arXiv:2304.02902v1 [stat.ML] 6 Apr 2023

通过利用对称性实现贝叶斯神经网络中的高效MCMC采样

Jonas Gregor Wiese*[1]、Lisa Wimmer*[2, 3]、Theodore Papamarkou[4]、Bernd Bischl[2,3]、Stephan G¨unnemann[1,3], 和 David R¨ugamer[2,3]

[1] 慕尼黑工业大学[2] 慕尼黑大学统计系[3] 慕尼黑机器学习中心[4] 曼彻斯特大学数学系

摘要。 深度神经网络中的贝叶斯推断由于高维且强多模态的参数后验密度景观而极具挑战性。马尔可夫链蒙特卡洛方法渐近地恢复真实后验分布，但对于大型现代架构而言，其成本通常高得令人难以承受。作为一种广受欢迎的替代方案，局部方法聚焦于特定的参数区域，这些区域可以用具有可处理积分的函数来近似。尽管这类方法通常能取得令人满意的经验结果，但根据定义，它们无法考虑参数后验分布的多模态性。在本文中，我们认为，通过利用后验分布景观中的对称性，可以缓解精确但难以负担的方法与廉价但不精确的方法之间的两难困境。这类由神经元可互换性和某些激活函数诱导的对称性表现为：不同的参数值会产生相同的函数输出值。我们从理论上证明，贝叶斯神经网络中的后验预测密度可以限制在一个无对称性的参数参考集内。通过进一步推导出捕获函数多样性所需的蒙特卡罗链数量上界，我们提出了一种可行的贝叶斯推断直接方法。我们的实验表明，高效采样确实可行，为深度学习中准确的不确定性量化开辟了一条有前景的道路。

关键词： 不确定性量化 · 预测不确定性 · 贝叶斯推断 · 蒙特卡罗采样 · 后验分布对称性

1 引言

尽管大数据是深度学习的主导范式，但由于观测数量并非无限，如何量化不确定性（UQ）仍是该领域的一项重要问题。在贝叶斯学习中，不确定性量化的一个关键组成部分是参数后验密度，它为每个参数取值分配后验概率[5] [23]。在后验概率质量全部

* 贡献相同。[5] 我们假设似然由单个参数向量参数化。对于神经网络（神经网络），该参数包含所有权重和偏置。

集中于单一取值（表明对模型参数完全确定）与均匀分布于所有可能取值（体现出完全无知）这两种极端情况之间，参数后验密度的形状对于量化预测不确定性至关重要。然而，贝叶斯神经网络（BNN）的参数后验分布通常具有高度多模态，且很少能以封闭形式获得。经典的马尔可夫链蒙特卡洛（MCMC）方法渐近地恢复真实后验分布，但由于大量后验模态妨碍链达到合理的混合效果，该方法被认为对于贝叶斯神经网络而言代价过高 [25]。因此，拉普拉斯近似（LA； [9,29]）或深度集成（DE； [28]）等常用近似技术都聚焦于后验分布景观的局部区域。尽管这些方法比传统的马尔可夫链蒙特卡洛更快，且在许多应用中表现良好，但它们会系统性地忽略参数空间中可能对有效的不确定性量化至关重要的区域 [25]（第5.3节也展示了这一点）。

在这项工作中，我们质疑了“神经网络无法使用马尔可夫链蒙特卡洛”的普遍假设，并提出利用神经网络的不可辨识性这一在此背景下鲜少受到关注的特性，即存在两个或更多等价参数值，能够描述相同的输入输出映射。我们将这些等价值称为等输出参数状态。等输出参数状态源于某些激活函数 [6,27,37], 以及隐藏层中神经元参数可自由置换的性质 [20], ，并且可以相互转换。

这一现象所产生的功能冗余会随着网络的深度和宽度迅速增长（参见图1），并在后验密度中引入对称性。为了实现精确推断（误差仅来自蒙特卡罗误差），我们需要纳入所有会导致不同输入输出映射的非等输出参数状态。反过来说，只考虑这些功能各异的 映射，意味着有效参数空间仅占网络原始参数空间中相对较小的一部分。由于数量众多的等输出状态并不能为预测不确定性带来任何新信息，因此，在通过蒙特卡罗积分近似后验预测密度（PPD）时，我们需要的马尔可夫链蒙特卡罗样本要少得多。通过在事后显式移除样本中的对称性，我们甚至能够揭示后验中与功能相关的部分，并为在约化后的有效参数空间中进行解释和解析近似提供可能。

我们的贡献。我们分析后验空间冗余在量化 BNN 不确定性中的作用，并作出以下贡献：1）我们证明，可以通过一个规模小得多的参考集获得完整的后验预测分布（PPD）；该参考集包含在函数空间中被唯一标识的参数状态。2）我们提出一种估算所需蒙特卡罗链数量的方法，以发现功能多样模式，并为多层感知机（MLP）的参数空间采样提供实用指南。3）我们提供实验结果，证明与标准马尔可夫链蒙特卡洛方法和局部近似方法相比，我们的方法具有更优的预测性能。4）最后，我们展示了通过显式移除对称性可以实现后验分布的可解释性和解析近似，并为此提出了一种算法概念验证，

该方法可在事后获得。

2 相关工作

本节回顾多层感知机中对称性的存在、消除与利用方面的文献，并探讨此类对称性与统计推断之间的关系。

参数状态对称性的存在。 文献此前已讨论过网络参数状态的不唯一性。 [20] 是最早指出等输出状态会在多层感知机的参数空间中诱导对称性的研究者之一。他们在参数空间的一般线性群中，聚焦于保持输入输出映射不变的变换子群，推导出等输出参数状态的等价类，并证明对于每个多层感知机，都存在一个最小且完备的代表集，涵盖所有在函数上彼此不同的参数状态。 [42] 和 [27]沿着这一研究方向，考察了具有特定激活函数的单隐藏层多层感知机，并将研究从双曲正切推广到更一般的自仿射激活函数。 [6] 在双曲正切激活函数的背景下，将研究扩展至任意深度的多层感知机。近期， [37] 刻画了 ReLU 激活函数下的等输出参数状态，同样聚焦于单隐藏层的情况； [1] 则针对任意有限正交群 $\mathbb{G}$，对所有具有 ReLU 激活函数且满足 $\mathbb{G}$不变性的单隐藏层多层感知机进行了分类。最后， [45] 在一个框架中推广了上述大部分研究，以探讨任意架构中仿射对称性的可辨识性。

对称性消除。 贝叶斯模型参数后验密度中的对称性会产生不利影响，而统计学和机器学习的多个研究领域已经对此进行了处理。一个典型例子是有限混合模型中的标签切换，其中，将标签分配给混合成分的可置换性会诱导出与贝叶斯神经网络中类似的对称性。为了使混合模型具有可识别性， [3] 提出了一种带有在线重新标记的自适应 Metropolis 算法，通过在离散的置换矩阵集合上进行优化，有效地消除了置换对称性。然而，这类穷举搜索方法难以扩展到具有大量参数的现代神经网络，因为等输出状态的数量会随着参数数量呈指数增长。

在贝叶斯神经网络中，已知对称性会因浪费计算预算访问对称模式而减缓马尔可夫链蒙特卡洛向平稳参数后验密度的收敛 [31, 34]。 [25], 报告了大规模广泛实验的结果，确实发现马尔可夫链蒙特卡罗链在函数空间中的混合往往优于在参数空间中的混合。因此，通过施加参数约束并为部分潜变量定义锚定点来减弱对称性的影响，已被证明能够改善混合效果 [41]。例如， [39],提出了一种受约束采样方法，并将其应用于 ReLU激活的多层感知机。

利用对称性。 然而，参数后验密度中的对称性未必总是令人头疼的问题。恰恰相反，它们可以用于增强泛化能力，并使推断变得可行。越来越多的研究开始探索在模式连通性背景下消除对称性的做法；这种方法通过找出损失近乎恒定的连通区域，而非孤立的局部最优点，从而获得更稳健的神经网络解 [10, 17]。 [43] 、 [2], 等研究聚焦于隐藏层神经元的等输出置换，提出对齐多个网络的逐层嵌入，从而提升单个模型的性能。沿用类似思路， [38]对 ReLU 激活神经网络中的参数向量进行事后 标准化，所依据的是等输出等价类的概念。

在贝叶斯深度学习领域，利用由置换群表示的参数对称性（精确的或近似的）这一理念，推动了提升式马尔可夫链蒙特卡罗 [5,33]的发展。轨道马尔可夫链已被引入，以利用参数对称性来减少混合时间 [5,32]。提升式马尔可夫链蒙特卡罗主要是在概率图模型的背景下进行研究的。由于多层感知机可以表示为图模型，因此在多层感知机的背景下利用提升式马尔可夫链蒙特卡罗具有应用空间 [26,45]。

我们认为，尽管等输出对称性显然增加了参数后验密度的复杂性，但它们有可能促进MCMC推断。基于采样空间的很大一部分由某个最小搜索集的对称副本构成这一洞见，我们得出结论：并行运行多条较短的MCMC链，其中每条链都可以采样一个函数上不同的模式，相较于从单条链中收集大量样本，代表了对可用预算更高效的利用。在第4节中，我们为观察所有函数上多样的后验模态所需的MCMC链数量提出了一个上界，这是成功推断的关键标准。因此，参数后验对称性的视角为以往多链马尔可夫链蒙特卡罗工作提供了新的理论依据，这些工作主要受到利用并行计算资源的驱动 [30, 40]。我们在第5节中的实验表明，对于参数后验密度中具有许多对称性的贝叶斯神经网络而言，这种方法确实比单链马尔可夫链蒙特卡罗更有效。与 [25],我们的发现一致，我们主张在MCMC采样期间关注函数空间，而不是参数空间混合。

因此，我们认为等输出参数状态的存在是一种无害现象。尽管如此，消除对称性仍有其益处：将参数后验密度缩减至足以表示其全部函数多样性的最小参数集合后，我们便有机会获得更好的可解释性，甚至可能实现解析近似。我们在第5节的第5.3节中，通过一种定制算法（补充材料 B）展示了对称性消除的潜力。下一节将介绍相关数学背景，并引入等输出变换的刻画方式及形式化记号。

3 背景与符号说明

多层感知机架构。在本文中，我们考虑以下形式的神经网络。令$f\colon\ \mathcal{X} \to \mathcal{Y}$表示一个具有 K 层的多层感知机，其中第 $l \in \{1, \ldots, K\}$ 层由 M_l 个神经元组成，将特征向量 $\boldsymbol{x} = (x_1, \ldots, x_n)^\top \in \mathcal{X} \subseteq \mathbb{R}^n,\ n \in \mathbb{N}$，映射到结果向量

$$f(\boldsymbol{x}) =: \hat{\boldsymbol{y}} = (\hat{y}_1, \ldots, \hat{y}_m)^\top \in \mathcal{Y} \subseteq \mathbb{R}^m,\ \ m \in \mathbb{N},$$

用于估计 $\boldsymbol{y} = (y_1, \ldots, y_m)^\top \in \mathcal{Y}$。多层感知机第 i 层中的第 l 个神经元与权重 w_{lij}, $j = 1, \ldots, M_{l-1}$, 以及偏置 b_{li} 相关联。我们将所有多层感知机参数汇总到向量中

$$\boldsymbol{\theta} := (w_{211}, \ldots, w_{KM_KM_{K-1}}, b_{21}, \ldots, b_{KM_K})^\top \in \Theta \subseteq \mathbb{R}^d$$

并写作 $f_{\boldsymbol{\theta}}$ 以明确多层感知机由 $\boldsymbol{\theta}$ 参数化。对于每个隐藏层 $l \in \{2, \ldots, K-1\}$, 输入先经过线性变换，再由函数 a 激活。更具体地，我们将第 l 个隐藏层中第 i 个神经元的预激活定义为

$$o_{li} = \sum_{j=1}^{M_{l-1}} w_{lij} z_{(l-1)j} + b_{li}$$

其后激活值为来自前一层的 $z_{(l-1)i} = a(o_{(l-1)i})$。对于输入层，我们有 $z_{1i} = x_i$, $i = 1, \ldots, n$; 对于输出层，则有 $z_{Ki} = \hat{y}_i, i = 1, \ldots, M_K$。

