



中国矿业大学 (北京)

China University of Mining & Technology, Beijing

硕士学位论文

空间广义线性混合效应模型及其应用

作者：黄湘云

学院：理学院

学号：TSP150701029

学科专业：统计学

导师：李再兴

2018 年 6 月

中图分类号: _____

单位代码: _____

密 级: _____

硕 士 学 位 论 文

中文题目: 空间广义线性混合效应模型及其应用

英文题目: Spatial Generalized Linear Mixed Models and Its Applications

作 者: 黄湘云

学 号: TSP150701029

学科专业: 统计学

研究方向: 数据分析与统计计算

导 师: 李再兴

职 称: 教授

论文提交日期: 2018 年 月 日 论文答辩日期: 2018 年 月 日

学位授予日期: 2018 年 月 日

中国矿业大学(北京)

独创性声明

本人声明所呈交的学位论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得中国矿业大学或其他教学机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

作者签名:_____ 日期:_____

关于论文使用授权的说明

本人完全了解中国矿业大学有关保留、使用学位论文的规定，即：学校有权保留送交论文的复印件，允许论文被查阅或借阅；学校可以公布论文的全部或部分内容，可以采用影印、缩印或其他复制手段保存论文。

（保密的论文在解密后应遵守此规定）

作者签名:_____ 导师签名:_____ 日期:_____

摘 要

在空间广义线性混合效应模型（简称 SGLMM 模型）下，估计所需的高效计算方法一直是研究的重要方向。论文创新点有三个方面，其一是在 R 语言编程环境下实现了估计 SGLMM 模型参数的三类算法，分别是低秩近似、蒙特卡罗最大似然和贝叶斯马尔科夫链蒙特卡罗（简称贝叶斯 MCMC）算法；其二是在数值模拟结果上比较了三类算法的优劣；其三是在软件 Stan 上实现了贝叶斯 MCMC 算法（简称 STAN-MCMC），并将其应用到喀麦隆地区患眼线虫病和朗格拉普岛上核残留的数据分析上，STAN-MCMC 取得了和贝叶斯 MCMC 算法几乎一样的效果，由于软件 Stan 本身具有的高可扩展性和适应性，STAN-MCMC 算法同样具有这些优点。

关键词：低秩近似，蒙特卡罗最大似然，马尔科夫链蒙特卡罗

Abstract

Effective and efficient algorithms of spatial generalized linear mixed effects models are always pursued by reseachers. The innovation of the thesis has three parts. One is to realize three kinds of parameter estimation methods of spatial generalized linear mixed effects model under the R language programming environment. They are low rank approximation, Monte Carlo Maximum Likelihood and Bayesian Markov Chain Monte Carlo (shorted by Bayes MCMC) algorithms, respectively. The second is the numerical simulation for comparing the advantages and disadvantages of above algorithms. The third is to implement Bayesian MCMC algorithm based on Stan software (referred to as STAN-MCMC), which also applied to analyze malaria data in the Gambia and loa loa data in Cameroon. In conclusion, STAN-MCMC has achieved performance of Bayesian MCMC algorithm almost. Furthermore, due to the high scalability and adaptability of Stan software, the STAN-MCMC algorithm also has these advantages.

Key words: low-rank, monte carlo maximum likelihood, markov chain monte carlo

目 录

1	绪论	1
1.1	研究意义	1
1.2	文献综述	1
1.3	论文结构	2
1.4	符号说明	2
2	基础知识	5
2.1	指数族	5
2.2	最小二乘估计	7
2.3	极大似然估计	8
2.4	随机过程的连续性和可微性	9
2.5	平稳高斯过程	9
2.6	修正的第三类贝塞尔函数	10
2.7	拉普拉斯近似	10
2.8	Jeffreys 先验分布	11
2.9	贝叶斯定理与先验分布	12
2.10	维数灾难与蒙特卡罗积分	13
2.11	采样器与 Stan	14
2.12	弱先验信息	17
2.13	方差缩减因子	17
3	统计模型	21
3.1	线性模型	21
3.2	广义线性模型	21
3.3	线性混合效应模型	22
3.4	广义线性混合效应模型	22
3.5	空间线性混合效应模型	22
3.6	空间广义线性混合效应模型	22
3.6.1	模型结构	22
3.6.2	协方差函数	23
3.6.3	模型识别	24
4	参数估计	27
4.1	极大似然估计	28

4.2	剖面似然估计	29
4.3	参数估计的算法.....	30
4.3.1	拉普拉斯近似算法.....	31
4.3.2	蒙特卡罗极大似然算法.....	34
4.3.3	贝叶斯 MCMC 算法	35
4.3.4	低秩近似算法.....	36
4.4	贝叶斯 STAN-MCMC 算法.....	38
4.4.1	算法提出的背景和意义.....	38
4.4.2	STAN-MCMC 算法实现过程.....	39
4.4.3	求解模型的 R 包.....	41
5	数值模拟	43
5.1	平稳空间高斯过程	43
5.1.1	一维平稳空间高斯过程.....	43
5.1.2	二维平稳空间高斯过程.....	45
5.2	空间广义线性混合效应模型	46
5.2.1	响应变量服从二项分布.....	46
5.2.2	响应变量服从泊松分布.....	47
6	数据分析	49
6.1	空间线性混合效应模型	49
6.2	喀麦隆及周边地区盘尾丝虫病的空间分布	52
6.3	朗格拉普岛放射性残留物的空间分布	54
6.4	冈比亚儿童疟疾流行强度的空间分布	56
7	总结展望	59
	参考文献.....	65
	致谢.....	67
	作者简介.....	69
	附录.....	71
	表格	71
	数据模拟	73
	7.0.1 模拟平稳高斯过程.....	73
	7.0.2 模拟空间广义线性混合效应模型.....	74

HMC 算法	75
--------------	----

1 绪论

空间统计的内容非常丰富，主要分为地质统计 (geostatistics)、离散空间变差 (discrete spatial variation) 和空间点过程 (spatial point processes) 三大块^[1]。地质统计这个术语最初来自南非的采矿业^[2]，并由 Georges Matheron 及其同事继承和发展，用以预测黄金的矿藏含量和质量，空间广义线性混合效应模型 (Spatial Generalized Linear Mixed Model, 简称 SGLMM) 在空间数据建模中具有重要的地位，除了采矿业，还可应用于卫生统计、气象统计和空间经济统计等领域，如分析地区范围内的疟疾分布，有限气象站点条件下，预测地区的污染物浓度分布等具有非常广泛的应用。

1.1 研究意义

空间广义线性混合效应模型有着广泛的应用，如评估岩心样本石油含量，分析核污染物浓度的空间分布^[3]，预测冈比亚儿童疟疾流行度的空间分布，冈比亚疟疾空间分布^[4:5]和喀麦隆及其周边地区的热带眼线虫流行病的空间分布^[6]，对热带疾病预防和控制项目提供决策支持^[7]。在热带地区，淋巴丝虫病和盘尾丝虫病是严峻的公共卫生问题，据世界卫生组织统计，在非洲撒哈拉以南、阿拉伯半岛和南美洲的 34 个国家约 2000 ~ 4000 万人感染河盲病^[8]。例如，喀麦隆中部省份 Loa loa 是导致河盲病的寄生虫，它感染强度与疾病流行度之间存在线性关系，即 Loa loa 感染强度越大流行度越高^[9]。1997 年，研究表明 Loa loa 流行度对应的高感染强度的临界值为 20%^[10]，而研究个体水平的感染情况与群体水平流行度之间的关系有助于大规模给药^[7]。

1.2 文献综述

研究估计空间广义线性混合效应模型参数的算法，实际上，SGLMM 是一种比较复杂的广义线性混合效应模型，其参数估计的计算方法是一个很大的挑战，一直以来都是研究的一个重点方向。

在贝叶斯方法框架下，Peter J. Diggle 等 (1998 年)^[3] 使用随机游走的 Metropolis 算法实现马尔科夫链蒙特卡罗获得模型参数的后验密度分布及后验量的估计值，

Paulo J. Ribeiro Jr. 和 Peter J. Diggle (2001 年)^[11] 提出 Langevin-Hastings 算法，相比于随机游走的 Metropolis 算法，取得了更好的效果。

Ole F Christensen (2004 年)^[12] 实现的蒙特卡罗极大似然算法，

Håvard Rue 等 (2009 年)^[13] 提出用高斯马尔科夫随机场近似平稳空间高斯过程，提出集成嵌套拉普拉斯算法。

在基于似然的框架下，

更加高效的算法或算法的实现可以更加快速准确有效地做疾病预防和医疗资源分配。

1994 年 Charles J. Geyer 在最大似然估计指数族中的参数时，用蒙特卡罗积分近

似参数的对数似然函数，提出并证明了蒙特卡罗最大似然 (MCML) 算法的收敛性和相应估计的相合性和渐近正态性^[14]，这为算法的应用打开了局面。2002 年 Hao Zhang 将 MCML 算法应用到 SGLMM 模型中，同样证明了参数估计的相合性和渐近正态性^[15]。1998 年 Peter J. Diggle 等人在贝叶斯方法下，将基于 Metropolis-Hastings 采样器的 MCMC 算法应用于 SGLMM 模型的参数估计，通过该模型和算法分析了南太平洋朗格拉普岛上的核残留分布情况（SGLMM 模型的响应变量服从泊松分布），北拉纳克郡和南坎布里亚郡的弯曲杆菌感染情况（SGLMM 模型的响应变量服从二项分布）^[3]。随后，似然估计的统计性质和随机模拟算法的收敛性和应用成为研究的重点，2004 年 Ole F Christensen 将 MCML 方法继续应用于朗格拉普岛的数据分析，但是添加了非空间的随机效应，取得了更好的拟合效果^[12]。2016 年 Peter J. Diggle 和 Emanuele Giorgi 在 SGLMM 模型的基础上进行了扩展，其一将响应变量的范围扩大到混合分布，其二在模型中使用了多源数据，组合了来源于随机和非随机的调查数据（即潜在有偏的数据），数据分别来自学校和社区的调查，以分析肯尼亚疟疾流行情况；其三将时间因素考虑进模型，建立时空广义线性混合效应模型，并分析了马拉维 2010 年 5 月至 2013 年 6 月的疟疾流行数据^[16]。

