arXiv:2603.02437v1 [stat.CO] 2 Mar 2026

利用稀疏性提升分层贝叶斯模型的无转向采样效率

Cole C. Monnahan
资源生态学与渔业管理 阿拉斯加渔业科学中心 美国国家海洋和大气管理局渔业局 西雅图，华盛顿州，美国
cole.monnahan@noaa.gov

Kasper Kristensen
DTU Aqua 丹麦技术大学 哥本哈根，丹麦
kaskr@dtu.edu

James T. Thorson
资源生态与渔业管理 阿拉斯加渔业科学中心 美国国家海洋和大气管理局渔业局 西雅图，华盛顿州，美国
james.thorson@noaa.gov

Bob Carpenter
计算数学中心 Flatiron研究所 纽约州纽约市，美国
bcarpenter@flatironinstitute.org

2026年3月4日

摘要

分析人员通常使用贝叶斯分层模型来理解自然过程。无转向采样器（NUTS）是用于对高维连续可微模型进行采样的最广泛使用的算法。但高相关性会拖慢 NUTS，尤其是在高维情况下，从而限制应用分析的复杂程度。本文介绍稀疏NUTS（SNUTS），它使用稀疏精度矩阵对后验分布进行预条件（去相关和去尺度化）Q。我们使用模板模型构建器（TMB），从边缘后验分布的拉普拉斯近似众数高效计算 Q，然后通过贝叶斯软件Stan将预条件化后验分布传递给NUTS进行采样。我们将SNUTS应用于17个不同的案例研究，结果表明，使用 Q 进行预条件，相较于Stan业界标准的对角或稠密预条件器，收敛速度快一个到两个数量级。SNUTS的表现也优于使用Pathfinder变分推断所估计协方差矩阵的逆进行预条件。对于漏斗、宽尾或多峰中出现曲率剧烈变化的模型，SNUTS无法提高采样效率。当模型具有高维、稀疏和高相关性（这些特征在应用统计学中十分普遍）时，SNUTS最具优势，且可扩展至超过 10^4 个参数。SNUTS的开源实现已收录于R软件包 `SparseNUTS`中。

关键词 贝叶斯推　　断 · 层级模型 · 马尔可夫链蒙特卡洛 · 无转向采样器 · 稀疏掩　　s
矩阵 · 稀疏去相关　elation ·模板模型构建器

1 引言

层级模型因其能够灵活地对多层次过程进行建模，常用于研究各种过程（Cressie 等人，2009 年）。其中许多是贝叶斯层次分析，通常使用马尔可夫链蒙特卡洛（MCMC）算法生成后验样本，以进行推断（Gelman 等人，2014 年）。无转向采样器（NUTS；Hoffman 和 Gelman (2014 年)）是一种广泛用于复杂层级模型的 MCMC 算法（例如，Monnahan 等人 (2017年)），但其效率（单位时间内生成的有效后验样本数）可能受到各种棘手后验几何结构的影响，进而因运行时间过长且缺乏稳健的收敛性而限制其实际应用（Betancourt 和 Girolami，2015 年）。因此，开发能够提高 NUTS 采样效率并缩短运行时间的算法和策略，对各领域的应用研究都至关重要。

一个挑战是参数之间存在较大的（线性）相关性，且边缘尺度差异明显（图1）。这可用条件数因子 $\kappa = (\sum_{d=1}^{D}(\frac{\max(\lambda)}{\lambda_d})^4)^{1/4}$ 表示，其中 λ 是在当前位置评估的对数密度的逆负黑塞矩阵（即局部协方差）的特征值序列（Langmore 等，2019年）。条件因子可以衡量 NUTS抽样的难度：其值越接近 1，抽样越容易；较强的相关性和差异显著的尺度都会导致较大的 κ，使采样效率降低。[1] 全局协方差可用于对模型进行预条件处理（即去相关和去尺度化），从而提高采样效率，但通常在采样前未知；而且在高维情况下，矩阵代数的计算成本可能令人难以承受（Betancourt，2017年；Neal，2011年；Bales 等，2019年）。在这类情况下，可采用自动微分变分推断（ADVI；Kucukelbir 等（2017年））或 Pathfinder（Zhang 等，2022年）等近似算法，避免完整的 NUTS抽样；但由于算法不稳定且难以验证拟合结果，这些方法尚未得到广泛应用。另一种选择是在预热阶段估计全局协方差，但即便采用低秩近似，这仍是一项挑战（Bales 等，2019年）。还有一种方法是使用 NUTS，通过嵌入式拉普拉斯近似方法（ELA；例如，Margossian 等人（2020年））从边缘后验分布中采样。该方法前景可期，但尚未得到广泛

经过测试。因此，Stan 和 PyMC 等主流平台使用的默认设置是进行较长时间的 NUTS 预热阶段，以估计并调整参数尺度，但不调整相关性（Carpenter 等人，2017年；Stan 开发团队，2025年；Abril-Pla 等人，2023年）。因此，若能在 NUTS 抽样之前，以可扩展的方式调整相关性和尺度，将会非常有用。

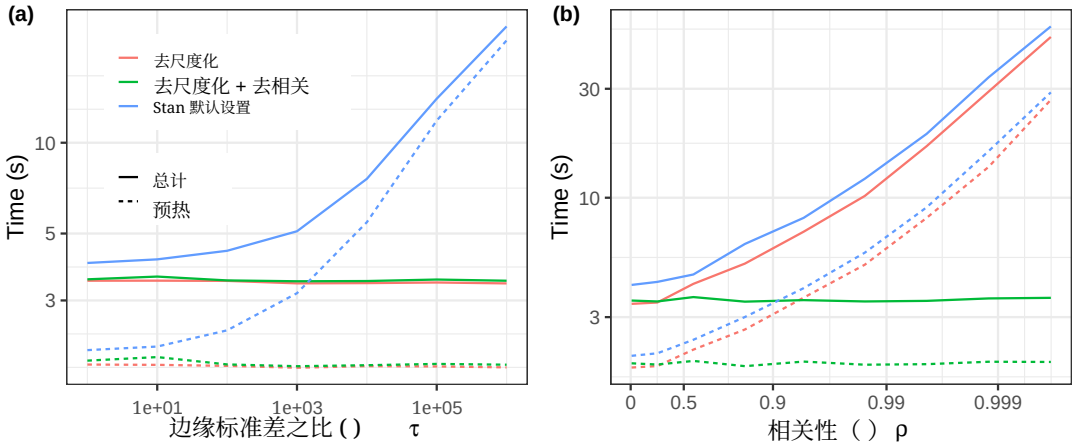


图*1.* 预热时间比较。简单示意边际标准差之比（a）和相关性（b）对简单二元正态模型 $\mathbf{X} \sim \mathcal{N}(\mathbf{0}, \Sigma)$的预热时间（虚线）和 *NUTS* 总运行时间（实线）的影响。在（a）中，$\Sigma = \left(\begin{smallmatrix} 1 & 0 \\ 0 & \tau \end{smallmatrix}\right)$ 和 τ变化；在（b）中，$\Sigma = \left(\begin{smallmatrix} 1 & \rho \\ \rho & 1 \end{smallmatrix}\right)$ 和 ρ 变化。*NUTS* 运行了三次（不同颜色），首先采用 *Stan* 默认设置，随后仅进行去尺度化，或在抽样前对后验分布进行去尺度化和去相关。所有情况下，均先进行 *1000* 次预热迭代，再进行 *1000* 次抽样迭代，并采用 *Stan* 默认设置。预热期内启用了对角质量矩阵自适应。

近年来，层级模型最大边际似然估计（MMLE）方面的创新扩大了可构建模型的规模和复杂度。尤其是，Template ModelBuilder(`TMB`；Kristensen 等(2016年)) 被用于多种常见的层级模型，例如广义线性混合模型（GLMMs；Brooks 等人(2017年) 以及时空回归和物种分布模型（Thorson 等人，2025a；Anderson 等人，2025年；Kellner 等，

[1]例如，标准差为 1、相关性为 ρ 的二维多元正态分布，其特征值为 $1-\rho$ 和 $1+\rho$，两者的比值没有上界。类似地，当相关性为 0、尺度为 σ_1、σ_2 时，特征值为 σ_1^2 和 σ_2^2，两者的比值同样没有上界。

2023年），以及状态空间种群动态、状态空间动物运动、标记重捕和隐马尔可夫模型、时间序列和系统发育结构方程模型等小众应用（表S1）。这些 TMB 模型通常被构建为具有稀疏精度（即逆协方差）矩阵的多元正态随机效应，例如高斯马尔可夫随机场（GMRF；Rue 和 Held (2005年)）以及在空间模型中广泛使用的随机偏微分方程（SPDE）近似（Lindgren 等人，2011年）。 TMB日益普及，得益于它能够自动检测并利用稀疏性来高效进行最大边际似然估计，同时还能计算稀疏精度矩阵（Q），以表示参数的近似精度。 TMB利用 Q在各种层级模型中进行推断的成功表明，它可用于处理全局相关性和变量尺度问题，而这些问题给应用贝叶斯分析带来了挑战。

在此，我们将 TMB现有的计算工具与 Stan 的 NUTS 算法相结合，开发 SNUTS 算法，同时实现一种新颖的稀疏预条件器，以提高 NUTS 的采样效率。我们将 TMB 与 Pathfinder 在生成近似样本方面进行基准测试，随后针对两个维度逐渐增加、最高达到 10^4 的模拟模型，将 SNUTS 效率与使用 Stan 默认设置的 NUTS 进行比较。接着，我们在 17 个复杂程度、维度和稀疏程度各异的案例研究中，对 SNUTS 和嵌入式拉普拉斯近似 SNUTS 进行基准测试。最后，我们在一个新的 R 软件包 `SparseNUTS` 中实现 SNUTS，并展示其在 `TMB` 和 `RTMB` (Kristensen，2025年) 模型中的应用。

2 方法

首先，我们回顾参数相关性为何会降低 NUTS 的采样效率。其次，我们提出一种利用稀疏矩阵代数对后验分布进行预条件处理的新方法，并回顾 TMB 如何估计 Q。第三，我们介绍一个新的 R 软件包 `SparseNUTS`，它为贝叶斯分析提供了用户友好的工作流程，其中包括自动选择预条件器和所需预热长度的启发式方法。最后，我们使用 `SparseNUTS` 评估 SNUTS 比标准 NUTS 的效率提升幅度。

2.1 NUTS中的稀疏预条件

2.1.1 哈密顿蒙特卡洛概述

为理解相关性为何会降低 NUTS 采样效率，我们首先回顾其基础算法。NUTS 属于哈密顿蒙特卡洛（HMC）这一类 MCMC 算法，它们利用哈密顿动力学高效生成提议（Neal，2011年；Betancourt，2017年）。哈密顿动力学描述一个虚构的冰球如何随时间在无摩擦表面上滑动，其运动由哈密顿函数 $H(p,q) = U(q) + K(p)$ 控制。 $K(p)$ 是动量向量 p 上的动能函数，通常为 $K(p) = p^\top M^{-1}p/2$，其中 M 是对称正定的“质量矩阵”。 $U(q)$ 是位置向量 q 的势能。在 HMC 中，使用 $U(q) = -f(q)$，其中 $f(q)$ 是未归一化对数后验密度函数， q 是参数向量。

哈密顿系统的解通常通过蛙跳算法用离散步近似，该算法交替更新 p 和 q （Neal，2011年）。单个蛙跳步骤包含三次更新，步长为 $\epsilon > 0$，并以位置和动量M 作为起点（q_t， p_t），

$$\begin{aligned} p_{t+\epsilon/2} &= p_t + \frac{\epsilon}{2}\nabla f(q) \\ q_{t+\epsilon} &= q_t + \epsilon M^{-1} p_{t+\epsilon/2} \\ p_{t+\epsilon} &= p_{t+\epsilon/2} + \frac{\epsilon}{2}\nabla f(q). \end{aligned} \tag{1}$$

连续执行 L 个蛙跳步骤，即可生成一条随时间演进的近似 HMC“轨迹”，并可从中选取一个点来生成 MCMC 样本（Betancourt，2016年）。NUTS 算法无需指定 ϵ，因为它会自适应地确定该值；它也无需指定 L，该值通过在轨迹出现“U 形转向”时停止来动态确定（Hoffman 和 Gelman，2014 年）。其目标是在后验分布中进行有方向的移动。生成期望平方跳跃距离较大的点，可以避免扩散式随机游走行为，从而直接提高有效样本量，使 NUTS 能够高效生成后验样本（Neal，2011年；Hoffman 和 Gelman，2014 年）。重要的是，NUTS 需要计算梯度函数 $\nabla f(q)$ 来生成轨迹（方程（1））， 这构成了计算瓶颈，在高度优化的实现中会消耗超过 90% 的计算预算。因此，任何能够减少轨迹移动相似距离所需计算次数的方法，都能提升 NUTS 的性能。

存在相关性时，由于振荡行为、蛙跳步数上限以及过早出现的 U 形转弯，NUTS 轨迹很少能够遍历整个后验分布。轨迹通常需要进行许多次梯度评估

（图 S1），这会降低采样速度。只要能够对后验分布进行预条件处理，使其条件数更接近 1，就能以较低成本延长 NUTS 轨迹，并大幅提高采样效率。理想情况下，尺度为 1 且不存在相关性的后验分布，其条件数为 1。如果已知后验分布的全局协方差的先验信息，且该协方差在后验分布中的变化不太大，就可以利用它对后验分布进行预条件处理。具体而言，可以对参数空间进行线性变换，使其变为 $q' = L^{-1}q$ 其中 $L = \text{chol}(\Sigma)$ 是下三角 Cholesky 分解，使得 $\text{cov}(q) = \Sigma = LL^\top$；同时，令 M 为单位矩阵，因此动能函数为 $K(p) = p^\top p/2$ (Neal, 2011年）。这是因为，仿照 Gelman 等人(2014 年）稍微放宽随机变量的记法，我们有

$$\text{cov}(q') = \text{cov}(L^{-1}q) = L^{-1}\text{cov}(q)L^{-\top} = L^{-1}LL^\top L^{-\top} = \text{I},$$