预测不确定性。 在贝叶斯范式中，参数上会施加一个先验密度 $p(\boldsymbol{\theta})$，通常作为数据贝叶斯模型的一部分。利用贝叶斯法则，参数后验密度

$$p(\boldsymbol{\theta}|\mathcal{D}) = \frac{p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathcal{D})}$$

会根据数据 $\mathcal{D}$ 所提供并编码在似然 $p(\mathcal{D}|\boldsymbol{\theta})$ 中的信息更新这一先验信念。在监督学习中，数据通常由一组 N 特征向量 $\boldsymbol{x} \in \mathcal{X}$ 和结果向量 $\boldsymbol{y} \in \mathcal{Y}$ 给出，构成数据集 $\mathcal{D} = \{(\boldsymbol{x}^{(1)}, \boldsymbol{y}^{(1)}), \ldots, (\boldsymbol{x}^{(N)}, \boldsymbol{y}^{(N)})\}$。PPD $p(\boldsymbol{y}^*|\boldsymbol{x}^*, \mathcal{D})$ 量化了模型对新观测 $(\boldsymbol{x}^*, \boldsymbol{y}^*) \in \mathcal{X} \times \mathcal{Y}$ 的预测不确定性或函数不确定性。由于

$$p(\boldsymbol{y}^*|\boldsymbol{x}^*, \mathcal{D}) = \int_\Theta p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{D})\, \mathrm{d}\boldsymbol{\theta},$$

要推导这种不确定性，需要获取后验密度 $p(\boldsymbol{\theta}|\mathcal{D})$，该密度可通过 MCMC采样估计。

等输出变换。 现在，我们将更正式地刻画等输出参数状态的概念，以及在这些状态之间进行转换的变换。若两个参数状态 $\boldsymbol{\theta}, \boldsymbol{\theta}'$ 对所有可能来自 $\mathcal{X}$ 的输入都产生相同输出，则称它们等输出 。我们用 $\sim$ 表示这一等价关系（证明见补充材料 A.2），并记作：

$$\boldsymbol{\theta} \sim \boldsymbol{\theta}' \iff f_{\boldsymbol{\theta}}(\boldsymbol{x}) = f_{\boldsymbol{\theta}'}(\boldsymbol{x}) \, \forall \, \boldsymbol{x} \in \mathcal{X}, \quad \boldsymbol{\theta}, \boldsymbol{\theta}' \in \Theta.$$

等输出关系始终是针对特定的多层感知机 f 定义的；当上下文清楚时，我们会在记号中省略它。

至少有一个隐藏层包含多个神经元的所有多层感知机，都存在此类等输出参数状态，这些状态源自输入输出映射 $[20, 27]$ 的置换不变性。由于第 l 层第 i 个神经元的预激活运算彼此可交换[6], ，因此可以通过置换相应的参数，自由交换隐藏层中的 $M_l > 1$ 神经元 l。此外，使用具有内在对称性质的某些激活函数也会产生等输出变换。例如，对于双曲正切函数，可以翻转相应参数的符号，因为 $\tanh(x) = -\tanh(-x)$。对于 ReLU 激活，可以应用缩放变换，使网络的映射保持不变，即对于 $|c| > 0$，$\text{ReLU}(x) = c^{-1} \cdot \text{ReLU}(c \cdot x)$（另见补充材料 A.1）。

我们考虑在 $\boldsymbol{\theta}$ 中为线性的变换映射，这些映射会产生有限数量的等输出变换矩阵，例如双曲正切激活函数。ReLU 激活函数允许以无限多种方式重新缩放权重，因此不在我们的研究范围内。具体而言，令

$$\mathcal{F}_{\boldsymbol{T}} : \Theta \to \Theta, \boldsymbol{\theta} \mapsto \boldsymbol{T}\boldsymbol{\theta}, \quad \boldsymbol{T} \in \mathbb{R}^{d \times d},$$

是参数向量的一种与激活函数相关的变换，例如可以编码保持输出不变的符号翻转。若 $f_{\boldsymbol{\theta}}(\cdot) = f_{\mathcal{F}_{\boldsymbol{T}}(\boldsymbol{\theta})}(\cdot)$，则 $\mathcal{F}_{\boldsymbol{T}}$ 构成一种等输出变换。我们将所有保持输出不变的变换矩阵 $\boldsymbol{T}$ 收集到集合 $\mathcal{T}$ 中，即，

$$\mathcal{T} = \left\{ \boldsymbol{T} \in \mathbb{R}^{d \times d} \mid f_{\boldsymbol{\theta}}(\cdot) = f_{\mathcal{F}_{\boldsymbol{T}}(\boldsymbol{\theta})}(\cdot) \right\}.$$

类似地，令

$$\mathcal{F}_{\boldsymbol{P}} : \Theta \to \Theta, \boldsymbol{\theta} \mapsto \boldsymbol{P}\boldsymbol{\theta}, \quad \boldsymbol{P} \in \{0, 1\}^{d \times d},$$

是对参数向量中的元素进行置换的变换。我们将产生等输出参数状态的置换矩阵集合定义为

$$\mathcal{P} = \left\{ \boldsymbol{P} \in \mathbb{R}^{d \times d} \mid f_{\boldsymbol{\theta}}(\cdot) = f_{\mathcal{F}_{\boldsymbol{P}}(\boldsymbol{\theta})}(\cdot) \right\}.$$

[6] 回顾一下，神经元的预激活 i 在层 l 中为 $o_{li} = \sum_{j=1}^{M_{l-1}} w_{lij} z_{(l-1)j} + b_{li}$。根据求和的交换律，元素的任意排列 π： $J \to J$ 来自集合 $J = \{1, \ldots, M_{l-1}\}$ 都将得到相同的预激活：$o_{li} = \sum_{j \in J} w_{lij} z_{(l-1)j} + b_{li} = \sum_{j \in \pi(J)} w_{lij} z_{(l-1)j} + b_{li}$.

$\mathcal{P}$ 的基数至少为 $\prod_{l=2}^{K-1} M_l!$ [20] 以顺序方式从第一层遍历神经网络时，向每一层应用置换，以补偿其前一层中的置换。

由于激活函数逐个作用于神经元，与激活函数相关的变换和置换相关的等输出变换彼此不会相互影响（例如，我们可以先置换两个神经元对应的权重，之后再翻转其符号）。因此，我们可以将激活函数变换和置换变换任意组合，定义如下：

$$\mathcal{E} = \left\{ \boldsymbol{E} = \boldsymbol{T}\boldsymbol{P} \in \mathbb{R}^{d\times d}, \boldsymbol{T} \in \mathcal{T}, \boldsymbol{P} \in \mathcal{P} \mid f_{\boldsymbol{\theta}}(\cdot) = f_{\mathcal{F}_{\boldsymbol{E}}(\boldsymbol{\theta})}(\cdot) \right\}.$$

$\mathcal{E}$ 中的变换矩阵将呈现分块对角结构，各个块对应网络层。这是因为置换 $\boldsymbol{P}$ 会同时影响输入权重和输出权重，但仅表现为两组输入权重和两组输出权重交换位置，绝不会改变其所属层。与激活函数相关的符号翻转或缩放逐个作用于神经元，因此 $\boldsymbol{T}$ 是一个对角矩阵，不会改变 $\boldsymbol{P}$的分块对角结构。

对于等输出变换集合 $\mathcal{E}$ 的基数，我们可以根据 $\mathcal{P}$的最小基数推导出一个下界：

$$|\mathcal{E}| \geq \prod_{l=2}^{K-1} M_l! \cdot |\mathcal{T}_l|,$$

其中，$|\mathcal{T}_l|$ 表示适用于第 l层神经元的与激活函数相关的变换数量。由此可立即看出，功能冗余会随着网络规模迅速增加（另见图1）。由于存在等输出参数状态，对于常用先验分布（见第4节），多层感知机参数后验密度会呈现对称性形式的功能冗余：

$$p(\boldsymbol{\theta}|\mathcal{D}) = \frac{p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathcal{D})} \overset{10,13}{=} \frac{p(\mathcal{D}|\boldsymbol{E}\boldsymbol{\theta})p(\boldsymbol{E}\boldsymbol{\theta})}{p(\mathcal{D})} = p(\boldsymbol{E}\boldsymbol{\theta}|\mathcal{D}), \quad \boldsymbol{\theta} \in \Theta, \boldsymbol{E} \in \mathcal{E}. \quad (1)$$

4 高效采样

这些参数后验密度中的对称结构表明，采-样可以变得更加高效。在下一节中，我们将展示，理论上可以从一个由少量非等价参-数状态组成的参考集中获得后验预测分布，并提出一个足以对所有非对称后验模态进行采样的马尔可夫链数量上界。

4.1 后验分布参考集

如第3节所述，对于神经网络的每个参数状态 $\boldsymbol{\theta}$，存在与 $\boldsymbol{\theta}$ 通过等输出变换相关的功能冗余对应项 $\boldsymbol{\theta}'$，使得 $f_{\boldsymbol{\theta}}(\cdot) = f_{\boldsymbol{\theta}'}(\cdot)$。我们可以利用这一等价关系将参数空间 Θ 划分为互不相交的等价类。为此，令参考集

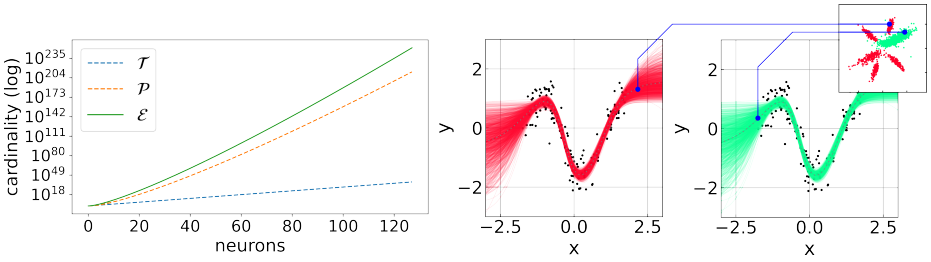

图 1.双曲正切激活多层感知机示例。 *Left*: 对于包含 1至 128个神经元的单隐藏层，等输出变换集合的基数下界；包含 128个神经元的网络的函数冗余因子为 $1.31 \cdot 10^{254}$。 *Right*: 一个十维多层感知机参数后验分布（右上角，以二元边际密度表示）呈现出对称性，因此所有红色样本簇都与绿色样本簇具有等输出关系。相关的函数空间完全相同，即许多后验模态是冗余的。

$\mathcal{S}_1$ 是每个等价类的最小代表集（参见 [6]中的开放的最小充分搜索集）。 $\mathcal{S}_1$ 中的所有参数状态在功能上都是多样的，即 $\boldsymbol{\theta}, \boldsymbol{\theta} \in \mathcal{S}_1 \Rightarrow \boldsymbol{\theta} \not\sim \boldsymbol{\theta}$，并且 Θ 中的每个元素恰好与 $\mathcal{S}_1$中的一个元素等价。对于有限数量的等输出变换，例如双曲正切激活多层感知机的情况（隐藏神经元符号翻转组合的有限种可能性），神经网络参数空间随后可以被剖分为 $|\mathcal{E}|$ 个不相交的代表集，这些代表集包含参考集元素的等输出变换，具体方式如下。

命题1（参数空间剖析）。 令 $\mathcal{S}_1$ 为唯一确定的网络参数状态的参考集。然后，对于有限数量的等输出变换，参数空间可以剖析为 $|\mathcal{E}|$ 个互不相交的非空代表集，除去集合 $\mathcal{S}^0 \subset \Theta$，即，

$$\Theta = \left(\dot{\bigcup}_{j=1}^{|\mathcal{E}|} \mathcal{S}_j \right) \dot{\cup} \mathcal{S}^0, \; where \; \mathcal{S}_j \cong \{ \boldsymbol{\theta} \mid \boldsymbol{\theta} = \boldsymbol{E}_j \boldsymbol{\theta}' \quad \forall \boldsymbol{\theta}' \in \mathcal{S}_1, \boldsymbol{E}_j \in \mathcal{E} \}, \tag{2}$$