文献综述分三个方向：基于似然的频率学派，基于随机模拟的贝叶斯学派，计算和实践问题

1.3 论文结构

在下面的章节中，第 1 章介绍空间广义线性混合效应模型的相关背景和研究意义，综述了目前广泛使用的贝叶斯 MCMC 算法、MCML 算法研究现状。第 3 章回顾了一般线性模型到 SGLMM 模型的结构，指出了模型从简单到复杂的变化过程中的区别和联系，尤其是指出了模型求解的不同方法和基于 R 实现的软件包。第 4 章介绍了求解 SGLMM 模型的三类算法细节，分别是 MCML 算法、低秩近似算法和贝叶斯 MCMC 算法，并在贝叶斯 MCMC 算法的基础上，提出基于 Stan 程序库实现的贝叶斯 MCMC 算法，简记为贝叶斯 STAN-MCMC 方法。此外，还补充介绍了 Stan 程序及相关 R 包。第 5 章详细介绍了模拟的过程，比较了 MCML、Low-Rank、贝叶斯 MCMC 和贝叶斯 STAN-MCMC 算法的优劣，如算法的准确度、效率，还给出实际数据分析时的建议。第 6 章使用贝叶斯 STAN-MCMC 算法给出基于 SGLMM 模型的案例分析，分别是喀麦隆周边地区人群感染 Loa loa 的流行度分析和冈比亚 5 岁以下儿童感染疟疾的流行度分析。第 7 章总结论文的主要工作和结论，不足之处，以及有待研究的方向。

1.4 符号说明

重要说明：凡是论文中公式、图表与文字之间的距离明显变大，就是该在附近添加一两行话填充一下公式中加粗表示向量，如 $\mathbf{u}, \boldsymbol{\beta}$

参考 <https://stefvanbuuren.name/fimd/symbol-description.html>

1. 数学公式：随机过程 \mathcal{S} 大写字母 `\mathcal` 花体，
2. 随机向量 $\boldsymbol{\theta}, \boldsymbol{\beta}, \mathbf{x}, \mathbf{y}$ 向量，矩阵都大写加粗，将向量看作一维的矩阵，用加粗方式分别表示未知参数，
3. 模型回归参数向量 `\boldsymbolsymbol`，数据向量/矩阵 `\mathbfbf`，
4. 矩阵 \mathbf{X} 用大写字母加粗表示；
5. n 维实（复）数域 $\mathbb{R}^n, \mathbb{C}^2$ 大写字母空心体；
6. 期望方差 E, Var 用 `\mathsf` 表示
7. 正文：R 包和 LaTeX 包名加粗 **nlme**
8. 代码：等宽字体，输出 `#>`，代码注释 `#`
9. 转置 A^\top 和多元函数求二阶导 $Q''(\mathbf{s})$
10. 在贝叶斯环境下，术语混合效应模型用贝叶斯分层模型替换
11. 章引用 `#chap`：自定义名称节引用 `#sec`：自定义名称小节引用 `#subsec`：即都加上前缀章节的自定义名称要尽可能短，可读性差关系不大，因为引用次数不多且集中

论文当中出现的符号和相应的描述，及在论文中首次出现的位置，见表 1.1

表 1.1: 论文中出现的数学符号及说明

符号	描述	章. 节
\mathbf{y}	响应变量的观测向量	2.1
\mathbf{X}	协变量数据矩阵	2.1
$\boldsymbol{\beta}$	模型回归系数向量，如 $\boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2)$	2.1
$\boldsymbol{\theta}$	模型其它参数集，如 $\boldsymbol{\theta} = (\phi, \tau^2, \sigma^2)$	2.1
\top	向量或矩阵转置，如 $\mathbf{y}^\top, \mathbf{X}^\top$	2.1
$'$	撇号表示一元函数求导，如 $b'(\theta), b''(\theta)$	2.1
\mathbb{R}^n	n 维实数域	
$\mathcal{S}(x), x \in \mathbb{R}^2$	二维空间随机过程 $\mathcal{S}(x)$	
$E(\mathbf{Y})$	随机向量 \mathbf{Y} 的期望	
$\text{Var}(\mathbf{Y})$	随机向量 \mathbf{Y} 的方差	
$\text{Cov}(\mathbf{X}, \mathbf{Y})$	随机向量 \mathbf{X} 与 \mathbf{Y} 的协方差	

2 基础知识

本章概要

2.1 指数族

一般地, 样本 \mathbf{Y} 的分布服从指数族, 即形如

$$f_Y(y; \theta, \phi) = \exp \left\{ (y\theta - b(\theta)) / a(\phi) + c(y, \phi) \right\} \quad (2.1)$$

其中, $a(\cdot), b(\cdot), c(\cdot)$ 是某些特定的函数。如果 ϕ 已知, 这是一个含有典则参数 θ 的指数族模型, 如果 ϕ 未知, 它可能是含有两个参数的指数族。对于正态分布

$$\begin{aligned} f_Y(y; \theta, \phi) &= \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left\{ -\frac{(y - \mu)^2}{2\sigma^2} \right\} \\ &= \exp \left\{ (y\mu - \mu^2/2) / \sigma^2 - \frac{1}{2} (y^2/\sigma^2 + \log(2\pi\sigma^2)) \right\} \end{aligned} \quad (2.2)$$

通过与 (2.1) 式对比, 可知 $\theta = \mu$, $\phi = \sigma^2$, 并且有

$$a(\phi) = \phi, \quad b(\theta) = \theta^2/2, \quad c(y, \phi) = -\frac{1}{2} \{ y^2/\sigma^2 + \log(2\pi\sigma^2) \}$$

记 $l(\theta, \phi; y) = \log f_Y(y; \theta, \phi)$ 为给定样本点 y 的情况下, 关于 θ 和 ϕ 的对数似然函数。样本分布 Y 的均值和方差具有如下关系

$$E\left(\frac{\partial l}{\partial \theta}\right) = 0 \quad (2.3)$$

和

$$E\left(\frac{\partial^2 l}{\partial \theta^2}\right) + E\left(\frac{\partial l}{\partial \theta}\right)^2 = 0 \quad (2.4)$$

从 (2.1) 式知

$$l(\theta, \phi; y) = y\theta - b(\theta)/a(\phi) + c(y, \phi)$$

因此,

$$\begin{aligned} \frac{\partial l}{\partial \theta} &= y - b'(\theta)/a(\phi) \\ \frac{\partial^2 l}{\partial \theta^2} &= -b''(\theta)/a(\phi) \end{aligned} \quad (2.5)$$

从 (2.3) 式和 (2.5), 可以得出

$$0 = E\left(\frac{\partial l}{\partial \theta}\right) = \{\mu - b'(\theta)\}/a(\phi)$$

所以

$$E(Y) = \mu = b'(\theta)$$

根据 (2.4) 式和 (2.5) 式，可得

$$0 = -\frac{b''(\theta)}{a(\phi)} + \frac{\text{Var}(Y)}{a^2(\phi)}$$

所以

$$\text{Var}(Y) = b''(\theta)a(\phi)$$

可见， Y 的方差是两个函数的乘积，一个是 $b''(\theta)$ ，它仅仅依赖典则参数，被叫做方差函数，另一个是 $a(\phi)$ ，它独立于 θ ，仅仅依赖 ϕ ，方差函数可以看作是 μ 的函数，记作 $V(\mu)$ 。

函数 $a(\phi)$ 通常形如

$$a(\phi) = \phi/w$$

其中 ϕ 可由 σ^2 表示，故而也叫做发散参数 (dispersion parameter)，是一个与样本观察值相关的常数， w 是已知的权重，随样本观察值变化。对正态分布模型而言， w 的分量是 m 个相互独立的样本观察值的均值，我们有

$$a(\phi) = \sigma^2/m$$

所以， $w = m$ 。

根据 (2.1) 式，正态、泊松和二项分布的特征见表 2.1，其它常见分布见 Peter McCullagh 等 (1989 年)^[17]。

表 2.1: 指数族内常见的一元分布的共同特征及符号表示¹

	正态分布	泊松分布	二项分布
记号	$N(\mu, \sigma^2)$	$P(\mu)$	$B(m, \pi)/m$
y 取值范围	$(-\infty, \infty)$	$0(1)\infty$	$\frac{0(1)m}{m}$
ϕ	$\phi = \sigma^2$	1	$1/m$
$b(\theta)$	$\theta^2/2$	$\exp(\theta)$	$\log(1 + e^\theta)$
$c(y; \theta)$	$-\frac{1}{2}\left(\frac{y^2}{\phi} + \log(2\pi\phi)\right)$	$-\log(y!)$	$\log\left(\frac{m}{my}\right)$

	正态分布	泊松分布	二项分布
$\mu(\theta) = E(Y; \theta)$	θ	$\exp(\theta)$	$e^\theta / (1 + e^\theta)$
联系函数: $\theta(\mu)$	identity	log	logit
方差函数: $V(\mu)$	1	μ	$\mu(1 - \mu)$

2.2 最小二乘估计

考虑如下线性模型的最小二乘估计

$$EY = X\beta; \text{Var}(Y) = \sigma^2 I_n \quad (2.6)$$

其中, Y 为 $n \times 1$ 维观测向量, X 为已知的 $n \times p (p \leq n)$ 阶设计矩阵, β 为 $p \times 1$ 维未知参数, σ^2 未知, I_n 为 n 阶单位阵。

定义 2.1 (最小二乘估计). 在模型 (2.6) 中, 如果

$$(Y - X\hat{\beta})^\top (Y - X\hat{\beta}) = \min_{\beta} (Y - X\beta)^\top (Y - X\beta) \quad (2.7)$$

则称 $\hat{\beta}$ 为 β 的最小二乘估计 (Least Squares Estimate, 简称 LSE)。

定理 2.1 (最小二乘估计). 若模型 (2.6) 中的 X 是列满秩的矩阵, 则 β 的最小二乘估计为

$$\hat{\beta}_{LS} = (X^\top X)^{-1} X^\top Y, \quad \text{Var}(\hat{\beta}_{LS}) = \sigma^2 (X^\top X)^{-1}$$

σ^2 的最小二乘估计为

$$\hat{\sigma}_{LS}^2 = (Y - X\hat{\beta}_{LS})^\top (Y - X\hat{\beta}_{LS}) / (n - p)$$