其条件数为1，正如预期。只要后验曲率的变化不大，预条件就能提高效率；对于近似多元正态分布的分布而言，情况正是如此，因为正态分布具有恒定的曲率。

2.1.2 使用稠密矩阵进行预条件

为了使用 NUTS 从 q' 中采样，需要对数密度函数 $f_{q'}(q')$ 和梯度函数 $g_{q'}(q')$ 来生成轨迹。它们可推导如下：

$$f_{q'}(q') = f_q(Lq') \tag{2}$$
$$g_{q'}(q') = \nabla f_{q'}(q') = g_q(Lq')L, \tag{3}$$

变换产生的常数变量变换调整因子 $\left|\det(L^{-1})\right|$ 可以省略（参见 Neal (2011) 第 4.1 节）。Σ 通常是稠密矩阵，因此 L 也会是稠密矩阵；计算维度为 D 时 $\mathcal{O}(D^2)$ 的乘积 Lq'，计算成本为平方级。这种随维度呈平方增长的计算成本促使我们将 Σ 的非对角元素设为零，使 L 成为对角矩阵，并使矩阵向量乘积 Lq' 化为逐元素向量乘积，其计算规模为 $\mathcal{O}(D)$。我们将 Σ 的这两种情况分别称为“稠密”和“对角”，沿用 Stan 使用的术语（Stan 开发团队，2025年）。对角矩阵虽然计算速度更快，但只能考虑变量尺度，而无法考虑相关性，这可能导致 NUTS 轨迹很长、采样效率低下（图1；Neal (2011)）。

2.1.3 利用稀疏矩阵进行高效预条件

我们可以用类似的方式进行预条件，但利用 Q 的稀疏性，令 $q' = L^\top q$，其中 $L =$ 为chol(Q)且$Q = LL^\top$。重要的是，L 的稀疏程度既取决于 Q的稀疏性，也取决于其排序，因为 L 中可能的非零元素数量大于 Q 下三角中的非零元素数量（参见Rue 和 Held（2005年）第2.4节）。通过对 Q 用置换矩阵 P重新排序，可以增加 L 的稀疏程度；左乘该矩阵会重排各行，右乘其转置会重排各列，即$Q_P = PQP^\top$。这种置换可以显著提高 L的稀疏性，并降低下文中稀疏矩阵乘法的计算成本（Rue 和 Held（2005年）；图 2）。

为实现这种稀疏预条件，我们首先使用R函数 `Cholesky`，该函数来自 `Matrix` R软件包Bates等人（2025年），以确定 P。然后令 $L =$chol（Q_P），从而 $Q_P = LL^\top$，并定义 $q' = L^\top Pq$，同时再次忽略变量变换常数，将稀疏预条件器定义为

$$\begin{aligned} f_{q'}(q') &= f_q((L^\top P)^{-1}q') \\ g_{q'}(q') &= \nabla f_{q'}(q') = g_q((L^\top P)^{-1}q')(L^\top P)^{-1}. \end{aligned} \tag{4}$$

要计算 cov(q')，首先注意到 $Q_P^{-1} = (LL^\top)^{-1} = L^{-\top}L^{-1}$，因此

$$\text{cov}(q) = Q^{-1} = (P^{-1}Q_P P^{-\top})^{-1} = P^\top L^{-\top}L^{-1}P,$$

代入预条件变量后得到

$$\text{cov}(q') = \text{cov}(L^\top Pq) = L^\top PQ^{-1}(L^\top P)^\top = L^\top P(P^\top L^{-\top}L^{-1}P)P^\top L = \text{I},$$

因为置换矩阵满足 $PP^\top =$I。因此，q'具有所期望的条件因子1。这些稀疏矩阵计算的高效实现是在R中使用 `Matrix` 包的 `solve` 函数完成的，示例代码见附录 C。

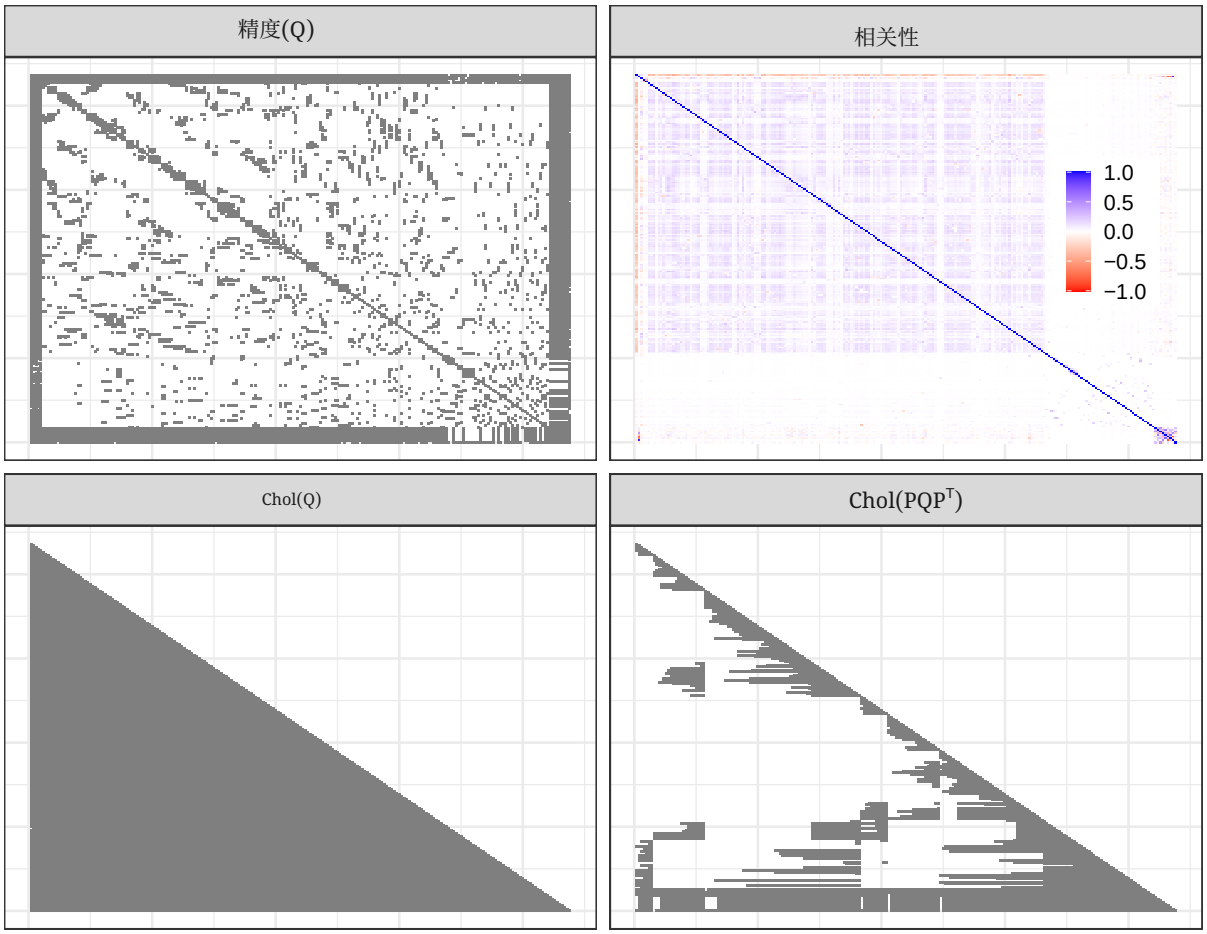


图 *2.* ***Q*** 的可视化。*TMB* 为 *sdmTMB* 空间模型自动检测到的精度矩阵 Q 的稀疏模式（左上）、由 $\Sigma = Q^{-1}$ 推导出的相关矩阵（右上）、Q 本身的下三角 *Cholesky* 分解（左下），以及通过矩阵 P 对 Q 进行置换以最大化稀疏性（即 $PQP^{\top}$，右下）后的 *Cholesky* 分解。非零元素以灰色显示，零元素以白色显示（以突出稀疏性），相关矩阵除外，它按相关值着色。Q 有 *81%* 的非对角零元素，而相关矩阵完全稠密，体现了 Q 与 Q^{-1} 之间稀疏性的差异。在此情况下，Q 中的稀疏模式会导致 *Cholesky* 分解几乎稠密（左下），但先对 Q 进行置换即可恢复稀疏性（右下）

公式（4）的计算和内存成本取决于（$L^{\top}P$）$^{-1}$的维度与稀疏程度，因此我们在NUTS之外，对空间SPDE 模型（该模型在 Q中具有高稀疏性）上的对角、稠密和稀疏矩阵，随维度增加对梯度计算进行了基准测试。稀疏去相关在较低维度下比稠密去相关更慢，但对于具有 $d > 100$ 的模型更快（图3），总体而言，$g_{q'}$（q'）的计算成本低于 g_q（q），这意味着稀疏预条件为 $d > 10^4$增加的计算成本可以忽略不计，至少对该特定模型如此。在2.3.4 中，我们将更详细地评估NUTS 采样效率方面的表现。

接下来，我们将探讨如何在 NUTS 抽样之前求得稀疏精度矩阵 Q，以便将稀疏预条件与稠密和对角版本进行比较。就我们所知，`TMB` 是唯一能够针对任意统计模型计算具有稀疏性的 Q 的软件平台，因此本文将重点介绍它。`TMB` 本身及其采用的频率学派技术，对一些读者而言可能并不熟悉，因此我们先简要介绍，然后详细说明如何在贝叶斯背景下计算和使用 Q。

2.1.4 模板模型构建器概述

模板模型构建器（`TMB`）是一个统计软件包，用于对高维非线性层级模型执行最大边际似然估计（Kristensen 等人，2016年）。用户用模板化的 C++ 编写负对数似然，随后将其

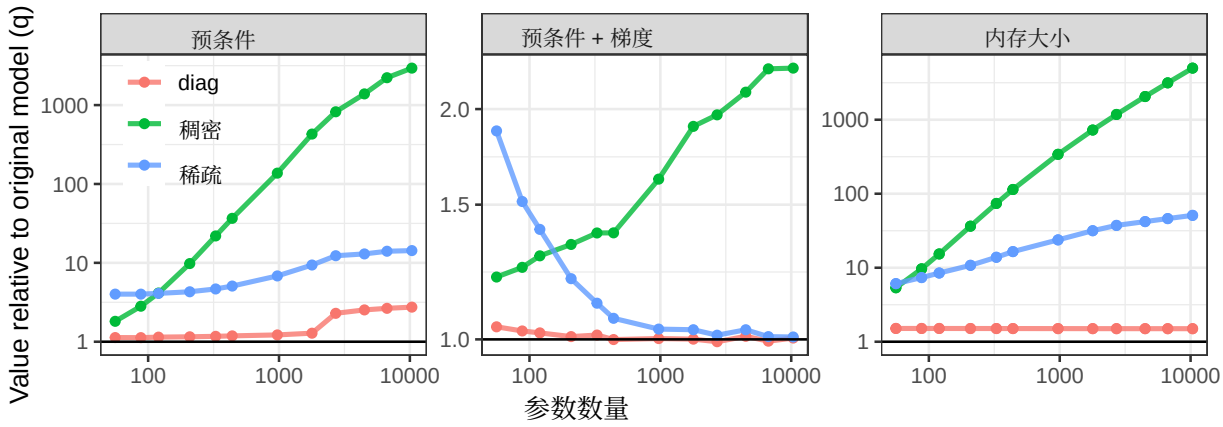


图*3.* 基准测试结果。对观测值数量不断增加、空间随机效应稀疏性按设计设为较高水平（通常为 $> 99\%$）的模拟泊松随机偏微分方程模型进行的基准测试结果。图中展示：左）仅对 $g_q \to g_{q'}$ 进行变换的计算成本（即评估 g_q的成本可忽略不计）；中）变换及梯度 $g_q(q)$的计算成本；右）变换所用的内存。图中以原始模型的值为基准，展示了三种变换（不同颜色）。

编译并通过 R 接口链接。或者，用户也可以直接用 R 编写模型，经由 `RTMB` 通向`TMB` 的接口（Kristensen, 2025年）。用户还会声明哪些参数是“随机效应”（即通常假定具有高斯先验的潜在状态），从而使参数空间 $x = (u, \theta)$ 被划分为随机（u）效应和固定（θ）效应，并且 $f_x(u, \theta)$ 是数据与随机效应的负对数似然，而 $g_x(u, \theta) = \nabla f_x(u, \theta)$ 是梯度函数。

固定效应 θ 的最大边际似然估计通过最大化边际似然$L^*(\theta) = \sqrt{2\pi}^n \det(H(\theta))^{-1/2} \exp(-f_x(\hat{u}, \theta))$的拉普拉斯近似得到（Skaug 和 Fournier, 2006）。其中，$\hat{u}(\theta) = \arg\min_u f_x(u, \theta)$是在固定 θ 时使目标函数达到最小值的条件众数，而 $H(\theta) = f''_{uu}(\hat{u}(\theta), \theta)$是 $f_x(u, \theta)$关于u 的海森矩阵，并在$\hat{u}(\theta)$处求值。 `TMB` 通过自动微分计算 $f_\theta(\theta) = -\log L^*(\theta)$及其梯度 $g_\theta(\theta)$，并利用 u的稀疏性。 θ 称为边际众数，$\hat{u}(\theta)$是 u的经验贝叶斯估计值，而$\hat{x} = (\hat{u}(\theta), \theta)$是条件众数。

