\mathbb{1}b)$ 的秩至多为 $\binom{p+d}{p}$。

证明。由于这些 $N$点互不相同，存在一个方向 $(w_1,\ w_2,\ \ldots,\ w_d)^T$，使得各点在该向量上的标量投影互不相同。也就是说，以下序列中的每一项

$$c_k:=x_{k1}w_1+x_{k2}w_2+\cdots+x_{kd}w_d,\ k=1,2,\ldots,N$$

具有一个不同的非零值。

首先考虑这样的情形：$g$ 在 $I$ 上无穷次可微，且不是多项式。在这种情形下，存在 $b_0\in I$，使得导数 $g^{(k)}(b_0)\neq 0$ 对 $0\le k\le N$ 成立。我们证明，对于任意 $d<m\le N$，都可以选择一个 $d\times m$ 矩阵 $W=(w_1,w_2,\ldots,w_d)^T(t_1,t_2,\ldots,t_m)$和 $b=(b_0,b_0,\ldots,b_0)$，使矩阵 $g(XW+\mathbb{1}b)$的秩为 $m$。假设这不成立。令 $S$ 为 $g(XW+\mathbb{1}b)$前$m-1$列的线性张成空间，并考虑矩阵 $W(t)=(w_1,w_2,\ldots,w_d)^T(t_1,t_2,\ldots,t_{m-1},t)$。由于对所有 $t$，矩阵 $g(XW(t)+\mathbb{1}b)$的最后一列，即向量 $v(t):=(g(c_1t+b_0)\ g(c_2t+b_0)\ldots,g(c_Nt+b_0))^T$都属于空间 $S$，而该空间与 $t$ 无关。$v(t)$的导数也如此。对 $v(t)$关于 $t$ 求 $k$次导数，其中 $k=0$取1, ..., $m-1$，并在 $t=0$处求值，得到$v^{(k)}(0)=g^{(k)}(b_0)(c_1^k,c_2^k,\ldots,c_N^k)^T$。由于对所有 $k=0,\ g^{(k)}(b_0)\neq 0$均成立，其中 $k=0$取1, 2, ... $m-1$，因此有$(c_1^k,c_2^k,\ldots,c_N^k)^T\in S$, $0\le k\le m-1$。这就产生了矛盾，因为 $m$ 个向量 $(c_1^k,c_2^k,\ldots,c_N^k)$，$0\le k\le m-1$ 线性无关，而 $S$ 的维数至多为 $m-1$。因此，对于某个 $d\times m$矩阵 $W$ 和某个 $m$维行向量 $b$，$g(XW+\mathbb{1}b)$的秩为 $m$。

现在，我们考虑 $g(t)=\max\{t,0\}$ 的情况。不失一般性，我们假设 $c_1>c_2>\cdots>c_N$。通过选择 $W=(w_1,\ w_2,\ \ldots\ldots,\ w_d)^T(1,\ 1,\ \ldots\ldots,\ 1)$，并令 $b=(-s_1,\ -s_2,\ \ldots\ldots,\ -s_m)$，其中 $c_k>s_k\ge c_{k+1}$、$1\le k\le m$，我们可以看到，$g(XW+\mathbb{1}b)$ 的第 $k$ 列是向量 $(c_1-s_k,\ c_2-s_k,\ \ldots\ldots,\ c_k-s_k,\ 0,\ 0,\ \ldots\ldots,\ 0)^T$，其第 $k$ 个坐标非零，且第 $k$ 个坐标之后的所有坐标均为 0。显然，这类列向量的线性张成空间的维数为 $m$。

反过来，假设 $g$ 是次数为 $p$ 的多项式，那么 $g(XW+\mathbb{1}b)$ 的所有列向量都可以表示为如下形式的逐元素乘积的线性组合：$u_{j_1}u_{j_2}\cdots u_{j_k}$，其中 $1\le k\le p$，$u_j=(x_{1j},\ x_{2j},\ \ldots\ldots,\ x_{Nj})^T$。因此，$g(XW+\mathbb{1}b)$ 的秩至多为 $\binom{d+p}{p}$。

注*1*。 众所周知，连续非多项式激活函数能够保证在紧域上一致可逼近（ [15]中的定理3.1）。这里我们仅在一个更强的假设下给出了一个简要的线性代数证明，即该函数在某一点具有无穷次可微性，为下一节构造新的激活函数铺平了道路。此外，关于 ReLU 函数的结果也是已知的（例如， [23]中的定理1）。

## 3 定制激活函数

虽然除多项式外，所有函数都可以作为激活函数，但选用合适的激活函数可以大幅减少所需的组合项数量。因此，定制激活函数可能非常重要。Diaconis 和 Shahshahani 的以下定理 [3] 可视为定制激活函数的首次推动：任意有界闭区域 $\mathbb{R}^d$ 上的多元连续函数 $f$ 都可以近似表示为：

$$f(x_1, x_2, \ldots, x_d) \approx \sum_{i=1}^{m} g_i \left(a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{id}x_d\right), \tag{4}$$

其中 $g_i$ 是非线性连续函数（[3] 定理 4）。因此，选择合适的激活函数 $g_i$对于准确近似非常重要。

当前的现成神经网络很少关注如何选择激活函数，而是简单地使用少数几种固定的激活函数之一，例如 ReLU（即 max($x$,0)）。这种做法似乎效果不错，尤其适用于分类问题，但对于回归问题而言，可能并非最高效的方式。这极大地限制了神经网络的潜力。