其中， $\dot{\cup}$ 表示不相交集合上的并集。我们使用 $\mathcal{S}^0$ 作为残差量，以处理无法明确归入集合 $\mathcal{S}_j$中任何一个的情况，因为即使在使用非单位矩阵 $\boldsymbol{E}_j \in \mathcal{E}$ 进行变换时，它们仍保持不变。

构成 $\mathcal{S}^0$ 的边界情况确实存在，例如位于两个类别的边界 [6]，或零向量等退化情况 [42, 45]。有关所涉及集合的特征描述以及证明概要，请参阅补充材料 A.4。

如果先验具有变换不变性，那么等输出参数状态具有相同的后验概率 $p(\boldsymbol{\theta}|\mathcal{D}) = p(\boldsymbol{E}\boldsymbol{\theta}|\mathcal{D})$；见补充材料 A.3，公式 (10)-(13)。此外，根据定义，等输出参数状态对任意 $\boldsymbol{E} \in \mathcal{E}$ 都会产生相同的预测 $p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{\theta}) = p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{E}\boldsymbol{\theta})$。因此，以下推论成立。

推论1（重新表述的后验预测密度）。令 $\mathcal{E}$ 为有限集。如命题*1*所述，考虑互不相交的非空集合 $\mathcal{S}_j$、 $j \in \{1, \ldots, |\mathcal{E}|\}$以及剩余空间 $\mathcal{S}^0$。如果先验密度 $p(\boldsymbol{\theta})$ 具有变换不变性，则后验预测密度表示为

$$
\begin{aligned}
p(\boldsymbol{y}^*|\boldsymbol{x}^*, \mathcal{D}) &= \int_{\Theta} p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{D}) \, \mathrm{d}\boldsymbol{\theta} && (3) \\
&= |\mathcal{E}| \int_{\mathcal{S}_j} p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{D}) \, \mathrm{d}\boldsymbol{\theta} + \int_{\mathcal{S}^0} p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{D}) \, \mathrm{d}\boldsymbol{\theta} \\
&\approx |\mathcal{E}| \int_{\mathcal{S}_j} p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{D}) \, \mathrm{d}\boldsymbol{\theta}. && (4)
\end{aligned}
$$

推论1的证明见补充材料A.3。该结论由命题1以及变换不变先验密度的假设推出，而这一假设在实践中通常成立（例如，对于广泛应用的各向同性高斯先验）。我们还可以用（4）进一步近似（3），因为集合 $\mathcal{S}^0 \subset \mathbb{R}^d$ 的大小可忽略不计（取决于 Θ，甚至可能具有零勒贝格测度）。

根据推论1，只需对集合 $\mathcal{S}_j$ 中一个集合里唯一确定的参数状态进行积分，即可在残差集合范围内获得 PPD，并乘以因子 $|\mathcal{E}|$ ，以根据后验分布中的冗余程度校正概率值。换言之，只需对后验分布的 $1/|\mathcal{E}|$ 部分进行采样，就能推断出 NN 中一组唯一确定的参数状态，进而获得完整的 PPD。如图1所示，这会大幅缩小目标采样空间。例如，对于含有128个神经元的单层双曲正切激活网络，后验空间可有效缩减至原始大小的 10^{254} 分之一。

对于具有无限多个等输出变换的情况，例如 ReLU 激活的多层感知机（缩放因子 $|c| > 0$ 可以任意选取），我们可以采用类似的推理。由于两个代表集上的积分相同，只需观察后验密度的一个代表集，就能捕获网络的全部函数多样性。关于 ReLU 对称性的更深入讨论，请参阅例如 [4]。

如何获得代表集？ 实际上，在使用蒙特卡罗近似公式 (4) 时，没有必要真的将采样过程限制在特定的集合 $\mathcal{S}_j$中，这实际上可能并不简单[7]。由于任何等输出变换都是先验已知的$a\ priori$，我们只需要意识到这样一个事实：运行采样过程后，每个样本在理论上都可以映射到不同的代表集。因此，对于后验预测分布积分的计算，只要样本覆盖所有功能上具有多样性的参数状态，就可以让它们继续分散在各个代表集中。对于

[7] [12] 证明了，寻找在作用于输入空间的群下的不变表示是一个 NP难问题。虽然我们尚未发现参数空间存在此类结果，但 [12] 中关于输入置换的 NP难性表明，我们的情况可能也具有类似性质。

为了提高可解释性并对后验分布进行解析近似，明确消除对称性或许仍然值得。在第5节的第 3 小节中，我们展示了如何使用一种适用于双曲正切激活网络的定制算法来消除这类对称性（补充材料 B）。

4.2 马尔可夫链的一个上界

问题在于，仍需确定需要多少样本才能充分近似一组唯一确定的参数状态。即使在无对称性的情况下，BNN 后验分布也可能呈现多个功能多样模式，代表结构不同的假设，其具体情况取决于网络架构和底层数据生成过程。例如，在第5节.3中，我们讨论了一个参数化不足的网络案例，该网络因容量受限而保留了三个鲜明的模式。

下文中，我们假设存在 $\nu \in \mathbb{N}$ 个功能多样模式，目标是在运行马尔可夫链蒙特卡洛时至少访问每个模式或其局部邻近区域一次。由于在一条链中从一个模式切换到另一个模式的能力取决于多种因素，例如接受概率以及其他参数的当前状态，因此增加每条链的样本数并不一定与访问的模式数量相关。因此，我们建议关注独立链的数量，而非每条链的样本数，以有效控制访问的模式数量。

这进一步使我们能够推导出所需独立链的数量上界，从而确保每个模式至少被访问一次。随后，每条链的样本数量最终将决定近似质量。在计算后验预测分布时，我们同时对所有链的全部样本进行蒙特卡罗积分 [30]。在实践中，给定用户定义的最大资源数量 ρ （例如，CPU 核心数），以下命题给出了链的数量$\mathcal{G}$ 低于用户资源限制（即 $\mathcal{G} < \rho$）且足以访问每个模式的概率下界。

命题2（足够数量马尔可夫链的概率界）。设 π_1、…、π_ν分别为由独立启动的马尔可夫链访问这 ν个功能不同模式的概率，且 $\Pi_J := \sum_{j\in J}\pi_j$。那么，给定 ρ 条链，

$$\mathbb{P}(\mathcal{G} < \rho) \geq 1 - \rho^{-1}\left\{\sum_{q=0}^{\nu-1}(-1)^{\nu-1-q}\sum_{J:|J|=q}(1-\Pi_J)^{-1}\right\}. \tag{5}$$

证明见补充材料 A.6。请注意，该上界与神经网络架构无关，只取决于对功能多样模式数量及其概率的假设 ν，不考虑对称副本。命题2可用于在特定假设下计算马尔可夫链蒙特卡罗链的数量——例如，根据领域知识，或在最坏情况计算中——从而为多层感知机的 MCMC 采样提供实用指导。鉴于

表 1. 测试集上的平均逐点对数预测密度（LPPD）值（越大越好；括号内为一个标准误）。每个数据集和网络中表现最佳的结果以粗体标出。

	较小的网络 f_1			**较大的网络** f_2		
	MCMC（我们的方法）	MCMC（s.c.）	DE	MCMC（我们的方法）	MCMC（s.c.）	DE
$\mathcal{D}_S$	**-0.53** (± 0.09)	-0.56 (± 0.11)	-0.58 (± 0.11)	**-0.59** (± 0.12)	**-0.59** (± 0.12)	-2.13 (± 0.03)
$\mathcal{D}_I$	**0.79**(± 0.06)	0.65（± 0.07）	0.56（± 0.06）	**0.91** （± 0.09）	**0.91** （± 0.09）	-2.02（± 0.02）
$\mathcal{D}_R$	0.64 (± 0.10)	**0.75** (± 0.11)	-1.46 (± 0.06)	**0.95** (± 0.08)	**0.95** (± 0.08)	-2.20 (± 0.02)
翼型	**-0.74** (± 0.04)	-0.80 (± 0.05)	-1.62 (± 0.03)	**0.92** (± 0.05)	0.72 (± 0.10)	-2.17 (± 0.01)
混凝土	**-0.41** (± 0.05)	-0.44 (± 0.06)	-1.59 (± 0.03)	**0.26** (± 0.07)	0.25 (± 0.07)	-2.03 (± 0.01)
糖尿病	**-1.20** (± 0.07)	**-1.20** (± 0.07)	-1.47 (± 0.07)	**-1.18** (± 0.08)	-1.22 (± 0.09)	-2.09 (± 0.04)
能源	**0.92**(± 0.04)	0.69 (± 0.12)	-1.76 (± 0.02)	2.07 (± 0.46)	**2.38** (± 0.11)	-1.99 (± 0.02)
ForestF	**-1.37** (± 0.07)	**-1.37** (± 0.07)	-1.60 (± 0.06)	**-1.43** (± 0.45)	-1.69 (± 0.49)	-2.20 (± 0.02)
游艇	**1.90**(± 0.16)	1.29 (± 0.56)	-1.14 (± 0.14)	**3.31** (± 0.21)	0.15 (± 0.09)	-2.18 (± 0.03)

LA 和差分进化等局部近似方法 [28, 29]，具有相当高的预测性能，我们认为，在实践中假设功能模式数量较少是合理的。我们在第5节中的定性实验支持这一推断。

作为应用命题2的示例，假设参考集中有 $\nu = 3$ 个功能多样模式，$\pi_1 = 0.57$, $\pi_2 = 0.35$，$\pi_3 = 0.08$（所选模式旨在代表一个相当多样的功能模式集）。链数上限为 $\rho = 1274$，可确保我们以 $\mathbb{P}(\mathcal{G} < \rho) \geq 0.99$ 的概率观测到所有功能多样模式。

5 实验

现在，我们来考察理论发现，并将由此得到的方法与单链马尔可夫链蒙特卡洛和差分进化进行比较。在所有实验中[8]，，我们采用贝叶斯回归模型，其似然函数为正态似然函数，参数 $\boldsymbol{\theta}$ 服从标准正态先验，正态似然的方差（视为干扰参数）则服从限制在正实数范围内的截断标准正态先验。根据任务不同，我们使用无转向采样器 [22]，经 2^{10} 步预热后从后验分布中采集一个样本，或者使用基于梯度的方法求得最大后验估计量（详情见补充材料 C.2、C.3）。

5.1 性能比较

在我们的第一个实验中，我们使用推导出的链数上界（我们的方法），展示贝叶斯神经网络的预测性能，其中后验预测分布基于 MCMC采样计算。在这种情况下，我们对 G 条链各收集一个样本，因此总计 G 个样本。我们将其与从单条链（s.c.）收集 G 个样本的 MCMC采样，以及在三个合成数据集（$\mathcal{D}_S$、 $\mathcal{D}_I$ 和 $\mathcal{D}_R$）上使用十个集成成员的差分进化进行比较；此外，还使用来自 [11] 的基准数据（有关数据集的详细信息及 LA 上的补充结果，

[8] https://anonymous.4open.science/r/efficientmcmc sampling in bnns symmetry-6FAF/

分别参见补充材料 C.1 和E.1）。我们使用一个较小的神经网络f_1 ，其包含一个隐藏层和三个神经元；以及一个较大的网络 f_2，其包含三个隐藏层，每层有16个神经元；二者均采用双曲正切激活。如第 4.2节所述，我们假设存在三个功能多样模式 $\nu = 3$ 以及模式概率 $\pi_1 = 0.57, \pi_2 = 0.35, \pi_3 = 0.08$，如给定示例中所示。为了展示基于马尔可夫链蒙特卡洛的后验预测分布近似的性能，我们使用逐点对数预测密度（LPPD； [18]）在测试数据上测量拟合优度

$$\mathrm{LPPD} = \log \int_\Theta p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{D}) \, \mathrm{d}\boldsymbol{\theta} \approx \log \left(\tfrac{1}{G} \sum_{g=1}^{G} p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{\theta}^{(g)}) \right), \quad (6)$$

