arXiv:2305.05089v2 [cs.NE] 7 Jun 2023

可约双曲正切网络的函数等价与路径连通性

Matthew Farrugia-Roberts计算与信息
系统学院 墨尔本大学
matthew@far.in.net

摘要

理解人工神经网络的学习过程，需要厘清学习发生所在的参数空间结构。神经网络参数的函数等价类是实现相同输入输出函数的参数集合。对于许多架构，几乎所有参数都具有简单且有充分文献记载的函数等价类。然而，也存在极少数可约参数，其函数等价类更为丰富，这是由网络单元之间的冗余造成的。本文针对单隐藏层双曲正切架构，给出了单元冗余和可约函数等价类的算法刻画。我们证明，这类函数等价类是分段线性路径连通集；对于冗余单元占多数的参数，其集合的直径至多为 7 条线性线段。

1 引言

深度学习算法通过在高维参数空间中进行局部搜索，为人工神经网络架构构建参数。这一搜索由某种损失景观的地形引导。而这种地形又由神经网络参数与神经网络输入输出函数之间的关系所决定。因此，理解这些参数与函数之间的关系，是理解深度学习的关键。

众所周知，神经网络参数往往无法唯一确定一个输入输出函数。例如，在两个相邻的隐藏单元之间交换权重通常会保持函数等价（Hecht-Nielsen，1990）。对于许多架构而言，几乎所有参数都有一类简单的函数等价参数。对于具有各种非线性函数的多层前馈架构，这些类别已经得到了刻画（例如，Sussmann，1992；Albertini 等，1993；K˚urková 和 Kainen，1994；Phuong 和 Lampert，2020；Vla˘ci´c 和 Bölcskei，2021）。
然而，所有现有关于函数等价的研究都将某些参数的零测集排除在考察范围之外，而对于这些参数，函数等价类可能更加丰富。其中一类这样的参数被称为 可约参数。这些参数表现出某些结构冗余，使得同一个函数可以通过更少的隐藏单元来实现（Sussmann，1992；Vla˘ci´c 和 Bölcskei，2021），从而形成更加丰富的函数等价类。
尽管可约参数并不典型，但它们在深度学习中可能发挥重要作用。学习会对参数施加非随机的选择压力，而可约参数因其简约性而成为有吸引力的解（参见 Farrugia-Roberts，2023）。这些参数是信息奇异性的来源（参见Fukumizu，1996年），与深度学习统计理论相关（Watanabe，2009；Wei et al.，2022）。此外，函数等价类的结构会影响损失景观的拓扑，进而影响学习动力学。

预印本。评审中。

本文基于 Sussmann (1992) 关于双曲正切非线性函数网络可约性的奠基性研究，探讨了具有双曲正切非线性函数的单隐层网络的函数等价类。尽管这种架构与现代深度学习并无直接关联，但结构冗余对所有架构中的函数等价性具有尚未得到研究的影响。对这一简单情形进行全面研究，是朝这一研究方向迈出的第一步。为此，我们提出以下理论贡献。[1]

1. 在第4节中，我们给出一种形式化算法，可从任意函数等价类生成典范代表参数，并系统地消除所有结构冗余来源。这扩展了以往仅适用于不可约参数的算法。2. 在第5节中，我们反转这一典范化算法，将任意参数的函数等价类刻画为若干简单参数流形的并集。这一刻画扩展了关于不可约参数的著名结果。3. 我们证明，在可约情形下，函数等价类是一个由分段线性路径连通的集合——也就是说，任意两个函数等价的可约参数之间，都存在一条仅由等价参数构成的分段线性路径（定理6.1）。4. 我们证明，如果某个参数具有很高程度的可约性（特别是，如果用一半数量的可用隐藏单元就能实现同一函数），那么连接任意两个等价参数所需的线性线段数量至多为 7（定理6.3）。

在第7节中，我们将讨论这些结果对于理解参数空间结构的意义，并概述未来的研究方向，包括将结果推广到现代架构。

2 相关工作

Sussmann（1992）研究了单隐层双曲正切网络中的函数等价，并证明两个不可约参数函数等价，当且仅当它们之间可以通过交换和取反隐藏单元权重等简单操作相互转换。此后，这一结果被推广到具有更广泛非线性函数类别的架构（Albertini 等，1993；K˚urková 和 Kainen，1994）、具有多个隐藏层的架构（Fefferman 和 Markel，1993；Fefferman，1994），以及某些循环架构（Albertini 和 Sontag，1992、1993a、b、c）。近年来，研究者也在 ReLU 网络中发现了类似结果（Phuong 和 Lampert，2020；Bona-Pellissier 等，2021；Stock 和 Gribonval，2022），而 Vlaˇci´c 和 Bölcskei（2021、2022）则将 Sussmann 的结果推广到了非常广泛的一类架构和非线性函数。然而，所有这些结果都以排除某些参数零测度子集为代价，而这些子集具有更丰富的函数等价类。

另一类相关研究记录了参数空间的全局对称性——即对整个参数空间进行整体变换，同时保留所有已实现函数。这类对称性的研究由 Hecht-Nielsen (1990) 发起。 Chen 等人 (1993，另见 Chen 和 Hecht-Nielsen，1991) 证明，对于多层双曲正切网络，所有解析对称性均由单元交换和取反生成。Rüger and Ossen (1997) 将这一结果推广到了其他 S 形非线性函数。解析性条件排除了不连续对称性，例如选择性作用于可约参数、使其具有更丰富等价类的不连续对称性（Chen 等人，1993）。

Rüger and Ossen(1997) 提出了一种典范化算法。该算法对每个隐藏单元的权重取反，直至偏置为正，然后按照偏置权重的非降序对每个隐藏层中的单元进行排序。该算法恰好对上述单元交换和取反保持不变，但无法正确地将以更复杂方式不同的等价参数规范化。

据我们所知，只有一项研究直接探讨了可约参数的函数等价类。Fukumizu and Amari (2000) 和Fukumizu 等(2019) 总结了在保持神经网络函数不变的情况下添加单个隐藏单元的方法，而Sim.sek 等人(2021) 则将这项工作扩展到添加多个隐藏单元的情形。尽管这些方法是在不同的框架下推导出来的，但事实证明，通过此类单元添加所能到达的参数空间子集对应于函数等价类，与我们研究的等价类类似（尽管架构略有不同）。我们在备注2) 和3) 中的贡献，尤其是其中的相似之处，详见备注5.4 和5.5 以及备注6.2。

[1]贡献（1）、（2）和（3）也见于作者的学士论文（Farrugia-Roberts，2022，§5）。

3 预备知识

我们考虑一类全连接前馈神经网络架构，该架构具有一个输入单元、一个带偏置的输出单元，以及一个包含 $h \in \mathbb{N}$ 个带偏置隐藏单元的隐藏层，其非线性函数为双曲正切函数双曲正切函数$(z) = (e^z - e^{-z})/(e^z + e^{-z})$。此类架构具有参数空间 $\mathcal{W}_h = \mathbb{R}^{3h+1}$。我们的结果可直接推广到具有多维输入和输出的神经网络，详见附录A。

网络各单元的权重和偏置以如下格式编码在参数向量中：$(a_1, b_1, c_1, \ldots, a_h, b_h, c_h, d) \in \mathcal{W}_h$ 其中，对于每个隐藏单元 $i = 1, \ldots, h$，都有一个输出权重 $a_i \in \mathbb{R}$、一个输入权重 $b_i \in \mathbb{R}$ 和一个偏置 $c_i \in \mathbb{R}$，而 $d \in \mathbb{R}$ 是一个输出单元偏置。因此，每个参数 $w = (a_1, b_1, c_1, \ldots, a_h, b_h, c_h, d) \in \mathcal{W}_h$ 都对应一个数学函数f_w：$\mathbb{R} \to \mathbb{R}$，定义如下：

$$f_w(x) = d + \sum_{i=1}^{h} a_i \tanh(b_i x + c_i).$$

当且仅当两个参数 $w \in \mathcal{W}_h, w' \in \mathcal{W}_{h'}$ 作为 $\mathbb{R}$ 上的函数相等时（即 $\forall x \in \mathbb{R}, f_w(x) = f_{w'}(x)$），它们才是函数等价的。显然，函数等价是 $\mathcal{W}_h$ 上的一个等价关系。给定一个参数 $w \in \mathcal{W}_h$，w的函数等价类记为 $\mathrm{F}[w]$，它是 $\mathcal{W}_h$ 中所有与 w函数等价的参数构成的集合：

$$\mathfrak{F}[w] = \{\, w' \in \mathcal{W}_h \mid f_w = f_{w'} \,\}.$$

对于这类架构，几乎所有参数的函数等价类都是一个离散集合，可由简单的单元取反与交换变换 σ_i、$\tau_{i,\ j}$ 完全刻画：$\mathcal{W}_h \to \mathcal{W}_h$对于 i、$j = 1$、……、h，其中

$$\begin{aligned}
\sigma_i(a_1, b_1, c_1, \ldots, a_h, b_h, c_h, d) &= (a_1, b_1, c_1, \ldots, -a_i, -b_i, -c_i, \ldots, a_h, b_h, c_h, d) \\
\tau_{i,j}(a_1, b_1, c_1, \ldots, a_h, b_h, c_h, d) &= (a_1, b_1, c_1, \ldots, c_{i-1}, a_j, b_j, c_j, a_{i+1}, \\
&\qquad \ldots, c_{j-1}, a_i, b_i, c_i, a_{j+1}, \ldots, a_h, b_h, c_h, d).
\end{aligned}$$