恰当选择激活函数可以显著提高神经网络的效率。例如，为了逼近函数 $z = \sin(xy)$，神经网络可以采用以下逐步策略（5），将（$x, y$）$^T$ 变为 sin($xy$):

$$\begin{pmatrix} x \\ y \end{pmatrix} \xrightarrow{\text{linear}} \begin{pmatrix} x+y \\ x-y \end{pmatrix} \xrightarrow{(\cdot)^2} \begin{pmatrix} (x+y)^2 \\ (x-y)^2 \end{pmatrix} \xrightarrow{\text{linear}} xy \xrightarrow{\sin(\cdot)} \sin(xy) \tag{5}$$

换言之，如果在第一层和第二层分别将非线性函数 $f(t) = t^2$ 和 $g(s) = \sin(s)$ 用作激活函数，那么甚至可以发现精确的函数。当然，如果不知道 $f$的闭式表达式，就不可能像上面所举例的那样选择精确的激活函数。不过，借助领域知识中可能存在的一些部分信息，我们可以据此设计更好的激活函数。这正是本文的目标。需要指出的是，这种解析设计方法不同于 Japtap 等人采用的自适应激活函数方法，他们将逐层和逐神经元的局部自适应激活函数引入物理信息神经网络 [18], ，并表明这类自适应激活函数能够提升一组基准数据集上的训练速度和准确率。

定制激活函数的主要标准，是它们能否轻松生成输入向量线性张成空间之外的向量。我们在上文分析过，只要激活函数不是多项式，就能生成这样的向量，其仿射组合可以近似所有目标向量。除此之外，激活函数的表现通常取决于数据集、模型超参数和网络架构。在图像分类中，ReLU 和 Leaky ReLU 是常见选择。循环神经网络则更常使用 sigmoid 和双曲正切函数。冒着招致批评的风险，我们在此提出三项标准：

**1. 光滑性**。对于目标函数光滑的回归问题，假设激活函数具有光滑性是很自然的。考虑在有界区域上，使用采用基于梯度的优化器的神经网络来近似未知多元连续函数 $f(x_1, x_2, \ldots\ldots, x_d)$ 的回归问题。要使基于梯度的算法有效运行，函数$f$ 的偏导数必须存在。因此，函数 $f$ 必须是有界利普希茨的，即存在常数$L$，使得对于有界区域中的任意 $X = (x_1, x_2, \ldots\ldots, x_d)$ 和 $Y = (y_1, y_2, \ldots\ldots, y_d)$，都有

$$|f(X) - f(Y)| \leq L\sqrt{\Sigma_{i=1}^{d}(x_i - y_i)^2}. \tag{6}$$

如果我们允许在豪斯多夫距离上存在一些误差 $\varepsilon$ 来近似目标函数的水平集 $K$，那么我们近似的就是集合 $K + B_{d-1}(\varepsilon)$，其中 $B_{d-1}(\varepsilon)$ 是 $\mathbb{R}^{d-1}$ 中半径为 $\varepsilon$的球。已知 $K + B_{d-1}(\varepsilon)$ 的曲面具有 $C^{1,\,1}$（但 $C^2$）的光滑性，即使 $K$ 是 $c^\infty$ 光滑的 [12]。要近似这样的集合，最好使用具有相同光滑程度的形状。因此，激活函数最好具有连续一阶导数。没有必要具备更高的光滑程度。换言之，没有必要使用 $\log(1 + x^4)$ 这类函数来提高 $x = 0$ 附近的光滑性。

**2. 局部曲率**。尽管根据定理1的证明，我们只需要存在一个点 $x_0$，使激活函数 $g$ 在所有 $k$处都满足 $g^{(k)}(x_0) \neq 0$，但为了免去在$g(XW + \mathbb{1}b)$中选择最佳 $W$ 和 $b$ 以匹配点 $x_0 = XW + \mathbb{1}b$的负担，我们要求 $g^{(k)}(x) \neq 0$ 在$x$ 尽可能多的位置对所有 $k$ 都成立。换句话说，$g$ 不是分段多项式。乍看之下，这似乎与使用三次样条逼近平滑函数的常见做法相矛盾。区别在于，这个激活函数用于变换中间变量，而不是对函数本身进行局部逼近。

**3. 增长率**。从降低泛化误差的角度来看，好的激活函数不应对少数特定变量赋予过大的权重。$X$中的离群值可能会导致值 $XW + \mathbb{1}b$ 显著增大。如果 $g'(t)$在 $|t|$ 较大时也很大，那么基于梯度的优化器很可能会大幅调整权重矩阵 $W$中的相应参数，使神经网络难以训练；少数变量的微小扰动也可能显著改变预测值。因此，如果激活函数用于多个层，其在无穷远处的增长速度不应快于线性增长。常见的激活函数有几十种。分析这些函数后，我们注意到，除二值阶跃函数、Sigmoid和双曲正切函数等少数有界函数外，其余激活函数（如 ReLU、Leaky ReLU、PReLU、ELU、GELU、SELU、SiLU、Softplus、Mish）都随 $x \to \infty$线性增长。