若将模型 (2.6) 的条件 $\text{Var}(Y) = \sigma^2 I_n$ 改为 $\text{Var}(Y) = \sigma^2 G$, $G(> 0)$ 为已知正定阵, 则 β 的最小二乘估计为

$$\tilde{\beta}_{LS} = (X^\top G^{-1} X)^{-1} X^\top G^{-1} Y$$

称 $\tilde{\beta}_{LS}$ 为广义最小二乘估计 (Generalized Least Squares Estimate, 简称 GLSE), 特别地, 当 $G = \text{diag}(\sigma_1^2, \dots, \sigma_n^2)$, $\sigma_i^2, i = 1, \dots, n$ 已知时, 称 $\tilde{\beta}_{LS}$ 为加权最小二乘估计 (Weighted Least Squares Estimate, 简称 WLSE)^[18]

¹均值参数用 μ 表示, 二项分布里用 π 表示; 典则参数用 θ 表示, 定义见 (2.1) 式, μ 和 θ 的关系在表 2.1 的第 6 和第 7 行给出。

2.3 极大似然估计

定义 2.2 (极大似然估计). 设 $p(\mathbf{x}; \boldsymbol{\theta})$, $\boldsymbol{\theta} \in \Theta$ 是 $(\mathbb{R}^n, \mathcal{P}_{\mathbb{R}^n})$ 上的一族联合密度函数, 对给定的 \mathbf{x} , 称

$$L(\boldsymbol{\theta}; \mathbf{x}) = kp(\mathbf{x}; \boldsymbol{\theta})$$

为 $\boldsymbol{\theta}$ 的似然函数, 其中 $k > 0$ 是不依赖于 $\boldsymbol{\theta}$ 的量, 常取 $k = 1$ 。进一步, 若存在 $(\mathbb{R}^n, \mathcal{P}_{\mathbb{R}^n})$ 到 $(\Theta, \mathcal{P}_{\Theta})$ 的统计量 $\hat{\boldsymbol{\theta}}(\mathbf{x})$ 使

$$L(\hat{\boldsymbol{\theta}}(\mathbf{x}); \mathbf{x}) = \sup_{\boldsymbol{\theta}} L(\boldsymbol{\theta}; \mathbf{x})$$

则 $\hat{\boldsymbol{\theta}}(\mathbf{x})$ 称为 $\boldsymbol{\theta}$ 的一个极大似然估计 (Maximum Likelihood Eestimate, 简称 MLE)。

概率密度函数很多可以写成具有指数函数的形式, 如指数族, 采用似然函数的对数通常更为简便。称

$$l(\boldsymbol{\theta}, \mathbf{x}) = \ln L(\boldsymbol{\theta}, \mathbf{x})$$

为 $\boldsymbol{\theta}$ 的对数似然函数。对数变换是严格单调的, 所以 $l(\boldsymbol{\theta}, \mathbf{x})$ 与 $L(\boldsymbol{\theta}, \mathbf{x})$ 的极大值是等价的。当 MLE 存在时, 寻找 MLE 的常用方法是求导数。如果 $\hat{\boldsymbol{\theta}}(\mathbf{x})$ 是 Θ 的内点, 则 $\hat{\boldsymbol{\theta}}(\mathbf{x})$ 是下列似然方程组

$$\partial l(\boldsymbol{\theta}, \mathbf{x}) / \partial \theta_i = 0, \quad i = 1, \dots, m \quad (2.8)$$

的解。 $p(\mathbf{x}; \boldsymbol{\theta})$ 属于指数族时, 似然方程组 (2.8) 的解唯一。

定理 2.2 (相合性). 设 x_1, \dots, x_n 是来自概率密度函数 $p(x; \theta)$ 的一个样本, 叙述简单起见, 考虑单参数情形, 参数空间 Θ 是一个开区间, $l(\theta; x) = \sum_{i=1}^n \ln p(x_i; \theta)$ 。

若 $\ln(p; \theta)$ 在 Θ 上可微, 且 $p(x; \theta)$ 是可识别的 (即 $\forall \theta_1 \neq \theta_2, \{x : p(x; \theta_1) \neq p(x; \theta_2)\}$ 不是零测集), 则似然方程 (2.8) 在 $n \rightarrow \infty$ 时, 以概率 1 有解, 且此解关于 θ 是相合的。

定理 2.3 (渐近正态性). 假设 Θ 为开区间, 概率密度函数 $p(x; \theta), \theta \in \Theta$ 满足

1. 在参数真值 θ_0 的邻域内, $\partial \ln p / \partial \theta, \partial^2 \ln p / \partial \theta^2, \partial^3 \ln p / \partial \theta^3$ 对所有的 x 都存在;
2. 在参数真值 θ_0 的邻域内, $|\partial^3 \ln p / \partial \theta^3| \leq H(x)$, 且 $\mathbb{E}H(x) < \infty$;
3. 在参数真值 θ_0 处,

$$\mathbb{E}_{\theta_0} \left[\frac{p'(x, \theta_0)}{p(x, \theta_0)} \right] = 0, \quad \mathbb{E}_{\theta_0} \left[\frac{p''(x, \theta_0)}{p(x, \theta_0)} \right] = 0, \quad I(\theta_0) = \mathbb{E}_{\theta_0} \left[\frac{p'(x, \theta_0)}{p(x, \theta_0)} \right]^2 > 0 \quad (2.9)$$

其中撇号表示对 θ 的微分。记 $\hat{\theta}_n$ 为 $n \rightarrow \infty$ 时，似然方程组的相合解，则

$$\sqrt{n}(\hat{\theta}_n - \theta_0) \longrightarrow \mathcal{N}(\mathbf{0}, I^{-1}(\theta))$$

2.4 随机过程的连续性和可微性

为记号简便起见，考虑一维空间下，随机过程 $S(x)$ 的均方连续性和可微性。

定义 2.3 (连续性). 随机过程 $S(x)$ 满足

$$\lim_{h \rightarrow 0} \mathbb{E}[\{S(x+h) - S(x)\}^2] = 0$$

则称 $S(x)$ 是均方连续 (mean-square continuous) 的。

定义 2.4 (可微性). 随机过程 $S(x)$ 满足

$$\lim_{h \rightarrow 0} \mathbb{E}\left[\left\{\frac{S(x+h) - S(x)}{h} - S'(x)\right\}^2\right] = 0$$

则称 $S(x)$ 是均方可微的，并且 $S'(x)$ 就是均方意义下的一阶导数。如果 $S'(x)$ 是均方可微的，则 $S(x)$ 是二次均方可微的，随机过程 $S(x)$ 的高阶均方可微性可类似定义。

Bartlett (1955 年)^[19] 得到如下结论

定理 2.4 (平稳随机过程的可微性). 自相关函数为 $\rho(u)$ 的平稳随机过程是 k 次均方可微 (mean-square differentiable) 的，当且仅当 $\rho(u)$ 在 $u = 0$ 处是 $2k$ 次可微的。

2.5 平稳高斯过程

一般地，空间高斯过程 $\mathcal{S} = \{S(x), x \in \mathbb{R}^2\}$ 必须满足条件：任意给定一组空间位置 $x_1, x_2, \dots, x_n, \forall x_i \in \mathbb{R}^2$ ，每个位置上对应的随机变量 $S(x_i), i = 1, 2, \dots, n$ 的联合分布 $\mathcal{S} = \{S(x_1), S(x_2), \dots, S(x_n)\}$ 是多元高斯分布，其由均值 $\mu(x) = \mathbb{E}[S(x)]$ 和协方差 $G_{ij} = \gamma(x_i, x_j) = \text{Cov}\{S(x_i), S(x_j)\}$ 完全确定，即 $\mathcal{S} \sim \text{MVN}(\mu_S, G)$

平稳空间高斯过程需要空间高斯过程满足平稳性条件：其一， $\mu(x) = \mu, \forall x \in \mathbb{R}^2$ ，其二，自协方差函数 $\gamma(x_i, x_j) = \gamma(u), u = \|x_i - x_j\|$ 。可见均值 μ 是一个常数，而自协方差函数 $\gamma(x_i, x_j)$ 只与空间距离有关。注意到平稳高斯过程 \mathcal{S} 的方差是一个常数，即 $\sigma^2 = \gamma(0)$ ，然后可以定义自相关函数 $\rho(u) = \gamma(u)/\sigma^2$ ，并且 $\rho(u)$ 满足对称性， $\rho(u) = \rho(-u)$ ，因为对 $\forall u, \text{Corr}\{S(x), S(x-u)\} = \text{Corr}\{S(x-u), S(x)\} = \text{Corr}\{S(x), S(x+u)\}$ ，这里的第二个等式是根据平稳性得来的，根据协方差的定义不难验证。在本论文中如果不特别说明，平稳就指上述协方差意义下的平稳，这种平稳性条件广泛应用于空间统计数据建模。

2.6 修正的第三类贝塞尔函数

空间过程的协方差函数是梅隆族时，需要用到修正的第三类贝塞尔函数 $\mathcal{K}_\kappa(u)$ ，它是修正的贝塞尔方程的解^[20]，函数形式如下

$$\begin{aligned} I_{-\kappa}(u) &= \sum_{m=0}^{\infty} \frac{1}{m! \Gamma(m + \kappa + 1)} \left(\frac{u}{2}\right)^{2m+\kappa} \\ \mathcal{K}_\kappa(u) &= \frac{\pi}{2} \frac{I_{-\kappa}(u) - I_\kappa(u)}{\sin(\kappa\pi)} \end{aligned} \quad (2.10)$$

其中 $u \geq 0$, $\kappa \in \mathbb{R}$ ，如果 $\kappa \in \mathbb{Z}$ ，则取该点的极限值，实际上 R 内置的函数 `besselK` 可以计算 $\mathcal{K}_\kappa(u)$ ^[21]