更正式地说，对于所有所谓的不可约参数，这些变换生成完整的函数等价类（Sussmann，1992）。参数 $w = (a_1,\ b_1,\ c_1,\ \ldots\ldots,\ a_h,\ b_h,\ c_h,\ d) \in \mathcal{W}_h$ 是可约的，当且仅当其满足以下任一条件（否则，w 是不可约的）：

(i) $a_i = 0$对于某个 i，或 (ii) $b_i = 0$ 对于某个 i，或 (iii) $(b_i,\ c_i) = (b_j,\ c_j)$，对于某个 $i \neq j$，或 (iv) $(b_i,\ c_i) = (-b_j,\ -c_j)$，对于某个 $i \neq j$。

Sussmann（1992）还指出，在这一类架构中，可约性对应于非最小性：当且仅当某个参数 $w \in \mathcal{W}_h$ 可约，当且仅当 w 在函数上等价于某个$w' \in \mathcal{W}_{h'}$，且后者所需的隐藏单元更少 $h' < h$。我们将秩定义为 w的秩，记作$\mathrm{rank}(w)$，即实现 f_w所需的最少隐藏单元数：

$$\mathrm{rank}(w) = \min \{\, h' \in \mathbb{N} \mid \exists w' \in \mathcal{W}_{h'};\ f_w = f_{w'} \,\}.$$

最后，我们使用以下参数集的连通性概念。给定一个集合$W \subseteq \mathcal{W}_h$，定义在 W 中的分段线性路径为一个连续函数 ρ：$[0, 1] \to W$，由有限条线性线段构成。若两个参数 $w,\ w' \in \mathcal{W}_h$ 在中分段线性路径连通，记作 $w \leftrightsquigarrow w'$（其中 W 为隐含参数），则存在一条位于 W 中的分段线性路径，使得$\rho(0) = w$ 且 $\rho(1) = w'$。注意，$\leftrightsquigarrow$ 是 W上的一个等价关系。当且仅当 $\leftrightsquigarrow$ 是满的，即 W 中所有参数对都在 W中分段线性路径连通时，集合 $W \subseteq \mathcal{W}_h$ 本身才是分段线性路径连通。

分段线性路径的长度 是构成该路径的最大线性线段数。两个分段线性道路连通参数之间的距离 是连接它们的最短路径的长度。分段线性道路连通集合的直径 是集合中任意两个参数之间的最大距离。

4 参数规范化

一个参数典范化算法 将函数等价类中的每个参数映射到该类中的典范代表参数。因此，典范化算法可作为函数等价的计算测试。

先前的研究已经为某些不可约参数描述了典范化算法（Rüger andOssen，1997）；但是，当这些算法应用于函数等价的可约参数时，可能无法产生相同的输出。我们引入了一种能够正确规范化可约参数和不可约参数的典范化算法。该算法基于类似的取反和排序阶段，并结合了一个新颖的约简阶段。该阶段通过各种操作有效地移除或"归零"冗余单元，从而分离出一个函数等价但不可约的子参数。

算法 4.1（参数规范化）。给定参数空间 $\mathcal{W}_h$，执行以下步骤：

1: 过程 规范化($w = (a_1, b_1, c_1, \ldots, a_h, b_h, c_h, d) \in \mathcal{W}_h$) 2: ▷ **阶段 1：约简参数，将冗余隐藏单元归零** ◁**3:** $Z \leftarrow \{\}$ ▷ **跟踪"归零"的单元 4: 当以下四个条件中的任意一个成立时执行 5: 如果对于某个隐藏单元** $i \notin Z, a_i = 0$ **则 6:** $b_i, c_i \leftarrow 0$**7:** $Z \leftarrow Z \cup \{i\}$**8: 否则，如果对于某个隐藏单元** $i \notin Z, b_i = 0$ **则 9:** $d \leftarrow d + a_i$**双曲正切函数(c_i) 10:** $a_i, c_i \leftarrow 0$**11:** $Z \leftarrow Z \cup \{i\}$**12: 否则，如果对于某些隐藏单元** $i, j \notin Z, i \neq j$，$(b_i, c_i) = (b_j, c_j)$ **则 13:** $a_j \leftarrow a_j + a_i$**14:** $a_i, b_i, c_i \leftarrow 0$**15:** $Z \leftarrow Z \cup \{i\}$**16: 否则，如果对于某些隐藏单元** $i, j \notin Z, i \neq j$，$(b_i, c_i) = (-b_j, -c_j)$ **则** ▷ **—— (iv) 17:** $a_j \leftarrow a_j - a_i$**18:** $a_i, b_i, c_i \leftarrow 0$**19:** $Z \leftarrow Z \cup \{i\}$**20: 结束 if 21: 结束 while 22:** ▷ **阶段 2：对非零单元取反，使输入权重为正** ◁**23: 对每个隐藏单元** $i \notin Z$ **执行 24:** $a_i, b_i, c_i \leftarrow$ **符号(b_i) ⋅** (a_i, b_i, c_i) **25: 结束 for 26:** ▷ **阶段 3：按照输入权重和偏置对单元进行排序** ◁**27:** $\pi \leftarrow$ **一个排列，按降序对** $i = 1, \ldots, h$ **进行排序，依据为递减的** b_i**，若相等则依据递减的** c_i**打破平局 28:** $w \leftarrow (a_{\pi(1)}, b_{\pi(1)}, c_{\pi(1)}, \ldots, a_{\pi(h)}, b_{\pi(h)}, c_{\pi(h)}, d)$ **29:** ▷ **现在，** w **已被变换为典范等价参数** ◁**30: 返回** w**31: 结束过程**

下面的定理确立了算法4.1的正确性。

The理 4.2。设 w，$w' \in \mathcal{W}_h$。设 $v =$ 规范化(w) 和 $v' =$ 规范化(w')。则 n

(i) v 与 w函数等价；且 *(ii)* 若 w 与 w' 函数等价，则 $v = v'$。

证明。对于 (i)，注意到 f_w 在阶段 1 和阶段 2 的循环每次迭代中，以及阶段 3 的排列中均保持不变。对于 (ii)，注意到阶段 1 会将函数等价且不可约的子参数 $u \in \mathcal{W}_r$ 和 $u' \in \mathcal{W}_{r'}$ 从输入参数 w 和 w' 中分离出来（不包括置零的单元）。我们有 $f_u = f_w = f_{w'} = f_{u'}$，因此根据 Sussmann(1992）的结果，$r = r' =$ 秩(w)，且 u 和u' 通过单元取反和交换变换相关联。即使存在零单元，这一结论仍然成立。根据构造，阶段 2 和阶段 3 对且仅对这类变换保持不变。 □

5 完整函数等价类

算法4.1 对于给定函数等价类中的所有参数，都会生成一致的输出。它是以下刻画完整函数等价类的基础。

该刻画背后的思路是，枚举参数的单元在整个算法 4.1 中被约化、取反和排序的各种方式。每条这样的规范化轨迹 都对应一个简单的参数集合，该集合恰好沿着这条路径经过算法，如下所示。

定义 5.1 (规范化轨迹)。 设 $r, h \in \mathbb{N}, r \le h$。 r 阶的规范化轨迹 r 在h 个单元上 是一个元组 (σ, τ)，其中 $\sigma \in \{-1,+1\}^h$ 是一个符号向量（用于跟踪算法中的单元取反）； τ: $\{1,\dots,h\} \to \{0,1,\dots,h\}$ 是一个值域包含$\{1,\dots,r\}$ 的函数（用于跟踪算法中的单元约化和排列）。