为同时考虑这三个标准，并填补有界增长与线性增长之间的空白，我们提出考虑以下激活函数：$g_1(x) = \log(1+|x|^\alpha)$、$g_2(x) = \text{sign}(x)\log(1+|x|^\alpha)$，以及当 $x > 0$ 时的函数 $g_3(x) = \log(1 + |x|^\alpha)$；当 $x \leq 0$ 时的 $g_3(x) = 0$。对于 $1 \leq \alpha \leq 2$，当正的 $x$ 接近 0 时，函数 $g_1(x)$ 的表现类似于 $x^\alpha$；当 $|x|$ 较大时，其表现类似于 $\alpha\log x$。函数 $g_2$ 和 $g_3$ 是相应的修改形式。具体而言，我们引入以下两个函数：$\log(1 + x^2)$ 和 $\text{sign}(x)\log(1 + |x|)$。为便于引用，我们将 $\log(1+x^2)$ 称为 Seagull 激活函数，因为其图像看起来像一只飞翔的海鸥；将 $\text{sign}(x)\log(1 + |x|)$ 称为线性对数单元（LLU）。我们将在本文后面展示 Seagull 激活函数的有效性。 [4] 的图3展示了另一个例子，表明具有对数增长的激活函数明显优于具有有界增长或线性增长的激活函数。

## 4 案例研究：部分可交换目标

在本节中，我们考虑满足以下关系的 $f$

$$f(u, v, w) = f(v, u, w) \tag{7}$$

对于 $u = (x_1, x_2, \ldots, x_k)$ $v = (x_{k+1}, x_{k+2}, \ldots, x_{2k}) \in [-1, 1]^k$和 $w = (x_{2k+1}, x_{2k+2}, \ldots, x_d) \in [-1, 1]^{d-2k}$。为方便起见，我们称这类函数相对于两组变量是部分可交换的。这类函数在实际中十分常见。例如，如果函数 $f$ 取决于空间中两个观测点之间的距离，那么作为这两个点的坐标（带因子或不带因子）的多元函数，$f$ 是部分可交换的。再如，如果图像的标签在左右翻转下保持不变，那么它就是部分可交换的。同样，三维结构[21] 的旋转不变性和人脸 [14] 的视角等价性，也可以用部分可交换特征来描述。实际上，若 $x_1, x_2, \ldots, x_m$ 是像素的常规向量化形式，记作 $u = (x_1, x_2, \ldots, x_k)$、$v = (x_m, x_{m-1}, \ldots, x_{m-k})$，其中，当 $m$ 为偶数时，$k = m/2$；当 $m$ 为奇数时，$k = (m - 1)/2$。那么，当 $m$ 为偶数时，图像的标签可表示为 $f(u, v)$；当 $m$ 为奇数时，可表示为 $f(u, v, x_{k+1})$。无论哪种情况，该函数都相对于 $u$和 $v$部分可交换，即 $f(u, v) = f(v, u)$，或 $f(u, v, x_{k+1}) = f(v, u, x_{k+1})$。

一种特殊情形是，$f$ 也满足

$$f(u, v, w) = f\left(\frac{u+v}{2}, \frac{u+v}{2}, w\right). \tag{8}$$

后一种情形的限制更多，也较少有趣，我们将其称为平凡情形。平凡情形的一个例子是，函数 $f$ 取决于空间中 $u$ 和 $v$ 的中点，而不是各个点的实际位置。

我们首先证明以下结论：

**定理 2.** 假设训练一个多层神经网络，以预测定义在 $\mathbb{R}^d$ 中某个区域上的未知有界利普希茨函数 $f(x)$，且该区域包含原点作为其内点。假设目标函数相对于变量的某两个子集具有部分可交换性。令 $f(x) = \psi \circ g(xW)$ 为对 $f$ 的神经网络预测值，其中 $W$ 是一个 $d \times m$，是第一个隐藏层的权重矩阵，激活函数为 $g$，且不含偏置。若 $f$ 保持了 $f$ 的非平凡部分可交换性，且自身非平凡，则 $\psi \circ g(\cdot)$ 是偶函数；当 $g$ 是偶函数时，即可实现这一点。

证明。令 $\alpha_1, \alpha_2, \ldots, \alpha_k, \beta_1, \beta_2, \ldots, \beta_k, \gamma_1, \gamma_2, \ldots, \gamma_{d-2k}$ 为 $W$的行向量。记 $u = (x_1, x_2, \ldots, x_k)$，$v = (x_{k+1}, x_{k+2}, \ldots, x_{2k})$ 和 $w = (x_{2k+1}, x_{2k+2}, \ldots, x_d)$。我们有

$$\widehat{f}(u, v, w) = \psi \circ g \left( \sum_{i=1}^{k} \alpha_i x_i + \sum_{i=1}^{k} \beta_i x_{i+k} + \sum_{j=1}^{d-2k} \gamma_j x_{2k+j} \right).$$

特别地，由这些表达式可得

$$\widehat{f}(u, -u, 0) = \psi \circ g \left( \sum_{i=1}^{k} (\alpha_i - \beta_i) x_i \right),$$

$$\widehat{f}(-u, u, 0) = \psi \circ g \left( -\sum_{i=1}^{k} (\alpha_i - \beta_i) x_i \right).$$

根据非三角部分可交换性假设，我们有 $f(u, -u, 0) = f(-u, u, 0)$。因此，

$$\psi \circ g(\sum_{i=1}^{k} (\alpha_i - \beta_i) x_i) = \psi \circ g(-\sum_{i=1}^{k} (\alpha_i - \beta_i) x_i),$$