2.1.5 计算稀疏近似精度矩阵

为简化记号，我们假设 u 是参数向量的第一个分量。那么， Q 是在 $\hat{x}$ 处计算的二阶导数矩阵，其分块形式如下：

$$Q = \begin{bmatrix} H_{uu} & H_{u\theta} \\ H_{\theta u} & H_{\theta\theta} \end{bmatrix} \tag{5}$$

其中 H_{uu} 是 u 在 $\hat{x}$处的稀疏海森矩阵， $H_{u\theta}$ 而 $H_{\theta u} = H_{u\theta}^\top$ 是海森矩阵中固定效应和随机效应对应的分块，在 $\hat{x}$处计算。最后， $H_{\theta\theta} = H_{\theta u} H_{uu}^{-1} H_{u\theta} + H_{\hat{\theta}\hat{\theta}}$ 其中 $H_{\hat{\theta}\hat{\theta}}$ 是边际似然 f_θ 在边际众数 $\hat{}$处的海森矩阵 θ （Kass 和 Steffey, 1989 年）。 `TMB` 在整个过程中利用稀疏性和自动微分，最终得到的是来自 `Matrix` 包的稀疏矩阵类型。

当 $Q_{ij} = 0$ 满足以下条件时，该元素为零：在给定所有其他参数的固定值后， i^{th} 与 j^{th} 参数条件独立。这一关系通过表示模型计算的图，使用“subgraph”算法自动检测（Bell 和 Kristensen, 2021）。因此， Q的稀疏模式反映了模型的条件独立结构（例如图 2）。在频率学派框架中， Q 基于给定数据时 u 分布的多元正态近似，提供均方误差估计，而不是基于对数据重复采样得出的方差估计； u （Zheng 和 Cadigan, 2021）。在贝叶斯背景下，我们假设 Q 是后验精度的一个近似。

2.2 稀疏 NUTS 算法

`TMB` 可用于贝叶斯背景：将先验和参数变换的雅可比修正纳入目标函数。这会将未归一化的负对数似然转换为未归一化的对数后验，

$f_q(q) = -f_x(u,\theta)$ 和 $g_q(q) = -g_x(u,\theta)$，同时保持贝叶斯参数向量 $q = (u,\theta)$ 中固定效应（θ）和随机效应（u）的划分。$\hat{}q = (u(\theta)\ \theta)$ 因此是条件后验众数（如上所定义），并假设 Q 可将后验分布近似为 $q \sim \mathcal{N}(\hat{}q, Q^{-1})$。同样，广义 delta 方法可用于求得参数函数的近似后验分布（Kristensen 等，2016），附录 C 中提供了一个完整示例。

我们提出 SNUTS 的以下步骤：

1. 写出未归一化对数后验密度 $f_q(q)$，指定随机效应 u，从而划分$q = (u,\theta)$。2. 对边缘后验的拉普拉斯近似进行数值最大化，并在众数 $\hat{}q = (\hat{}u(\theta)\ \theta)$ 处计算层级模型的 Q（见公式 (5)）$\hat{}\hat{}$，或不含随机效应的模型的 Σ。3. 使用 Q 和 Σ 自动选择预条件器及预热长度，并通过公式 (2) 或公式 (4) 得到 $f_{q'}$ 和 $g_{q'}$。4. 将 $f_{q'}$ 和 $g_{q'}$ 传入 NUTS 算法，以生成 q' 的后验样本。此处我们通过 `StanEstimators` 接口使用 Stan 的 NUTS 算法（Johnson，2025）。5. 对参数空间进行重新条件化（$q' \to q$），通过诊断评估 MCMC 收敛，并像标准贝叶斯分析一样，使用后验样本对参数和派生量进行推断。

我们在开源 R 软件包 `SparseNUTS` 的 `sample_snuts` 函数中实现了 SNUTS。该软件包可从 XXXX 获取，并提供用户友好的接口，包括自动选择预条件器和预热时长（更多详情和一个完整示例请参见附录 A）。

2.3 评估

2.3.1 案例研究

我们在 17 个具有不同结构、数据量和维度的模型上评估 SNUTS（表 1）。表 S2 提供了更多详情、所做的任何修改以及原始引文。

2.3.2 量化 Q的准确性

一个关键主张是 $\mathcal{N}(\hat{}q,\ Q)$ 能够很好地近似许多层级模型的后验几何结构。我们通过将其与 Pathfinder 进行比较来检验这一主张；Pathfinder 能够高效地生成后验分布的近似样本，其开发初衷主要是用于初始化 NUTS 链（Zhang 等人，2022年）。研究发现，该方法在多种模型上优于 ADVI 算法，因此我们将其用作参考方法（Zhang 等人，2022年）。我们将 Pathfinder 的表现与 TMB 的多元正态样本方法进行比较 $q^{sim} \sim \mathcal{N}(\hat{}q,\ Q^{-1})$ 并将其称为 “ 精度抽样 ” 方法。我们使用 1-Wasserstein 距离（ McCann，1995 ；Craig，2016 年）衡量这两种方法的准确性，距离越大表示后验近似越差。我们还比较了计算成本（运行时间），具体细节见附录 A。

2.3.3 比较算法之间的采样效率

MCMC算法的采样效率是生成一个有效后验样本所需的时间，也是比较不同算法或软件平台性能的常用度量（例如，Monnahan等人（2017年））。有效样本量（ESS）定义为所有模型参数和对数后验密度中的最小值，并由 `ess_bulk` 函数计算，该函数来自 `posterior` R软件包（Bürkner等人，2025年），其实现了Vehtari等人（2021年）中的方法。运行时间不包括任何模型编译，而是各并行链的预热时间与采样时间之和，以及对 Q 进行优化、计算和求逆所需的时间，这属于自动选择算法的一部分（即SNUTS开销）。

为了在下文相同的框架中（即 `sample_snuts`）将 SNUTS 与 NUTS 进行基准比较，我们对每项分析运行了两次。第一次使用 Stan 默认设置，包括 1000 次预热迭代（期间会自适应对角质量矩阵）和 1000 次采样迭代，以下称为“NUTS”。第二次使用 SNUTS 算法并自动选择预条件器（即对角、稠密或稀疏），进行 150 次预热采样，禁用质量矩阵自适应，并进行 1000 次采样迭代。所有链均从 $\mathcal{N}(\hat{}q, Q^{-1})$ 中随机抽样的值初始化。我们进行了三次包含四条链的独立分析，以量化不同初始值和 NUTS 随机性所导致的结果差异。

Name	模型类型	参数数量	稀疏性	最大相关性	标准差比	预条件器
因果	因果时间序列	264 (232)	78.7	0.96, 0.97	101.8, 111.0	稀疏
钻石	线性混合模型	26 (24)	0	0.89, 0.89	189.6, 189.7	稠密
dlm	动态线性模型	118 (114)	91.7	0.67, 0.67	9.2, 9.5	稠密
gp_pois_回归	泊松高斯过程	13（11）	0	0.972，0.988	8.1，8.4	稠密
irt_2pl	项目反应理论	148 (144)	56.3	0.84, 0.68	8.4, 13.2	稠密
基尔皮斯耶尔维	线性回归	3 (0)	0	0.9999887, 0.9999884	3982.5, 3982.5	稠密
lynx	龙格-库塔常微分方程求解器	50 (42)	63	0.89, 0.88	19.5, 19.2	稠密
海燕	隐马尔可夫模型	274 (266)	94.2	0.97, 0.97	115.8, 114.3	稠密
狭鳕	非线性种群动力学	351 (55)	28.1	0.9999821, 0.9999817	66.3, 65.2	稀疏
氡	线性混合模型	389 (386)	98.5	0.27, 0.12	79.4, 75.7	diag
蝾螈	零膨胀负二项分布	39 (23)	44.5	0.963, 0.958	7.6, 6.9	稠密
sam	非线性种群动力学	1226 (1209)	96.6	0.91, 0.91	11.9, 12.0	稀疏
学校	线性混合模型	10 (8)	66.2	0.26, 0.29	6.1，6.45	diag
sdmTMB	空间 Tweedie 广义线性混合模型	219（216）	81.3	0.94，0.92	451.0，352.2	稠密
燕子	状态空间标记重捕法	177 (148)	75.4	0.94, 0.96	37.3, 37.5	稠密
wham	非线性种群动力学	385（322）	66.5	0.93、0.91	26.5、9.9	稀疏
wildf	二项广义线性混合模型	1101 (1092)	98.4	0.96, 0.95	15.6, 11.8	稀疏

表 ***1.*** 案例研究。参数总数及其中随机效应的数量分别列出，随机效应数量放在括号中。 Q 中的稀疏性是 Q 下三角中已知为零的元素所占百分比，不包括对角线。最大相关性是后验分布中所有参数之间成对相关性的绝对值最大值，其后是根据 Q估计的最大值。标准差比是后验分布中的边际参数标准差与 Q中的边际参数标准差之比。预条件器一栏显示了自动选定的方法。

2.3.4 更高维度下的性能

我们在两种不同的模型上评估了 SNUTS 随维度增加时的性能。首先，我们复现了 Brooks 等人 (2017年) 的分析，其中数据由负二项广义线性混合模型模拟得出（见表 *1* 中的蝾螈 ），随着地点数量增加，随机效应也随之增加。其次，我们模拟了一个参数化为 SPDE 模型的泊松空间广义线性混合模型，如上文图 3所示，并逐步增加观测值和随机效应的数量。

2.3.5 稀疏NUTS性能基准测试

接下来，我们在案例研究（表 1）中评估了 SNUTS 的性能。首先，通过比较预条件前后的梯度计算成本，以及相对于完整 SNUTS 运行估算 Q 所需的时间，量化 SNUTS 的额外开销。随后，我们在每个案例研究中运行 NUTS 和 SNUTS，同样使用三次分析，每次分析包含四条链。

2.3.6 嵌入式拉普拉斯近似

使用 NUTS 对边缘后验分布（f_θ）进行采样，最早由 Monnahan 和 Kristensen (2018年) 探索，随后 Margossian 等人 (2020年) 对其进行了更深入的研究，并将这种方法称为“嵌入式拉普拉斯近似”（ELA）HMC。在 `SparseNUTS` 中，只需设置 `sample_snuts(...,laplace=TRUE)`即可轻松实现 ELA 方法。我们排除了模型 海燕、狭鳕和 *WHAM*，因为它们速度太慢。我们比较了 ELA-NUTS 和 ELA-SNUTS 相较于完整 SNUTS 的采样效率。ELA-NUTS 与 NUTS 的固定效应后验分布差异可能揭示拉普拉斯近似不准确所导致的偏差（Monnahan 和 Kristensen，2018年）。我们的重点是 NUTS 的计算效率，因此没有研究偏差，但我们指出，使用 ELA-NUTS 时必须考虑这一重要因素。

3 结果

对后验分布的最大相关性和边缘标准差之比进行比较后发现， Q 能够准确捕捉这一几何结构特征，但*wildf* 和*irt_2pl* 模型除外（表1，图 S4）。我们发现，

对所有模型而言，精度抽样对初始值的敏感性较低，且速度显著快于 Pathfinder（图4a）；在大多数模型中，其 Wasserstein 距离也相当或更优（即更小）（图 4b ）。这些结果表明，从 $\mathcal{N}(\hat{}q,\ Q^{-1})$ 进行采样既更快，而且通常也优于从 Pathfinder 抽取样本；而 Pathfinder 又优于其他 ADVI 模型（Zhang 等人，2022）。 Q 因此是多种复杂层级后验分布的一种很有前景的近似，这些样本可用于初始化 SNUTS 链，如 `init="random"` 在 `sample_snuts` 中所示。由于Q 无论如何都必须为 SNUTS 计算，因此这些随机抽样几乎无需额外成本。

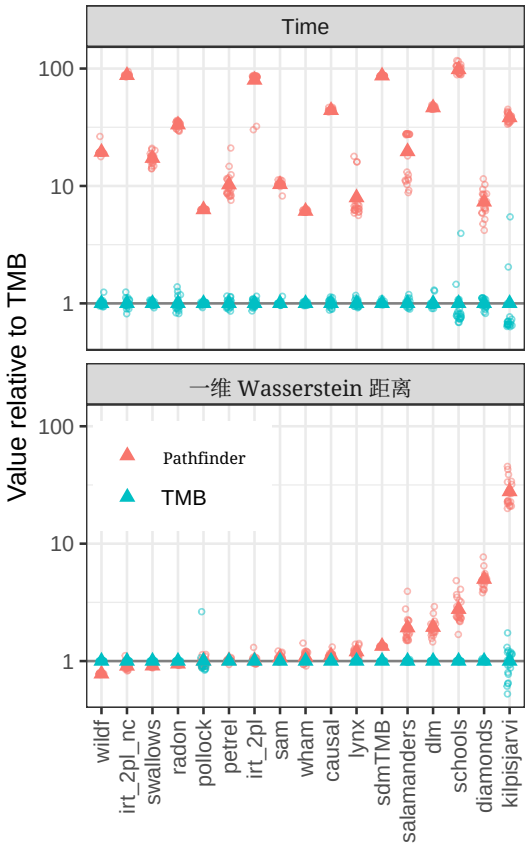


图 ***4.*** 与 ***Pathfinder*** 的比较。 这些图比较了 *TMB* 在表 *1*所列案例研究中的近似性能与 *Pathfinder* 的近似性能。 *TMB* 方法假设后验分布近似为 $\mathcal{N}(\hat{x},\ Q^{-1})$ 其中 $\hat{x}$ 是条件众数，Q 是根据方程 *5*计算得到的稀疏精度矩阵。*20* 个点分别代表一次独立分析，每次分析使用不同的随机初始参数向量（一个后验样本）和用于生成样本的随机种子。实心三角形表示均值；由于性能是相对于该方法衡量的，因此 *TMB*的方法始终为 *1*。*Pathfinder* 使用了 *4* 条串行路径，而 Q 只优化一次并生成样本。图中展示了两种方法的运行时间（上图；越低越好）和一维 *Wasserstein* 距离（下图；越低越好）。后者用于衡量 *1000* 个样本对后验分布的近似程度，两者均相对于 *TMB*给出。模型按 *Pathfinder* 相对于 *TMB*的性能排序。