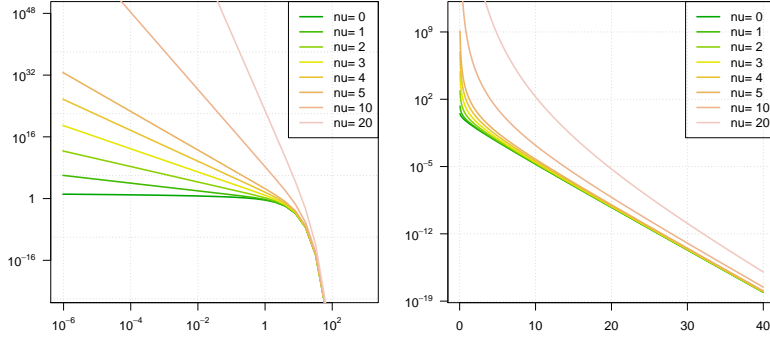


图 2.1: 贝塞尔函数图像

2.7 拉普拉斯近似

先回顾一下基本的泰勒展开，一个函数可以在点 a 处展开成和的形式，有时候是无穷多项，可以使用其中的有限项最为近似，通常会使用前三项，即到达二阶导的位置。

$$f(x) = f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f'''(a)}{3!}(x-a)^3 + \dots$$

以基本的抛物线为例， $f(x) = x^2$ ，在 $a = 2$ 处展开

$$f(x) = x^2, \quad f'(x) = 2x, \quad f''(x) = 2, \quad f'''(x) = 0$$

因此，

$$f(x) = x^2 = 2^2 + 2(2)(x-2) + \frac{2}{2}(x-2)^2$$

拉普拉斯近似用正态分布来估计任意分布，它使用泰勒展开的前三项近似 $\log g(x)$ ，

展开的位置是 \hat{x} , 则

$$\log g(x) \approx \log g(\hat{x}) + \frac{\partial \log g(\hat{x})}{\partial x}(x - \hat{x}) + \frac{\partial^2 \log g(\hat{x})}{2\partial x^2}(x - \hat{x})^2$$

在函数 $g(x)$ 的极值点 \hat{x} 展开, $x = \hat{x}$ 一阶导是 0, 用曲率去估计方差是 $\hat{\sigma}^2 = -1/\frac{\partial^2 \log g(\hat{x})}{2\partial x^2}$, 再重写上述近似

$$\log g(x) \approx \log g(\hat{x}) - \frac{1}{2\hat{\sigma}^2}(x - \hat{x})^2$$

现在, 用这个结果做正态近似, 将上式两端取指数和积分, 移去常数项

$$\int g(x)dx = \int \exp[\log g(x)]dx \approx \text{constant} \int \exp[-\frac{(x - \hat{x})^2}{2\hat{\sigma}^2}]dx$$

拉普拉斯方法用正态分布近似分布 $f(x)$, 其均值 \hat{x} , 可以通过求解 $f'(x) = 0$ 获得, 方差 $\hat{\sigma}^2 = -1/f''(\hat{x})$

以卡方分布 χ^2 为例,

$$f(x; k) = \frac{x^{k/2-1}e^{-x/2}}{2^{k/2}\Gamma(k/2)}, x \geq 0$$

$$\log f(x) = (k/2 - 1) \log x - x/2$$

$$\log f'(x) = (k/2 - 1)/x - 1/2 = 0$$

$$\log f''(x) = -(k/2 - 1)/x^2$$

所以

$$\chi_k^2 \stackrel{LA}{\sim} N(\hat{x} = k - 2, \hat{\sigma}^2 = 2(k - 2))$$

自由度越大, 近似效果越好, 对于多元分布的情况不难推广, 使用多元泰勒展开和黑塞矩阵即可表示。并且参数集 θ 有唯一的极大值点 $\hat{\theta}$ ^[22]

2.8 Jeffreys 先验分布

设 $\mathbf{x} = (x_1, \dots, x_n)$ 是来自密度函数 $p(x|\theta)$ 的一个样本, 其中 $\theta = (\theta_1, \dots, \theta_p)$ 是 p 维参数向量。在对 θ 无任何先验信息可用时, Jeffreys (1961 年) 利用变换群和 Harr 测度导出 θ 的无信息先验分布可用 Fisher 信息阵的行列式的平方根表示。这种无信息先验分布常称为 Jeffreys 先验分布。其求取步骤如下:

1. 写出样本的对数似然函数 $l(\theta|x) = \sum_{i=1}^n \ln p(x_i|\theta)$;

2. 算出参数 θ 的 Fisher 信息阵

$$\mathbf{I}(\theta) = \mathbb{E}_{x|\theta} \left(- \frac{\partial^2 l}{\partial \theta_i \partial \theta_j} \right)_{i,j=1,\dots,p}$$

在单参数场合， $\mathbf{I}(\theta) = \mathbb{E}_{x|\theta} \left(- \frac{\partial^2 l}{\partial \theta^2} \right)$;

3. θ 的无信息先验密度函数为 $\pi(\theta) = [\det \mathbf{I}(\theta)]^{1/2}$ ，在单参数场合， $\pi(\theta) = [\mathbf{I}(\theta)]^{1/2}$

2.9 贝叶斯定理与先验分布

非信息先验分布，扁平先验 flat prior，模糊先验

以标准线性模型为例介绍贝叶斯分析及其基本概念^[23]，为什么不用 RMSE 均方误差，WAIC pDIC 模型选择 loo K-CV

贝叶斯定理 2.2 贝叶斯定理 (2.11)

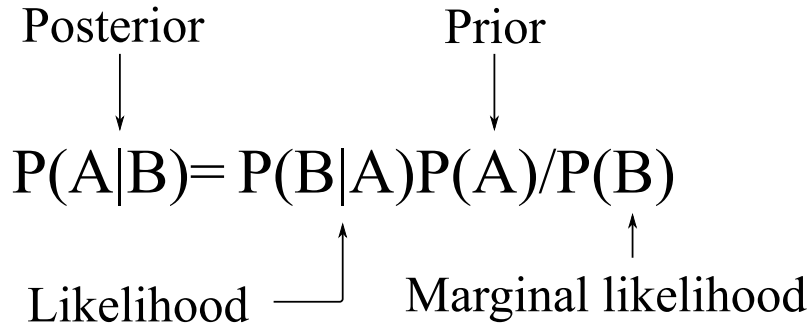


图 2.2: 贝叶斯定理

作为铺垫，先结合 SGLMM 模型介绍一下贝叶斯定理，其中， θ 代表 SGLMM 模型中的参数， \mathbf{Y} 是响应变量对应的观察值。

$$\begin{aligned}
 p(\theta|\mathbf{Y}) &= \frac{p(\theta, \mathbf{Y})}{p(\mathbf{Y})} && \text{[条件概率定义]} \\
 &= \frac{p(\mathbf{Y}|\theta)p(\theta)}{p(\mathbf{Y})} && \text{[链式法则]} \\
 &= \frac{p(\mathbf{Y}|\theta)p(\theta)}{\int_{\Theta} p(\mathbf{Y}, \theta) d\theta} && \text{[全概率公式]} \\
 &= \frac{p(\mathbf{Y}|\theta)p(\theta)}{\int_{\Theta} p(\mathbf{Y}|\theta)p(\theta) d\theta} && \text{[链式法则]} \\
 &\propto p(\mathbf{Y}|\theta)p(\theta) && \text{[Y 已知]}
 \end{aligned} \tag{2.11}$$

2.10 维数灾难与蒙特卡罗积分

一般地，混合效应模型的统计推断总是不可避免的要面对高维积分，处理高维积分的方法一个是寻找近似方法避免求积分，一个是寻找有效的随机模拟方法直接求积分。这里，介绍蒙特卡罗方法求积分，以计算 N 维超立方体的内切球的体积为例说明。

假设我们有一个 N 维超立方体，其中心在坐标 $\mathbf{0} = (0, \dots, 0)$ 。超立方体在点 $(\pm 1/2, \dots, \pm 1/2)$ ，有 2^N 个角落，超立方体边长是 1， $1^N = 1$ ，所以它的体积是 1。

如果 $N = 1$ ，超立方体是一条从 $-\frac{1}{2}$ 到 $\frac{1}{2}$ 的单位长度的线，如果 $N = 2$ ，超立方体是一个单位正方形，对角是 $(-\frac{1}{2}, -\frac{1}{2})$ 和 $(\frac{1}{2}, \frac{1}{2})$ ，如果 $N = 3$ ，超立方体就是单位体积的立方体，对角是 $(-\frac{1}{2}, -\frac{1}{2}, -\frac{1}{2})$ 和 $(\frac{1}{2}, \frac{1}{2}, \frac{1}{2})$ ，依此类推， N 维超立方体体积是 1，对角是 $(-\frac{1}{2}, \dots, -\frac{1}{2})$ 和 $(\frac{1}{2}, \dots, \frac{1}{2})$ 。

现在，考虑 N 维超立方体的内切球，我们把它称为 N 维超球，它的中心在原点，半径是 $\frac{1}{2}$ 。我们说点 y 在超球内，意味着它到原点的距离小于半径，即 $\|y\| < \frac{1}{2}$ 。

一维情形下，超球是从的线，包含了整个超立方体。二维情形下，超球是中心在原点，半径为 $\frac{1}{2}$ 的圆。三维情形下，超球是立方体的内切球。

我们知道单位超立方体的体积是 1，但是其内的内切球的体积是多少呢？我们已经学过如何去定义一个积分计算半径为 r 的二维球（即圆）的体积（即面积）是 πr^2 ，三维情形下，内切球是 $\frac{4}{3}\pi r^3$ 。但是更高维的欧式空间里，内切球的体积是多少呢？

我们当然可以去计算越来越复杂的多重积分，但是这里我们介绍采样的方法去计算积分，即所谓的蒙特卡罗方法，由乌拉姆 (S. Ulam)、冯·诺依曼 (J. von Neumann) 和梅特罗波利斯 (N. Metropolis) 等在美国核武器研究实验室创立，当时正值二战期间，为了研制原子弹，出于保密的需要，与随机模拟相关的技术就代号蒙特卡罗。现在，蒙特卡罗方法占据现代统计计算的核心地位，特别是与贝叶斯相关的领域。

用蒙特卡罗方法去计算单位超立方体内的超球，首先我们需要在单位超立方体内产生随机点，然后计算落在超球内点的比例，即超球的体积。随着点的数目增加，估计的体积会收敛到真实的体积。因为这些点都独立同均匀分布，根据中心极限定理，误差下降的比率是 $\mathcal{O}(1/\sqrt{n})$ ，这也意味着每增加一个小数点的准确度，样本量要增加 100 倍。

表 2.2 列出了前 10 维超球的体积，随着维数的增加，超球的体积迅速变小，超立方体内随机点的个数是 100000。这里有一个反直观的现象，内切球的体积竟然随着维数的增加变小，并且在 10 维的情形下，内切球的体积已不到超立方体的 0.3%。