定理 5.2。 设 $w \in \mathcal{W}_h$ 和 $v = (\alpha_1, \beta_1, \gamma_1, \dots, \alpha_h, \beta_h, \gamma_h, \delta) = \mathrm{C}_{\text{规范化}}(w)$。设$r = \text{秩}(w)$。那么，函数等价类$\mathfrak{F}[w] \subset \mathcal{W}_h$ 是若干子集的并集

$$\mathfrak{F}[w] = \bigcup_{(\sigma,\tau)\in\Gamma(h,r)} \left(X^{\delta}_{\tau^{-1}[0]} \cap \bigcap_{i=1}^{r} Y^{\alpha_i,\beta_i,\gamma_i}_{\sigma,\tau^{-1}[i]} \cap \bigcap_{i=r+1}^{h} Z_{\sigma,\tau^{-1}[i]} \right) \tag{1}$$

其中 $\Gamma(h, r)$ 表示在 h 单元上阶数为 r 的所有规范化轨迹的集合，并且

$$X^{\delta}_{I} = \left\{ (a_1,b_1,c_1,\dots,a_h,b_h,c_h,d) \in \mathcal{W}_h \;\middle|\; \begin{array}{c} \forall i \in I, b_i = 0 \text{ and} \\ d + \sum_{i\in I} a_i \tanh(c_i) = \delta \end{array} \right\};$$

$$Y^{\alpha,\beta,\gamma}_{\sigma,I} = \left\{ (a_1,b_1,c_1,\dots,a_h,b_h,c_h,d) \in \mathcal{W}_h \;\middle|\; \begin{array}{c} \forall i \in I, \sigma_i \cdot (b_i, c_i) = (\beta,\gamma) \\ \text{and } \sum_{i\in I} \sigma_i a_i = \alpha \end{array} \right\}; \text{ and}$$

$$Z_{\sigma,I} = \left\{ (a_1,b_1,c_1,\dots,a_h,b_h,c_h,d) \in \mathcal{W}_h \;\middle|\; \begin{array}{c} \forall i,j \in I, \sigma_i \cdot (b_i, c_i) = \sigma_j \cdot (b_j, c_j) \\ \text{and } \sum_{i\in I} \sigma_i a_i = 0 \end{array} \right\}.$$

证明。 假设 $w' = (a'_1,\ b'_1,\ c'_1,\ \dots\dots,\ a'_h,\ b'_h,\ c'_h,\ d) \in \mathcal{W}_h$ 属于（1）中的并集，因此也属于某个规范化轨迹 $(\sigma,\ \tau) \in \Gamma(h,\ r)$所对应的交集。那么 $f_{w'} = f_v = f_w$，如下所示：

$$\begin{aligned} f_{w'}(x) &= d' + \sum_{i\in\tau^{-1}[0]} a'_i \tanh(b'_i x + c'_i) + \sum_{j=1}^{r} \sum_{i\in\tau^{-1}[j]} a'_i \tanh(b'_i x + c'_i) + \sum_{j=r+1}^{h} \sum_{i\in\tau^{-1}[j]} a'_i \tanh(b'_i x + c'_i) \\ &= \delta + \sum_{j=1}^{r} \alpha_j \tanh(\beta_j x_i + \gamma_j) \text{ since } w' \in X^{\delta}_{\tau^{-1}[0]} \cap \bigcap_{j=1}^{r} Y^{\alpha_j,\beta_j,\gamma_j}_{\sigma,\tau^{-1}[j]} \cap \bigcap_{j=r+1}^{h} Z_{\sigma,\tau^{-1}[j]}. \end{aligned}$$

现在，假设 $w' \in \mathrm{F}[w]$。按照算法 4.1 对 w' 的执行过程，构造规范化轨迹 $(\sigma, \tau) \in \Gamma(h, r)$。令 $\sigma_i = -1$，其中 $\text{sign}(b'_i) = -1$，否则为 $+1$。按如下方式从恒等映射构造 τ。在每次阶段 1 迭代中，如果选择第二分支，则将 $\tau(i)$ 重新映射为 0。如果选择第三分支或第四分支，则对于 $k \in \tau^{-1}[i]$（包括其自身 i），将 $\tau(k)$ 重新映射为 j。最后，合并阶段 3 的排列 π: 对于 $k \notin \tau^{-1}[0]$，同时将 $\tau(k)$ 重新映射为 $\pi(\tau(k))$。

注意，根据定理 4.2，对 w' 进行规范化 $= v$。那么 $w' \in X^{\delta}_{\tau^{-1}[0]} \cap \bigcap_{\gamma_j}$，因为 $\tau^{-1}[0]$ 恰好包含并入 δ 的那些单元。此外，对于 $j = 1,\dots,r, w' \in Y^{\alpha_j,\beta_j,\sigma}$ $_{,\tau^{-1}[j]}$，因为 $\tau^{-1}[j]$ 恰好包含并入 v 的单元 j，且 σ 它们的相对符号（$\beta_j > 0$）。同样，对于$j \in r+1,\dots,h, w' \in Z_{\sigma,\tau^{-1}[j]}$（若 $\tau^{-1}[j]$ 为空则此条件自然成立）。 □

注 5.3。 如果 $w \in \mathcal{W}_h$ 是不可约的，那么 秩$(w) = h$。对于 $(\sigma, \tau) \in \Gamma(h, h)$， τ是一个排列（因为值域必须包含 $\{1,\dots,h\}$）。因此，迹的集合对应于由单元取反和对换生成的变换集合，如 Sussmann (1992所述。

注 5.4. 当 秩$(w) = h - 1$时，忽略符号向量和排列后，实质上有三种规范化轨迹，分别对应于为 $(h-1)$-单元网络添加一个额外单元的三种方式，详见 Fukumizu and Amari (2000) 和 Fukumizu 等 (2019)：引入一个新的常数单元或输出为零的单元，或者将一个现有单元拆分为两个。

注 5.5。 类似地，在 ‚Sim‚sek 等人 (2021，定义 3.2 和 3.3）中，$(r + j)$-元组与一个排列共同发挥作用，相当于 τ 在刻画扩张流形时的作用；这与函数等价类类似，但采取的是另一种视角，即向不可约参数添加单元。 ‚Sim‚sek 等人 (2021) 研究的是不存在单元取反对称性的情形，因此无需使用符号向量。

6 路径连通性

本节将证明，可约函数等价类是分段线性路径连通的（定理 6.1）；并且对于秩至多为可用隐藏单元数量一半的参数，其直径至多为 7 条线段（定理6.3）。

定理 6.1。 设 $w \in \mathcal{W}_h$。若 w 是可约的，则$\mathfrak{F}[w]$ 是分段线性道路连通的 d.

证明。 只需证明每个可约参数 $w \in \mathcal{W}_h$ 在 $\mathfrak{F}[w]$ 中都可通过分段线性路径连通到其典范代表 C典范化(w)。路径的构造通过追踪算法 4.1执行过程中参数的变化来进行。对于阶段 1 和阶段 2 中循环的每次迭代，以及阶段 3 中排列的每个对换，我们构造一条由多条线段组成的子路径。为描述这些子路径，我们将每条子路径起点处的参数记为 $w = (a_1, b_1, c_1, \ldots, a_h, b_h, c_h, d)$，注意，该参数会在算法执行过程中不断变化，但在所有这些中间点处都与原始参数 w 功能等价。

1. 在阶段 1 循环的每次迭代中，构造方式取决于所选分支，具体如下。图中展示了一些示例1。

   (i) 一条直接路径，将 b_i 和 c_i 插值至零。(ii) 一条由两段组成的路径，先将 a_i 插值至零，并将 d 插值至 $d + a_i$双曲正切函数(c_i)，再将 c_i 插值至零。(iii) 一条由两段组成的路径，先将 a_i 插值至零，并将 a_j 插值至 $a_j + a_i$，再将 b_i 和 c_i 插值至零。(iv) 一条由两段组成的路径，先将 a_i 插值至零，并将 a_j 插值至 $a_j - a_i$，再将 b_i 和 c_i 插值至零。

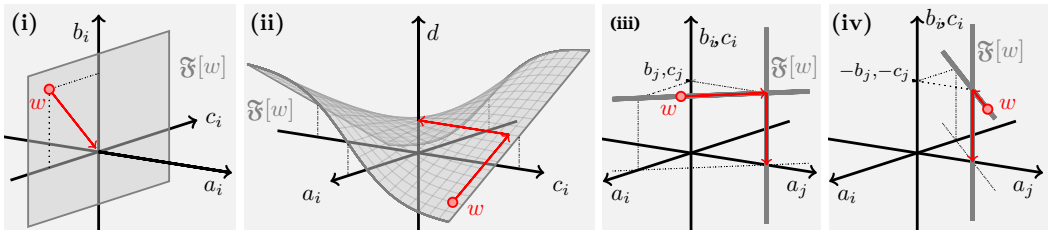


Figure 1：为阶段 1 的每个分支构造的路径示例。其他维度保持不变 d.