我们在两个模拟示例中进行测试后发现，除*glmmTMB*模型参数少于 100 个时选用稠密预条件器外，SNUTS 均选用了稀疏预条件器。对于*glmmTMB* 模型，SNUTS 所需时间约为 NUTS 的 6–13%，有效样本量约为其两倍。同样，对于*spde* 模型，SNUTS 所需时间约为 NUTS 的 30–70%，产生的有效样本量是 NUTS 的 15–20 倍。综合来看，运行时间更短、有效样本量更高，使 SNUTS 效率较 NUTS 提高了 20–30（*glmmTMB*）倍和 20–70（*spde*）倍，具体取决于维度（图 5）。

在我们的案例研究中，SNUTS 通常也优于 NUTS。我们发现，计算变换后的梯度 $g_{q'}(q')$ 的平均计算成本是 $g_q(q)$ 成本的 1.00 倍（钻石）至 1.43 倍（*sam*），大多数情况下低于 1.3 倍（表 S3）。对于许多此类模型，梯度计算成本主要取决于计算原始梯度$g_q(q)$ 的成本，而这取决于模型复杂度和数据量。对于某个模型，SNUTS 的额外开销也相对较小，通常占 SNUTS 总运行时间的 <2%。这意味着只需增加少量成本，就能在 NUTS 抽样之前获得 Q。

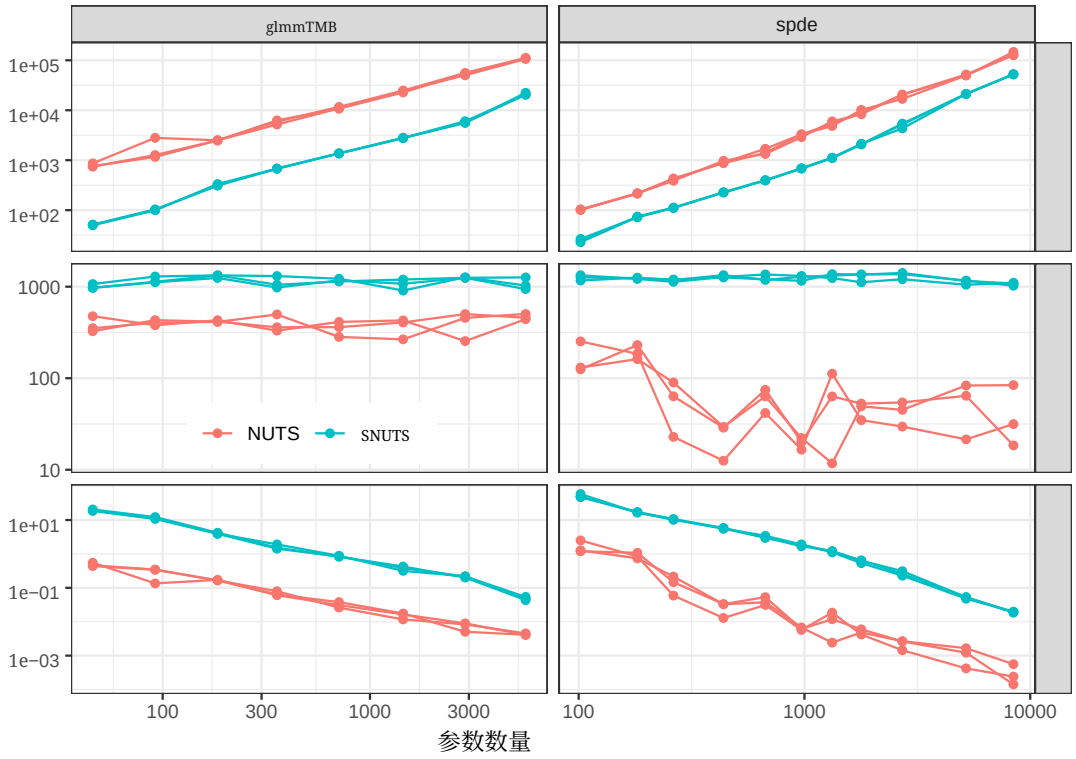


图*5.* 可扩展性评估。两个模型（列）随维度增加时的性能结果： *glmmTMB* 是负二项回归， *spde* 是空间泊松 *SPDE* 回归。每种算法（颜色）均展示了不同的性能指标（行）。三组分析各包含四条链（线），显示了估计值的变异性。

我们发现，在大多数案例研究中，SNUTS 的总耗时都更短，通常相差几个数量级，但两种算法的有效样本量差异较大（图 6）。总体而言，除*wildf* 和 *irt_2pl* 模型外，所有模型的效率（ESS/t）都更高；使用 SNUTS 后，性能提升最高可达几个数量级。例如，钻石 模型被列为Magnusson等人（2024年）中一个对 NUTS 而言几何结构较难处理的模型，但由于预热后平均轨迹长度从 NUTS 的 979 降至 SNUTS 的 7，SNUTS 的效率提高了 195 倍。减少梯度调用带来的成本节省，远远超过了该模型的 SNUTS 额外开销和较慢的梯度计算。该模型的参数相对较少（26 个），且 Q中没有稀疏性，但在规模更大且稀疏的模型中也有类似发现。例如，*WHAM* 模型是一种复杂的非线性状态空间种群动力学模型，其效率提高了 39 倍，平均轨迹长度从 256 降至 15，同时有效样本量提高了 2.19 倍。

wildf 模型值得注意，因为使用 SNUTS 导致其有效样本量在所有模型中降幅最大，而且与 Pathfinder 相比表现也最差（图4a、6）。在这种情况下，显然 Q 由于非线性相关性和边际方差（参数尺度）估计不准确，无法很好地近似后验分布，因此预条件反而降低了采样效率（图 S5）。除了稀疏预条件器外，我们还在 Stan 中启用了对角质量矩阵的自适应，使 Stan 能进一步调整尺度。结果确实如此：该版本的模型（*wildf_*自适应版，见图6）比 NUTS 快约 1.17 倍。

最后，在对 ELA 方法的测试中，我们发现结果参差不齐。对于学校和蝾螈模型，ELA-SNUTS 的效率始终高于 ELA-NUTS，分别为其 1.97 至 24.4 倍（图7），这表明使用 ELA 时对边缘后验进行预条件处理是值得的。此外，对于大多数但并非所有模型，完整的 SNUTS 都优于 ELA-NUTS；例如，因果和*sam* 这两个 ELA 模型都选择了稠密预条件器，其效率分别比完整的 SNUTS 高 2.79 倍和 1.74 倍。总体而言，我们的结果表明，SNUTS 通常优于 ELA-NUTS，而只有在某些情况下，ELA-SNUTS 才是更好的选择。

4 讨论

条件独立性自然存在于各种层级模型中，并体现为随机效应结构的稀疏性。诸如 `TMB` 和 `lme4` 这样的软件包利用这种稀疏性，高效地提供最大边缘

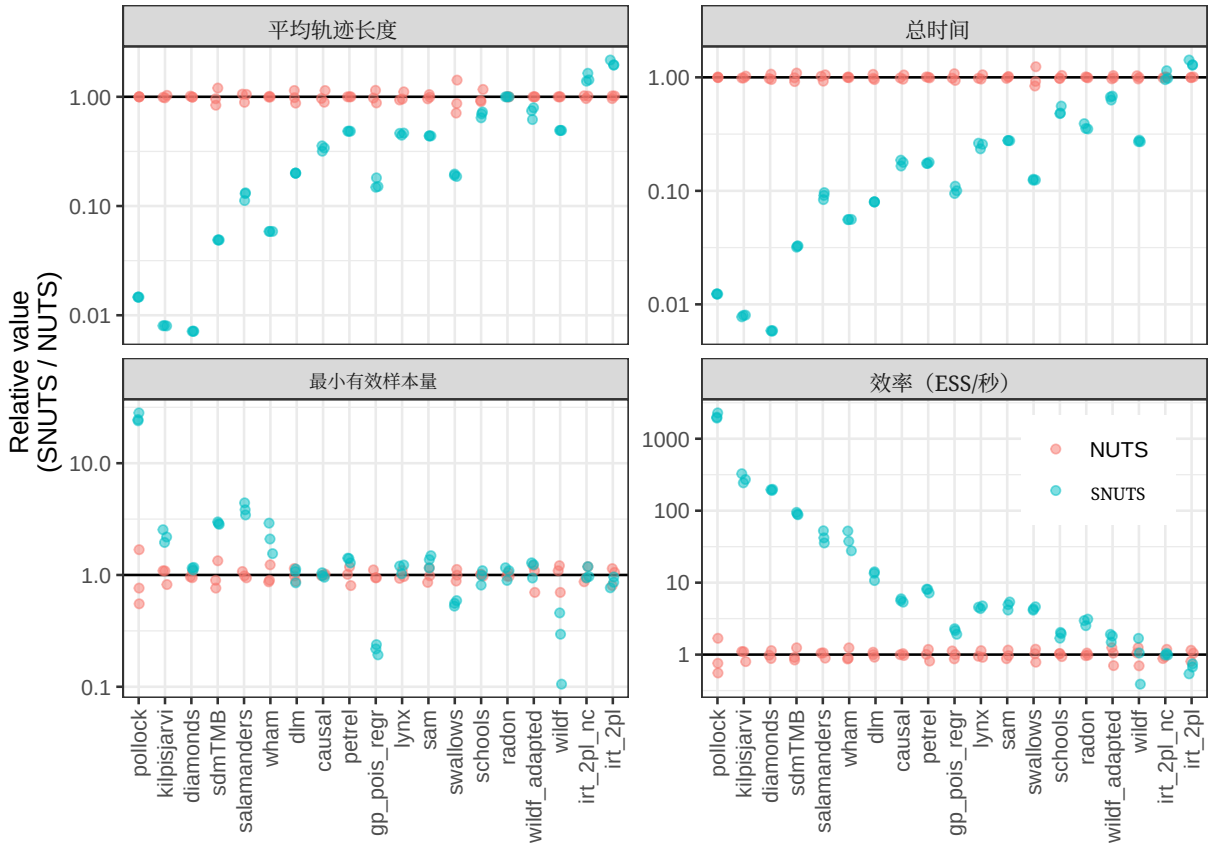


图 *6.* 与 ***NUTS*** 的比较。在多个模型/数据集组合上，对 *SNUTS* 与 *Stan* 中 *NUTS* 默认实现的四项性能指标进行比较。平均轨迹长度是预热后迭代中蛙跳步数的平均值。总时间（左）包括优化和计算 Q、采样和预热时间，但不包括编译时间。最小有效样本量（*ESS*）是所有参数中的最小值。采样效率是最小有效样本量与总时间的比值。*SNUTS* 自动选择的预条件器（*A1.2* 节）见表*1*。三组分别包含四条链的独立分析（点）展示了这种变化。

似然估计（最大边际似然估计），适用于多种层级模型形式。诸如 `INLA` (Lindgren 和 Rue，2015 年) 等软件包采用了相同的嵌套拉普拉斯近似方法进行两阶段贝叶斯推断。这里，我们采用了不同的方法，在最大边际似然估计处计算整个后验分布的全局近似。使用拉普拉斯近似来近似贝叶斯后验分布并非新鲜事（例如，参见 Kass 和 Steffey (1989 年) 以及 Gelman 等人(2014 年) 的第 13 章），但在最大边际似然估计处计算稀疏精度矩阵并用其进行预条件则是新的。这里我们展示，TMB 的速度和准确性都优于 Pathfinder 等变分推断技术（见图4）。这一准确性促使我们在贝叶斯工作流的探索阶段使用后验近似（Gelman 等人，2020年），甚至在需要高通量时用于推断。

我们重点探讨如何利用后验分布近似，对哈密顿蒙特卡罗采样器（如无转向采样器（NUTS））进行全局预条件。我们在此表明，预条件的效果非常显著，足以将面向几何结构复杂的层级模型的完整贝叶斯采样扩展到数万个随机效应；面对高相关性时，这远超标准推断软件包（如 `Stan` 或 `PyMC`）使用对角或稠密预条件器所能处理的规模。我们的方法是黑箱式的，无需针对特定层级模型结构进行定制，因此适用范围远广于 `INLA` 这类系统，后者需要为每种模型形式定制求积规则。

我们计算的核心——稀疏精度矩阵 Q——计算成本相对较低，仅占 SNUTS 运行时间的不到 2%（见表 S3）。因此，我们能够以极低的额外开销，在高维空间中考虑全局相关性（见图 3、5）。SNUTS 的采样效率更高，但这只是其中一部分。或许更重要的是，它消除了使用 NUTS 拟合模型时通常需要的漫长且不稳定的质量矩阵自适应预热期。归根结底，速度提升了一个到两个数量级

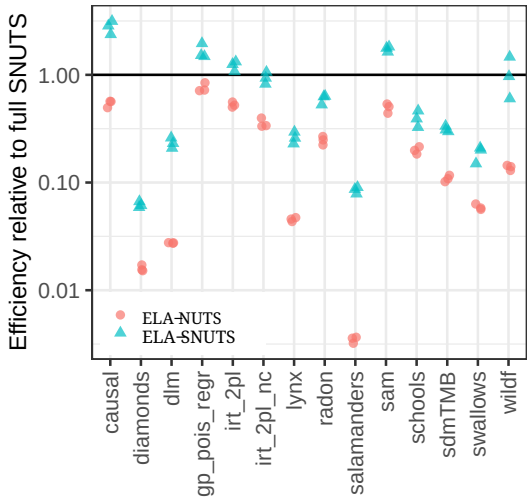


图*7.* 嵌入式拉普拉斯近似（***ELA***）的性能。 *ELA*是指由*NUTS*或*SNUTS*对经拉普拉斯近似边缘化的后验分布进行采样。*ELA-NUTS*仍使用*Stan*的默认设置，而*ELA-SNUTS*则采用自动选择的“稠密”或“对角”方法对边缘后验进行预条件化，并如前所述使用更短的预热期且不进行质量矩阵自适应（见第2.2节）。这两种*ELA*版本的效率（*ESS/t*）均相对于完整*SNUTS*采样显示，用以衡量对*NUTS*或*SNUTS*使用*ELA*时，相对完整*SNUTS*可能获得的改进。