其中， $\boldsymbol{\theta}^{(1)}, \ldots, \boldsymbol{\theta}^{(G)}$ 是通过马尔可夫链蒙特卡洛采样，从参数后验密度 $p(\boldsymbol{\theta}|\mathcal{D})$ 中跨所有链获得的 G 样本。对于每个测试点 $(\boldsymbol{x}^*, \boldsymbol{y}^*)$，都会计算公式(6)。表 1 报告了每种数据集和采样方案组合下， N^* 个独立测试点的平均 LPPD（详见附录C.1）。我们的结果清楚地表明，按照我们的方法，仅使用适量的马尔可夫链，在除两个实验之外的所有实验中，都能取得与单链马尔可夫链蒙特卡罗和差分进化相当甚至更好的性能。

5.2 推论1的实际评估

接下来，我们使用所提出的链的上界来研究推论1中推导出的性质，并再次采用第4.2节示例中的假设。为此，我们使用网络 f_2 分析数据集 $\mathcal{D}_I$ 的后验预测分布。对于马尔可夫链蒙特卡洛运行中新收集的每个样本，都会在二维（输入/输出）网格上近似计算更新后的后验预测分布。然后，在 f_2 的输入值网格上对连续密度之间的 Kullback-Leibler（KL）散度取平均值（详见补充材料 D.1）。如图2所示，尽管网络 f_2 的规模以及等输出参数状态的数量很大 $|\mathcal{E}| = \left(16! \cdot 2^{16}\right)^3 \approx 2.58 \cdot 10^{54}$，后验预测分布在远少于$|\mathcal{E}|$ 个样本后便已收敛，并且函数空间的图表表明，仅经过来自同样多条链的1274个样本，函数多样性就已经饱和。

5.3 后验分布对称性消除

迄今为止，我们主要关注 MCMC采样的预测性能。然而，如第 4.1 节所述，将所有样本映射到一个联合代表集，有望大幅缩减有效权重空间，从而提升可解释性，甚至可能实现解析近似。为此，我们针对双曲正切激活多层感知机提出了一种定制算法，作为概念验证。该算法以数据依赖方式消除对称性，从而将代表集中剩余模式的数量降至最低（详情见补充材料 B）。

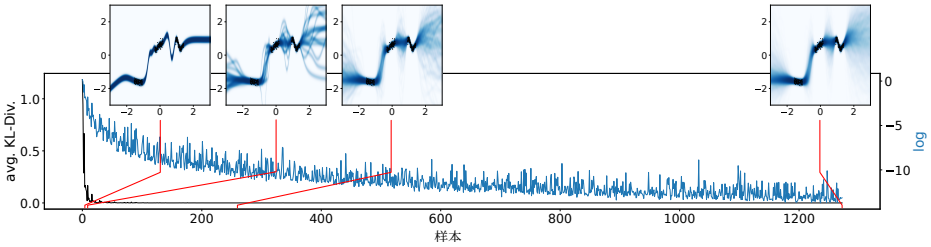

图 2. 马尔可夫链蒙特卡洛的收敛情况，以连续添加来自新的独立链的样本并重新估计后验密度时，KL散度在原始尺度（黑色）和对数尺度（蓝色）上的变化表示。叠加的小图：网络在获得 $2^0, 2^4, 2^8,$ 和 $G = 1274$ 个样本后的近似后验预测分布；颜色越深表示概率越高。

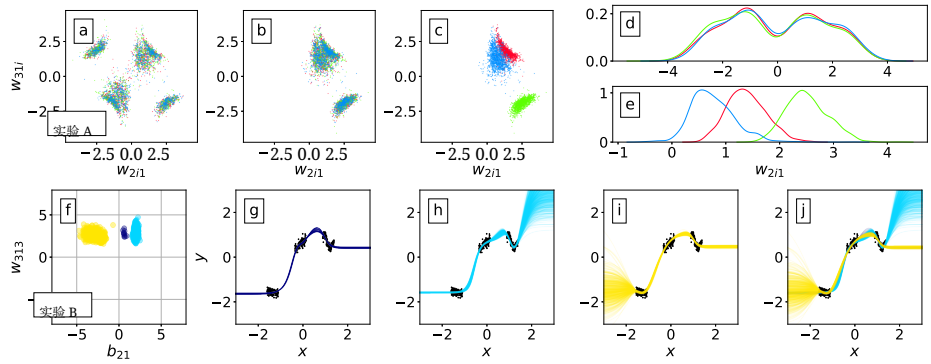

图3。对于实验A（顶行），在运行马尔可夫链蒙特卡洛（a）时获得的初始状态，以及在算法4的步骤中经过变换和重新分配后的状态（b、c）下，对隐藏神经元的参数子空间进行了可视化。不同颜色表示隐藏层中当前的神经元索引。通过重新分配神经元，算法4有效地找到了一个最优参考集，并将多峰复杂的单变量边缘密度（d）分离为三个单变量密度。对于实验B（底行），消除对称性后的多层感知机参数后验密度形成了一个三峰系统，此处以双变量图（f）进行了展示。研究这些模式后，我们发现它们在功能上各不相同，即分别代表了数据集的不同假设（g-i）。将它们结合起来，便形成了完整的函数空间（j）。

我们通过针对数据集 $\mathcal{D}_S$ 和 $\mathcal{D}_I$ 的实验A和B，分别展示了该方法对 f_1 的有效性。对于实验A（图 3 上方），我们沿着所提出算法的各个步骤，将神经元参数空间可视化（详情见补充材料 B 中的算法1-4）。不同颜色表示当前神经元索引（即 $\{1, 2, 3\}$ 之一），对应于相应神经元参数向量的隐藏层。起初，后验密度的对称性十分明显（图 3a），而神经元参数向量

具有相同的分布（图 3d）。在算法的第一步（图 3b）中，由于移除了双曲正切所诱导的符号翻转对称性，参数空间实际上缩减为原来的一半，并保留了三个簇。第二步（聚类）通过将该参数子空间的区域分配给模型中的神经元，移除了完整后验密度中的所有置换对称性。图 3c 展示了这一过程，并清楚地显示了由不同簇颜色区分的状态。在图 3d 和图 3e 中，每个神经元传入权重 w_{2i1} 的单变量边缘密度揭示了它们在参数空间中的重新分配。运行我们的算法后，每个神经元都呈现出独特的单峰密度，从而使完整参数空间中的密度也呈现单峰特征（可视化见补充材料 E.3）。我们可以得出结论，在这种情况下只存在一个功能多样的模态，局部近似方法（如 LA）也应能够恢复该模态。

对于实验B（图3下方），我们重点展示应用对称性消除算法后得到的完整参数后验密度（图3f）。BNN后验分布中仍保留着三个以不同颜色表示的功能多样模式。现在，我们可以通过进一步采用谱聚类方法对变换后的样本进行聚类，在函数空间中解释这些模式（详情见补充材料D.2）。图3g-j展示了函数空间中的网络参数状态，揭示了网络能够从给定数据集中学习到的三个功能多样假设。此类知识有助于获得更合适的近似，例如拉普拉斯近似混合（MoLA；[13]），如补充材料E.2所示。请注意，标准LA等只关注单一模式的方法，只能捕捉到三分之一的函数多样性。因此，要实现有意义的不确定性量化（UQ），必须考虑所有功能多样模式。

6 讨论

我们证明，由于等输出参数状态的存在，只需覆盖部分参数空间，就能获得贝叶斯多层感知机的后验预测分布（PPD）。结合保证恢复每个功能多样模式所需马尔可夫链蒙特卡洛链数量的上界，我们的方法为深度学习中的精确不确定性量化（误差范围仅限于蒙特卡罗误差）铺平了道路。此外，我们展示了对称性消除的应用，并提出一种概念验证方法，可将双曲正切激活多层感知机的样本映射到一个代表集。这一事后处理流程大幅提升了无对称性后验密度的可解释性，并有助于进行解析近似。作为未来的研究方向，我们计划探究能否通过集成训练，以信息丰富的方式初始化采样状态来改进我们的MCMC采样方法，并以 [19, 36, 46]中的见解为基础。

致谢。 LW 获得由德国联邦教育与研究部资助的 DAAD Konrad Zuse 卓越人工智能学院项目支持。

参考文献

1. Agrawal, D., Ostrowski, J.：G-不变浅层神经网络的分类。载于：神经信息处理系统进展（2022）2. Ainsworth, S., Hayase, J., Srinivasa, S.：Git re-basin：对置换对称性取模合并模型。载于：第十一届国际学习表征会议（2023），https://openreview.net/forum?id=CQsmMYmlP5T 3. Bardenet, R., K´egl, B.：一种用于从混合信号中进行推断的自适应蒙特卡罗马尔可夫链算法。《物理学杂志：会议系列》**368**（2012）4. Bona-Pellissier, J., Bachoc, F., Malgouyres, F.：深度前馈 ReLU 神经网络的参数可辨识性（2021）5. Van den Broeck, G., Kersting, K., Natarajan, S., Poole, D.：《提升概率推断导论》。MIT 出版社（2021），https://mitpress.mit.edu/ books/introduction-lifted-probabilistic-inference 6. Chen, A.M., Lu, H.m., Hecht-Nielsen, R.：前馈神经网络误差曲面的几何。《神经计算》**5**（6），910–927（1993）7. Cortes, C., Vapnik, V.：支持向量网络。《机器学习》**20**（3），273–297（1995）8. Cortez, P., Morais, A.d.J.R.：利用气象数据预测森林火灾的数据挖掘方法（2007）9. Daxberger, E., Kristiadi, A., Immer, A., Eschenhagen, R., Bauer, M., Hennig, P.：Laplace Redux——轻松实现贝叶斯深度学习。载于：第35届神经信息处理系统会议（NeurIPS 2021）（2021）10. Draxler, F., Veschgini, K., Salmhofer, M., Hamprecht, F.：神经网络能源景观中几乎不存在障碍。载于：第35届国际机器学习会议论文集。第1309–1318页。PMLR（2018）11. Dua, D., Graff, C.：UCI 机器学习资料库（2017）12. Ensign, D., Neville, S., Paul, A., Venkatasubramanian, S.：通过（群）不变量解释神经网络的复杂性。载于：机器学习研究论文集。第76卷（2017）13. Eschenhagen, R., Daxberger, E., Hennig, P., Kristiadi, A.：用于提升深度学习事后不确定性估计的拉普拉斯近似混合模型。载于：NeurIPS 2021 贝叶斯深度学习研讨会（2021）14. Falcon, W.：PyTorch lightning。https://github.com/PyTorchLightning（2019）15. Flajolet, P., Gardy, D., Thimonier, L.：生日悖论、优惠券收集问题、缓存算法与自组织搜索。《离散应用数学》**39**（3），207–229（1992）16. Fr¨uhwirth-Schnatter, S.：《经典与动态切换模型及混合模型的马尔可夫链蒙特卡罗估计》。《美国统计学会杂志》**96**（453），194–209（2001）17. Garipov, T., Izmailov, P., Podoprikhin, D., Vetrov, D.P., Wilson, A.G.：DNN 的损失曲面、模态连通性与快速集成。载于：第32届神经信息处理系统会议论文集（NeurIPS 2018）（2018）18. Gelman, A., Hwang, J., Vehtari, A.：理解贝叶斯模型的预测信息准则。《统计与计算》**24**（6），997–1016（2014）19. Graf, S., Luschgy, H.：《概率分布量化基础》。Springer（2007）20. Hecht-Nielsen, R.：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。Elsevier（1990）