这意味着 $\psi \circ g(\cdot)$ 是 $\mathrm{R}^m$ 上的偶函数。

注 *2*。 由于多元函数 $\psi$ 依赖于后续层，通常非常复杂，因此，确保 $\psi \circ g(\cdot)$ 为偶函数的一种简单有效的方法，是在可交换变量所连接的第一层选用偶激活函数 $g$。 $\psi \circ g$ 的偶性也可以通过 $\psi$的偶性来实现，而后者可以通过在后续层使用偶激活函数来达成。对于卷积神经网络，要实现左右翻转不变性，可以在第一个全连接层使用偶激活函数。要实现局部可交换性，可以在中间层使用偶激活函数。

使用偶激活函数显然并不常见。事实上，几乎所有流行的激$1 + x^2$活函数都不是偶函数。因此，我们将使用上文构造的 Seagull 激活函数，即函数 $\log(1 + x^2)$。请注意，定理 2 并未建议所有层都使用偶激活函数。此外，我们的实验似乎也表明，大量使用偶激活函数并不能带来更多益处。我们建议在可交换变量首次被整合到一起的那一层使用它。

## 5 实证评估

本节介绍所提出的激活函数定制方法在合成数据和真实世界数据上的应用。我们首先考察目标函数具有较强可交换性的低维情形。

**不同目标函数和不同原始激活函数的实验** s

考虑函数 $y = f(u, v, w)$，它表示顶点为 $u = (x_1, x_2, x_3)$、$v = (x_4, x_5, x_6)$ 和 $w = (x_7, x_8, x_9)$ 的三角形的面积。显然，$f(x_1, x_2, ..., x_9)$ 满足 $f(u, v, w) = f(v, u, w)$。事实上，我们有闭式公式 $f(u, v, w) = \frac{1}{2}\sqrt{A^2 + B^2 + C^2}$，其中

$$A = (x_4 - x_1)(x_8 - x_2) - (x_7 - x_1)(x_5 - x_2),$$

$$B = (x_4 - x_1)(x_9 - x_3) - (x_7 - x_1)(x_6 - x_3),$$

$$C = (x_5 - x_2)(x_9 - x_3) - (x_8 - x_2)(x_6 - x_3).$$

简而言之，$f$ 是一个由9个变量构成的正四次多项式的平方根。为确保性能提升并非源于目标函数 $f$的特定公式，我们还考虑了以下四个目标函数：$\log(1 + f)$、$e^f/100$、$\sin(f)$和 $Q(f) = \sqrt{\frac{f^2+3}{f+1}}$。因此，这五个函数涵盖了不同的增长速率，从对数增长、指数增长到波动函数。

为了使用神经网络发现近似公式，我们从 $[-2, 2]^9$中随机采样了10,000个九维向量，分别表示三个点 $u$、$v$、$w$ 在$\mathrm{R}^3$ 中的（$x$、$y$、$z$）坐标，并使用上述公式生成标签。同样，我们独立生成了2000个向量及其对应标签，构成测试集。

对于函数 $f$，我们随后构建了若干全连接神经网络，并选出整体性能最佳的网络。选定的模型是一个具有四个隐藏层的全连接神经网络（即多层感知机）。输入层有9个节点，输出层有1个节点。每个隐藏层有100个节点。对于另外四个函数，我们重复了相同的流程，发现相同的神经网络架构接近最佳。因此，我们对全部五个函数都采用了相同的架构。基于选定的神经网络架构，我们为每个目标函数分别使用五种常见激活函数（ReLU、ELU、Sigmoid、Tanh、Softplus）构建了五个模型，并使用RMSProp优化器对全部25个模型训练500轮，批次大小设为100。针对每个模型函数调整学习率，初始值为（0。001，0。003，0。005），并每100轮减半。采用其中表现最佳的模型。

为了展示使用定制激活函数的有效性，我们针对选出的 25 个模型，均将第一个隐藏层中的激活函数替换为 Seagull 激活函数 $\log(1 + x^2)$，同时保持神经网络的其余部分和超参数不变。

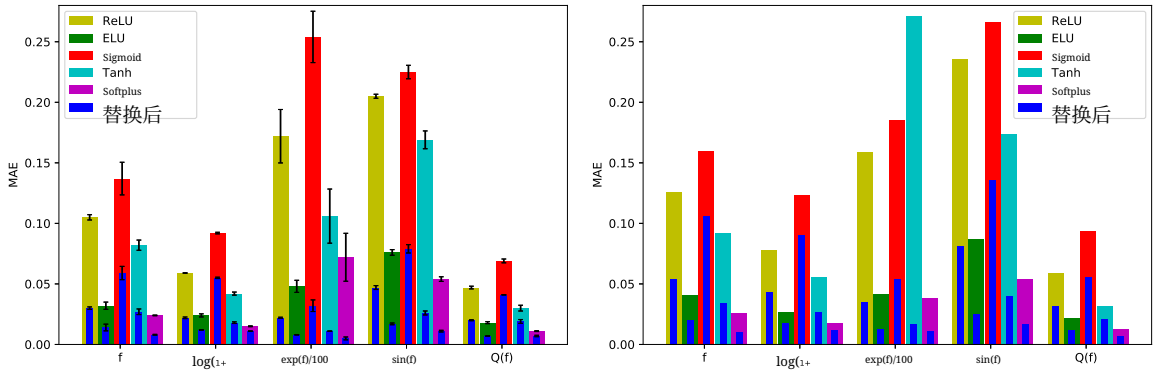


**图 1.** 在五个回归任务中使用 Seagull 激活函数的效果。较宽的柱：原始模型的平均绝对误差；较窄的蓝色柱：将第一个激活函数替换为 Seagull 后模型的平均绝对误差。左：每项实验均独立重复 5 次。右：添加了 5% 的噪声。

我们使用平均绝对误差（MAE）评估网络的性能，其计算方式如下：

$$MAE = \frac{1}{N}\sum_{j=1}^{N}|y_j - \hat{y}_j| \tag{9}$$

其中， $N$ 是测试样本的数量， $y_j$ 和$\hat{}y_j$ 分别是真实值和预测值。首先选择 MAE 而非 MSE，是因为 MAE 与 log$(1+x^2)$中新激活函数所包含的平方函数的相关性较低。