由于（原始的） w 是可约的，（当前的） w 必定至少经历过一次阶段 1 的迭代，并且至少有一个 空白单元 k 满足 a_k， b_k， $c_k = 0$。对于任意这样的参数w，F$[w]$ 中都存在一条由三段组成的路径，实现一次空白交换操作，将另一个单元 i 的权重转移到单元 k，并保持 a_i， b_i， $c_i = 0$不变：首先将 b_k 插值至 b_i ，并将 c_k 插值至c_i；然后将 a_k 插值至 a_i ，并将 a_i 插值至零；最后将 b_i 和 c_i 插值至零。同样，存在一条由三段组成的路径，可实现一次负空白交换操作，在将权重插值到空白单元的过程中对其取反。说明这些操作后，继续：

2. 对于阶段 2 循环中满足 sign$(b_i) = -1$ 的每次迭代，设 k 是一个空白单元，并构造一条由六条线段组成的路径。首先，将单元 i 空白交换到单元 k。然后，将单元 k 负空白交换到单元 i。净效果是对单元 i 取反。

3. 在阶段 3 中，为排列 π 分解为对换乘积时的每条线段构造一条路径。考虑对换 (i, j)。如果 i 或 j 是空白，则只需对它们执行空白交换操作。如果两者都不是空白，则令 k 为空白单元。使用三次空白交换操作构造一条由九条线段组成的路径，并使用 k 作为“临时存储”来实现该对换：首先，对换单元 i 和 k；然后，对换单元 i （现在为空白）和 j；最后，对换单元 j （现在为空白）和 k （其中包含 i 的原始权重）。

所得参数是典范代表，并且可以验证，每条子路径中的每条线段都按要求保留在F$[w]$ 中。 □

注 6.2. ,Sim,sek 等人 (2021，定理 B.4) 构造了类似的路径，以展示其扩张流形的连通性（参见注 5.5）。他们首先通过空单元交换操作连接约化形式参数，然后归纳证明每次单元添加都能保持连通性。

定理 6.3. 设 $w \in \mathcal{W}_h$。若 秩$(w) \le \frac{h}{2}$，则 $\mathfrak{F}[w]$ 的直径至多为 7。

证明。 设 $w \in \mathcal{W}_h$ 的 秩$(w) = r \le \frac{h}{2}$。设 $w' \in \mathfrak{F}[w]$。我们构造一条分段线性路径，从 w 到 w'，由 7 条线段组成。根据定理 6.1，通过典范代表参数 $v = \text{CANONICALISE}(w)$可得到一条路径。然而，这条路径过长。我们利用以下机会并行化线段并“抄近路”，将路径长度压缩至 7 。这些优化步骤如图2所示。

(a) 将算法 4.1 对 w 执行阶段 1 后得到的结果记为 u。将对 w'执行阶段 1 后得到的结果记为 u'。我们不再沿着从 u到 v、再返回 u'的单元取反与交换变换，而是直接将 u 变换为 u'，不一定经过 v。(b) 我们用两条线段连接 w 和 u，并行执行阶段 1 的所有迭代。第一条线段将空白单元的输出权重移至非空白单元和输出单元偏置。第二条线段将空白单元的输入权重和偏置插值至零。我们对连接 w 和 u'采用相同的优化。(c) 我们利用大多数空白单元作为“临时存储”，通过两次空白交换操作（6 条线段）连接 u 和 u'。首先，将 u 的非空白单元空白交换至 u'的空白单元中，得到一个与 u'没有共同非空白单元的参数 $\bar{u}'$。然后，我们对这些权重执行（取反）空白交换，将其移至u'中适当的非空白单元，从而实现关联 u、 $\bar{u}'$和 u'的单元取反与交换变换。

(d) (b) 和 (c) 中的操作开始和/或结束时，会在空白单元的输入权重和偏置之间进行插值，使其从零开始和/或归零，同时输出权重为零。我们将相邻的起始/结束线段合并在一起，而不一定经过零点。由此得到所需的七段路径，它依次经过参数 w， w^1， w^2，……， w^6， $w' \in W_h$。

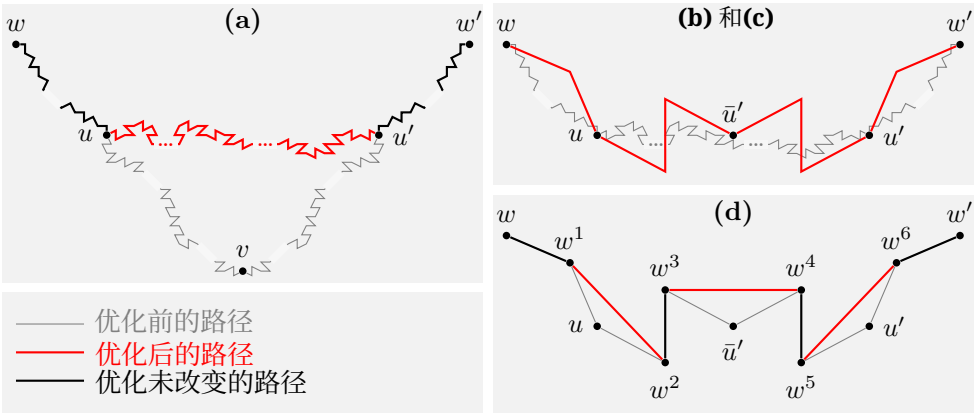


图2：四种路径优化的概念示意图，它们在高维参数空间中生成一条由等价参数构成的七段分段线性路径。**(a)** 在约简后的参数之间直接执行单元取反与交换变换，而不经过规范参数。**(b) & (c)** 将约简步骤并行化，并利用大多数空白单元使变换并行化。**(d)** 合并约简和空白交换操作的首尾线段。

为详细描述所构造的路径，我们为关键参数 w， w'， u， u'， w^1， w^2，……， $w^6 \in \mathcal{W}_h$的分量引入以下记号：

$$w = (a_1^w, b_1^w, c_1^w, \ldots, a_h^w, b_h^w, c_h^w, d^w) \qquad u = (a_1^u, b_1^u, c_1^u, \ldots, a_h^u, b_h^u, c_h^u, d^u)$$
$$w' = (a_1^{w'}, b_1^{w'}, c_1^{w'}, \ldots, a_h^{w'}, b_h^{w'}, c_h^{w'}, d^{w'}) \qquad u' = (a_1^{u'}, b_1^{u'}, c_1^{u'}, \ldots, a_h^{u'}, b_h^{u'}, c_h^{u'}, d^{u'})$$
$$w^k = (a_1^k, b_1^k, c_1^k, \ldots, a_h^k, b_h^k, c_h^k, d^k) \qquad (k = 1, \ldots, 6).$$

在 u 的 h 个单元中，恰好有 $h - r$ 个为空白——即 CANONICALISE(w) 中集合 Z 所包含的那些单元。将 r 个非空白单元的补集记为 $U = \{1, \ldots, h\} \setminus Z$。类似地，根据 u' 定义 Z' 和 U'。

明确了记号之后，我们现在可以详细描述关键点 $w^1, \ldots, w^6$，同时证明整条路径都包含在函数等价类 $\mathrm{F}[w]$ 中。

1. 第一条线段将每个输出权重从 a_i^w插值到 a_i^u，并将输出偏置从 d^w 插值到 d^u。也就是说，$w^1 = (a_1^u,\ b_1^w,\ c_1^w,\ \ldots\ldots,\ a_h^u,\ b_h^w,\ c_h^w,\ d^u)$。

要看出这条线段位于 $\mathrm{F}[w]$ 内，注意到两个参数之间的输入权重和偏置均未改变，因此，对于 $x \in \mathbb{R}$ 和 $t \in [0,1]$，$f_{tw^1+(1-t)w}(x) = tf_{w^1}(x) + (1-t)f_w(x)$。为证明 $f_w = f_{w^1}$，我们构造一个函数 $\tau\colon \{1,\ldots,h\} \to \{0,1,\ldots,h\}$，它根据 CANONICALISE($w$) 的阶段 1 每次迭代后所选分支的不同而定义：选择第二分支时，将 $\tau(i)$ 重新映射为 0；选择第三或第四分支时，对于 $k \in \tau^{-1}[i]$（包括其自身），将 $\tau(k)$ 重新映射为 j。此外，我们定义一个符号向量 $\sigma \in \{-1,+1\}^h$，其中若 $\mathrm{sign}(b_i^w) = -1$，则 $\sigma_i = -1$，否则为 $\sigma_i = +1$。于是：

$$\begin{aligned} f_w(x) &= d^w + \textstyle\sum_{j=0}^{k} \sum_{i \in \tau^{-1}[j]} a_i^w \tanh(b_i^w x + c_i^w) \\ &= d^w + \textstyle\sum_{i \in \tau^{-1}[0]} a_i^w \tanh(c_i^w) + \sum_{j=1}^{h} \left(\sum_{i \in \tau^{-1}[j]} \sigma_j \sigma_i a_i^w \right) \tanh(b_j^w x + c_j^w) \\ &= d^u + \textstyle\sum_{j=1}^{h} a_j^u \tanh(b_j^w x + c_j^w) = f_{w^1}(x). \end{aligned}$$