，超越了当前黑箱层级模型领域最先进的水平，同时也提升了稳健性（见图 6）。这些改进在很大程度上得益于对稀疏性的利用，也表明利用稀疏性能够在贝叶斯推断的许多计算环节中发挥重要作用。

在 Q 中的稀疏性以及全局相关性，在本文所考察的模型中都很常见（表1、表S1）。每当相关性是NUTS效率低下的主要原因时——在并非数据匮乏的层级模型中，这种情况往往更为常见——我们预期SNUTS能够提供稳健性、效率与可扩展性。对于高斯马尔可夫随机场等模型尤其如此，例如空间随机偏微分方程模型，它们通常表现出高相关性与高稀疏性（例如，见图2）。 `TMB`充当各类流行统计软件包的计算后端，尤其是广义线性混合模型、各类时间序列模型以及物种分布模型（此类软件包的部分列表见表S1）。现有的 `TMB` 集成意味着SNUTS可方便地用于广泛类别的模型。 `TMB`的受欢迎程度正在上升，随之适用于SNUTS的模型范围也将持续扩大。

强相关性只是导致 NUTS抽样低效的众多原因之一。性能受漏斗状几何结构、多峰、极端尾部或其他具有挑战性的几何结构所限的模型，无法被多元正态分布很好地近似，因此 SNUTS 很可能无法提高采样效率。事实上，项目反应理论双参数逻辑斯蒂（*irt_2pl*）模型具有众所周知的困难几何结构，除高相关性外，还因每名学生的数据量较低而涉及乘法不可辨识性和漏斗。即便采用该模型的非中心化版本，SNUTS 也无法解决这一问题 Papaspiliopoulos 等人 （2007）（见 irt_2pl_nc，图 6）。然而，我们此处的结果以及贝叶斯后验分布随样本量增加的渐近行为（见 Gelman 等人 （2014 年）第 4.2 节）表明，许多贝叶斯模型将足够接近正态，从而使 SNUTS 能够提高采样效率，或至少帮助分析人员更快地识别并解决由非正态几何结构引起的其他问题。SNUTS 还便于轻松实现 ELA-SNUTS，这对于更具挑战性的几何结构可能是一个有用的工具。

我们在 `TMB` 中实现了 SNUTS，因为它能够自动检测任意模型的稀疏性，使得稀疏精度矩阵 Q 能够在高维情况下计算（方程（5））。其他平台也会在类似情境下利用稀疏性。例如， `INLA` (Lindgren 和 Rue，2015 年）通过嵌套拉普拉斯近似进行近似贝叶斯推断，并实现稀疏精度矩阵以提高计算速度，但与 `TMB` （例如，Osgood-Zimmerman和 Wakefield （2023年））相比，其适用范围仅限于某些类别的潜在高斯模型。Stan 和 PyMC 等更灵活的平台可以借鉴 `TMB` 的自动稀疏性检测，以及 SNUTS 所依托的拉普拉斯机制。目前，SNUTS 适用于 `TMB` 或 `RTMB`能够拟合的模型，包括一些常见的模型类型；而且`TMB`可用于评估在其他贝叶斯统计软件包中采用稀疏性的益处。

5 致谢

感谢 Andrew Johnson 修改 `StanEstimators` 软件包，使其专门适用于 `TMB` 和 `RTMB` 模型。我们感谢 Théo Michelot 提供海燕示例。我们也感谢 Darcy Webber 和 Eric Ward 对稿件早期版本提出的宝贵意见。

参考文献

Noel Cressie, Catherine A. Calder, James S. Clark, Jay M. Ver Hoef, 和 Christopher K. Wikle。《生态学分析中的不确定性处理：分层统计建模的优势与局限》。Ecological Applications，19(3):553–70，2009 年。doi: 10.1890/07-0744.1。Andrew Gelman, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, 和 Donald B. Rubin。《贝叶斯数据分析》，第 3 卷。Taylor and Francis，2014 年。Matthew D. Hoffman 和 Andrew Gelman。《无转向采样器：在哈密顿蒙特卡罗中自适应设定路径长度》。Journal of Machine Learning Research，15(1):1593–1623，2014 年。Cole C. Monnahan, James T. Thorson, 和 Trevor A. Branch。《利用哈密顿蒙特卡罗加快生态学中贝叶斯模型的估计》。Methods in Ecology and Evolution，8(3):339–348，2017年。doi: 10.1111/2041-210x.12681。Michael Betancourt 和 Mark Girolami。《用于层级模型的哈密顿蒙特卡罗》。Current trends in Bayesian methodology with applications，79(30):2–4，2015 年。Ian Langmore, Michael Dikovsky, Scott Geraedts, Peter Norgaard, 和 Rob Von Behren。《哈密顿蒙特卡罗的条件数》。arXiv 预印本 arXiv:1905.09813，2019年。Michael Betancourt。《哈密顿蒙特卡罗的概念性介绍》。arXiv 预印本 arXiv:1701.02434，2017年。URL https://arxiv.org/abs/1701.02434。Radford M. Neal。《利用哈密顿动力学进行 MCMC》。Handbook of Markov Chain Monte Carlo，2:113–162，2011年。doi: 10.1201/b10905。Ben Bales, Arya Pourzanjani, Aki Vehtari, 和 Linda Petzold。《在哈密顿蒙特卡罗中选择度量》，2019年。Alp Kucukelbir, Dustin Tran, Rajesh Ranganath, Andrew Gelman, 和 David M. Blei。《自动微分变分推断》。Journal of Machine Learning Research，18(14):1–45，2017 年。Lu Zhang, Bob Carpenter, Andrew Gelman, 和 Aki Vehtari。《Pathfinder：并行拟牛顿变分推断》。Journal of Machine Learning Research，23(306):1–49，2022年。Charles Margossian, Aki Vehtari, Daniel Simpson, 和 Raj Agrawal。《使用伴随微分拉普拉斯近似的哈密顿蒙特卡罗：潜在高斯模型及更广泛领域的贝叶斯推断》。载于 H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, 和 H. Lin 编，Advances in Neural Information Processing Systems，第 33 卷，第 9086–9097 页。Curran Associates, Inc.，2020年。Bob Carpenter, Andrew Gelman, Matthew D. Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus Brubaker, Jiqiang Guo, Peter Li, 和 Allen Riddell。《Stan：一种概率编程语言》。Journal of Statistical Software，76(1):1–29，2017年。ISSN 1548-7660。doi: 10.18637/jss.v076.i01。Stan 开发团队。Stan参考手册。v2.37.0.，2025年。URL http://mc-stan.org。Oriol Abril-Pla, Virgile Andreani, Colin Carroll, Larry Dong, Christopher J. Fonnesbeck, Maxim Kochurov, Ravin Kumar, Junpeng Lao, Christian C. Luhmann, Osvaldo A. Martin, Michael Osthege, Ricardo Vieira, Thomas Wiecki, 和 Robert Zinkov。《PyMC：一种现代、全面的 Python 概率编程框架》。PeerJ Computer Science，9:e1516，2023年。Kasper Kristensen, Anders Nielsen, Casper W. Berg, Hans Skaug, 和 Bradley M. Bell。《TMB：自动微分与拉普拉斯近似》。Journal of Statistical Software，70(5):21，2016年。doi: 10.18637/jss.v070.i05。Mollie E. Brooks, Kasper Kristensen, Koen J. van Benthem, Arni Magnusson, Casper W. Berg, Anders Nielsen, Hans J. Skaug, Martin Maechler, 和 Benjamin M. Bolker。《glmmTMB 在零膨胀广义线性混合建模的软件包中兼顾速度与灵活性》。R Journal，9(2):378–400，2017年。doi: 10.32614/Rj-2017-066。James T. Thorson, Sean C. Anderson, Pamela Goddard, 和 Christopher N. Rooper。《tinyVAST：一个 R 软件包，提供表达式接口，用于指定多元时空模型中的滞后效应与同期效应》。Global Ecology and Biogeography，34(4):e70035，2025a。doi: https://doi.org/10.1111/geb.70035。Sean C. Anderson, Eric J. Ward, Philina A. English, Lewis A. K. Barnett, 和 James T. Thorson。《sdmTMB：一个用于空间与时空随机场广义线性混合效应模型的快速、灵活且易用的 R 软件包》。Journal of Statistical Software，115(2):1–46，2025年。doi: 10.18637/jss.v115.i02。URL https://www.jstatsoft.org/index.php/jss/article/view/v115i02。

Kenneth F. Kellner, Adam D. Smith, J. Andrew Royle, Marc Kery, Jerrold L. Belant, 和 Richard B. Chandler。《unmarked R 软件包：生态学中出现率与丰度建模的十二年进展》。Methods in Ecology and Evolution，14(6):1408–1415，2023年。doi: https://doi.org/10.1111/2041-210X.14123。Havard Rue 和 Leonhard Held。《高斯马尔可夫随机场：理论与应用》。Chapman and Hall/CRC，2005年。Finn Lindgren, Havard Rue, 和 Johan Lindstrom。《高斯场与高斯马尔可夫随机场之间的显式联系：随机偏微分方程方法》。英国皇家统计学会会刊 B 辑：统计方法学，73(4):423–498，2011年。ISSN 1369-7412。doi: 10.1111/j.1467-9868.2011.00777.x。Kasper Kristensen。《RTMB：用于 TMB 的 R 语言绑定》，2025年。URL https://github.com/kaskr/RTMB。R 软件包版本 1.7。Michael Betancourt。《确定哈密顿蒙特卡罗中的最优积分时间》。arXiv 预印本 arXiv:1601.00225，2016年。Douglas Bates, Martin Maechler, 和 Mikael Jagan。《Matrix：稀疏与稠密矩阵类和方法》，2025年。URL https://CRAN.R-project.org/package=Matrix。R 软件包版本 1.7-3。Hans J. Skaug 和 David A. Fournier。《非高斯层级模型中边际似然的自动近似》。计算统计与数据分析，51(2):699–709，2006年。doi: 10.1016/j.csda.2006.03.005。Robert E. Kass 和 David Steffey。《条件独立层级模型中的近似贝叶斯推断（参数经验贝叶斯模型）》。美国统计学会会刊，84(407):717–726，1989 年。doi: 10.2307/2289653。Bradley M. Bell 和 Kasper Kristensen。《使用算法微分计算稀疏雅可比矩阵和海森矩阵》。arXiv 预印本 arXiv:2111.05207，2021年。URL https://arxiv.org/abs/2111.05207。Nan Zheng 和 Noel Cadigan。《混合效应模型与 TMB 的频率学派德尔塔方差近似》。计算统计与数据分析，160:107227，2021年。doi: https://doi.org/10.1016/j.csda.2021.107227。Andrew R. Johnson。《StanEstimators：使用 Stan 为任意 R 函数估计参数》，2025年。URL https://github.com/andrjohns/StanEstimators。R 软件包版本 0.1.2.9000。Robert J. McCann。《单调保测映射的存在性与唯一性》。杜克数学杂志，80 (2):309–323，1995年。doi: 10.1215/S0012-7094-95-08013-2。Katy Craig。《Wasserstein 度量的指数公式》。ESAIM：控制、优化与变分法，22(1):169–187，2016年。doi: 10.1051/cocv/2014069。Paul-Christian Bürkner, Jonah Gabry, Matthew Kay, 和 Aki Vehtari。《posterior：用于处理后验分布的工具》，2025年。URL https://mc-stan.org/posterior/。R 软件包版本 1.6.1。Aki Vehtari, Andrew Gelman, Daniel Simpson, Bob Carpenter, 和 Paul-Christian Bürkner。《秩正态化、ˆ 折叠与局部化：一种用于评估 MCMC 收敛性的改进 R》。Bayesian Analysis，16(2):667–718，2021年。doi: 10.1214/20-ba1221。Cole C. Monnahan 和 Kasper Kristensen。《用于快速贝叶斯推断的无转向采样：介绍 adnuts 和 tmbstan 这两个 R 软件包》。PLOS ONE，13(5):e0197954，2018年。doi: 10.1371/journal.pone.0197954。Mans Magnusson, Jakob Torgander, Paul-Christian Bürkner, Lu Zhang, Bob Carpenter, 和 Aki Vehtari。《posteriordb：测试、基准评估和开发贝叶斯推断算法》。arXiv 预印本 arXiv:2407.04967，2024年。Finn Lindgren 和 Havard Rue。《使用 R-INLA 进行贝叶斯空间建模》。Journal of Statistical Software，63(19):1–25，2015 年。Andrew Gelman, Aki Vehtari, Daniel Simpson, Charles C. Margossian, Bob Carpenter, Yuling Yao, Lauren Kennedy, Jonah Gabry, Paul-Christian Bürkner, 和 Martin Modrák。《贝叶斯工作流》。arXiv 预印本 arXiv:2011.01808，2020年。Omiros Papaspiliopoulos, Gareth O. Roberts, 和 Martin Skold。《层级模型参数化的一般框架》。统计科学，22(1):59–73，2007年。doi: 10.1214/088342307000000014。Aaron Osgood-Zimmerman 和 Jon Wakefield。《模板模型构建器的统计综述：一种灵活的空间建模工具》。国际统计评论，91(2):318–342，2023年。Jonah Gabry 和 Tristan Mahr。《bayesplot：贝叶斯模型绘图》，2025年。URL https://mc-stan.org/bayesplot/。R 软件包版本 1.14.0。Dominic Schuhmacher, Björn Bähre, Nicolas Bonneel, Carsten Gottschlich, Valentin Hartmann, Florian Heinemann, Bernhard Schmitzer, 和 Jörn Schrieber。《transport：最优传输计划与 Wasserstein 距离的计算》，2024年。R 软件包版本 0.15-4。