21. Hinton,G., Srivastava, N., Swersky, K.：RMSProp：将梯度除以其近期幅度的运行平均值（机器学习神经网络课程）（2012）22.Hoffman,M.D., Gelman, A.：无转向采样器：在哈密顿蒙特卡洛中自适应地设置路径长度。机器学习研究杂志**15(47)**，1593–1623（2014） 23. H¨ullermeier,E.,Waegeman, W.：机器学习中的偶然不确定性与认知不确定性：概念与方法简介。机器学习（2021） 24. Izmailov, P., Maddox, W.J., Kirichenko, P.,Garipov,T., Vetrov,D., Wilson, A.G.：贝叶斯深度学习的子空间推断。载于：第35届人工智能不确定性会议论文集。第1169–1179页。PMLR（2020） 25. Izmailov, P., Vikram,S.,Hoffman,M.D.,Wilson, A.G.：贝叶斯神经网络后验分布究竟是什么样的？载于：第38届国际机器学习会议论文集，PMLR 139（2021）26. Kipf, T.N., Welling, M.：使用图卷积网络进行半监督分类。载于：ICLR 2017（2017）27. K◦urkov´a, V., Kainen, P.C.：函数等价的前馈神经网络。神经计算**6(3)**，543–558（1994） 28. Lakshminarayanan, B.,Pritzel,A., Blundell, C.：使用深度集成进行简单且可扩展的预测不确定性估计。载于：第31届神经信息处理系统会议论文集（NIPS2017）（2017） 29. MacKay, D.J.C.：贝叶斯插值。神经计算 **4**，415–447 （ 1992 ） 30. Margossian, C.C., Hoffman,M.D., Sountsov, P., Riou-Durand, L., Vehtari,A., Gelman,A.：嵌套的$\ hatR$：运行多个短链时评估马尔可夫链蒙特卡洛的收敛性（2022）31. Nalisnick, E.T.：关于贝叶斯神经网络的先验分布。博士论文，加利福尼亚大学欧文分校（2018）32. Niepert,M.：置换群轨道上的马尔可夫链。载于：第二十八届人工智能不确定性会议论文集。第624–633页。UAI'12，AUAI出版社，美国弗吉尼亚州阿灵顿（2012） 33.Niepert, M.：对称性感知边缘密度估计。载于：第二十七届AAAI人工智能会议论文集。第725–731页。AAAI'13，AAAI（2013） 34. Papamarkou, T.,Hinkle,J., Young,M.T.,Womble, D.：贝叶斯神经网络中的马尔可夫链蒙特卡洛挑战。统计科学 **37(3)**（2022）35. Paszke,A., 等：PyTorch：一种命令式风格的高性能深度学习库。载于：第33届神经信息处理系统会议（2019） 36. Pearce,T., Leibfried,F.,Brintrup, A.：神经网络中的不确定性：近似贝叶斯集成。载于：第二十三届国际人工智能与统计会议论文集。机器学习研究论文集，第108卷，第234–244页。PMLR（2020年8月26–28日） 37. Petzka, H., Trimmel,M.,Sminchisescu, C.：关于 2 层 ReLU 网络对称性的注记。Northern Lights深度学习研讨会**1** （2020） 38. Pittorino, F., Ferraro, A., Perugini, G., Feinauer,C.,Baldassi,C.,Zecchina,R.：环面上的深度网络：消除对称性揭示景观几何中平坦区域的结构。载于：第39届国际机器学习会议论文集，PMLR 162（2022） 39. Pourzanjani,A.A.,Jiang,R.M.,Petzold,L.R.：改进用于贝叶斯推断的神经网络的可辨识性。载于：第二届贝叶斯深度学习研讨会（NIPS）（2017）

40. Rosenthal, J.S.：并行计算与蒙特卡罗算法。远东理论统计学杂志**4**，207–236（2000） 41. Sen,D.、Papamarkou, T.、Dunson,D.：贝叶斯神经网络与降维（2020 08044 422008），arXiv：.. Sussmann,H.J.：具有给定输入-输出映射的最小前馈网络的权重唯一性。神经网络 **5**（4），589–593（1992） 43. Tatro, N.J.、Chen, P.Y.、Das,P.、Melnyk,I.、Sattigeri, P.、Lai, R.：通过神经元对齐优化模态连通性。载于：第34届神经信息处理系统会议论文集（NeurIPS 2020）（2020） 44. Tsanas, A.、Xifara,A.：使用统计机器学习工具准确量化估计住宅建筑的能源性能。能源与建筑 **49**，560–567（2012） 45.Vlaˇci´c,V.、B¨olcskei, H.：仿射对称性与神经网络可辨识性。数学进展 **376**，107485（2021）。https://doi.org/https: //doi.org/10.1016/j.aim.2020.107485, https://www.sciencedirect.com/science/article/pii/S0001870820305132 46. Wilson,A.G.、Izmailov, P.：贝叶斯深度学习与泛化的概率视角。载于：第34届神经信息处理系统国际会议论文集。NIPS’20，CurranAssociates Inc.，美国纽约州雷德胡克（2020） 47.Yeh,I.C.：使用人工神经网络对高性能混凝土强度进行建模。水泥与混凝土研究 **28**（12），1797–1808（1998）

A 推导

A.1 等输出参数状态的函数等价性

我们考虑参数状态的置换相关和激活函数相关的等输出变换。下面给出神经元互换以及两种激活函数——双曲正切和 ReLU——所对应的函数等价性。

置换相关等价性。 根据加法的交换律，在第 $l-1$ 层中置换两个神经元 v， w，不会改变第 l 层第 i 个神经元的预激活 o_{li}。

$$\begin{aligned} o_{li} &= (w_{li1}z_{(l-1)1} + b_{l1}) + \cdots + (w_{liv}z_{(l-1)v} + b_{lv}) + (w_{liw}z_{(l-1)w} + b_{lw}) \\ &\qquad + \cdots + (w_{liM_{l-1}}z_{(l-1)M_{l-1}} + b_{lM_{l-1}}) \\ &= (w_{li1}z_{(l-1)1} + b_{l1}) + \cdots + (w_{liw}z_{(l-1)w} + b_{lw}) + (w_{liv}z_{(l-1)v} + b_{lv}) \\ &\qquad + \cdots + (w_{liM_{l-1}}z_{(l-1)M_{l-1}} + b_{lM_{l-1}}) \end{aligned}$$

双曲正切相关等价性。 由于双曲正切是奇函数，将所有连接到输入神经元（来自第 $l-1$ 层）输出端以及输出神经元（通往第 $l+1$ 层）的参数的符号取反，同时将第 l 层对应的偏置参数符号取反，不会改变输出。

$$\begin{aligned} w_{(l+1)ki}\tanh(o_{li}) &= w_{(l+1)ki}\tanh\left(\sum_{j=1}^{M_{l-1}} w_{lij}z_{(l-1)j} + b_{li}\right) \\ &= -w_{(l+1)ki}\tanh\left(\sum_{j=1}^{M_{l-1}} -w_{lij}z_{(l-1)j} - b_{li}\right) \\ &= -w_{(l+1)ki}\tanh(-o_{li}) \end{aligned}$$

与 ReLU 相关的等式。 与双曲正切类似，在 ReLU 的情况下，对输入（来自第 $l-1$ 层）和输出神经元（传递至第 $l+1$ 层）的参数以及第 l 层的偏置参数进行缩放，会生成等输出参数状态。

$$\begin{aligned} w_{(l+1)ki}\mathrm{ReLU}(o_{li}) &= w_{(l+1)ki}\mathrm{ReLU}\left(\sum_{j=1}^{M_{l-1}} w_{lij}z_{(l-1)j} + b_{li}\right) \\ &= c\cdot w_{(l+1)ki}\mathrm{ReLU}\left(\sum_{j=1}^{M_{l-1}} c^{-1}\cdot w_{lij}z_{(l-1)j} + c^{-1}\cdot b_{li}\right) \\ &= c\cdot w_{(l+1)ki}\mathrm{ReLU}(c^{-1}\cdot o_{li}) \end{aligned}$$

A.2 等输出等价关系

设 $f\colon\ \mathcal{X} \to \mathcal{Y}$ 表示一个将特征向量 $\boldsymbol{x} \in \mathcal{X} \subseteq \mathbb{R}^n,\ n \in \mathbb{N}$，映射到结果向量 $f(\boldsymbol{x}) \in \mathcal{Y} \subseteq \mathbb{R}^m, m \in \mathbb{N}$ 的多层感知机。若两个参数状态 $\boldsymbol{\theta},\ \boldsymbol{\theta}' \in \Theta$ 对应的映射 $f_{\boldsymbol{\theta}},\ f_{\boldsymbol{\theta}'}$ 对所有可能的输入都产生相同的输出，则称它们具有等输出性。我们用 $\sim$ 表示这种关系，并写作：

$$\boldsymbol{\theta} \sim \boldsymbol{\theta}' \Leftrightarrow f_{\boldsymbol{\theta}}(\boldsymbol{x}) = f_{\boldsymbol{\theta}'}(\boldsymbol{x}) \forall\, \boldsymbol{x} \in \mathcal{X},\ \ \boldsymbol{\theta}, \boldsymbol{\theta}' \in \Theta.$$

$\sim$ 构成一个等价关系，这可由相等关系是等价关系立即推出：

E1（自反性）对于 $\boldsymbol{\theta} \in \Theta, \boldsymbol{\theta} \sim \boldsymbol{\theta}$ 因为 $f_{\boldsymbol{\theta}}(\boldsymbol{x}) = f_{\boldsymbol{\theta}}(\boldsymbol{x}) \forall\, \boldsymbol{x} \in \mathcal{X}$。
E2（对称性）对于 $\boldsymbol{\theta}, \boldsymbol{\theta}' \in \Theta$,

$$\boldsymbol{\theta} \sim \boldsymbol{\theta}' \Leftrightarrow f_{\boldsymbol{\theta}}(\boldsymbol{x}) = f_{\boldsymbol{\theta}'}(\boldsymbol{x}) \forall\, \boldsymbol{x} \in \mathcal{X} \Rightarrow f_{\boldsymbol{\theta}'}(\boldsymbol{x}) = f_{\boldsymbol{\theta}}(\boldsymbol{x}) \forall\, \boldsymbol{x} \in \mathcal{X} \Leftrightarrow \boldsymbol{\theta}' \sim \boldsymbol{\theta}$$

E3（传递性）令 $\boldsymbol{\theta}_1 \sim \boldsymbol{\theta}_2$ 和 $\boldsymbol{\theta}_2 \sim \boldsymbol{\theta}_3$ 对于 $\boldsymbol{\theta}_1,\ \boldsymbol{\theta}_2,\ \boldsymbol{\theta}_3 \in \Theta$。由于

$$\boldsymbol{\theta}_1 \sim \boldsymbol{\theta}_2 \Leftrightarrow f_{\boldsymbol{\theta}_1}(\boldsymbol{x}) = f_{\boldsymbol{\theta}_2}(\boldsymbol{x}) \forall\, \boldsymbol{x} \in \mathcal{X}$$

and

$$\boldsymbol{\theta}_2 \sim \boldsymbol{\theta}_3 \Leftrightarrow f_{\boldsymbol{\theta}_2}(\boldsymbol{x}) = f_{\boldsymbol{\theta}_3}(\boldsymbol{x}) \forall\, \boldsymbol{x} \in \mathcal{X},$$

由等式的传递性可得，

$$f_{\boldsymbol{\theta}_1}(\boldsymbol{x}) = f_{\boldsymbol{\theta}_3}(\boldsymbol{x}) \forall\, \boldsymbol{x} \in \mathcal{X} \Leftrightarrow \boldsymbol{\theta}_1 \sim \boldsymbol{\theta}_3.$$