2. 第二条线段完成归约，并开始第一次空白交换操作，以将非零单元存储在 Z' 中。对于 $i \in U \cap U'$，选择互不相同的“存储”单元 $j \in Z \cap Z'$。根据假设，数量足够，因此 $r \le \frac{h}{2}$ $|U \cap U'| = |U| - |Z \cap U'| = \overline{r} - |Z \cap U'| \le (h - r)$ $- |Z \cap U'| = |Z'| - |Z \cap U'| = |Z' \cap Z|$。将单元 j 的输入权重从 b_j^w 插值到 b_i^w，并将其偏置从 c_j^w 插值到 c_i^w。与此同时，对于所有其他 $j \in Z$，将输入权重和偏置插值为零。由于 Z 的定义，对于 $j \in Z$、$a_j^1 = a_j^u = 0$，该线段处于 $\mathrm{F}[w]$ 内。

3. 第三条线段将 $U \cap U'$ 中各单元的输出权重移至步骤 (2) 中为 $Z \cap Z'$ 准备的各单元。对于 $i \in U \cap U'$，选择与步骤 (2) 相同的存储单元 j。将单元 j 的输出权重从 $a_j^u = 0$ 插值至 a_i^u，并将单元 i 的输出权重从 a_i^u 插值至零。根据步骤 (2)，由于 $b_i^2 = b_j^2$ 和 $c_i^2 = c_j^2$，这条线段位于 $\mathrm{F}[w]$ 内。

4. 第四条线段完成第一次空白交换操作，并开始第二次操作，以形成 u' 的单元。对于 $i \in U'$，将单元 i 的输入权重从 b_i^3 插值至 $b_i^{u'}$，并将其偏置从 c_i^3 插值至 $c_i^{u'}$。这条线段位于 $\mathrm{F}[w]$ 内，因为对于 $i \in U' \cap Z$，根据 的定义，$a_i^3 = a_i^u = 0$ Z $i \in U' \cap U$ $a_i^3 = 0$ 3 ；而对于，根据步骤 ( )。

5. 第五条线段将所选单元在 Z' 中的输出权重转移到步骤 (4) 中准备好的 U'中的单元。我们只需将每个单元 i的输出权重线性插值到 $a_i^{u'}$。

要看出这条线段位于 $\mathrm{F}[w]$ 内，只需注意 u 和 u' 可通过某些单元取反与交换变换相互关联。因此，它们的非零单元集合之间存在一种对应关系，使得对应单元的输入权重和偏置相同（或互为相反数）。由于步骤 (2)–(4)，w^4 中有 r 个“存储”单元，其权重与 u中的单元相同，而且这种对应关系也扩展到这些存储单元。由于存储单元与 U'不相交，这第五条线段的效果是：将 w^4 中每个存储单元 $j \in Z'$ 的输出权重从 a_i^u插值到零（其中 i 如步骤 (3) 所述），同时将其对应单元 $k \in U'$ 的输出权重从零插值到 $\pm a_i^u = a_k^{u'}$（符号取决于将 u 与 u'相关联的单元取反变换）。

6. 第六个线段完成第二次空白交换操作，并开始逆转约简。对于 $i \in Z'$，将单元 i 的输入权重从 b_i^5 插值到 $b_i^{w'}$，并将其偏置从 c_i^5 插值到 $c_i^{w'}$。该线段位于 $\mathrm{F}[w]$ 内，因为对于 $i \in Z'$，$a_i^5 = a_i^{u'} = 0$ 根据 Z' 的定义成立。

7. 第七条线段当然是从 w^6 插值到 w'。要证明这条线段位于 $\mathrm{F}[w]$ 内，注意根据步骤 (5) 和 (6)，$w^6 = (a_1^{u'},\ b_1^{w'},\ c_1^{w'},\ \ldots\ldots,\ a_h^{u'},\ b_h^{w'},\ c_h^{w'},\ d^{u'})$（注意 $d^u = d^{u'}$，因为输出单元的偏置在单元变换中保持不变）。因此，这种情况与步骤 (1) 相反，类似的证明同样适用。

□

7 讨论

本文研究了可约神经网络参数的函数等价类及其连通性。这些可约函数等价类是流形的复杂并集，呈现出以下丰富的定性结构。

- 存在一个由约化形式参数构成的中心离散数组，其中不可约子网络中分布着数量最多的空白单元。这些约化形式参数通过单元取反与交换变换相互关联，正如不可约参数一样。
- 与不可约情形不同，这些约化形式参数通过一个分段线性路径网络相互连接。具体而言，这些是（负向）空白交换操作；当存在多个空白单元时，则是同时进行的并行空单元交换操作。
- 各种流形从这一中心网络分支延伸出来，反向追溯各种约化操作（可选择并行执行）。与之对偶，这些流形描述了添加单元的方法（参见 Fukumizu and Amari，2000；Fukumizu 等，2019；Simsek 等，2021）。

定理 6.3 证明，当空白单元占多数时，该参数网络的直径将缩小为一个较小的常数个线性线段。当空白单元较少时，遍历中心网络有时需要执行更多空白交换操作。未来的研究可以探讨不同单元置换下最短路径长度与秩之间的权衡关系。

迈向现代架构。 我们研究了单隐层双曲正切网络，但由零单元、常数单元或成比例单元引起的结构冗余（可约性条件(i)–(iii)）是前馈网络组件的普遍特征。单元取反对称性是奇非线性函数的特征；其他非线性函数也会因其自身的仿射对称性而表现出类似的冗余。在更复杂的架构中，还会存在其他冗余来源，例如层之间的相互作用或专门的计算结构。

我们呼吁未来的研究去寻找、编目并深入调查这类冗余来源，而不是将其视为参数空间中测度为零子集的一部分并假定它们无关紧要。我们的结果为这一方向的未来研究提供了起点。Vlačić 和 Bölcskei(2021）对Sussmann (1992）结果的显著推广，将会是一个有益的补充。

函数等价与深度学习。 函数等价参数具有相同的损失。可约函数等价类中的连续方向和分段线性路径（定理 5.2、6.1和 6.3）因此意味着损失景观中的平坦方向和等损失路径。更广泛地说，低损失或零损失参数的集合是函数等价类的并集，其中可能（或者在具有足够过参数化的情况下必然）包括可约函数等价类。
理解可约函数等价类或许是理解这些主题的关键。特别值得关注的是与涉及单元剪枝的理论研究（Kuditipudi 等人，2019）以及置换对称性的研究（Brea 等人，2019）之间的联系。当然，损失相同并不意味着函数等价——事实上，Garipov 等人(2018) 观察到低损失路径中存在函数非等价现象。可约参数与这些主题的确切关联仍有待阐明。
如果损失景观是光滑的，那么上述讨论对于仅仅接近某个可约参数的不可约参数也近似成立。未来的研究应开发衡量与低秩参数邻近性的方法（参见 Farrugia-Roberts，2022、2023），并通过实证研究学习过程中遇到的参数中近似可约性的普遍程度。

8 结论

虽然可约参数仅占参数空间中测度为零的子集，但其函数等价类仍可能是理解参数空间结构的关键，进而有助于理解深度学习所处的损失景观。我们通过研究单隐层双曲正切网络，迈出了理解不可约参数之外函数等价问题的第一步。由于存在结构冗余，可约函数等价类比其不可约对应类丰富得多。通过考虑各种类型的结构冗余，我们刻画了可约函数等价类，并研究了它们的分段线性连通性。

致谢

贡献（1）、（2）和（3）也见于 MFR 的硕士论文（Farrugia-Roberts，2022，§5）。
MFR 在完成本研究期间获得了墨尔本工程学院基金会奖学金和长期未来基金的资助。我们感谢 Daniel Murfet 在研究期间及准备本文稿时提供的宝贵反馈。

参考文献

Francesca Albertini 和 Eduardo D. Sontag。对于神经网络，函数决定形式。技术报告 SYCON-92-03，罗格斯系统与控制中心，**1992**。扩展版见Albertini 和 Sontag (1993a)。引文见第2页。