Ian D. Jonsen、W. James Grecian、Lachlan Phillips、Gemma Carroll、Clive McMahon、Robert G. Harcourt、Mark A. Hindell 和 Toby A. Patterson。aniMotum：用于动物移动数据的 R 软件包，支持快速质量控制、行为估计和模拟。*Methods in Ecology and Evolution*，14(3):806–816，2023年。Phillip Brinck Vetter、Jan K. Møller、Uffe Thygesen、Peder Bacher 和 Henrik Madsen。*ctsmTMB*：使用模板模型构建器进行连续时间随机建模，2025年。R 软件包版本 1.0。James T. Thorson、Albert G. Andrews III、Timothy E. Essington 和 Scott I. Large。动态结构方程模型综合了受生态机制约束的生态系统动态。*Methodsin Ecology and Evolution*，15(4):744–755，2024年。doi: 10.1111/2041-210x.14289。James T. Thorson、Kasper Kristensen、Kerim Y. Aydin、Sarah K. Gaichas、David G. Kimmel、Elizabeth A. McHuron、Jens M. Nielsen、Howard Townsend 和 George A. Whitehouse。层级生态系统模型的优势：使用新型状态空间质量平衡模型 EcoState 进行演示。*Fish and Fisheries*，26(2):203–218，2025b。Arni Magnusson 和 Mark Maunder。*fishgrowth*：拟合鱼类数据的生长曲线，2025年。R 软件包版本 1.0.4。Jenni Niku、Francis K. C. Hui、Sara Taskinen 和 David I. Warton。gllvm：在 R 中使用广义线性潜变量模型快速分析多变量丰度数据。*Methods in Ecology and Evolution*，10(12):2173–2182，2019年。doi: https://doi.org/10.1111/2041-210X.13303。Theo Michelot。hmmTMB：在 R 中使用灵活协变量效应的隐马尔可夫模型。*Journal of Statistical Software*，114(5):1–45，2025年。doi: 10.18637/jss.v114.i05。Sina Mews、Jan-Ole Koslik 和 Roland Langrock。如何构建潜在马尔可夫模型：时间与空间的作用。*Statistical Modelling*，25(6):481–507，2025年。doi: 10.1177/1471082X251355681。Jeff L. Laake、Devin S. Johnson 和 Paul B. Conn。marked：用于捕获—再捕获数据最大似然和马尔可夫链蒙特卡洛分析的 R 软件包。*Methods in Ecology and Evolution*，4(9):885–890，2013年。doi: https://doi.org/10.1111/2041-210X.12065。Daniel Sabanes Bove、Christian Stock、Liming Li、Jun He 和 Kayode Adebola。*mmrm*：重复测量混合模型，2025年。R 软件包版本 0.3.16。James T. Thorson 和 Wouter van der Bijl。phylosem：使用系统发育结构方程模型进行系统发育推断和性状插补的快速简便 R 软件包。*JournalofEvolutionary Biology*，36(10):1357–1364，2023年。Jens Wahl。*stochvolTMB*：随机波动模型的似然估计，2025年。R 软件包版本 0.3.0。Anders Nielsen 和 Casper W. Berg。使用状态空间模型估计种群资源评估中的时变选择性。*Fisheries Research*，158:96–101，2014 年。doi: 10.1016/j.fishres.2014.01.014。Alexios Galanos。*tsissm*：线性创新状态空间不可观测成分模型，2025年。R 软件包版本 1.0.2。James T. Thorson。使用 Vector Autoregressive Spatio-Temporal (VAST) 软件包进行种群、生态系统、栖息地和气候评估的决策指南。*Fisheries Research*，210:143–161，2019年。doi: https://doi.org/10.1016/j. fishres.2018.10.013。Brian C. Stock 和 Timothy J. Miller。Woods Hole Assessment Model (WHAM)：一种通用状态空间评估框架，通过随机效应纳入时间和年龄变化过程，并与环境协变量相关联。*Fisheries Research*，240:105967，2021年。doi: https://doi.org/10.1016/j.fishres.2021.105967。Sebastien Descamps、Arnaud Tarroux、Yves Cherel、Karine Delord、Olav R. Godo、Akiko Kato、Bjorn A. Krafft、Svein-Hakon Lorentsen、Yan Ropert-Coudert、Georg Skaret 和 Oystein Varpe。数据来自：南极海燕和商业磷虾渔业在海上的分布与猎物选择，2016年。网址 http://dx.doi.org/10.5441/001/1.q4gn4q56。Cole C. Monnahan、Bridget E. Ferriss、S. Kalei Shotwell、Zack Oyafuso、Mike Levine、James T. Thorson、Lauren Rogers、Jane Sullivan 和 Juliette Champagnat。阿拉斯加湾狭鳕种群评估。载于《阿拉斯加湾底层鱼类资源种群资源评估与渔业评价报告》。技术报告，由阿拉斯加湾底层鱼类计划小组编制，北太平洋渔业管理委员会，阿拉斯加州安克雷奇，2024年。Franzi Korner-Nievergelt、Tobias Roth、Stefanie von Felten、Jerome Guelat、Bettina Almasi 和 Pius Korner-Nievergelt。使用线性模型、*R*、*BUGS* 和 *Stan* 进行生态学贝叶斯数据分析：包括与频率学派统计学的比较。Academic Press，2015 年。ISBN 0128016787。Benjamin M. Bolker、Beth Gardner、Mark Maunder、Casper W. Berg、Mollie Brooks、Lizzie Comita、Elizabeth Crone、Sarah Cubaynes、Tyrell Davies、Perry de Valpine、John Ford、Olivier Gimenez、Marc Kery、Eun Jung Kim、Cleridy Lennert-Cody、Arni Magnusson、Steven Martell、John Nash、Anders Nielsen、Jim Regetz、Hans Skaug 和 Elise Zipkin。在 R、AD Model Builder 和 BUGS 中拟合非线性生态模型的策略。*Methods in Ecologyand Evolution*，4(6):501–512，2013年。ISSN 2041-210X。doi: 10.1111/2041-210X.12044。

A 补充章节

A.1 近似方法对初始化的敏感性

在正文中，我们使用 TMB从后验分布中抽取的样本初始化 Pathfinder。我们首先在简单的学校模型上测试了初始化选择的敏感性，由此得出这一做法。我们将所有参数随机初始化为 $U(-2,\ 2)$、 $U(-1,\ 1)$，或从参考后验分布初始化；此外还使用了两个固定向量：条件众数^q以及所有参数均设为 0 的向量。每种方法进行了 20 次重复，发现不同初始化方法的运行时间表现相近，精度抽样的速度约为 Pathfinder 的 100 倍（图S2）。值得注意的是，Pathfinder 的近似样本在 Wasserstein 距离上的变化更大，尤其是从后验分布初始化时；而从 0 初始化时方差最低。精度抽样的结果非常一致，因为边际优化能够找到相同的众数，因此 Wasserstein 距离的所有变异都来自随机样本生成。这表明 TMB 对初始化的敏感性低于 Pathfinder。我们使用了后验样本，因为这似乎是一种直接的折中方案；然而在实践中，后验样本并不可用，因此需要采用其他方法。

A.2 自动选择预条件器类型

对于给定模型，最佳预条件器（即对角、稠密或稀疏）取决于许多因素。许多模型没有边际众数，因此可能无法获得 Q、 Q，但它们可能没有稀疏性或全局相关性很弱（条件数较小）；也可能没有随机效应，因此只有 Σ 可用。这些情况意味着需要采用不同的预条件方法。幸运的是，通常在开始 NUTS 抽样之前就能获得这些信息，因此我们提出一种启发式方法来自动选择方案。
我们采用以下方案自动选择去相关类型：

- 如果 Q 和 Σ 均不可用，则采用 Stan 的默认设置，估计对角质量矩阵。
- 如果相关性较低，只需使用对角矩阵对后验分布进行去尺度化，因为稠密和稀疏预条件器只会增加计算开销。
- 如果相关性较高（成对相关性的最大值高于 0.3），则使用稠密或稀疏矩阵，具体取决于哪种梯度计算速度更快（即公式 2 或 3；图 3）。

由于预条件是在 Stan 外部应用的，因此 Stan 内部使用的质量矩阵 M 会作用于预条件化后验分布。理论上，如果 Q^{-1} 对后验分布的几何结构近似得足够准确，那么进一步去相关或去尺度化就没有必要，因此可以禁用 Stan 的质量矩阵自适应，从而将预热期大幅缩短至 150 个预热样本（参见附录 A 中的 A.3）。可以通过事后比较后验分布的边际方差和成对参数相关性的估计值与 Q所隐含的对应值来诊断这一点；若两者差异较大，则表明近似存在问题，并可能导致 SNUTS 效率下降。正文中使用 *wildf* 模型展示了这种做法如何进一步提高采样效率

值得注意的是，我们的自动选择流程会给 NUTS 分析增加额外的计算成本，具体包括边际优化、估计 Q、求逆 $Q^{-1} = \Sigma$ 以获得相关性；如果相关性较高，还需进行乔列斯基分解 chol(Σ) 和 chol(Q_P)，以比较稠密与稀疏梯度的计算成本。我们假设，与完整 NUTS 分析所需的时间相比，这些额外计算开销微不足道，并将在下文检验这一假设。

A.3 确定足够的预热期

Stan 中 NUTS 的默认常用配置是在预热阶段通过一系列扩展窗口，同时调整步长和对角质量矩阵（Stan 开发团队，2025年）。这一流程旨在适用于各种模型类型，但需要较长的预热期，通常为 1000 次预热迭代，或总迭代次数的 50%。在此我们指出， Q 同时表示边际方差和全局相关性；若能得到良好近似，则无需调整质量矩阵，因此预热期只需足够长，以便进入典型集并调整步长。

我们通过在 Stan 中对 SNUTS 链运行常规自适应，并考察扩展窗口期间质量矩阵自适应对步长的影响，来检验这一说法，以确定预热需要多长时间才能使步长估计值稳定。我们按照正文所述（精度抽样），从 Q 中随机抽取初始值，分别初始化 10 条链。我们发现，即使在第一个窗口之后，所有模型的步长都已稳定（图 S3），这表明对角质量矩阵自适应并未对后验分布进行实质性的重新缩放，或许只有 *wildf* 模型例外（见正文的结果部分）。因此，我们禁用了 Stan 的质量矩阵自适应，并将总迭代次数设为 1250，其中 150 次

次为所有 SNUTS 分析的预热迭代；与此相反，NUTS 分析采用 Stan 的默认设置，分别为 2000 次和 1000 次。

A.4 SparseNUTS R软件包的示例用法

主函数是 sample_snuts(obj)，它接受 TMB的动态列表对象（obj）及其他可选参数，并返回自定义 R S3 类 tmbfit 的一个成员；该类是一个列表，包含重要输出，例如条件众数 $\hat{q}$、Q以及参数相关性，还有后验样本和关于 SNUTS 分析的元数据。用户可以指定标准 NUTS 输入，例如链的数量、每条链的预热次数和样本数，以及用于指定（例如）最大树深度和目标接受率的控制列表；此外还可指定 SNUTS 特有选项，例如通过 metric 参数指定预条件器类型（默认值为“auto”），以及用于覆盖边际优化和内部 Q 计算默认行为的控制项。

我们演示基本用法，针对带有随机站点的 glmmTMB 负二项回归，如下所示（另见 o 蝾螈见表 1，S2）。

```
library(glmmTMB)
data(Salamanders)
mod <- glmmTMB(count~spp * mined + (1|site), Salamanders, family="nbinom2")

library(SparseNUTS)
fit <- sample_snuts(obj=mod$obj)
print(fit)
```

这会产生以下控制台输出。

```
Model 'glmmTMB' has 39 pars, and was fit using NUTS with a 'dense' metric
4 chain(s) of 1150 total iterations (150 warmup) were used
Average run time per chain was 5.5 seconds Minimum ESS=1284.6 (32.11%),
 and maximum Rhat=1.003 There were 0 divergences after warmup
```

在此示例中，Q 的稀疏度为 44.5%，最高的成对相关性为 0.96；我们可以看到，系统自动选择了稠密预条件器，并采用较短的预热期且不进行质量矩阵自适应。SparseNUTS提供了一组成员函数，让用户能够执行贝叶斯分析中的标准任务。SNUTS 的一个独特之处在于，它同时提供近似（即 Q）和完整 MCMC 抽样的估计值。可以对这些估计值进行比较，以评估近似的准确性，例如判断在模型开发初期近似是否足够，或更好地了解预条件为何可能无法提高采样效率。具体而言，SparseNUTS 有一个 pairs 成员函数，可绘制后验分布的成对图，并叠加以 $\hat{q}$ 为中心、来自 Q^{-1} 的 95% 置信椭圆；还有函数 plot_uncertainties 用于比较渐近估计值与后验边际方差估计值，以及 plot_marginals 用于绘制直方图。

后验样本可通过 as.data.frame() 提取到 R 数据框中，并放回 TMB 对象的report 函数，以从派生量获得后验样本。该输出也可接入外部贝叶斯 R 包，用于典型贝叶斯工作流的其他环节，例如通过 PSIS-LOO 进行模型选择（Vehtari等人，2021年），通过 bayesplot （Gabry 和 Mahr，2025年）进行专门的 MCMC 绘图，以及后验预测检验的计算与可视化。软件包示例文档提供了该软件包这些及其他功能的实例：[https://cole-monnahan-noaa.github.io/SparseNUTS/.