A.3 后验密度参考 et

先验不变性 对于神经网络参数状态 $\boldsymbol{\theta} \in \Theta \subseteq \mathbb{R}^d$ 和任意等输出参数状态 $\boldsymbol{E}_j \in \mathcal{E}$, 若假设先验分布在变换 $\mathcal{F}_{\boldsymbol{E}_j} : \Theta \to \Theta、\ \boldsymbol{\theta} \mapsto \boldsymbol{E}_j\boldsymbol{\theta}、\ \boldsymbol{E}_j \in \mathcal{E}$ 下保持不变，则 $\boldsymbol{\theta}$ 和 $\boldsymbol{E}_j\boldsymbol{\theta}$ 的先验概率相同，即 $p(\boldsymbol{\theta}) = p(\boldsymbol{E}_j\boldsymbol{\theta})$。根据定义，等输出参数状态会产生相同的函数映射，因此为它们赋予相同的先验概率是合理的。一个典型例子是普遍使用的各向同性高斯先验。以下展示了在这种情况下，原始权重向量与变换后权重向量的先验概率等价性。

$$\begin{aligned} p(\boldsymbol{\theta}) = \mathcal{N}(\boldsymbol{\theta}|\mathbf{0}, \boldsymbol{I}) &= \frac{1}{(2\pi)^{d/2}} \exp\left(-\frac{1}{2}\boldsymbol{\theta}^\top\boldsymbol{\theta}\right) && (7)\\ &= \frac{1}{(2\pi)^{d/2}} \exp\left(-\frac{1}{2}\boldsymbol{\theta}^\top\boldsymbol{E}_j^\top\boldsymbol{E}_j\boldsymbol{\theta}\right) && (8)\\ &= \frac{1}{(2\pi)^{d/2}} \exp\left(-\frac{1}{2}(\boldsymbol{E}_j\boldsymbol{\theta})^\top(\boldsymbol{E}_j\boldsymbol{\theta})\right) && (9)\\ &= \mathcal{N}(\boldsymbol{E}_j\boldsymbol{\theta}|\mathbf{0}, \boldsymbol{I}) = p(\boldsymbol{E}_j\boldsymbol{\theta}) && (10) \end{aligned}$$

$\boldsymbol{E}_j$ 的正交性源于 $\boldsymbol{E}_j = \boldsymbol{T}\boldsymbol{P}$ 是正交矩阵的乘积。 $\boldsymbol{P}$ 是一个置换矩阵，因此是正交的。

$\boldsymbol{T}$ 是一个对角矩阵，同时也是正交矩阵，因为其所表示的变换经过精心设计，使得 $\boldsymbol{T}^\top \boldsymbol{T}$ 是单位矩阵（例如，对于双曲正切，对角值等于 -1 ）。利用 $\boldsymbol{E}_j$ 的正交性，可以从方程(7)推导到方程(8)。不过请注意，诸如由 ReLU诱导的缩放变换通常无法用正交矩阵表示。由于可能存在无限多种缩放变换，这些变换构成的问题不太容易具体刻画。

似然不变性 同样，我们证明似然对于等输出参数状态是不变的，依据是 $\boldsymbol{\theta}$ 和 $\boldsymbol{E}_j\boldsymbol{\theta}$ 是等输出参数状态:

$$p(\mathcal{D}|\boldsymbol{\theta}) = \prod_{i=1}^{N} p(\boldsymbol{y}^{(i)}|\boldsymbol{x}^{(i)}, \boldsymbol{\theta}) = \prod_{i=1}^{N} p(\boldsymbol{y}^{(i)}|f_{\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})) \tag{11}$$

$$= \prod_{i=1}^{N} p(\boldsymbol{y}^{(i)}|f_{\boldsymbol{E}_j\boldsymbol{\theta}}(\boldsymbol{x}^{(i)})) \tag{12}$$

$$= \prod_{i=1}^{N} p(\boldsymbol{y}^{(i)}|\boldsymbol{x}^{(i)}, \boldsymbol{E}_j\boldsymbol{\theta}) = p(\mathcal{D}|\boldsymbol{E}_j\boldsymbol{\theta}) \tag{13}$$

后验不变性 根据公式 7 至 13，两个等输出参数状态的后验概率相同:

$$p(\boldsymbol{\theta}|\mathcal{D}) = \frac{p(\mathcal{D}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathcal{D})} \underset{(14)}{\overset{(13)}{=}} \frac{p(\mathcal{D}|\boldsymbol{E_j\theta})p(\boldsymbol{E}_j\boldsymbol{\theta})}{p(\mathcal{D})} = p(\boldsymbol{E}_j\boldsymbol{\theta}|\mathcal{D}) \tag{14}$$

A.4 命题1的证明概要

这里，我们给出命题1的证明概要，其基本依据是 [6]，并考虑了如何处理边缘和边界情况。设 $\mathcal{S}_1 \subset \Theta$ 是由等输出关系 $\sim$所诱导的等价类中的一组完整代表，满足 $\forall \boldsymbol{\theta} \in \Theta\ \exists_1 \boldsymbol{\theta}' \in \mathcal{S}_1 :\ \boldsymbol{\theta} \sim \boldsymbol{\theta}'$。我们将 $\mathcal{S}_1$ 称为唯一标识的网络参数状态的参考集 （参见 [6]中的开放式最小充分搜索集）。设 $\mathcal{S}_j$ 是 $\mathcal{S}_1$ 在由 $\boldsymbol{E}_j \in \mathcal{E}$诱导的变换 $\mathcal{F}_{\boldsymbol{E}_j}$ 下的像，满足

$$\mathcal{S}_j = \mathcal{F}_{\boldsymbol{E}_j}(\mathcal{S}_1) = \{\boldsymbol{E}_j\boldsymbol{\theta} : \boldsymbol{\theta} \in \mathcal{S}_1, \boldsymbol{E}_j \in \mathcal{E}\}, \quad j = 1, \ldots, |\mathcal{E}|.$$

现在，我们通过以下操作收集残集 $\mathcal{S}^0$ 中的所有边界和边缘情形:

$$\mathcal{S}^0 := \Theta \setminus \bigcup_{j=1}^{|\mathcal{E}|} \mathcal{S}_j$$

$$\mathcal{S}^0 = \mathcal{S}^0 \cup \bigcup_{i \neq j} (\mathcal{S}_i \cap \mathcal{S}_j)$$

$$\forall j = 1, \ldots, |\mathcal{E}| : \mathcal{S}_j = \mathcal{S}_j \setminus \mathcal{S}^0$$
$$\Rightarrow \forall i, j \in \{1, \ldots, |\mathcal{E}|\}, i \neq j : \mathcal{S}_i \cap \mathcal{S}_j = \emptyset,\ \mathcal{S}_j \cap \mathcal{S}^0 = \emptyset$$

由此，我们得到集合 $\mathcal{S}_i, \mathcal{S}_j$ ，它们两两不相交，且与残集 $\mathcal{S}^0$ 的交集为空。还需证明这些集合的并等于 Θ。为此，考虑满足 $\boldsymbol{\theta} \notin \left(\mathcal{S}^0 \cup \bigcup_{j=1}^{|\mathcal{E}|} \mathcal{S}_j\right)$的 $\boldsymbol{\theta} \in \Theta$。根据定义，参考集包含每个等价类的一个代表元，因此存在 $\boldsymbol{\theta}' \in \mathcal{S}_1$ ： $\boldsymbol{\theta} \sim \boldsymbol{\theta}'$。然而，由此可推出 $\exists j \in \{1, \ldots, |\mathcal{E}|\}$ ： $\boldsymbol{\theta} = \boldsymbol{E}_j \boldsymbol{\theta}' \Rightarrow \boldsymbol{\theta} \in \mathcal{S}_j$，尤其是 $\boldsymbol{\theta} \in \left(\mathcal{S}^0 \cup \bigcup_{j=1}^{|\mathcal{E}|} \mathcal{S}_j\right)$，从而得到矛盾。

A.5 推论1的证明

结合 A.3 和 A.4 中的结果，可得后验预测密度表示为:

$$p(\boldsymbol{y}^*|\boldsymbol{x}^*, \mathcal{D}) = \int_{\Theta} p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{D}) \, \mathrm{d}\boldsymbol{\theta} \tag{15}$$
$$= \sum_{i=1}^{|\mathcal{E}|} \int_{\mathcal{S}_i} p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{D}) \, \mathrm{d}\boldsymbol{\theta} \tag{16}$$
$$+ \underbrace{\int_{\mathcal{S}^0} p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{D}) \, \mathrm{d}\boldsymbol{\theta}}_{c}$$
$$= \sum_{i=1}^{|\mathcal{E}|} \int_{\mathcal{S}_i} p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{D}) \, \mathrm{d}\boldsymbol{\theta} + c \tag{17}$$
$$= \sum_{i=1}^{|\mathcal{E}|} \int_{\mathcal{S}_1} p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{E}_i\boldsymbol{\theta}) p(\boldsymbol{E}_i\boldsymbol{\theta}|\mathcal{D}) \, \mathrm{d}\boldsymbol{\theta} + c \tag{18}$$
$$= \int_{\mathcal{S}_1} \sum_{i=1}^{|\mathcal{E}|} p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{E}_i\boldsymbol{\theta}) p(\boldsymbol{E}_i\boldsymbol{\theta}|\mathcal{D}) \, \mathrm{d}\boldsymbol{\theta} + c \tag{19}$$
$$= \int_{\mathcal{S}_1} |\mathcal{E}| \cdot p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{D}) \, \mathrm{d}\boldsymbol{\theta} + c \tag{20}$$
$$= |\mathcal{E}| \cdot \int_{\mathcal{S}_1} p(\boldsymbol{y}^*|\boldsymbol{x}^*, \boldsymbol{\theta}) p(\boldsymbol{\theta}|\mathcal{D}) \, \mathrm{d}\boldsymbol{\theta} + c \tag{21}$$

公式 (1) 可将公式 (15) 转换为公式 (16)。如公式 (13) 和 (14) 所述，似然和后验不变性使我们能够将公式 (17) 重写为公式 (18)。

A.6 马尔可夫链的上界

现在我们证明公式 (5) 得出的上界。首先注意，根据马尔可夫不等式，链的数量 G 满足

$$P(G \geq \rho) \leq \mathbb{E}(G) \cdot \rho^{-1}. \tag{22}$$

我们可以将这个不等式改写为

$$P(G < \rho) \geq 1 - \mathbb{E}[G] \cdot \rho^{-1}. \tag{23}$$

至少访问每个模式一次所需的独立链数量，可以表述为广义优惠券收集问题 [15], ，其中需要的抽样次数（独立链数量）用于收集（访问）所有 ν 优惠券（模式），并对其进行估计。这为我们提供了所需链数量期望值的表达式：

$$\mathbb{E}(G) = \sum_{q=0}^{\nu-1} (-1)^{\nu-1-q} \sum_{J:|J|=q} (1 - \Pi_J)^{-1}, \quad \text{with } \Pi_J = \sum_{j \in J} \pi_j. \tag{24}$$

结合公式（23）和公式（24），我们得到公式（5）。

B 后验分布对称性消除

第4.2节利用等输出参数状态的知识，推导出观测多层感知机参数后验密度的所有功能多样模式所需的马尔可夫链数量上界。然而，获得的样本分散在集合 $\mathcal{S}_j$ 中；参见第4节和公式（2）。这掩盖了对网络参数后验密度几何结构的任何洞见，使得除了获得的函数不确定性之外，几乎无法进行任何解释。因此，理想情况下，所有样本都应映射到一个代表集。下面，我们将为双曲正切激活多层感知机构造一种对称性消除算法。

推理。 假设有 G 个可用的后验样本 $\boldsymbol{\theta}^{(g)}$, $g \in \{1, \ldots, G\}$。由于多层感知机中的等输出变换按层分解（见第 3 节），可以通过遍历 $K-2$ 隐藏网络层来消除对称性。我们提议按逆序处理各层，其依据是：前一层的输出 $l-1$ 可视为具有 M_l 个神经元的单层多层感知机的输入。因此，以这种方式处理多层感知机，相当于从 $K-2$ 个相互独立的单层多层感知机中消除对称性。

在每一步中，只需考虑当前隐藏层 l 中神经元的参数。对于第 l 层中的隐藏神经元 $i \in \{1, \ldots, M_l\}$，对应的参数是该神经元输出端的权重和偏置 o_{li}