Francesca Albertini 和 Eduardo D. Sontag。对于神经网络，函数决定形式。《神经网络》，6(7):975–990，**1993**a。通过Crossref访问。引文见第2 和第10页。

Francesca Albertini 和 Eduardo D. Sontag。离散时间神经网络的可辨识性。载于*1993*年欧洲控制会议论文集，第 2 卷，第 460–465 页。欧洲控制协会，**1993**b。通过 Francesca Albertini访问。引文见第 2页。

Francesca Albertini 和 Eduardo D. Sontag。循环网络权重的唯一性。载于 系统与网络：数学理论与应用：*1993*年国际 *MTNS* 研讨会论文集，第 II 卷，第 599–602 页。Akademie Verlag，**1993**c。通过 Francesca Albertini或通过Eduardo D. Sontag访问。另见扩展版，通过 Eduardo D. Sontag访问。引文见第 2页。

Francesca Albertini、Eduardo D. Sontag 和 Vincent Maillot。《神经网络权重的唯一性》。载于用于语音与视觉的人工神经网络，第 113–125 页。Chapman & Hall，伦敦，**1993**。研讨会论文集，研讨会于 1992 年在罗格斯大学举行。可通过EduardoD. Sontag获取。引用见第1 和2页。

Joachim Bona-Pellissier、François Bachoc 和 François Malgouyres。深层前馈 ReLU 神经网络的参数可辨识性。**2021**。预印本 arXiv:2112.12982 [math.ST]。引用见第 2页。

Johanni Brea、Berfin ̧Sim ̧sek、Bernd Illing 和 Wulfram Gerstner。深层网络中的权重空间对称性会产生置换鞍点，这些鞍点通过等损失谷连接，并贯穿整个损失景观。**2019**。预印本 arXiv:1907.02911 [cs.LG]。引用见第 9页。

An Mei Chen 和 Robert Hecht-Nielsen。论前馈神经网络权重空间的几何。载于第二届国际人工神经网络会议，第 1–4 页。IET，**1991**。访问途径：IEEE Xplore。引文见第2页。

An Mei Chen、Haw-minn Lu 和 Robert Hecht-Nielsen。论前馈神经网络误差曲面的几何。*Neural Computation*，5(6)：910–927，**1993**。访问途径：Crossref。引文见第2页。

Matthew Farrugia-Roberts。神经网络中的结构退化。硕士论文，计算与信息系统学院，墨尔本大学，**2022**。访问途径：MatthewFarrugia-Roberts。引文见第2、9和10页。

Matthew Farrugia-Roberts。确定与可压缩神经网络邻近性的计算复杂性。**2023**。预印本 arXiv:2306.02834 [cs.LG]。引文见第1 和9页。

Charles Fefferman。从输出重建神经网络。 *Revista MatemáticaIberoamericana*，10(3):507–555，**1994**。通过以下平台访问：Crossref。引用于第2页。

Charles Fefferman 和 Scott Markel。根据输出恢复前馈网络。载于 神经信息处理系统进展 *6*，第 335–342 页。Morgan Kaufmann，**1993**。通过以下方式访问：NeurIPS。引自第 2 页。

Kenji Fukumizu。多层感知机网络信息矩阵的正则性条件。《神经网络》，9(5):871–879，**1996年**。通过以下平台访问：Crossref。引用见第1 页和12页。

Kenji Fukumizu 和 Shun-ichi Amari。多层感知机分层结构中的局部极小值与平台。《神经网络》，13(3):317–327，**2000**。通过以下平台访问：Crossref。引用见第2、5和第9页。

Kenji Fukumizu、Shoichiro Yamaguchi、Yoh-ichi Mototake 和 Mirai Tanaka。通过将神经网络嵌入过参数化空间实现半平坦极小值与鞍点。在《神经信息处理系统进展》第*32*卷，第13868–13876页。Curran Associates，**2019**。通过以下途径访问：NeurIPS。引用于第2、5和第9页。

Timur Garipov、Pavel Izmailov、Dmitrii Podoprikhin、Dmitry P. Vetrov 和 Andrew G. Wilson。损失曲面、模式连通性与深度神经网络的快速集成。在神经信息处理系统进展第 *31* 卷，第 8789–8798 页。Curran Associates，**2018**。访问途径：NeurIPS。引用见第 9 页。

Egbert Harzheim。有序集合。Springer，**2005年**。通过 Crossref 获取。引用见第 12页。

Robert Hecht-Nielsen。论前馈网络权重空间的代数结构。载于高级神经计算机，第 129–135 页。北荷兰出版社，阿姆斯特丹，**1990**。访问途径：Crossref。引文见第1和2页。

V˘era K˚urková 和 Paul C. Kainen。功能等价的前馈神经网络。*NeuralComputation*，6(3):543–558，**1994**。通过以下途径访问：Crossref。引用见第 1 和第 2页。

Rohith Kuditipudi、Xiang Wang、Holden Lee、Yi Zhang、Zhiyuan Li、Wei Hu、Rong Ge 和 Sanjeev Arora。解释多层网络低成本解的景观连通性。载于 神经信息处理系统进展 *32*，第 14601–14610 页。Curran Associates，**2019**。通过以下途径访问：NeurIPS。引用见第 9页。

Mary Phuong 和 Christoph H. Lampert。ReLU 网络的函数等价与参数等价。载于 第八届国际学习表征会议。OpenReview，**2020**。通过以下途径访问：OpenReview。引用见第1 和第 2页。

Stefan M. Rüger 和 Arnfried Ossen。权重空间的度量结构。*NeuralProcessingLetters*，5(2):1–9，**1997**。通过以下途径访问：Crossref。引用见第 2 和第 4页。

Berfin ̧Sim¸sek、François Ged、Arthur Jacot、Francesco Spadaro、Clément Hongler、Wulfram Gerstner 和 Johanni Brea。过参数化神经网络中损失景观的几何：对称性与不变性。载于 第 *38* 届国际机器学习会议论文集，第 9722–9732 页。PMLR，**2021**。通过 PMLR获取。引用见第 2、5、6和 9页。

Pierre Stock 和 Rémi Gribonval。ReLU 网络的一种嵌入及其可识别性分析。《构造性逼近》，**2022**。通过 Crossref获取。引用见第 2页。

Héctor J. Sussmann。给定输入—输出映射的最小前馈网络权重的唯一性。《神经网络》，5(4):589–593，**1992**。通过Crossref获取。引用见第1、2、3、4、5、9、12、13、14和 15页。

VernerVla˘ci´c 和 Helmut Bölcskei。仿射对称性与神经网络的可识别性。《数学进展》，376:107485，**2021**。通过Crossref访问。引用见第1、2和第9页。

Verner Vla˘ci´c 和 Helmut Bölcskei。S 形非线性函数族的神经网络可识别性。《构造性逼近》，55(1):173–224，**2022**。可通过 Crossref访问。引用于第 2页。

Sumio Watanabe。《代数几何与统计学习理论》。剑桥大学出版社，**2009**。引用于第1页。

Susan Wei、Daniel Murfet、Mingming Gong、Hui Li、Jesse Gell-Redman 和 Thomas Quella。深度学习是奇异的，而这是件好事。《*IEEE* 神经网络与学习系统汇刊》，**2022**。通过 Crossref访问。即将在后续卷中刊载。引用见第1页。

推广到多维输入和输出

在本附录中，我们考虑的架构族比第 3 节所介绍的架构族略为一般。具体而言，我们考虑一族具有 $n \in \mathbb{N}^+$ 个输入单元、 $m \in \mathbb{N}^+$ 个带偏置的线性输出单元，以及单个隐藏层的全连接前馈神经网络架构；该隐藏层包含 $h \in \mathbb{N}$个带偏置的隐藏单元，并采用双曲正切非线性函数。经过本附录其余部分所述的细微修改，所有定义、算法、定理和证明都可直接从 $n = m = 1$推广到任意 n 和 m。