表 2.2: 前 10 维单位超立方体内切球的体积（已经四舍五入保留小数点后三位）

维数	1	2	3	4	5	6	7	8	9	10
体积	1.000	0.784	0.525	0.307	0.166	0.081	0.037	0.016	0.006	0.0027

2.11 采样器与 Stan

随机模拟的基础是有高质量的伪随机数，如何生成和检验伪随机数的质量参见黄湘云的文章^[24]。通过随机模拟的方式从总体中获取样本，需要一个抽样（也叫采样）的过程，不同的采样算法（也叫采样器）在适用范围和采样效率方面有不同。在贝叶斯计算中，常用的采样器有 Gibbs, Metropolis 和汉密尔顿蒙特卡罗 (Hamiltonian Monte Carlo, 简称 HMC) 三类。

Matthew D. Hoffman 和 Andrew Gelman (2014 年)^[25] 提出的 No-U-Turn 采样器属于 HMC 方法衍生的采样器。

Stan 是一门基于 C++ 的高级编程语言，用户只需提供数据、模型和参数初值，目标后验分布的 Markov 链的模拟过程是自动实现的。除了可以完全在 Stan 脚本中写模型外，Stan 还提供其他编程语言的接口，如 R, Python 和 MATLAB 等，使得熟悉其他编程语言的用户也可以比较方便地调用。与 Python、R 这类解释型编程语言不同，Stan 代码需要先翻译成 C++ 代码，然后编译执行。

Donald B. Rubin (1981 年)^[26] 分析了 Donald L. Alderman 和 Donald E. Powers^[27] 收集的原始数据，得出表 2.3，Andrew Gelman 和 John B. Carlin 等 (2003 年)^[28] 建立分层正态模型 (2.12) 分析 Eight Schools 数据集，这里再次以该数据集和模型为例介绍 Stan 的使用和算法实现。

$$\begin{aligned}
 \mu &\sim \mathcal{N}(0, 5) \\
 \tau &\sim \text{Half-Cauchy}(0, 5) \\
 p(\mu, \tau) &\propto 1 \\
 \eta_i &\sim \mathcal{N}(0, 1) \\
 \theta_i &= \mu + \tau \cdot \eta_i \\
 y_i &\sim \mathcal{N}(\theta_i, \sigma_i^2)
 \end{aligned} \tag{2.12}$$

由美国教育考试服务调查搜集，用以分析不同的辅导项目对学生考试分数的影响，调查结果用来帮助高校招生。分别随机调查了 8 所高中，输出变量是一个分数，培训效应的估计 y_j ，其样本方差 σ_j^2 ，数据集见表 2.3。

表 2.3: Eight Schools 数据集

School	A	B	C	D	E	F	G	H
y_i	28	8	-3	7	-1	1	18	12
σ_i	15	10	16	11	9	11	10	18

分层正态模型可以在 Stan 中写成如下形式，在工作目录下把它保存为

8schools.stan

```
// saved as 8schools.stan
data {
  int<lower=0> J; // number of schools
  real y[J]; // estimated treatment effects
  real<lower=0> sigma[J]; // s.e. of effect estimates
}
parameters {
  real mu; // population mean
  real<lower=0> tau; // population sd
  real eta[J]; // school-level errors
}
transformed parameters {
  real theta[J]; // schools effects
  for (j in 1:J)
    theta[j] = mu + tau * eta[j];
  // theta = mu + tau*eta;
}
model {
  // set prior for mu or uniform prior distribution default
  // target += normal_lpdf(mu | 0, 10);
  // target += cauchy_lpdf(tau | 0, 25); # the same as mu
  target += normal_lpdf(eta | 0, 1);
  target += normal_lpdf(y | theta, sigma); // target distribution
  // y ~ normal(theta, sigma);
}
```

上述 Stan 代码的第一段提供数据：学校的数目 J ，估计值 y_1, \dots, y_J ，标准差 $\sigma_1, \dots, \sigma_J$ ，数据类型可以是整数、实数，结构可以是向量，或更一般的数组，还可以带约束，如在这个模型中 J 限制为非负， σ_J 必须是正的，另外两个反斜杠 `//` 表示注释。

第二段代码声明参数：模型中的待估参数，学校总体的效应 θ_j ，均值 μ ，标准差 τ ，学校水平上的误差 η 和效应 θ 。在这个模型中，用 μ, τ, η 表示 θ 而不是直接声明 θ 作为一个参数，通过这种参数化，采样器的运行效率会提高，还应该尽量使用向量化操作代替 for 循环语句。

最后一段是模型：稍微注意的是，正文中正态分布 $N(\cdot, \cdot)$ 中后一个位置是方差，而 Stan 代码中使用的是标准差。`target += normal_lpdf(y | theta, sigma)` 和 $y \sim \text{normal}(\text{theta}, \text{sigma})$ 对模型的贡献是一样的，都使用正态分布的对数概率密度函数，只是后者扔掉了对数密度函数的常数项而已，这对于 Stan 的采样、近似或优化算法没有影响^[29]。

算法运行的硬件环境是 16 核 32 线程主频 2.8 GHz 英特尔至强 E5-2680 处理器，系统环境 CentOS 7，R 软件版本 3.5.1，RStan 版本 2.17.3。

HMC 算法参数主要设置了 4 条链，每条链迭代 10000 次，为复现模型结果随机数种子设为 2018

表 2.4: 对 Eight Schools 数据集建立分层正态模型 (2.12)，采用 HMC 算法估计模型参数值

	mean	se_mean	sd	2.5%	25%	50%	75%	97.5%	n_eff	Rhat
μ	7.99	0.05	5.02	-1.65	4.75	7.92	11.15	18.10	8455	1
τ	6.47	0.06	5.44	0.22	2.45	5.18	9.07	20.50	7375	1
η_1	0.40	0.01	0.93	-1.49	-0.21	0.42	1.02	2.19	16637	1
η_2	0.00	0.01	0.87	-1.73	-0.58	0.00	0.57	1.70	16486	1
η_3	-0.20	0.01	0.93	-1.99	-0.82	-0.20	0.41	1.66	20000	1
η_4	-0.04	0.01	0.88	-1.80	-0.60	-0.04	0.53	1.74	20000	1
η_5	-0.36	0.01	0.88	-2.06	-0.94	-0.38	0.20	1.42	15489	1
η_6	-0.22	0.01	0.90	-1.96	-0.82	-0.23	0.37	1.57	20000	1
η_7	0.34	0.01	0.89	-1.49	-0.24	0.36	0.93	2.04	16262	1
η_8	0.05	0.01	0.94	-1.81	-0.57	0.06	0.69	1.91	20000	1
θ_1	11.45	0.08	8.27	-1.86	6.07	10.27	15.50	31.68	11788	1
θ_2	7.93	0.04	6.15	-4.45	3.99	7.90	11.74	20.44	20000	1
θ_3	6.17	0.06	7.67	-11.17	2.07	6.74	10.89	19.94	16041	1
θ_4	7.66	0.05	6.51	-5.63	3.75	7.72	11.62	20.78	20000	1
θ_5	5.13	0.05	6.41	-9.51	1.37	5.66	9.43	16.41	20000	1
θ_6	6.14	0.05	6.66	-8.63	2.35	6.58	10.40	18.47	20000	1
θ_7	10.64	0.05	6.76	-1.14	6.11	10.11	14.52	25.88	20000	1
θ_8	8.42	0.06	7.86	-7.24	3.91	8.26	12.60	25.24	16598	1
lp__	-39.55	0.03	2.64	-45.41	-41.15	-39.31	-37.67	-35.12	6325	1

表 2.4 的列为后验量的估计值：依次是均值 $E(\mu|Y)$ 、标准误 (standard error) $\text{Var}(\mu|Y)$ 、标准差 (standard deviation) $E(\sigma|Y)$ 、后验分布的 5 个分位点、有效样本数 n_{eff} 和潜在尺度缩减因子 (potential scale reduction factor)，最后两个量用来分析采

样效率；最后一行表示每次迭代的未正则的对数后验密度 (unnormalized log-posterior density) \hat{R} ，当链条都收敛到同一平稳分布的时候， \hat{R} 接近 1。