该实验使用不同数量的随机种子独立重复了 5 次。结果如图 1 左侧所示。不同目标函数的结果一致。

在现实世界应用中，数据总是包含噪声。为了测试我们的方法在含噪数据上的有效性，我们在训练标签中添加了均值为 0、标准差为 5% 的随机噪声。图 1 右侧的结果表明，偶激活函数具有鲁棒性且有效。

学习曲线也能直观展现这种改进。为此，我们考虑由 9 个变量 $3\times3$构成的行列式 $x_1$， $x_2$、……、 $x_9$。令 $f_1$ 为该行列式的余弦。函数 $f_1$是部分可交换的。为了测试替换为 Seagull 的有效性，我们生成了 50000 个训练向量，每个坐标均从方差为 1 的中心高斯分布中随机生成；并以相同方式独立生成了 2000 个测试向量。经过实验和参数调优，我们构建了一个全连接的 8 层神经网络，所有中间层均包含 100 个神经元，并交错设置了两个批归一化层和五个 Dropout 层，Dropout 率为 0.5，采用 Sigmoid 作为激活函数。我们使用 RMSprop 训练模型 500 个训练轮次，批次大小为 100，学习率为 0.003，每 100 个训练轮次减半，以最小化均方误差。然后，对每个模型，我们将第一层中的 Sigmoid 替换为 Seagull，其余超参数保持不变。从头开始重新训练这些模型。结果见图 2。

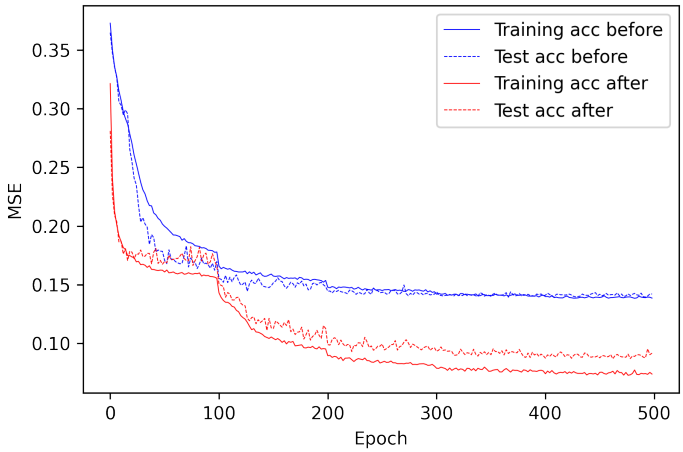


**图 2.** 学习曲线的比较。

**改变可交换程度的实验**

前两个实验中使用的函数 $f$ 具有很强的可交换性。事实上，当三组变量（$x_1, x_2, x_3$）、（$x_4, x_5, x_6$）和（$x_7, x_8, x_9$）进行置换时，该函数显然保持不变。不那么明显的是，当三组变量（$x_1, x_4, x_7$）、（$x_2, x_5, x_8$）和（$x_3, x_6, x_9$）进行置换时，该函数也保持不变。该函数 $f$ 在总共 12 种不同的置换下保持不变。

为了了解偶数 Seagull 对具有不同可交换程度的目标函数的影响，我们还考虑以下目标函数。为了尽量减少维数造成的伪影，我们还考虑一个 9 维函数。令 $\phi$ 为单位球面上由三个向量 $u=(x_1, x_2, x_3)$、$v=(x_4, x_5, x_6)$，

以及 $w = (x_7,\ x_8,\ x_9)$ 所围成的立体角。目标函数 $\phi$ 具有一个闭式公式

$$\phi(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9) = \phi(u, v, w) = \begin{cases} 2\tan^{-1} z & z \geq 0 \\ \pi + 2\tan^{-1} z & z < 0 \end{cases}$$

其中

$$z = \frac{|(u \times v) \cdot w|}{1 + u \cdot v + v \cdot w + w \cdot u}.$$

请注意，$\phi$ 与 $f$本质上不同，而且内函数 $z$ 甚至在曲面$u \cdot v + v \cdot w + w \cdot u = 1$上存在奇点。当三组变量（$x_1$，$x_2$，$x_3$）、（$x_4$，$x_5$，$x_6$）和（$x_7$，$x_8$，$x_9$）进行置换时，函数 $\phi$保持不变；但当三组变量（$x_1$，$x_4$，$x_7$）、（$x_2$，$x_5$，$x_8$）和（$x_3$，$x_6$，$x_9$）进行置换时，函数 $\phi$并不保持不变。共有 6 种不同的置换可使 $\phi$ 保持不变。