以及连接该神经元与下一层的权重。神经元参数向量 $\boldsymbol{\phi}^{(g,l,i)} \in \mathbb{R}^{(M_{(l-1)}+M_{(l+1)}+1)}$ 定义为

$$\boldsymbol{\phi}^{(g,l,i)} = (w_{li1}^{(g)}, \ldots, w_{liM_{(l-1)}}^{(g)}, w_{(l+1)1i}^{(g)}, \ldots, w_{(l+1)M_{(l+1)}i}^{(g)}, b_{li}^{(g)})^\top,$$

其中 $w_{lij}^{(g)}$， $b_{li}^{(g)}$ 对于 $2 \leq l \leq K-1$， $1 \leq i \leq M_l$， $1 \leq j \leq M_{(l-1)}$。该向量属于 Θ 的一个子空间，其维数取决于前一层和后一层的宽度。由于神经元可以自由互换（见第 3 节），同一隐藏层中可自由置换神经元的边缘后验分布相同。这意味着，每个 $\boldsymbol{\phi}^{(g,l,i)}$ 的边缘密度至少包含 M_l 个不同区域，神经元可能被映射到参考集中的任一区域，因此存在 $M_l!$ 种不同的置换配置。通过为隐藏神经元找到一个有效的参考状态分配，可以消除与置换相关的对称性，从而将参数子空间有效地划分为 M_l 个区域。对于下文讨论的双曲正切激活多层感知机，由于存在符号翻转的等输出参数状态，状态数量还会加倍。因此，在消除置换对称性之前，我们以遵循几何结构的方式将参数子空间一分为二，以消除与双曲正切相关的对称性（详情如下）。接下来，我们将介绍一种用于双曲正切激活多层感知机的自动对称性消除示例算法，无需显式指定 $\mathcal{E}$。

详细算法。 我们需要找到可辨识性约束，即在参数空间中选择一组超平面，将可用空间唯一地分配给某一层的各个神经元。这样一来，神经元就不可能具有对称配置，从而有效地消除对称性。为此，我们遵循 [16] 中的原则，并逐层处理神经元参数子空间；这些参数涉及该层中相应的等输出变换。令 $\mathcal{M}^{(g,l)} := \{\boldsymbol{\phi}^{(g,l,1)}, \ldots, \boldsymbol{\phi}^{(g,l,M_l)}\}$ 表示样本 g对应的第l层所有参数的集合，并令 $\mathcal{M}^{(*,l)} = \bigcup_{g=1}^{G} \mathcal{M}^{(g,l)}$。为了消除某一层 l中的与 tanh 相关的对称性，必须将相应神经元的参数子空间缩减一半。在遵循后验密度几何结构的同时将空间减半，会带来一个依赖于数据的优化问题，即寻找恰当的约束。在此，我们提出一个以零为中心的线性超平面$\boldsymbol{\beta}_l \in \mathbb{R}^{M_{(l-1)}+M_{(l+1)}+1}$ ，使其尽可能少地与神经元参数向量 $\boldsymbol{\phi} \in \mathcal{M}^{(*,l)}$ 的状态或簇相交。我们利用一种针对无标签数据的支持向量机（SVM； [7]）变体来优化该超平面，使其与所有神经元参数向量之间的距离最大。损失函数遵循 SVM 的合页损失形式，但将标签替换为$\boldsymbol{\beta}_l^\top \boldsymbol{\phi}$ 对于 $\boldsymbol{\phi} \in \mathcal{M}^{(*,l)}$的绝对值，形式为

$$\mathcal{L}(\boldsymbol{\beta}_l) = \tfrac{1}{2}\boldsymbol{\beta}_l^\top \boldsymbol{\beta}_l + C \cdot \textstyle\sum_{\boldsymbol{\phi} \in \mathcal{M}^{(*,l)}} \max\left(0, 1 - \left|\boldsymbol{\beta}_l^\top \boldsymbol{\phi}\right|\right),$$

其中， $C > 0$ 是与成本相关的超参数， $|\cdot|$ 是用户定义的范数（我们使用 L_1 范数，以此类比铰链损失）。损失最小超平面被选作遵循几何结构的约束，并用于翻转参数

将向量映射到超平面的一侧，即应用与双曲正切相关的等输出变换，并翻转对应参数向量条目的符号（算法 1）。在实践中，我们会使用随机初始化的超平面运行 $K_{\beta} > 1$ 次优化，因为这种支持向量机变体无法保证找到全局最优解。

算法 1 几何保持型双曲正切移除算法。

过程 TanhRemoval($\mathcal{M}^{(*,l)}$, K_{β})
  $\mathcal{R} \leftarrow \emptyset$
  对于每个 k 在范围 K_{β} **执行**
    $\beta_k \leftarrow \arg\min_{\beta} \mathcal{L}(\beta)$
    $\mathcal{R} \leftarrow \mathcal{R} \cup \{\beta_k\}$
  结束循环
  $\beta^* \leftarrow \beta \in \mathcal{R}$ 以最小损失 $\mathcal{L}(\beta)$
  对于每个 $\phi \in \mathcal{M}^{(*,l)}$ **执行**
    如果 $\beta^{*\top} \phi < 0$ **那么**
      $\phi \leftarrow -\phi$
    结束 if
  结束循环
结束过程

使用 SVM 方法移除与 tanh 相关的对称性后，必须将参数 l 所在层的边际后验密度中剩余的参数子空间划分给隐藏层神经元，以消除与置换相关的对称性。为此，我们每次考虑一个样本 $\boldsymbol{\theta}^{(g)}$，其元素为 $\phi^{(g,l,i)} \in \mathcal{M}^{(g,l)}$，并分配分类标签 $c^{(g,l,i)}$，$i \in \{1, \ldots, M_l\}$。初始时，每个元素都标记为其神经元索引 i。然后，我们使用高斯相似度函数 $s(a,\ b) = \exp\left(-||a-b||^2 \cdot (2\sigma^2)^{-1}\right)$，根据各集合 $\mathcal{M}^{(g,l)}$ 中元素之间的成对空间距离，对每个集合中的元素执行贪心约束 k-NN 分类（算法 2），以便将每个神经元参数向量 $\phi^{(g,l,i)}$ 分配到其在隐藏层中最可能的神经元位置。应谨慎选择超参数 k。通常，我们建议取较大的值，使分类尽可能基于更多邻居。所有实验中，我们均设定 $k = 1024$。

算法 2 执行贪心约束 k-神经网络分类的算法。

过程 GreedyConstrainedKNNClassification($\mathcal{M}^{(g,l)}$, $\mathcal{M}^{(*,l)}$, k)
  $\mathcal{R} \leftarrow \emptyset$
  对 范围 M_l 内的每个 i**执行**
    $\phi^{(g,l,i)} \in \mathcal{M}^{(g,\,l)}$
    $(p_i(c=1)..., p_i(c=M_l)) \leftarrow$ KNNClassification($\phi^{(g,l,i)}$, $\mathcal{M}^{(*,l)}$, k)
    $\mathcal{R} \leftarrow \mathcal{R} \cup \{(p_i(c=1)..., p_i(c=M_l))\}$
  结束循环
  $i \leftarrow 1$
  重复
    从 $\mathcal{R}$ 中找到概率最高的类别 c^* 对应的概率向量 p_j。
    $c^{(g,l,j)} \leftarrow$ 最大化$_c\, p_j(c=c^*)$
    $i \leftarrow i+1$
  直到 $i = M_l$
  对所有 j，设置 $p_j(c=c^*) = 0$。
  返回 $\{c^{(g,l,1)}, \ldots, c^{(g,l,M_l)}\}$
结束过程

接下来，根据分类结果对该层的神经元参数向量进行置换，即在 PERMUTE 操作中，将神经元重新索引为其所属类别（算法 3）。这实际上将参数子空间聚类为 M_l 个区域，这些区域隐式定义了约束并消除了与置换相关的对称性。k-NN 分类受到如下约束：集合 $\mathcal{M}^{(g,l)}$ 中任意一对神经元参数向量（$\phi^{(g,l,i)}$，$\phi^{(g,l,j)}$）都不能属于同一类别。具体做法是，优先将任意类别中概率最高的元素分配到相应类别，再按概率从高到低依次分配其余元素，直至所有向量均完成分配。由于贪心约束 k-NN 分类算法依据局部关系进行分类，因此需要对所有 $K-1$ 个隐藏层以及 G 个蒙特卡罗样本重复该过程 I 次迭代，以获得全局一致的神经元参数向量聚类。所有实验中，我们均设定 $I = 256$。

算法 3 隐藏层 l 的置换对称性消除。

过程 PermutationRemoval($\mathcal{M}^{(1,l)}$, ..., $\mathcal{M}^{(G,l)}$, $\mathcal{M}^{(*,l)}$, k, I)
  重复 I 次**执行**
    对 g 范围内的每个 G **执行**
      $\{c^{(g,l,1)}, \ldots, c^{(g,l,M_l)}\} \leftarrow$ GreedyConstrainedKNNClassification($\mathcal{M}^{(g,l)}$, $\mathcal{M}^{(*,l)}$, k)
    结束循环
    对 s 范围内的每个 S **执行**
      PERMUTE($\mathcal{M}^{(g,l)}$, $\{c^{(g,l,1)}, \ldots, c^{(g,l,M_l)}\}$)
    end for
  结束 for
结束过程

算法4概述了从多层感知机参数后验密度中采样的马尔可夫链中移除对称性的完整算法。算法4迭代调用算法1和算法3。

算法4 完整的几何约束对称性消除算法。

过程 GeometryRemoval($\{\boldsymbol{\theta}^1, \ldots, \boldsymbol{\theta}^G\}$, I, k, K_{β})
  对于每一层 l （反向） **执行**
    从马尔可夫链蒙特卡罗样本中构建神经元参数集 $M^{(1,l)}, \ldots, M^{(G,l)}$
$\{\boldsymbol{\theta}^1, \ldots, \boldsymbol{\theta}^G\}$
    $\mathcal{M}^{(*,l)} \leftarrow \mathcal{M}^{(1,l)} \cup \cdots \cup \mathcal{M}^{(G,l)}$
    TanHRemoval($\mathcal{M}^{(*,l)}$, K_{β}); 见算法 1
    PermutationRemoval($\mathcal{M}^{(1,l)}, \ldots, \mathcal{M}^{(G,l)}, \mathcal{M}^{(*,l)}, k, I$); 参见算法 3
  结束循环
结束过程。

C 实验设置

C.1 数据集

正弦数据集 $\mathcal{D}_S$ 取自 [9], 中提供的示例，而数据集 $\mathcal{D}_I$ 是来自 [24]的合成数据集。二维回归数据集 $\mathcal{D}_R$ 是我们按以下方式生成的另一个合成数据集:

$$
\begin{aligned}
p(x_1) &= \mathcal{U}(a = -2.0, b = 2.0) \\
p(x_2) &= \mathcal{U}(a = -2.0, b = 2.0) \\
f(x_1, x_2) &= x_1 \cdot \sin(x_1) + \cos(x_2) \\
p(y|x_1, x_2) &= \mathcal{N}(\mu = f(x_1, x_2), \sigma = 0.1) \\
p(\mathcal{D}) &= \prod_{i=1}^{256} p(y^{(i)}|x_1^{(i)}, x_2^{(i)}) p(x_1^{(i)}) p(x_2^{(i)})
\end{aligned}
$$

三个合成数据集 $\mathcal{D}_S$, $\mathcal{D}_I$ 和 $\mathcal{D}_R$ 均经过标准化处理，并用于所有实验，其可视化结果见图 4。我们将数据划分为 80% 的训练观测和 20% 的测试观测。除合成数据集外，我们还在表 2 中提供了基准测试所用 UCI 数据集的描述和参考文献。