A.5 Pathfinder 与 TMB 测试细节

我们测试了两种近似方法的计算时间。对于 Pathfinder，运行时间包括优化、生成随机正态样本以及评估下界证据。我们使用了四条路径（默认设置），但为便于比较而未并行运行。精度抽样的计算成本包括边缘后验的优化、Q的计算（公式 4）、Q的求逆，以及随机正态样本的生成。对于大型、高度稀疏的 Q 矩阵，若直接从 Q 生成样本（即不求 Q的逆；Rue 和 Held （2005年）的算法 2.4），可能会更高效，但此处未作探讨，因为在我们的案例研究中，这一计算方面的开销很小。对案例研究集合（表 1）的初步评估导致 Pathfinder 在从分散点初始化时出现许多数值问题且估计值不稳定，因此我们对所有模型使用了后验抽样。

为计算1-Wasserstein 距离，我们使用了 wasserstein 函数，该函数来自 transport R软件包（Schuhmacher等人，2024年）。估计的Wasserstein距离表示一个经验分布需要

发生多大程度的扭曲才能与另一个分布相匹配；数值越大表示差异越大，数值越接近零则表示分布越相似。我们计算了两种算法各自的1000个近似样本与10,000个NUTS样本之间的距离，并假定这些NUTS样本代表真实后验分布。

B 补充图表

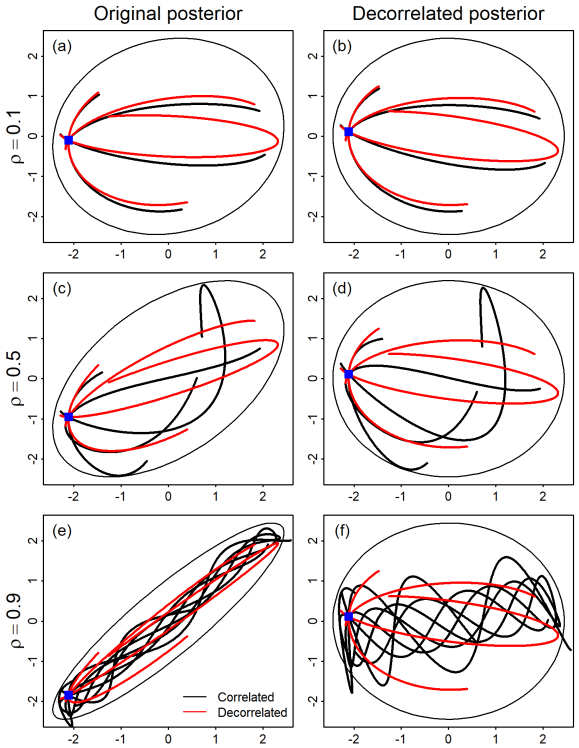


图 ***S1. NUTS*** 轨迹示例。这些图展示了单位方差、但相关性水平不同的二元正态后验密度 ρ（各行）。图中展示了 *5* 条轨迹（线条），它们从同一点（蓝色方块）出发，但随机动量各不相同。轨迹分别使用和不使用稠密预条件器计算（不同颜色），展示在原始后验分布上（q；左图）或预条件化后验分布上（q'；右图）。两个面板中的预条件化轨迹完全相同

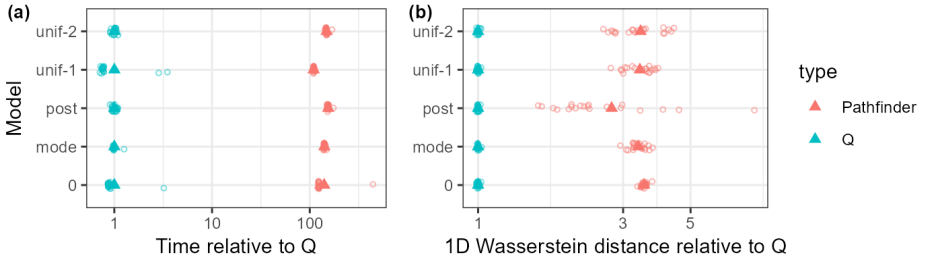


图 ***S2.*** 更多 ***Pathfinder*** 评估。这些图比较了针对 *schools* 模型的精度（Q）近似与 *Pathfinder* 算法在五种不同初始值生成方法下的近似性能。方法包括从 *U(-2,2)*（*"unif-2"*）或 *U(-1,1)*（*"unif-1"*）中随机抽样、从后验分布中随机抽样（*"post"*）、`TMB` 的经验贝叶斯条件众数（*"mode"*），以及将所有参数设为零（*"0"*）。*20* 个点中的每个点代表一次单独的分析，使用不同的随机初始参数向量（在适用时，两种方法使用相同的向量）和用于生成样本的随机种子。均值以实心三角形表示。*Pathfinder* 使用了 *4* 条串行路径，而 Q 优化一次并生成样本。

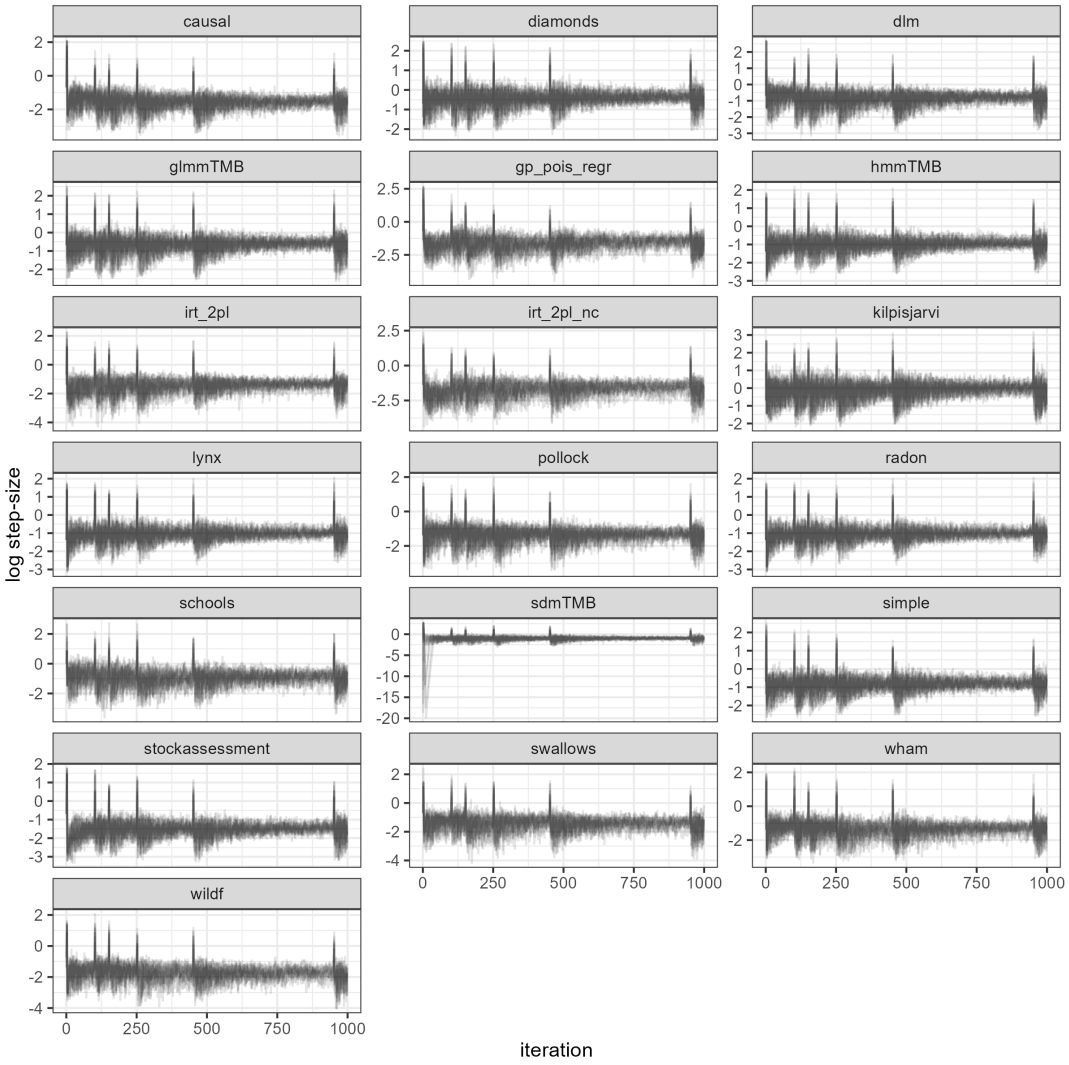


图 ***S3.*** 适应结果。图中显示了 *10* 条链在 *1000* 次预热迭代期间的 *NUTS* 步长对数；各链以使用精度抽样得到的随机值初始化。*Stan* 的适应方案会按逐渐扩大的间隔更新质量矩阵（本例中为 `diag_e`），随后相应地更新步长。每个模型均使用*"auto"*设置自动确定预条件器，因此，当后验分布可由 Q 良好近似时，预计质量矩阵在预热迭代期间不会发生显著变化。

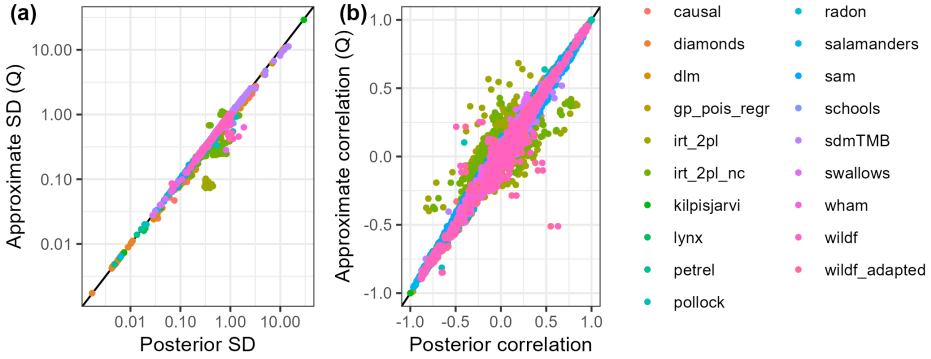


图 ***S4.*** 评估 Q 作为后验分布。图中按模型（颜色）比较了 Q 对后验分布的近似效果，分别考察 *(a)* 边际标准差 *(SD)* 和 *(b)* 成对相关性。偏离 *1:1* 线的点表明，精度为 Q 的多元正态分布对后验分布的近似效果较差。

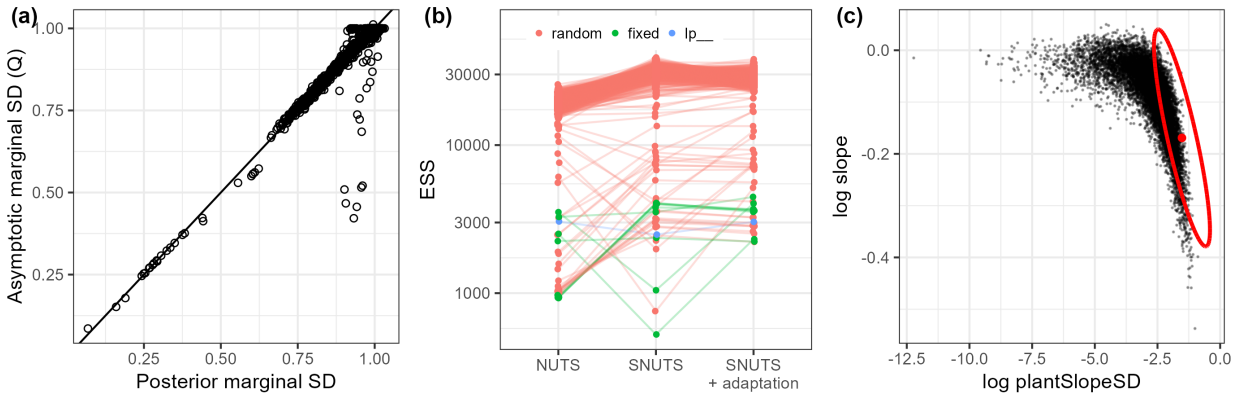


图 ***S5.*** 探究 ***wildf*** 的不良表现。*wildf* 模型重新运行了三次：标准 *NUTS*（即 *Stan* 默认设置），其在预热期间包含对角质量矩阵自适应；*SNUTS* 默认设置，其跳过质量矩阵自适应；以及再次使用 *SNUTS*，但开启对角质量矩阵自适应。三者均使用四条链以及 `adapt_delta=0.999` 和 *4000* 个总样本（*1000* 次预热），以获得更好的收敛和稳定的估计值。*(a)* 针对 *SNUTS*，比较了每个参数（点）来自后验样本的边际标准差与来自 Q^{-1} 对角元素平方根的渐近近似。许多参数的一致性较差，表明 Q 并不能很好地近似后验分布。*(b)* 三次运行中每个参数的有效样本量（*ESS*）（线段），按该参数是固定效应还是随机效应、或是未归一化对数后验密度（*"lp__"*）着色。少数参数在 *SNUTS* 下的有效样本量低得多，但在开启自适应后再次升高，证实对 Q 的不良近似有损采样效率，而通过在 *Stan* 中自适应对角质量矩阵可以恢复这一点。*(c)* 有效样本量最低的两个参数的后验点散点图，以及来自 Q 的二元渐近正态 *95%* 椭圆（红线），其中心位于估计众数。对这两个参数而言，该近似较差。