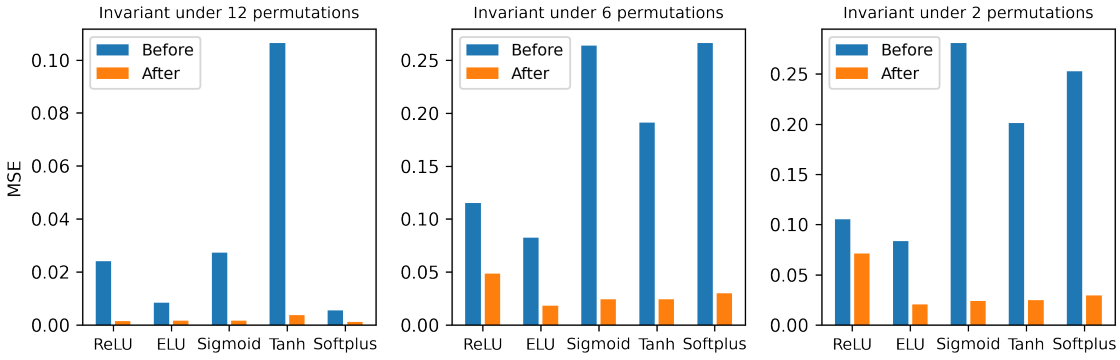


**图 3.** 在可交换程度不同的若干个 9 维目标函数上使用 Seagull 激活函数的效果。

此外，若我们定义 $\psi(x_1,\ x_2,\ \ldots,\ x_9) = \phi(x_1,\ x_2,\ \ldots,\ x_8,\ -x_9)$。则 $\psi$仅当（$x_1$，$x_2$，$x_3$）与（$x_4$，$x_5$，$x_6$）互换时才保持不变。因此，在三个函数 $f$、$\phi$ 和 $\psi$中，$\psi$ 的可交换性最弱（2 种置换），而$f$ 的可交换性最强（12 种置换）。

我们采用相同的方法，测试在第一层中用 Seagull 替换原始激活函数的效果。对于每个函数，我们构建了五个模型。各模型的层数和每层的神经元数量相同，但激活函数不同（ReLU、ELU、Sigmoid、Tanh、Softplus）。我们使用均方误差作为损失函数，并测试了三种不同的优化器（随机梯度下降、Adam 和 RMSprop），每种优化器都采用三种不同的学习率。图3中的结果表明，虽然改进幅度似乎与可交换程度呈正相关，但在低维情况下，即使可交换性较弱，改进也可能十分显著。

如果不存在可交换性，预计用 Seagull 替换不会带来任何好处，甚至可能使结果变差。为验证这一说法，我们使用了与图2相关的模型和函数，但将余弦函数替换为正弦函数。新函数不再是部分可交换的，而替换也没有带来任何明显改进。事实上，对于某些激活函数，结果甚至略有变差，部分原因是替换后没有调整超参数。

### 在不同层上应用 Seagull 的实验

Seagull 函数应用于哪一层很重要。最有效的做法是将其放在可交换变量首次连接的那一层。图3左图显示，效果会迅速减弱。如果将激活函数放在最后一层，效果可能会消失。在一种情况下，性能甚至变得更差。我们认为，部分原因在于参数是针对原始激活函数调优的，但替换后并未进行调整。

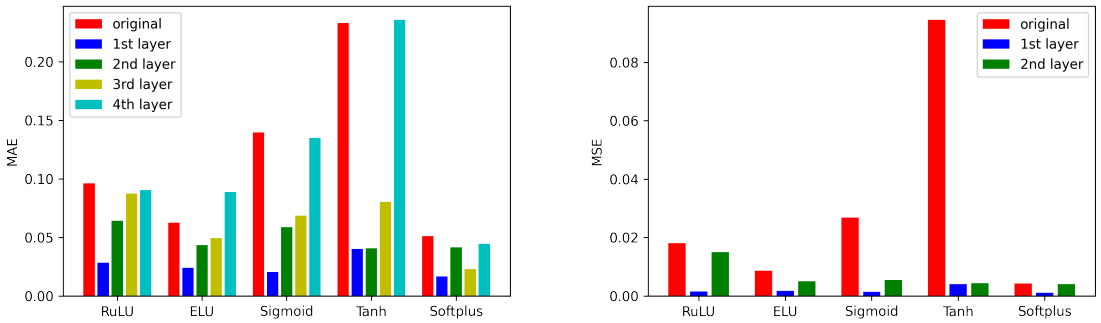


**图 4.** 在不同层使用 Seagull 激活函数的效果。左：平均绝对误差损失。右：均方误差损失。

这种改进并非损失函数平均绝对误差造成的假象。图3右图显示，将平均绝对误差损失替换为均方误差（MSE）后，改进同样显著。

尽管经过更长时间的训练并对学习率进行更精细的调整，经过更周密设计的神经网络可能会产生误差更小的预测值，但所有示例都清楚地表明，在第一层使用 Seagull 激活函数的效果显著。

### 维数实验

随着数据维数增加，部分可交换性的效果会减弱。事实上，在高维情况下，变量有许多组合，而满足部分可交换性条件的组合只占极小一部分。因此，预期随着维数增加，使用 Seagull 激活函数带来的改进会减小。此前实验的维数相对较低（9 维）。为考察维数对结果的影响，我们针对 16、25、36 和 64 维目标函数进行了实验。随着维数增加，改进幅度迅速减小。例如，对于 25 维函数，我们对一个 5 维单纯形的体积进行了实验，该单纯形由原点和五个 5 维点生成，这些点的坐标服从均值为 0、方差为 1 的正态分布。该函数在 10 种不同置换下保持不变。我们采用了与前述示例相同的实验策略，但平均改进幅度降至 12.7%。当维数增加到 64 时，在替换激活函数后，如果不调整参数，我们观察不到任何改进。