多维架构。 令 $n \in \mathbb{N}^+$、 $m \in \mathbb{N}^+$和 $h \in \mathbb{N}$。定义广义参数空间 $\mathcal{W}^n{}_{,h}{}^m = \mathbb{R}^{(n+m+1)h+m}$。网络各单元的权重和偏置按以下格式编码在参数向量中：$(a_1, b_1, c_1, \ldots, a_h, b_h, c_h, d) = w \in \mathcal{W}^n{}_{,h}{}^m$ 其中，对于每个隐藏单元 $i = 1, \ldots, h$，都有一个输出权重向量 $a_i \in \mathbb{R}^m$、一个输入权重向量$b_i \in \mathbb{R}^n$和一个偏置 $c_i \in \mathbb{R}$；而 $d \in \mathbb{R}^m$ 是一个输出单元偏置向量，其中每个输出单元对应一个偏置值。这次， w 表示一个多维数学函数 f_w：$\mathbb{R}^n \to \mathbb{R}^m$，其定义如下：

$$f_w(x) = d + \sum_{i=1}^{h} a_i \tanh(b_i \cdot x + c_i). \tag{2}$$

请注意，我们使用相同的元组记号和排列顺序$(a_1, b_1, c_1, \ldots, a_h, b_h, c_h, d)$ 但现在 a_i、 b_i和 d 都表示多分量向量。因此，在公式（2）中， b_i 和 x 现在使用内积（点积）相乘，而不是标量乘法，因为它们都是 $\mathbb{R}^n$中的向量。此外， $a_i \in \mathbb{R}^m$ 作为向量，需要与标量$\tanh($ $b_i \cdot x + c_i)$相乘。也就是说，对各个隐藏单元所产生的、作用于输出单元的贡献向量求和。要将主论文的结果推广到这一情形，首先需要把所有提及标量权重的地方替换为这些权重向量，并进行其他类似的更改，例如在适当情况下将字面上的零理解为向量零。

输入权重向量的符号与排序。 $\mathbb{R}^n$上的字典序，记作 $\preceq$，是一种关系，对于 u， $v \in \mathbb{R}^n$, $u \preceq v$，当且仅当 $u = v$，或者在 $i = 1$处，即u 与 v 不同的第一个索引， $u_i < v_i$。根据此定义，我们采用定义 $\prec$、 $\succ$和$\succeq$时的通常约定。最后，将 $v \in \mathbb{R}^n$的字典序符号记作 $\text{sign}_{\text{lex}}(v)$，定义如下：

$$\text{sign}_{\text{lex}}(v) = \begin{cases} +1 & (v \succ 0), \\ 0 & (v = 0), \\ -1 & (v \prec 0). \end{cases}$$

参数规范化算法以及其他一些定理和证明会反复用到输入权重向量的符号。在这些用法中，字典序符号满足标量符号函数所需的性质，因此，推广这些结果所需的第二项更改是将 $\text{sign}(\cdot)$ 的用法替换为 $\text{sign}_{\text{lex}}(\cdot)$。

当然，这种字典序关系也是全序（例如，参见 Harzheim，2005年，定理 4.1.11）。因此，可以据此对向量列表进行排序。按输入权重递减的顺序对单元进行排序，是算法4.1阶段 3 中的关键步骤，因此，第三项必要的改动是在该阶段采用递减字典序（$\succeq$）。

推广 Sussmann 等价定理。正文中的证明依赖于Sussmann (1992) 关于可约性与非最小性之间等价关系的结果，以及不可约的函数等价参数可通过单元取反与交换变换相互关联这一事实。Sussmann (1992) 研究的是包含多个输入单元但仅有一个输出单元的情形。引理A.1 和 A.2 将这些结果推广到了多输出情形。[2] 要将正文中的结果推广到此处，最后一项必要修改是将所有对 Sussmann 结果的引用替换为对引理A.1 或引理A.2的引用。

[2]证明将多输出情形化归为单输出情形，因此仍然依赖于 Sussmann (1992）的结果。 引理 A.1 的类似推广见 Fukumizu (1996年）。

单元取反与交换变换、可约性和非最小性的定义都可推广到任意 n 和 m，只需进行上述修改。为方便起见，这些定义在此重述。

一种单元取反变换是一个函数 σ_i：$\mathcal{W}^{n,\ h}{}^{m} \to \mathcal{W}_h^{n.m}$对于 $i = 1, \ldots, h$，其中

$$\sigma_i(a_1, b_1, c_1, \ldots, a_h, b_h, c_h, d) = (a_1, b_1, c_1, \ldots, -a_i, -b_i, -c_i, \ldots, a_h, b_h, c_h, d).$$

一种单元交换变换是一个函数$\tau_{i,\ j}$：$\mathcal{W}^{n,\ h}{}^{m} \to \mathcal{W}^{n}{}_{,\ h}{}^{m}$ 对于 $i, j = 1, \ldots, h$，其中

$$\tau_{i,j}(a_1, b_1, c_1, \ldots, a_h, b_h, c_h, d) = (a_1, b_1, c_1, \ldots, c_{i-1}, a_j, b_j, c_j, a_{i+1}, \ldots, c_{j-1}, a_i, b_i, c_i, a_{j+1}, \ldots, a_h, b_h, c_h, d).$$

参数 $w = (a_1,\ b_1,\ c_1,\ \ldots,\ a_h,\ b_h,\ c_h,\ d) \in \mathcal{W}^{n,\ h}{}^{m}$ 是可约的，当且仅当它满足以下任一条件（否则，w 是不可约的）：

(i) 对某个 i，$a_i = 0$；(ii) 对某个 i，$b_i = 0$；(iii) 对某个 $i \neq j$，$(b_i,\ c_i) = (b_j,\ c_j)$；或 (iv) 对某个 $i \neq j$，$(b_i,\ c_i) = (-b_j,\ -c_j)$。

参数 $w \in \mathcal{W}_h^{n\ m}$ $\mathcal{W}^n$ $h' < h$ w $w' \in$ ，是非最小的，当且仅当它在函数上等价于某个 ，$_{h'}{}^{m}$ 且其隐藏单元更少的参数。

引理 A.1. 对于 $w \in \mathcal{W}^{n,\ h}{}^{m}$，$w$ 可约，当且仅当 w 非最小。

证明。(⇒)：可按如下方式构造一个更小的函数等价参数。

(i) 如果对于某个 i，$a_i = 0$，那么隐藏单元 i 对该函数没有贡献。构造一个函数等价参数 $w' \in \mathcal{W}^{n,\ m}_{\ h-1}$，其中省略隐藏单元 i：

$$w' = (a_1, b_1, c_1, \ldots, a_{i-1}, b_{i-1}, c_{i-1}, a_{i+1}, b_{i+1}, c_{i+1}, \ldots, a_h, b_h, c_h, d).$$

(ii) 如果对于某个 i，$b_i = 0$，那么隐藏单元 i 对该函数的贡献仅为一个常数。构造一个函数等价参数 $w' \in \mathcal{W}^{n,\ m}_{\ h-1}$，其中省略隐藏单元 i，并更改输出单元偏置向量以作补偿：

$$w' = (a_1, b_1, c_1, \ldots, a_{i-1}, b_{i-1}, c_{i-1}, a_{i+1}, b_{i+1}, c_{i+1}, \ldots, a_h, b_h, c_h, d + a_i \tanh(c_i)).$$

(iii) 如果对于某个 $i \neq j$，$(b_i,\ c_i) = (b_j,\ c_j)$，那么隐藏单元 i 和 j 的贡献成比例。可将它们合并为一个单元（例如 j），使其具有相同的输入权重和偏置，并使用合并后的输出权重向量。相应地构造一个函数等价参数 $w' \in \mathcal{W}^{n,\ m}_{\ h-1}$：

$$w' = (a_1, b_1, c_1, \ldots, c_{i-1}, a_{i+1}, \ldots, c_{j-1}, a_j + a_i, b_j, c_j, a_{j+1}, \ldots, a_h, b_h, c_h, d).$$

(iv) 如果 $(b_i,\ c_i) = -(b_j,\ c_j)$ 对某个 $i \neq j$ 成立，那么隐藏单元 i 和 j 的贡献成负比例。由于 tanh 的奇函数性质，它们可以合并为一个单元（例如，j），其输入权重和偏置向量为 $(b_j,\ c_j)$，并具有合并后的输出权重向量。据此构造一个新的参数 $w' \in \mathcal{W}^{n,\ m}_{\ h-1}$：

$$w' = (a_1, b_1, c_1, \ldots, c_{i-1}, a_{i+1}, \ldots, c_{j-1}, a_j - a_i, b_j, c_j, a_{j+1}, \ldots, a_h, b_h, c_h, d).$$