这里我们对 τ 采用的非信息先验分布是均匀先验，参数 τ 的 95% 的置信区间是 (0.22, 20.5)，数据支持 τ 的范围低于

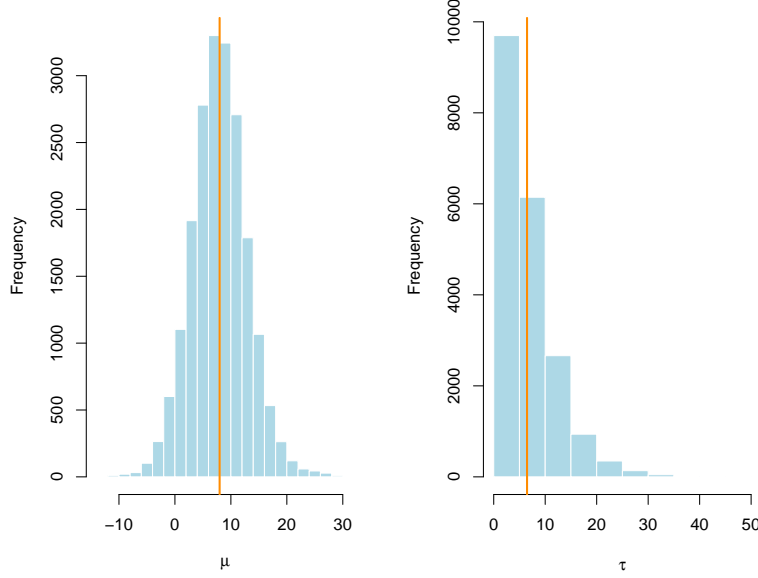


图 2.3: 对 μ, τ 给定均匀先验，后验均值 μ 和标准差 τ 的直方图

为了得到可靠的后验估计，做出合理的推断，诊断序列的几何遍历性是必不可少的部分

2.12 弱先验信息

2.13 方差缩减因子

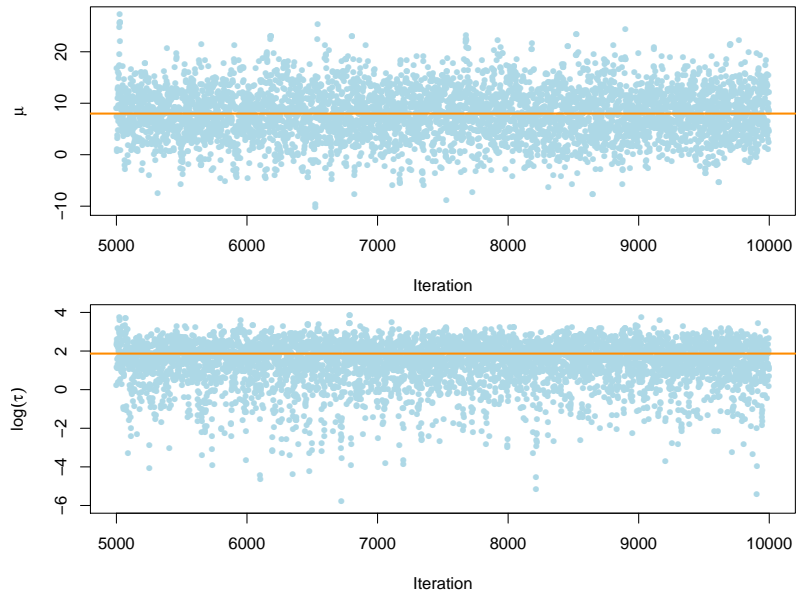
为了评估链条之间和内部的混合效果，我们引入潜在尺度缩减因子 \hat{R} ，对每个待估的量 ω ，模拟产生 m 条链，每条链有 n 次迭代值 $\omega_{ij} (i = 1, \dots, n; j = 1, \dots, m)$ ，用 B 和 W 分别表示链条之间（不妨看作组间方差）和内部的方差（组内方差）

$$\begin{aligned} B &= \frac{n}{m-1} \sum_{j=1}^m (\bar{\omega}_{\cdot j} - \bar{\omega}_{\cdot\cdot}), \quad \bar{\omega}_{\cdot j} = \frac{1}{n} \sum_{i=1}^n \omega_{ij}, \quad \bar{\omega}_{\cdot\cdot} = \frac{1}{m} \sum_{j=1}^m \bar{\omega}_{\cdot j} \\ W &= \frac{1}{m} \sum_{j=1}^m s_j^2, \quad s_j^2 = \frac{1}{n-1} \sum_{i=1}^n (\omega_{ij} - \bar{\omega}_{\cdot j})^2 \end{aligned} \quad (2.13)$$

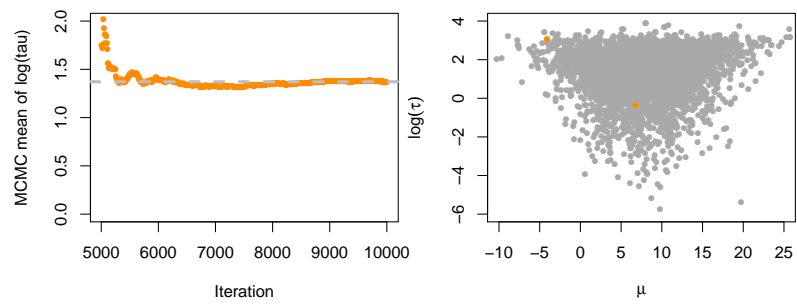
ω 的边际后验方差 $\omega|Y$ 是 W 和 B 的加权平均

$$\widehat{\text{Var}}^+(\omega|Y) = \frac{n-1}{n} W + \frac{1}{n} B \quad (2.14)$$

当初始分布发散 (overdispersed) 时，这个量会高估边际后验方差，但在链条平稳或



(a) 诊断序列的平稳性



(b) 蒙特卡罗均值误差和发散点

图 2.4: 诊断图

$n \rightarrow \infty$ 时，它是无偏的。同时，对任意有限的 n ，组内方差 W 应该会低估 $\text{Var}(\omega|Y)$ ，因为单个链条没有时间覆盖目标分布；在 $n \rightarrow \infty$ ， W 的期望会是 $\text{Var}(\omega|Y)$ 。

我们通过估计潜在尺度缩减因子 \hat{R} 检测链条的收敛性

$$\hat{R} = \sqrt{\frac{\widehat{\text{Var}}^+(\omega|Y)}{W}} \quad (2.15)$$

随着 $n \rightarrow \infty$ ， \hat{R} 下降到 1。如果 \hat{R} 比较大，我们有理由认为需要增加模拟次数以改进待估参数 ω 的后验分布^[30]。

3 统计模型

重新安排逻辑顺序第三章空间广义线性混合效应模型 1. 模型结构

第 3.1 节, 第 3.2 节和第 3.4 节简略介绍了线性模型, 广义线性模型和广义线性混合效应模型及其数学表示, 并随同模型给出了模型求解的 R 包或函数, 为空间广义线性混合效应模型的介绍做铺垫。第 3.6 节重点介绍了空间广义线性混合效应模型 (以下简称 SGLMM 模型), 并分三小节介绍模型中的重要成分, 第 3.6.1 小节介绍 SGLMM 模型中的空间效应, 即平稳空间高斯过程, 第 3.6.2 小节介绍影响空间效应结构的关键部分 — 自协方差函数和平稳空间高斯过程的核函数或自相关函数, 第 3.6.3 小节介绍非空间的随机效应, 在地质统计中常称为块金效应, 以及它带来的 SGLMM 模型可识别问题与相应处理方式。

3.1 线性模型

线性模型的一般形式为

$$Y = X^\top \beta + \epsilon, E(\epsilon) = 0, \text{Cov}(\epsilon) = \sigma^2 \mathbf{I} \quad (3.1)$$

其中, $Y = (y_1, y_2, \dots, y_n)^\top$ 是 n 维列向量, 代表对响应变量 Y 的 n 次观测; $\beta = (\beta_0, \beta_1, \dots, \beta_{p-1})^\top$ 是 p 维列向量, 代表模型 (3.1) 的协变量 X 的系数, β_0 是截距项; $X^\top = (1_{(1 \times n)}, X_{(1)}^\top, X_{(2)}^\top, \dots, X_{(p-1)}^\top)$, $1_{(1 \times n)}$ 是全 1 的 n 维列向量, 而 $X_{(i)}^\top = (x_{1i}, x_{2i}, \dots, x_{ni})^\top$ 代表对第 i 个自变量的 n 次观测; $\epsilon = (\epsilon_1, \epsilon_2, \dots, \epsilon_n)^\top$ 是 n 维列向量, 代表模型的随机误差, 并且假定 $E(\epsilon_i \epsilon_j) = 0, i \neq j$, 即模型误差项之间线性无关, 且方差齐性, 都是 $\sigma^2 (> 0)$ 。估计模型 (3.1) 的参数常用最小二乘和最大似然方法, 求解线性模型 (3.1) 的参数可以用 R 函数 `lm`, 近年来, 高维乃至超高维稀疏线性模型成为热门的研究方向, 相关的 R 包也越来越多, 比较流行的有 `glmnet`^[31] 和 `SIS`^[32]。

3.2 广义线性模型

广义线性模型的一般形式

$$g(\mu) = X^\top \beta \quad (3.2)$$

其中, $\mu \equiv E(Y)$, g 代表联系函数, 特别地, 当 $Y \sim N(\mu, \sigma^2)$ 时, 联系函数 $g(x) = x$, 模型 (3.2) 变为一般线性模型 (3.1)。当 $Y \sim \text{Binomial}(n, p)$ 时, 响应变量 Y 的期望 $\mu = E(Y) = np$, 联系函数 $g(x) = \ln(\frac{x}{1-x})$, 模型 (3.2) 变为 $\log(\frac{p}{1-p}) = X^\top \beta$ 。当 $Y \sim \text{Poisson}(\lambda)$ 时, 响应变量 Y 的期望 $\mu = E(Y) = \lambda$, 联系函数 $g(x) = \ln(x)$, 模型 (3.2) 变为 $\log(\lambda) = X^\top \beta$ 。指数族下其余分布对应的联系函数此处不一一列举, 完整列表可以参看 1989 年 McCullagh 和 Nelder 所著的《广义线性模型》^[17]。模型 (3.2) 最早

由 Nelder 和 Wedderburn 在 1972 年提出^[33]，它弥补了模型 (3.1) 的两个重要缺点：一是因变量只能取连续值的情况，二是期望与自变量只能用线性关系联系^[34]。求解广义线性模型 (3.2) 的 R 函数是 `glm`，常用拟似然法去估计模型 (3.2) 的参数。

3.3 线性混合效应模型

模型结构，参数估计，惩罚最小二乘和广义最小二乘极大似然估计和限制极大似然估计 `lme4` 包^[35] 剖面似然估计 `profile` 似然

3.4 广义线性混合效应模型

广义线性混合模型的一般形式

$$g(\mu) = X^T \beta + Z^T \mathbf{b} \quad (3.3)$$

其中， Z^T 是 q 维随机效应 \mathbf{b} 的 $n \times q$ 的数据矩阵，其它符号含义如前所述。广义线性混合效应模型中既包含固定效应 β 又包含随机效应 \mathbf{b} 。线性模型 (3.1) 和广义线性模型 (3.2) 中的协变量都是固定效应，而随机效应是那些不能直接观察到的潜效应，但是对响应变量却产生显著影响。特别是在基因变异位点与表现型的关系研究中，除了用最新科技做全基因组扫描获取显著的基因位点，还应该把那些看似不显著，联合在一起却显著的位点作为随机效应去考虑^[36]。求解模型 (3.3) 的 R 包有 `nlme`，`mgcv` 和 `lme4`，参数估计的方法一般有限制极大似然法。除了求解模型 (3.3) 外，`nlme` 可以拟合一些非线性混合效应模型，`mgcv` 可以拟合广义可加混合效应模型，`lme4` 使用了高性能的 Eigen 数值代数库，可以加快模型的求解过程。

MCMC 方法应用于 GLMM 模型^[37]

3.5 空间线性混合效应模型

Kriging 插值^[38;39] `gstat: Spatial and Spatio-Temporal Geostatistical Modelling, Prediction and Simulation`

3.6 空间广义线性混合效应模型

在提出 SGLMM 模型之前，先介绍空间高斯过程，然后是平稳空间高斯过程和 SGLMM 模型结构，以步步推进的方式引入各成分的假设条件，其后着重阐述了空间效应的自相关函数，它决定空间效应，包含模型中的待估参数。