R软件包	标题 / 描述	参考文献
aniMotum	将潜变量移动模型拟合于动物追踪数据，以进行位置质量控制和行为推断。	Jonsen 等 （2023年）
ctsmTMB	使用拉普拉斯近似进行 TMB中的连续时间随机建模。	Vetter 等人 (2025年)
dsem	将动态结构方程模型应用于时间序列数据，并采用通用且简化的规范来表示同时效应和滞后效应。	Thorson 等人 (2024年)
ecostate	海洋生态系统的状态空间质量平衡模型。	Thorson 等人（2025年b）
fishgrowth	将 Git 生长曲线应用于鱼类数据。	Magnusson 和 Maunder （2025年）
gllvm	使用广义线性潜变量模型分析多变量生态数据（GLLVM）。	Niku 等人 (2019年)
glmmTMB	使用不同的分布（泊松分布、负二项分布、零膨胀分布）和模板模型构建器拟合广义线性混合模型。	Brooks 等人 (2017年)
hmmTMB	用于带随机效应和平滑样条。	Michelot （2025年）
LaMa	潜在马尔可夫模型的快速数值最大似然估计。	Mews 等 (2025年)
marked	用于生存和丰度估计的标记重捕分析。	Laake 等 (2013)
mmrm	重复测量混合模型（MMRM）是分析-用于随机临床试验及其他领域的纵向连续结局。	Bove 等人(2025年)
phylosem	实现结合比较方法与混合模型的系统发育结构方程模型（SEM），替代方法和混合模型。	Thorson 和 van der Bijl (2023年)
sdmTMB	快速、灵活且易于使用的广义线性混合效应模型，支持空间与时空随机场。	Anderson 等 （2025年）
stochvolTMB	使用最大似然进行随机波动模型的参数估计。	Wahl (2025年)
stockassessment	用于渔业资源评估的状态空间评估模型。	Nielsen 和 Berg （2014 年）
tinyVAST	用于指定多元时空模型。	(Thorson等，2025年a)
tsissm	采用线性创新状态空间表示的不可观测成分时间序列模型（单一误差源），可选择误差分布，并可选择动态方差。	Galanos (2025年)
unmarked	使用以下方法为野生动物调查数据（占域、丰度）拟合层级模型 TMB 以快速进行边际似然估计。	Kellner 等 (2023年)
VAST	对多个类别（物种、）的数据进行时空分析，大小等），并纳入密度的空间变异和时空变异，适用于各类别之间采用因子分析或自回归相关，类别，并据此计算丰度指数、重心和占据面积。	Thorson （2019年）
wham	一种通用的状态空间评估框架，通过随机效应纳入随时间和年龄变化的过程，并与环境协变量相连接。	Stock 和 Miller （2021）
tinyVAST	用于指定多元时空模型。	(Thorson 等，2025a)

表***S1.*** 使用 ***TMB***的***R***软件包。这是一份非详尽列表，列出了使用模板模型构建器（*TMB*）或 *R* 接口 *RTMB* 作为计算引擎的 *R* 软件包。

Name	描述和修改
因果	多因果生态系统综合时间序列模型（由 'dsem' 实现，Thorson 等，2024年）。$\mathcal{N}(0,\ 0.25)$ 先验已添加到 beta_z[9:16] 以及 $\mathcal{N}(0,\ 0.5)$ 添加到 beta_z[20]。这些参数的识别度较低，且尾部长度不切实际地长。
钻石	含相关协变量的钻石线性混合效应模型。收录于Magnusson 等人 (2024年)，作为一个对 NUTS 而言几何结构较复杂的模型。
dlm	动态线性模型 https://james-thorson-noaa.github.io/dsem/articles/features.html#comparison-with-dynamic-linear-models 按"dsem"中的方式实现（Thorson 等人，2024年）。
gp_pois_回归	泊松高斯过程回归。数据来自 posteriordb 数据库（Magnusson 等人，2024年）。
irt_2pl	取自 posteriordb 数据库的项目反应理论模型（Magnusson等人，2024年）。这是中心化版本，非中心化版本 irt_2pl_nc 见一些图中。
基尔皮斯耶尔维	年度夏季气温与未中心化协变量的线性回归导致较高的后验相关性。摘自Bales 等人（2019年）。
lynx	表示为常微分方程并使用四阶龙格-库塔方法求解的猞猁与野兔动力学数据取自 https://jmahaffy.sdsu.edu/courses/f09/math636/lectures/lotka/qualde2.html。
海燕	包含个体随机效应的两状态运动隐马尔可夫模型使用 hmmTMB 的转移概率和伽马观测值（Michelot，2025年）根据南极海燕数据拟合（Descamps 等人，2016年）。
狭鳕	一个定制的非线性状态空间渔业种群动态模型，经调整自 Monnahan 等人（2024年）。我们为调查2，并关闭了对 log_DMpars[6] 的估计，该参数对应于狄利克雷-多项分布的过度离散项，用于调查6的年龄组成。
氡	按国家划分的氡水平线性混合效应模型。摘自 Bales 等人 (2019年)。
蝾螈	按地点统计蝾螈物种计数的零膨胀负二项模型，拟合结果如下在 glmmTMB（Brooks 等人，2017年）。
sam	在"sam"软件包中拟合的非线性状态空间渔业种群动态（Nielsen 和 Berg，2014 年），取自在线示例 https://github.com/fishfollower/SAM?tab=readme-ov-file#a-quick-example。
学校	学生成绩八校线性混合模型的经典非中心化版本，如例如，Gelman 等人（2014 年）。
sdmTMB	一个带有深度协变量样条的空间 SPDE Tweedie 广义线性混合模型，通过 `sdmTMB` 拟合包（Anderson 等人，2025年）。根据一个示例修改，使参数 κ 未进行估计。
燕子	使用环境协变量拟合鸟类标记重捕数据的状态空间生存与检测——鸟类重捕数据，取自Korner-Nievergelt 等人（2015 年），也曾由 Monnahan 等人（Monnahan 等人（2017年）中也进行了研究。sigmayear 的正态先验收紧为 $\mathcal{N}(0,\ 2)$。否则，当该参数取值较高时，NUTS 会出现许多发散。
wham	使用"wham"软件包的非线性状态空间资源评估模型（Stock and Miller，2021），基于教程示例中的例子，其中对数数量采用二维一阶自回归结构，位于年龄状态。未估计选择性和协变量回归参数（Ecov）因为可辨识性极差，且缺少用于指定先验的接口。
wildf	开花成功率的二项广义线性混合模型，其中截距具有年份效应，并包含交叉效应截距和协变量斜率上，并取自Bolker等人（2013年），也进行了分析深入探讨，见Monnahan等人（2017年）。

表 ***S2***。案例研究详情。本表进一步介绍了所采用的案例研究（见表 *1*）

模型	度量	参数	% 稀疏性（Q）	梯度比	Q 的 %
因果	稀疏	264	78.7	1.19	0.2
钻石	稠密	26	0.0	1.00	2.2
dlm	稠密	33	8.1	1.06	1.5
gp_pois_回归	稠密	13	0.0	1.10	0.0
irt_2pl	稀疏	144	56.3	1.21	0.1
irt_2pl_nc	稀疏	144	56.2	1.24	0.1
基尔皮斯耶尔维	稠密	3	0.0	1.12	0.1
lynx	稠密	50	63.0	1.09	2.4
海燕	稀疏	274	94.3	1.00	1.1
狭鳕	稀疏	351	28.3	1.11	2.1
氡	diag	389	98.5	1.01	0.2
蝾螈	稠密	39	44.5	1.03	1.4
sam	稀疏	1226	96.6	1.43	1.2
学校	diag	10	62.2	1.08	1.0
sdmTMB	稀疏	219	81.3	1.01	0.0
燕子	稀疏	177	75.4	1.05	1.0
wham	稀疏	385	66.2	1.02	4.6
wildf	稀疏	1101	98.4	1.14	1.0

表 ***S3. SNUTS*** 基准测试。该表展示了 *SNUTS* 在各案例研究中的计算成本。梯度比是 $g_q(q)$ 的成本除以 $g_{q'}(q')$ 的成本；它取决于所选的预条件器，表示变换后的梯度比梯度慢多少倍。最后一列表示优化 Q、将其求逆以得到 $Q^{-1} = \Sigma$，以及对 Q 和 Σ进行 *Cholesky* 分解所需的计算成本占 *SNUTS* 运行时间总量的百分比。有关模型的更多详情，请参见表 ***1***。

C R源代码

本节介绍 R 实现的相关细节。

```
library(RTMB)
rm(list=ls())# set up 8 schools example:dat <- list(J = 8,
y = c(28,  8, -3,  7, -1,  1, 18, 12),
sigma = c(15, 10, 16, 11,  9, 11, 10, 18))
pars <- list(eta=rep(1,8),mu=0, logtau=1)f <- function(pars){
RTMB::getAll(dat, pars)theta <- mu + exp(logtau) * eta;
lp <- sum(dnorm(eta, 0,1, log=TRUE))+ # hyperprior
sum(dnorm(y,theta,sigma,log=TRUE))+ # likelihood
logtau                              # jacobian
gq <- sum(exp(logtau)*eta[2:4])       # generated quantity fixed + random
ADREPORT(gq)                          # request delta method SE
REPORT(gq)                            # simple output
return(-lp)                           # TMB uses negative lp}
obj <- MakeADFun(func=f, parameters=pars, random='eta', silent=TRUE)



# Optimize and get Q via sdreport function
opt <- with(obj, nlminb(par,fn,gr))
sdrep <- sdreport(obj, getJointPrecision=TRUE)
str(sdrep$jointPrecision)
```

然后，我们手动使用 TMB的自动微分机制和 Matrix 包功能计算 Q 。

```
# This code block shows how to calculate Q in R. It reproduces
# the 'sdreport' functionality in a minimal way (see equations
# 9-10) and only for RTMB models. It is meant only as a
# demonstration for the model above (schools).

library(Matrix)
q_hat <- obj$env$last.par.best # joint mode
n <- length(q_hat)
theta_hat <- opt$par          # marginal mode
r <- obj$env$random           # index of random effects
nonr <- setdiff(seq_along(q_hat), r)

# Hessian block for fixed effects using finite differences
H_Bhat <- optimHess(theta_hat, obj$fn, obj$gr) # Hessian of marginal posterior

# Hessian of random effects at joint mode using AD.
H_AA <- obj$env$spHess(q_hat, random = TRUE)

# Second derivatives of the joint posterior at the joint
# mode for the fixed:random effect elements only. Uses AD.
H_AB <- obj$env$f(q_hat, order = 1, type = "ADGrad", keepx=nonr, keepy=r) ## TMBad only !!!
H_BA <- t(H_AB)
H_BB <- H_BA %*% solve(t(H_AA)) %*% H_AB + H_Bhat
Q <- Matrix(0, nrow = n, ncol=n, sparse=TRUE)
Q[r,r] <- H_AA
Q[r,nonr] <- H_AB
Q[nonr,r] <- H_BA
Q[nonr, nonr] <- H_BB

# this matches TMB's internal calculations in sdreport:
max(abs((Q-sdrep$jointPrecision))) # [1] 2.775558e-17
```

接下来我们展示如何使用 Q 对参数空间进行去相关（公式 3）。

```
# Use sparse metric Q to transform from original ‘x‘ to reparamterized ‘y‘
# first rebuild object without the Laplace appoximation (random=NULL)
obj2 <- RTMB::MakeADFun(func=obj$env$data, parameters=obj$env$parList(),
map=obj$env$map,random=NULL, silent=TRUE,DLL=obj$env$DLL)
# Do Cholesky on permuted Q
chd <- Matrix::Cholesky(sdrep$jointPrecision, super=TRUE, perm=TRUE)
L <- as(chd, "sparseMatrix")
# Drop all numerical zeros and convert to triangular storage
L <- tril(drop0(L)) ## class(L) == "dtCMatrix"
Lt <- Matrix::t(L) ## class(Lt) == "dtCMatrix"
perm <- chd@perm + 1L            # the new order of the system
iperm <- Matrix::invPerm(perm)   # the inverse
# assume an initial value in x space, like the joint mode
x0 <- q_hat      # the joint mode
x0perm <- x0[perm]               # permuted
# initial value in the transformed, permuted space
y0perm <- as.vector(Lt %*% x0perm)
# redefine the objective and gradient functions
fn.y <- function(y)  -obj2$fn(Matrix::solve(Lt, y)[iperm])
# carefully implement this to optimize sparsity and reduce
# computations since the gradient call is the expensive part of# NUTS
gr.y <- function(y){x <- Matrix::solve(Lt, y)[iperm]
-Matrix::solve(L, as.numeric(obj2$gr(x))[perm])}# back transform parameters
y.to.x <- function(y) as.numeric(Matrix::solve(Lt, y)[iperm])



# test them out
fn.y(y0perm)
obj2$fn(x0)  # matches but for sign
gr.y(y0perm) # gradient at joint mode is 0 for fixed effects only
```

最后，我们通过 `StanEstimators` 接口实现 SNUTS，并展示如何使用广义 delta 方法

```
## ------------------------------------------------------------
# Now show how to link this through StanEstimatorslibrary(StanEstimators)
fit <- stan_sample(fn=fn.y, par_inits=x0perm, grad_fun=gr.y,num_chains=4,
 seed = 12345)# posterior draws in transformed space
post.y <- unconstrain_draws(fit) |> as.data.frame()
# recorrelate the draws into untransformed space x
post.x <- t(apply(post.y[,2:11], 1, FUN=y.to.x))
cbind(postmean=colMeans(post.x), postmode=x0)
## ------------------------------------------------------------
# Now compare approximate (asymptotic normal) estimate of a
# generated quantity using the generalized delta method (via
# TMB::sdreport) against the posterior
# The mean and SE from the delta methd are assumed to be normal
gq.mle <- c(mean=sdrep$value, sd=sdrep$sd)
# push each posterior draw through and extract the generated
# quantity ’gq’ to get posterior of gq
gq.post <- apply(post.x, 1, \(x) obj2$report(x)$gq)
hist(gq.post, freq=FALSE, breaks=30, main=’Generated quantity example’,
xlab=’gq’)x <- seq(min(gq.post), max(gq.post))
```

```
y <- dnorm(x, gq.mle[1], gq.mle[2])
lines(x,y, col=2, lwd=2)
```