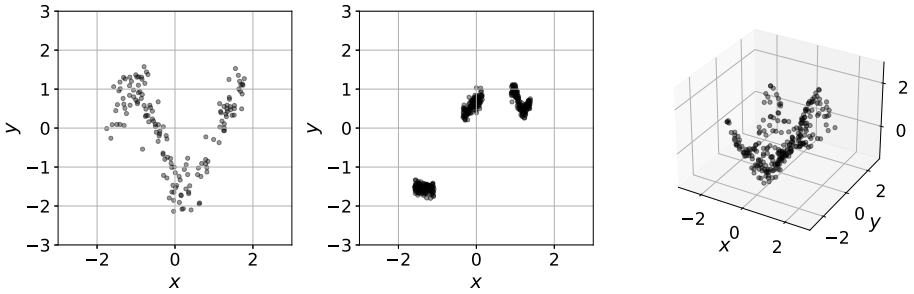

图 4. 合成数据集 $\mathcal{D}_S$ （左）、 $\mathcal{D}_I$ （中）和 $\mathcal{D}_R$ （右）的可视化。

表 2. 数据集特征及参考文献。

数据集	观测总数	训练集数量	测试集数量	特征	参考文献
正弦型 ($\mathcal{D}_S$)	150	120	30	1	改编自 [9]
Izmailov（$\mathcal{D}_i$）	400	320	80	1	改编自 [24]
二维回归 ($\mathcal{D}_R$)	256	304	52	2	–
翼型	1503	1202	301	5	[11]
混凝土	1030	824	206	8	[47]
糖尿病	442	353	89	10	[11]
能源	768	614	154	8	[44]
森林火灾	517	413	104	12	[8]
游艇	308	246	62	6	[11]

C.2 马尔可夫链蒙特卡罗

使用 NumPyro 实现的 No U-Turn Sampler（NUTS），并采用默认设置，对多层感知机参数进行 MCMC采样。具体而言，我们仅向 NUTS 提供模型和数据，将步长设为 1。0，并允许在预热阶段进行自适应。我们使用对角逆质量矩阵，该矩阵以单位矩阵初始化，并允许在预热阶段进行自适应。目标接受概率设为 0。8。

C.3 点估计

对于LA和差分进化估计，较小的架构 f_1 （一个包含三个神经元的隐藏层）和较大的架构 f_2 （三个隐藏层，每层包含16 个神经元）分别使用 RMSProp 优化器 [21] 训练500 和1000 个周期，使用恒定学习率 10^{-4} 且不进行权重衰减。在差分进化的情况下，我们用不同的随机

随机种子，且不使用数据自助法，遵循 [28] 原著中描述的方法。LA 样本直接从参数后验分布中抽取。由于数据集规模较小，我们采用全批量训练。我们的损失函数设为负对数后验分布。

$$\mathcal{L}(\boldsymbol{\theta}, \sigma) = -\log p(\boldsymbol{\theta}|\mathcal{D}) = \frac{1}{2\sigma^2}\sum_{i=1}^{N}(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i) - \boldsymbol{y}_i)^2 + N \cdot \log\sigma + \tfrac{1}{2}\boldsymbol{\theta}^\top\boldsymbol{\theta}. \quad (25)$$

损失函数需要针对网络参数 $\boldsymbol{\theta}$ 和干扰参数 σ进行最小化。LA 和差分进化（DE）估计所用的代码主要基于 PyTorch[35] 和 PyTorch Lightning [14] 库，以及 [9]开发的 Laplace 库。

D 方法

D.1 后验预测密度的 KL散度

两个连续随机变量的概率密度 p，q 之间的 KL散度定义为

$$D_{KL}(p||q) = \int_{\mathcal{X}} p(x)\log\frac{p(x)}{q(x)}\,\mathrm{d}x. \quad (26)$$

离散随机变量的概率密度 p，q 的相应表达式为

$$D_{KL}(p||q) = \sum_{x\in\mathcal{X}} p(x)\log\frac{p(x)}{q(x)}. \quad (27)$$

在第5.2节中，连续后验预测密度的KL散度是在特定输入x的值 $y \in [-3.0, 3.0]$ 规则网格上计算的。此外，该KL散度在输入值$x \in [-3.0, 3.0]$ 的规则网格上取平均，以获得最终值。

D.2 谱聚类

第 5.3 节中的谱聚类首先使用高斯相似度函数 $s(a,b) = \exp\big(-||a-b||^2 \cdot (2\sigma^2)^{-1}\big)$ 构建 4-NN 图，其中 $\sigma = 1$ 作为距离度量，并通过邻接矩阵 $\boldsymbol{A}$ 进行编码。然后，我们根据 $\boldsymbol{A}$ 计算归一化图拉普拉斯矩阵 $\boldsymbol{L}_{norm}$，其形式为

$$\boldsymbol{D}_{ij} = \begin{cases}\sum_{k=1}^{N}\boldsymbol{A}_{ik} & \text{if } i = j\\ 0 & \text{else}\end{cases}, \quad (28)$$

$$\boldsymbol{L}_{norm} = \boldsymbol{D}^{-1/2}\boldsymbol{L}\boldsymbol{D}^{1/2}, \quad (29)$$

其中，$\boldsymbol{D}_{ij}$ 是图的度矩阵 $\boldsymbol{D}$ 中的第（i,j）个元素。随后，确定 $\boldsymbol{L}_{norm}$ 的特征值谱，并对 $K = 3$执行 K-means 聚类。

E 补充结果

E.1 我们的方法与 LA 的性能比较

我们根据预测性能，将基于 MCMC采样计算后验预测分布的方法与 LA 进行比较。对于 MCMC采样，我们生成了 1274 条链，这是推导出的链数上界。表 3 总结了这一比较。

表 3. 测试集上的平均逐点对数预测密度（LPPD）值（越大越好；括号内为一个标准误）。每个数据集和网络中性能最高的结果以粗体标出。

	较小的网络 f_1 马尔可夫链蒙特卡洛(我们的方法)	马尔可夫链蒙特卡洛 (s.c.)	LA	**更大的网络** f_2 马尔可夫链蒙特卡洛 (我们的方法)	马尔可夫链蒙特卡洛 (s. c.)	LA
$\mathcal{D}_S$	**-0.53**(± 0.09)	-0.56 (± 0.11)	-0.57 (± 0.10)	**-0.59** (± 0.12)	**-0.59** (± 0.12)	-2.42 (± 0.01)
$\mathcal{D}_I$	**0.79**(± 0.06)	0.65 (± 0.07)	0.53 (± 0.07)	**0.91** (± 0.09)	**0.91** (± 0.09)	-1.81 (± 0.01)
$\mathcal{D}_R$	0.64 (± 0.10)	**0.75** (± 0.11)	-27.39 (± 3.65)	**0.95** (± 0.08)	**0.95** (± 0.08)	-2.33 (± 0.00)
翼型	**-0.74**(± 0.04)	-0.80 (± 0.05)	-1.78 (± 0.13)	**0.92**(± 0.05)	0.72 (± 0.10)	-3.57 (± 0.18)
混凝土	**-0.41** (± 0.05)	-0.44 (± 0.06)	-14.49 (± 1.02)	**0.26**(± 0.07)	0.25 (± 0.07)	-4.36 (± 0.47)
糖尿病	**-1.20** (± 0.07)	**-1.20** (± 0.07)	-1.46 (± 0.09)	**-1.18** (± 0.08)	-1.22 (± 0.09)	-2.61 (± 0.00)
能源	**0.92**(± 0.04)	0.69 (± 0.12)	-31.74 (± 1.88)	2.07 (± 0.46)	**2.38** (± 0.11)	-1.39 (± 0.06)
ForestF	**-1.37** (± 0.07)	**-1.37** (± 0.07)	-2.39 (± 0.16)	**-1.43** (± 0.45)	-1.69 (± 0.49)	-2.80 (± 0.00)
游艇	**1.90**(± 0.16)	1.29 (± 0.56)	-5.60 (± 1.51)	**3.31**(± 0.21)	0.15 (± 0.09)	-2.69 (± 0.00)

E.2 可解释性示例近似

第5.3节可解释性示例中，BNN 后验分布剩余的三个模式，在聚类后可使用 LA 混合（MoLA； [13]）进行解析近似。图5按分量及混合形式展示了近似的函数空间。

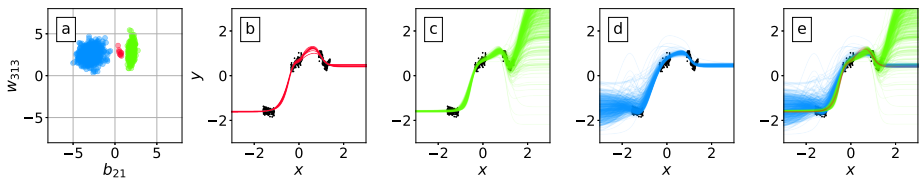

图5。 从左到右：经 MoLA 近似聚类后，BNN 后验分布中剩余的三个模式，在两个权重的二元边缘空间中可视化（a）；基于三个高斯分量得到的功能多样化网络参数状态（b-d）；将 b-d 组合后，三个高斯分布的样本构成的函数空间（e）。

E.3 参数后验密度剖面

图6展示了在第5.3节中研究的BNNf_1 在数据集 $\mathcal{D}_S$ 上的参数后验密度成对剖面，分别对应应用对称性消除算法之前（红色）和之后（绿色）的情况。所得的近似参数后验密度呈单峰分布，与原始参数后验密度相比简单得多，但在函数层面上二者是相同的。

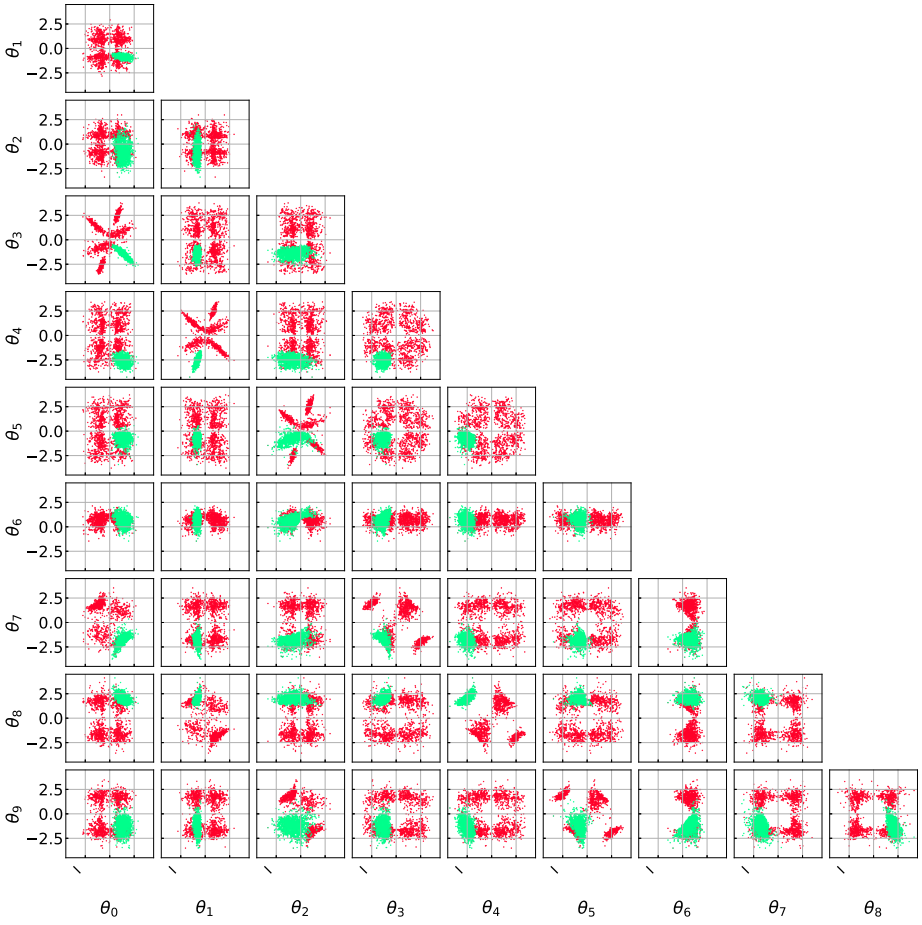

图 6. BNN f_1 的参数后验密度成对剖面可视化。通过马尔可夫链蒙特卡洛采样得到的参数后验密度（红色）呈现对称性，而变换后的近似参数后验密度（绿色）为单峰分布。