为了探究在替换后调整参数是否还能进一步改善超高维部分可交换目标函数的表现，我们在 CIFAR 10 上进行了实验。CIFAR 10 是一个广受欢迎的数据集，包含 60000 $32 \times 32$ 10 个类别的彩色图像，每个类别有 6000 张图像。关于 CIFAR-10 数据集的更多详情，请参阅 [13]。）

数据的维数为 3072。由于具有左右翻转不变性，因此存在部分可交换性。也可以认为，通过打乱 RGB 通道，分类很可能保持不变。尽管在如此高维的数据中，部分可交换性十分有限，但我们仍能看到使用 Seagull 激活函数的效果。

测试了以下深度卷积神经网络：Regnet [16], Resnet [7], VGG [20], 和 DPN [2]。这些网络的具体配置各不相同。具体而言，我们采用了计算量为 2 亿次浮点运算的 Regnet（Regnet200MF）、50 层 Resnet（ResNet-50）、包含 13 个卷积层和 3 个全连接层的 VGG 网络（VGG-16），以及 DPN-26。

将偶激活函数引入深度卷积神经网络有多种方法。具体来说，我们将 Regnet200MF、ResNet-50 和 VGG-16 中每个模块输出处的 ReLU 替换掉。以 ResNet-50 为例，其中每个模块由若干加权层和一个跳跃连接组成。我们将最后一个直接影响模块输出的 ReLU 替换为 Seagull 激活函数 $\log(1+x^2)$，其余激活函数保持不变。

训练设置方面，使用 50,000 张图像作为训练样本，10,000 张图像用于测试。训练轮次总数限制为 350。我们选择了动量为 0.9 的随机梯度下降优化器，并将学习

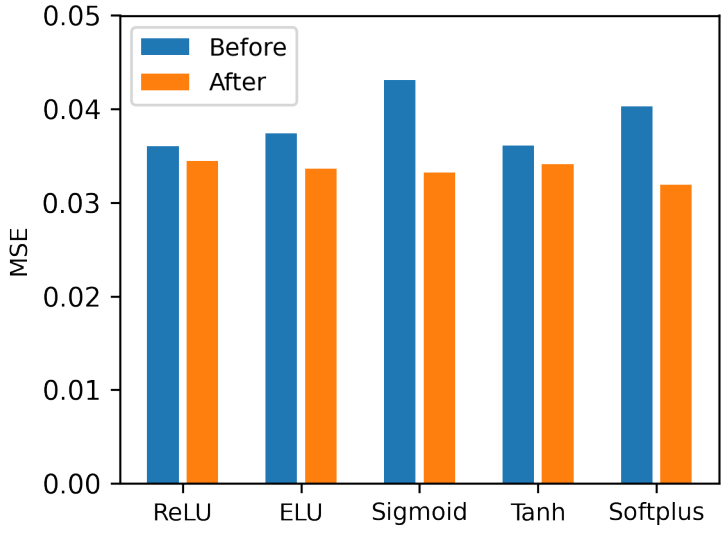


**图 5。** 在 25 维目标函数上使用 Seagull 激活函数的效果。

率设置为每 $[0, 150)$ 轮 0.1，设置为每 $[150, 250)$ 轮 0.01，设置为每 $[250, 350)$ 轮 0.001。所有训练均从头开始。考虑到深度卷积神经网络的随机初始化，每个训练过程都执行了 5 次，并在图 6 中汇总了测试数据上的平均准确率和标准差。

图6 中的结果揭示了一些有趣的信息。尽管存在一定差异，Seagull 函数在三种深度卷积神经网络上的表现均较 ReLU 函数有显著提升。网络中的特征图保留了图像的部分可交换性。考虑到深度卷积神经网络的结构，卷积运算和激活函数在保持空间关系不变的同时，也保留了部分可交换性。根据定理 2，偶激活函数相比 ReLU 能够更好地捕捉这一特征。尽管有人可能会认为，通过数据增强和滤波器旋转可以实现旋转不变性和翻转不变性，[5]，但使用所提出的 Seagull 函数可以获得相同的特征，同时还能显著减少网络规模和训练时间。

对于 DPN-26，我们采用了不同的策略。首先，我们从头开始在原始 DPN-26 上训练 300 个训练轮次。然后，我们插入一个带有偶激活函数的全连接层，并使用相同的超参数训练神经网络 300 个训练轮次。结果显示出明显的改进：对于未使用偶激活函数的网络，平均准确率达到 95。10% $\pm 0.11\%$；而使用偶激活函数的网络达到了 95。35% $\pm 0.04\%$。95。35% 的准确率也超过了 DPN-92，后者包含更多参数。这一显著改进进一步证明了在部分可交换目标函数中使用偶激活函数的有效性。

## 6 结论

本文强调了研究神经网络中激活函数的重要性。我们从理论上证明并通过合成数据集和真实世界数据集上的实验验证：当目标函数具有部分可交换性时，在可交换变量首次相连的层中使用 Seagull 激活函数，可以提升神经网络的性能。通过一个特殊但常见的案例，这些结果展现了定制激活函数的巨大潜力。