在所有情况下，新的参数 $w' \in \mathcal{W}^{n,\ m}_{\ h-1}$ 都具有 $f_{w'} = f_w$，因此 w 不是最小的。

(⇐): 我们将问题归约到单输出情形，并应用Sussmann (1992)的结果来证明 w至少满足一个可约性条件。

为化归到单输出情形，我们引入一些记号。对于函数 f_w：$\mathbb{R}^n \to \mathbb{R}^m$，定义一系列分量函数 $f_{(1)w}$、$f_w^{(2)}$、……、$f_{(m)w}$：$\mathbb{R}^n \to \mathbb{R}$，使得对于 $x \in \mathbb{R}^n$，

$$f_w(x) = \left(f_w^{(1)}(x), f_w^{(2)}(x), \ldots, f_w^{(m)}(x)\right).$$

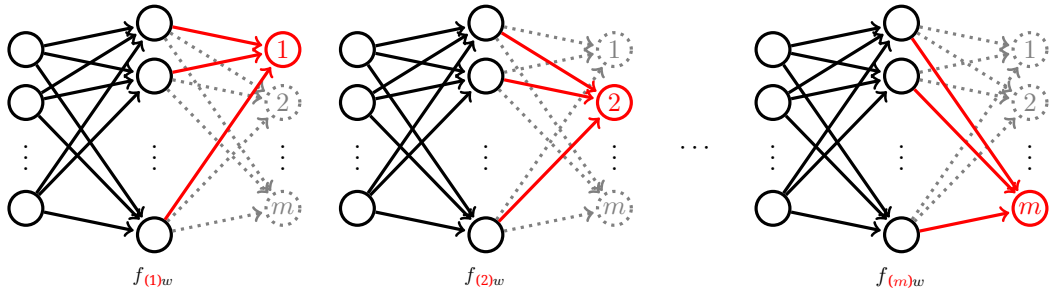


图 3: f_w的分量函数的连接图。包含的单元和权重以实线表示。每个网络的隐藏单元共享相同的输入权重（以及偏置，图中未显示）。

这些分量函数都是简单的神经网络函数，其架构具有 n 个输入单元和 1 个输出单元，并对应于原始神经网络连接图的一个子图，如图3所示。

将 $w \in \mathcal{W}^{n,\,m}_{\,h}$ 中对应的（重叠）子向量记为$w_{(1)},\ \ldots,\ w_{(m)} \in \mathcal{W}^{n,\,1}_{\,h}$。也就是说，对于 $\mu = 1,\ \ldots,\ m$,

$$w_{(\mu)} = (a_{1,\mu}, b_1, c_1, \ldots, a_{h,\mu}, b_h, c_h, d_\mu) \in \mathcal{W}^{n,1}_h.$$

现在，令 $w' = (a'_1,\ b'_1,\ c'_1,\ \ldots,\ a'_h,\ b'_h,\ c'_h,\ d') \in \mathcal{W}_{h'}$使得 $f_{w'} = f_w$其中 h'是实现 f_w 所需的最少隐藏单元数（$h' < h$ 依据非最小性假设）。对 $f_{w'}$应用相同的分解，以定义 $f^{(1)}_{w'},\ \ldots,\ f_{(m)w'}$，并定义 $w'_{(1)},\ \ldots,\ w'_{(m)} \in \mathcal{W}^{n,\,1}_{\,h'}$。
应用Sussmann (1992）的结果如下。由于 $f_w = f_{w'},\ \ f^{(\mu)}_w = f^{(\mu)}_{w'}$ 对于 $\mu = 1,\ \ldots\ldots,\ m$。因此，对于每个$w_{(\mu)},\ \ w'_{(\mu)}$ 都是使用更少单元的函数等价参数。因此，可约性条件（在 $m = 1$的特殊情况下）必须对每个$w_{(\mu)}$ （Sussmann，1992）成立。
由于条件（ii–iv）仅取决于输入权重和偏置，因此，如果这些条件中的任意一个对某个 $w_{(\mu)}$ 成立，那么对于其自身的 w 也必然成立（因为二者具有相同的输入权重和偏置），从而证明完成。现在只需考虑条件（ii–iv）对任何 $w_{(\mu)}$ 均不成立的情况，并证明在这种情况下，条件（i）对其自身的 w 成立。

我们还需要引入更多记号。对于 $i = 1, \ldots, h$，用 φ_i: $\mathbb{R}^n \to \mathbb{R}$ 表示函数 $\varphi_i(x)$ = 双曲正切函数$(b_i x + c_i)$。类似地，对于 $j = 1, \ldots, h'$，用 ψ_j: $\mathbb{R}^n \to \mathbb{R}$ 表示函数 $\psi_j(x)$ = 双曲正切函数 $(b'_j x + c'_j)$。然后，由于我们已经排除了 w 的可约性条件 (ii–iv)，因此没有任何 φ_i 是常数 (ii)，且任意两个都不成比例 (iii, iv)。对于 ψ_j 也是如此——条件 (i–iv) 不适用于 $w'_{(\mu)}$，因为我们假设 h' 是极小的。然而，对于 $\mu = 1, \ldots, m$，函数的线性组合

$$d_\mu + \sum_{i=1}^{h} a_{i,\mu}\varphi_i - d'_\mu - \sum_{j=1}^{h'} a'_{j,\mu}\psi_j = f^{(\mu)}_w - f^{(\mu)}_{w'} = 0$$

产生零函数。排除包含 $a_{i,\,\mu} = 0$ 或 $a'_{j,\,\mu} = 0$的项后，这个线性组合仍然成立。按照 Sussmann (1992) 中的相同推理，由于双曲正切函数的独立性（Sussmann，1992，引理 3.1），剩余项必须形成双射，使得

$$\varphi_i = \pm\psi_j \tag{3}$$

对某些 j，满足 $a'_{j,\mu} \neq 0$；对于每个 i，满足 $a_{i,\mu} \neq 0$。

为完成证明，请注意，w 与 w' 的单元之间的这些关系（3）独立于 μ。然而，这些关系具有“排他性”，因为不可能有两个 φ_i 与同一个 ψ_j成比例，否则它们彼此也会成比例（这在上文中已排除）。由于只有h' 个单元 $\psi_1, \ldots, \psi_{h'}$，因此必定存在一个隐藏单元 i（实际上至少存在 $h - h'$ 个单元），使得对于所有 $\mu = 1, \ldots, m$ 都有 $a_{i,\mu} = 0$（从而允许 φ_i避开任何此类关系）。也就是说，$a_i = (a_{i,1}, \ldots, a_{i,m}) = 0$，满足 w的条件 (i)，如所要求。 □

引理 A.2. 设 $w \in \mathcal{W}^n{}_{,h}{}^m$ 不可约，且设 $w' \in \mathcal{W}^n{}_{,h}{}^m$。若 w 和 w' 函数等价，则存在一个由单元取反与交换变换组成的复合链，整体构成变换 $T\colon \mathcal{W}^n{}_{,h}{}^m \to \mathcal{W}^n{}_{,h}{}^m$，使得 $w' = T(w)$。

证明。 再次，我们将问题化归为 $m = 1$的情形，并援引 Sussmann (1992)。

设 $w' \in \mathfrak{G}[w]$。引入两个神经网络的分解，与引理 A.1 的证明中相同，即由参数子向量 $w_{(1)}, \ldots, w_{(m)}, w'_{(1)}, \ldots, w'_{(m)} \in \mathcal{W}^n{}_{,h}{}^1$ 实现的分量函数 $f_{(1)w}, \ldots, f_{(m)w}, f^{(1)}_{w'}, \ldots, f_{(m)w'}$（参见图 3）。

对于 $\mu = 1, \ldots, m$，由于 $f_w = f_{w'}$，我们有 $f^{(\mu)}_w = f^{(\mu)}_{w'}$。现在，$w_{(\mu)}$ 和 $w'_{(\mu)}$ 不一定不可约；但若它们可约，则只能是因为条件 (i)，因为 $w_{(\mu)}$ 和 $w'_{(\mu)}$ 分别具有与 w 和 w' 相同的输入权重和偏置（根据假设，w 不可约；w' 不可约，因为它与 w 的单元数相同，故必然是最小的，而由引理 A.1 可知它不可约）。从 $w_{(\mu)}$ 和 $w'_{(\mu)}$ 中移除这类输出权重为零的单元，得到新的函数等价不可约参数 $u_{(\mu)}, u'_{(\mu)} \in \mathcal{W}^n$,1 rank($w_{(\mu)}$)。根据 Sussmann（1992，定理 2.1），存在一条单元取反与交换变换链 T_μ，使得 $u_{(\mu)} = T_\mu(u'_{(\mu)})$。

对于每个 μ, T_μ 都意味着 $w_{(\mu)}$ 和 $w'_{(\mu)}$ 中输出权重非零的单元之间存在一种关系，其中可能涉及这些单元的取反和排列。由于 w 和 w' 中的这些单元与 $w_{(\mu)}$和 $w'_{(\mu)}$共享输入权重和偏置，因此它们之间也必定存在相同的关系；并且（由于 w 是不可约的，条件 (ii–iv)），这些输入权重均非零，且同一参数下各单元的输入权重和偏置向量彼此绝不相同。此外，由于根据可约性条件 (i)，w 或 w' 中没有任何单元的输出权重向量可能为零，因此所有单元都参与了某种此类关系。

因此，可以根据这些隐含关系构造由单元取反和交换变换组成的复合变换，从而按要求关联 w 和 w'。 □