3.6.1 模型结构

空间广义线性混合效应模型是对模型 (3.3) 的进一步延伸，其一般形式为

$$g(\mu_i) = d(x_i)^T \beta + S(x_i) + Z_i \quad (3.4)$$

其中, $d^\top(x_i)$ 表示协变量对应的观测数据向量, 即 p 个协变量在第 i 个位置 x_i 的观测值。此外, 假定 $\mathcal{S} = \{S(x) : x \in \mathbb{R}^2\}$ 是均值为 0, 方差为 σ^2 , 平稳且各向同性的空间高斯过程, $\rho(x, x') = \text{Corr}\{S(x), S(x')\} \equiv \rho(\|x, x'\|)$, $\|\cdot\|$ 表示距离, 样本之间的位置间隔不大就用欧氏距离, 间隔很大可以考虑用球面距离; $S(x_i)$ 代表了与空间位置 x_i 相关的随机效应, 简称空间效应; 这里, $Z_i \stackrel{i.i.d}{\sim} N(0, \tau^2)$ 的非空间随机效应, 也称块金效应 (nugget effect), 一般解释为测量误差 (measurement error) 或微观变化 (micro-scale variation)^[12], 即 $\tau^2 = \text{Var}(Y_i|S(x_i)), \forall i = 1, 2, \dots, N$, N 是采样点的数目, 其它符号含义不变。

3.6.2 协方差函数

随机过程的协方差函数, 也称核函数, 如无特殊说明, 文中所指都是自协方差函数, 模型 (3.4) 的空间效应结构由相关函数决定, 在给出相关函数之前, 先计算一下空间效应的理论变差 $V(x, x')$, 即空间过程的协方差函数的一半, 变差源于采矿术语, 和线性预测 T_i 的变差 $V_T(u_{ij})$ 。为方便起见, 记 $T_i = d(x_i)^\top \beta + S(x_i) + Z_i$

$$\begin{aligned}
 V(x, x') &= \frac{1}{2} \text{Var}\{S(x) - S(x')\} \\
 &= \frac{1}{2} \text{Cov}(S(x) - S(x'), S(x) - S(x')) \\
 &= \frac{1}{2} \{E[S(x) - S(x')][S(x) - S(x')] - [E(S(x) - S(x'))]^2\} \\
 &= \sigma^2 - \text{Cov}(S(x), S(x')) = \sigma^2 \{1 - \rho(u)\} \\
 V_T(u_{ij}) &= \frac{1}{2} \text{Var}\{T_i(x) - T_j(x)\} \\
 &= \frac{1}{2} E[(T_i - T_j)^2] = \tau^2 + \sigma^2(1 - \rho(u_{ij}))
 \end{aligned} \tag{3.5}$$

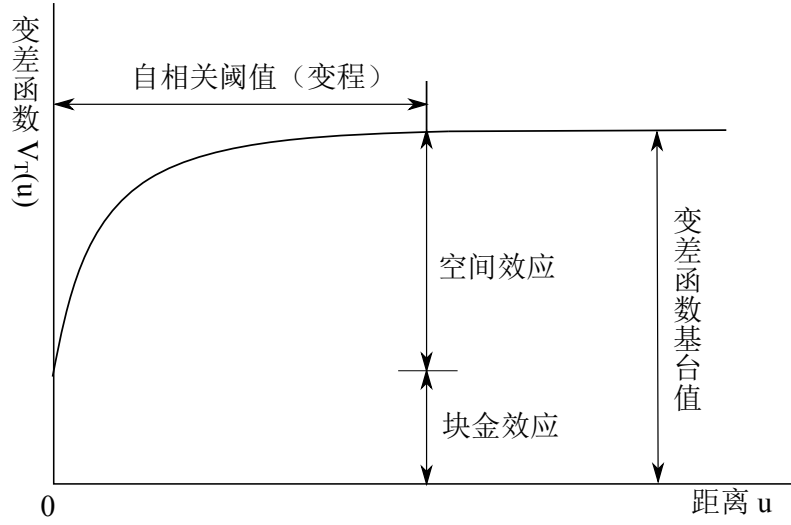
从方程 (3.5) 不难看出系数 $\frac{1}{2}$ 的化简作用, 类似地, 根据协方差定义可推知随机向量 $T = (T_1, T_2, \dots, T_n)$ 的协方差矩阵如下:

$$\begin{aligned}
 \text{Cov}(T_i(x), T_i(x)) &= E[S(x_i)]^2 + EZ_i^2 = \sigma^2 + \tau^2 \\
 \text{Cov}(T_i(x), T_j(x)) &= E[S(x_i)S(x_j)] = \sigma^2 \rho(u_{ij})
 \end{aligned} \tag{3.6}$$

相关函数 $\rho(u)$ 的作用和地位就显而易见了, 它是既决定理论变差又决定协方差矩阵的结构。

常见的自相关函数族有高斯协方差函数、球面协方差函数和 Matérn 协方差函数:

$$\rho(u) = \{2^{\kappa-1} \Gamma(\kappa)\}^{-1} (u/\phi)^\kappa \mathcal{K}_\kappa(u/\phi), u > 0 \tag{3.7}$$

图 3.1: 变差函数 $V_T(u)$ 示意图

一般地，假定 $\rho(u)$ 单调不增，即任何两样本之间的相关性应该随距离变大而减弱，尺度参数 ϕ 控制函数 $\rho(u)$ 递减到 0 的速率。方便起见，记 $\rho(u) = \rho_0(u/\phi)$ ，则方程 (3.7) 可简记为

$$\rho_0(u) = \{2^{\kappa-1}\Gamma(\kappa)\}^{-1}(u)^\kappa \mathcal{K}_\kappa(u), u > 0 \quad (3.8)$$

其中， $\mathcal{K}_\kappa(\cdot)$ 是阶数为 κ 的第二类修正的贝塞尔函数， $\kappa(> 0)$ 是平滑参数，满足这些条件的空间过程 \mathcal{S} 是 $\lceil \kappa \rceil - 1$ 次均方可微的。

值得注意的是 Matérn 族其实包含幂指数族当 $\kappa = 0.5$ 时， $\rho_0(u) = \exp(-u)$ ， $S(x)$ 均方连续但是不可微，当 $\kappa \rightarrow \infty$ 时， $\rho_0(u) = \exp(-u^2)$ ， $S(x)$ 无限次均方可微。

实际操作中，估计 κ ，为了节省计算，又不失一般性，经验做法是取离散的 κ 先验，如 $\kappa = 0.5, 1.5, 2.5$ ，分别对应 $S(x)$ 均方连续、一次可微和二次可微。实际上， $\rho(u)$ 的可微性表示了空间过程 \mathcal{S} 的曲面平滑程度。为更加直观地描述 $\rho(u)$ ，作图 3.2

从图上可以看出，相比于贝塞尔函数的阶 κ ，尺度参数 ϕ 对相关函数的影响大些，所以在实际应用中，先固定下 κ 是可以接受的。

为简化编程和描述，在 1998 年 Diggle 等人在数据分析中使用幂指数族 $\rho_0(u) = \exp(-(\alpha u)^\delta)$ ， $\alpha > 0, 0 < \delta \leq 2$ 作为相关函数^[3]。虽然其形式大大简化，函数图像和性质却与梅隆族相似，即当 $0 < \delta < 2$ 时， $S(x)$ 均方连续但不可微，当 $\delta = 2$ 时， $S(x)$ 无限次可微。

3.6.3 模型识别

在 SGLMM 模型的实际应用当中，一般先不添加非空间的随机效应，而是基于模型 (3.9) 估计参数，估计完参数，代入模型，观察线性预测 $T(\hat{x}_i)$ 和真实的 $T(x_i)$ 之间的残差，如残差表现不平稳，说明还有非空间的随机效应没有提取，因此添加块金效

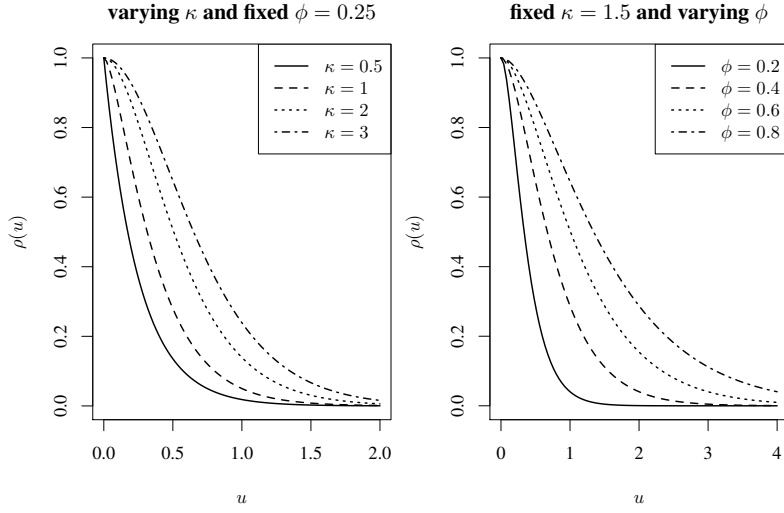


图 3.2: 固定尺度参数, 相关函数随距离的变化 (左); 固定贝塞尔函数的阶, 相关函数随距离的变化 (右)

应是合理的, 此时在模型 (3.4) 中有两个来源不同的随机效应 Z_i 与 $S(x_i)$ 。

$$g(\mu_i) = d(x_i)^\top \beta + S(x_i) \quad (3.9)$$

如何区分开 Z_i 与 $S(x_i)$, 或者更直接地说, 如何估计这两个随机效应的参数 τ^2, σ^2, ϕ , 即为可识别问题。向量 $T = (T_1, T_2, \dots, T_n)^\top$ 是协方差为矩阵 $\tau^2 I + \sigma^2 R$ 的多元高斯分布, 其中, 自相关函数 $R_{ij} = \rho(u_{ij}; \phi)$, u_{ij} 是 x_i 与 x_j 之间的距离, 由线性预测 T_i 的变差公式 (3.5) 知, 随机过程 $T(x)$ 的变差 $\tau^2 + \sigma^2(1 - \rho(u_{ij}))$ 和自相关函数 (3.10)

$$\rho^*(u) = \begin{cases} 1 & : x_i = x_j \\ \sigma^2 \rho(u_{ij}) / (\sigma^2 + \tau^2) & : x_i \neq x_j \end{cases} \quad (3.10)$$