致谢 我们谨此感谢美国国家科学基金会资助项目 OCA-1940270 和 2019609，以及美国国立卫生研究院资助项目 P20GM104420 提供的支持。

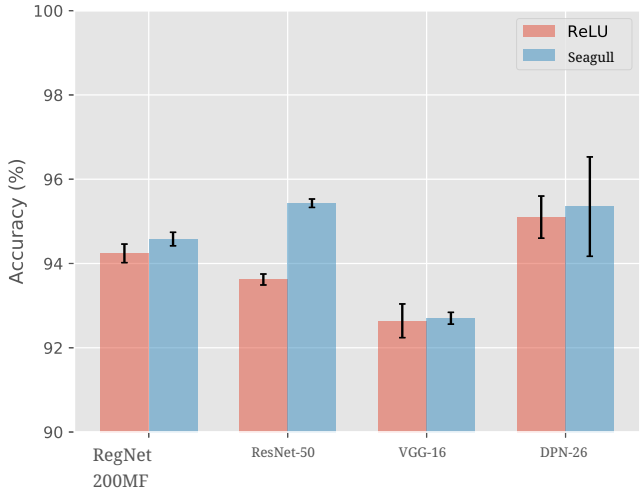


**图 6.** 不同神经网络结构下，Seagull 激活函数与 ReLU 在 CIFAR-10 上的表现。红色柱表示 ReLU 的平均准确率，蓝色柱表示偶激活函数在 CIFAR-10 数据集上的平均准确率。柱顶的黑线表示准确率的标准差。

## 参考文献

1. Carleo,G.,Cirac,I.,Cranmer,K.,Daudet,L.,Schuld,M.,Tishby,N.,Vogt-Maranto,L.,Zdeborov´a, L.：机器学习与物理科学。《现代物理评论》**91**(4), 045,002 (2019) 2. Chen, Y., Li, J., Xiao, H., Jin, X., Yan, S., Feng, J.：双路径网络。载于：《神经信息处理系统进展》，第 4467–4475 页（2017）3. Diaconis, P., Shahshahani, M.：关于线性组合的非线性函数。《SIAM科学与统计计算杂志》**5**(1), 175–191 (1984) 4. Gao, F.：Ae$^2$I：一种用于缺失值插补的双重自编码器。预印本（2021）5. Gao, H., Ji, S.：用于密集预测的高效且不变的卷积神经网络。载于：2017 IEEE国际数据挖掘会议（ICDM），第 871–876 页。IEEE（2017）6. Gybenko, G. 等：通过S形函数的叠加进行逼近。《控制、信号与系统数学》**2**(4), 303–314 (1989) 7. He, K., Zhang, X., Ren, S., Sun, J.：用于图像识别的深度残差学习。载于：IEEE计算机视觉与模式识别会议论文集，第 770–778 页（2016）8. Hornik, K.：多层前馈网络的逼近能力。《神经网络》**4**(2), 251–257 (1991) 9. Ismailov, V.E.：关于使用隐藏层中神经元数量有限的神经网络进行逼近。《数学分析与应用杂志》**417**(2), 963–969 (2014) 10. Iten, R., Metger, T., Wilming, H., Del Rio, L., Renner, R.：利用神经网络发现物理概念。《物理评论快报》**124**(1), 010,508 (2020) 11. Jagtap, A.D., Kawaguchi, K., Karniadakis, G.E.：用于深度神经网络和物理信息神经网络的局部自适应激活函数及斜率恢复项。arXiv 预印本，arXiv:1909.12228（2019）12. Kiselman, C.O.：光滑凸体的影子有多光滑？《伦敦数学学会杂志》**s2-33**(1), 101–109 (1986)。DOI 10.1112/jlms/s2-33.1.101。URL https://doi.org/10.1112/jlms/s2-33.1.10113. Krizhevsky, A., Hinton, G. 等：从微小图像中学习多层特征（2009）14. Leibo, J.Z., Liao, Q., Anselmi, F., Freiwald, W.A., Poggio, T.：视角容忍的人脸识别与赫布学习表明，神经调谐对头部朝向呈镜像对称。《当代生物学》**27**(1), 62–67 (2017) 15. Pinkus, A.：神经网络中 MLP 模型的逼近理论。《数值学报》**8**，143–195（1999）。DOI 10 0962492900002919 16 1017. /S . Radosavovic, I., Kosaraju, R.P., Girshick, R., He, K., Doll´ar, P.：设计网络设计空间。载于：IEEE/CVF计算机视觉与模式识别会议论文集，第 10,428–10,436 页（2020）17. Raissi, M., Karniadakis, G.E.：隐藏物理模型：利用机器学习求解非线性偏微分方程。《计算物理学杂志》**357**，125–141（2018）

18. Raissi, M., Perdikaris, P., Karniadakis, G.E.: 物理信息神经网络：用于求解涉及非线性偏微分方程的正问题和逆问题的深度学习框架。《计算物理学杂志》**378**, 686–707（2019年）19. de Silva, B.M., Higdon, D.M., Brunton, S.L., Kutz, J.N.: 从数据中发现物理规律：普适定律与偏差。arXiv 预印本 arXiv:1906.07906（2019年）20. Simonyan, K., Zisserman, A.: 用于大规模图像识别的超深卷积网络。arXiv 预印本 arXiv:1409.1556（2014年）21. Thomas, N., Smidt, T., Kearnes, S., Yang, L., Li, L., Kohlhoff, K., Riley, P.: 张量场网络：适用于三维点云、具有旋转和平移等变性的神经网络。arXiv 预印本 arXiv:1802.08219（2018年）22. Wang, R., Walters, R., Yu, R.: 将对称性融入深度动力学模型以提升泛化能力。arXiv 预印本 arXiv:2002.03061（2020年）23. Zhang, C., Bengio, S., Hardt, M., Recht, B., Vinyals, O.: 理解深度学习（仍然）需要重新思考泛化。《美国计算机学会通讯》**64**, 107 – 115（2021年）