在原点不连续, 只有当 $\tau^2 = \text{Var}(Y_i | S(x_i))$ 已知或者在同一位置可以用重复测量的方法直接估计时, 参数 τ^2, σ^2, ϕ 是可识别的^[5,40]。

4 参数估计

模型参数估计是建模分析的重要步骤，鉴于空间广义线性混合效应模型（以下简称 SGLMM 模型）的复杂性，文献综述中用的参数估计方法，比如最小二乘估计（简称 LSE）和极大似然估计（简称 MLE），其参数估计没有显式的表达式，必须发展有效的算法。目前，文献中的常用算法有低秩近似算法、蒙特卡罗极大似然算法和马尔科夫链蒙特卡罗算法，但这些尚未应用到 SGLMM 模型的参数估计中，我们应用这些算法去估计模型参数，特别提出了新的算法。本章具体结构如下：第 4.3 节介绍求解 SGLMM 模型 (4.1) 参数的三类算法，分别是第 4.3.2 小节介绍的蒙特卡罗极大似然算法（以下简称为 MCML 算法），第 4.3.4 小节介绍的低秩近似算法（以下简称为 Low-Rank 算法），第 4.3.3 小节介绍的贝叶斯框架下的马尔科夫链蒙特卡罗算法（以下简称为贝叶斯 MCMC 算法）。在贝叶斯 MCMC 算法的基础上，改进算法实现形式，提出 STAN-MCMC 算法，在第 4.4 节予以详细介绍。

$$g(\mu_i) = T_i = d(x_i)' \beta + S(x_i) + Z_i \quad (4.1)$$

其中，设置 $d_i = d'(x_i)$ ， $d'(x_i)$ 主要是强调协变量的空间内容，这里表示采样点的观测数据向量，即 p 个协变量在第 i 个位置 x_i 的观察值。 $\mathcal{S} = \{S(x) : x \in \mathbb{R}^2\}$ 是均值为 0，方差为 σ^2 ，平稳且各向同性的空间高斯过程，自相关函数是 $\rho(u; \phi)$ ， $S(x_i)$ 是空间效应， $Z_i \stackrel{i.i.d}{\sim} N(0, \tau^2)$ 的块金效应， g 是联系函数， $x_i \in \mathbb{R}^2$ 是采样点的位置。综上，模型 (4.1) 待估计的参数有 β 和 $\theta' = (\sigma^2, \phi, \tau^2)$ 。特别地，当响应变量分别服从二项分布和泊松分布时，模型 (4.1) 分别变为模型 (4.2) 和模型 (4.3)。

$$\log\left\{\frac{p_i}{1-p_i}\right\} = T_i = d(x_i)' \beta + S(x_i) + Z_i \quad (4.2)$$

$$\log(\lambda_i) = T_i = d(x_i)' \beta + S(x_i) + Z_i \quad (4.3)$$

模型 (4.2) 中，响应变量 Y_i 服从二项分布 $Y_i \sim \text{Binomial}(m_i, p_i)$ ，均值 $E(Y_i | S(x_i), Z_i) = m_i p_i$ ， m_i 表示在 x_i 的位置抽取的样本量，总的样本量就是 $M = \sum_{i=1}^N m_i$ ， N 表示采样点的个数。模型 (4.3) 中，响应变量 Y_i 服从泊松分布 $Y_i \sim \text{Poisson}(\lambda_i)$ 。在获取响应变量 Y 的观测的过程中，与广义线性或广义线性混合效应模型 (3.2) 和 (3.3) 不同的在于：在每个采样点 x_i ， Y_i 都服从参数不同但同类的分布。下面如无特殊说明，算法描述对象都是模型 (4.2)。

4.1 极大似然估计

研究区域 $D \subseteq \mathbb{R}^d$ ，对于第 i 次观测，让 s_i 表示区域 D 内特定的位置， $Y(s_i)$ 表示响应变量， $\mathbf{x}(s_i)$ 是一个 p 维的固定效应，这里 $i = 1, \dots, n$ 我们如下定义 SGLMM 模型：

$$E[Y(s_i)|u(s_i)] = g^{-1}[\mathbf{x}(s_i)' \boldsymbol{\beta} + \mathbf{u}(s_i)], \quad i = 1, \dots, n$$

其中

- $g(\cdot)$ 是实值可微的逆联系函数， $\boldsymbol{\beta}$ 是 p 维的回归参数。
- $\{U(\mathbf{s}) : \mathbf{s} \in D\}$ 是二阶平稳的高斯随机场，其均值为 0 ，空间自协方差函数 $\text{COV}(U(\mathbf{s}), U(\mathbf{s}')) = C(\mathbf{s} - \mathbf{s}'; \boldsymbol{\theta})$ ， $\boldsymbol{\theta}$ 是相关参数向量。 $\mathbf{u} = (u(s_1), u(s_2), \dots, u(s_n))'$ 是 $U(\cdot)$ 的一个实例。
- $Y(\cdot)|U(\cdot)$ 是独立随机过程，如给定 \mathbf{u} ，观察值 $\mathbf{y} = (y(s_1), y(s_2), \dots, y(s_n))'$ 是相互独立的。
- 给定 $u_i = u(s_i), i = 1, \dots, n$ ， $y_i = y(s_i)$ 的条件概率密度函数是

$$f(y_i|u_i; \boldsymbol{\beta}) = \exp[a(\mu_i)y_i - b(\mu_i)]c(y_i)$$

其中 $\mu_i = Y_i|u_i$ ， $a(\cdot), b(\cdot)$ 和 $c(\cdot)$ 是特定的函数。具体举例^[17]

那么，SGLMM 模型的边际似然函数为

$$L(\boldsymbol{\psi}; \mathbf{y}) = \int \prod_{i=1}^n f(y_i|u_i; \boldsymbol{\beta}) \phi_n(\mathbf{u}; 0, \Sigma_{\boldsymbol{\theta}}) d\mathbf{u} \quad (4.4)$$

其中 $\boldsymbol{\psi} = (\boldsymbol{\beta}, \boldsymbol{\theta})$ 是 SGLMM 模型的参数， $\phi_n(\cdot; 0, \Sigma_{\boldsymbol{\theta}})$ 是 n 元正态密度函数，其均值为 0 ，协方差矩阵为 $\Sigma_{\boldsymbol{\theta}} = (c_{ij}) = (C(s_i - s_j; \boldsymbol{\theta})), i, j = 1, \dots, n$ 。此边际似然函数中几乎总是卷入一个难以处理的积分，这是主要面临的问题，并且计算量随观测 y_i 的数量增加，因为随机场的维数等于观测的个数。

令 $\mathbf{y} = (y(s_1), \dots, y(s_n))'$ 表示观测值， $\pi(\boldsymbol{\psi})$ 表示模型参数的联合先验密度，那么联合后验密度为

$$\pi(\boldsymbol{\psi}, \mathbf{u}|\mathbf{y}) = \frac{f(\mathbf{y}|\mathbf{u}, \boldsymbol{\psi}) \phi_n(\mathbf{u}; 0, \Sigma_{\boldsymbol{\theta}}) \pi(\boldsymbol{\psi})}{m(\mathbf{y})}$$

其中

$$m(\mathbf{y}) = \int f(\mathbf{y}|\mathbf{u}, \boldsymbol{\psi}) \phi_n(\mathbf{u}; 0, \Sigma_{\boldsymbol{\theta}}) \pi(\boldsymbol{\psi}) d\mathbf{u} d\boldsymbol{\psi}$$

遭遇同样难以处理的高维积分问题，所以 $m(\mathbf{y})$ 亦不会有显式表达式。若取 $\pi(\boldsymbol{\psi})$ 为扁平先验 (flat priors)，特别地，如 $\pi(\boldsymbol{\psi}) \propto 1$ ，后验分布将简化为似然函数 (4.4) 的常数倍。如果导出的后验是合适的，MCMC 算法可以用来研究似然函数，但是对很多 GLMM 模型扁平先验 (flat priors) 会导出不合适的后验 (improper posteriors)^[41]，所以选用 diffuse prior 来导出合适的后验 (proper posteriors)，并且导出的后验接近似然函数，但它的使用不要求 posterior mode 是 MLE^[42]。

一般地，我们假定在给定 β 和 $S(x_i)$ 的条件下，响应变量 Y_i 服从指数族，这里不失一般性地讨论空间广义线性混合效应模型的极大似然估计。条件概率密度函数 $Y_i|\beta, \mu_i$ 为

$$f(y_i|\beta, S(x_i)) = \exp\{[y_i\mu_i - b(\mu_i)] + c(y_i)\}$$

其中， $\mu_i = E[Y_i|\beta, S(x_i)]$ 是典则参数， $Y_i \sim N(\mu, \sigma^2)$ 时，

$T = (T_1, T_2, \dots, T_n)'$ 是多元正态分布 $N(D\beta, \Sigma(\boldsymbol{\theta}))$ ，参数 $\beta, \boldsymbol{\theta}$ 的似然函数

$$L(\beta', \boldsymbol{\theta}'; y) = \int \prod_{i=1}^n f(y_i|t_i) f(T|\beta', \boldsymbol{\theta}') dt \quad (4.5)$$

4.2 剖面似然估计

极大似然估计是一种被广泛接受的统计方法，因其优良的大样本性质。在宽松的正则条件下，极大似然估计服从渐进正态分布，满足无偏性，而且是有效的估计。在空间线性混合效应模型下，响应变量服从高斯分布

$$\mathbf{y} \sim N(D\beta, \sigma^2 R(\phi) + \tau^2 \mathbf{I}) \quad (4.6)$$

其中 D 是 $n \times p$ 的观测数据矩阵， β 是相应的回归参数向量， R 依赖于标量或向量 ϕ 。模型 (4.6) 的对数似然函数

$$\begin{aligned} L(\beta, \tau^2, \sigma^2, \phi) = & -0.5\{n \log(2\pi) + \log\{(\sigma^2 \mathbf{R}(\phi) + \tau^2 \mathbf{I})\} \\ & + (\mathbf{y} - D\beta)^\top (\sigma^2 \mathbf{R}(\phi) + \tau^2 \mathbf{I})^{-1} (\mathbf{y} - D\beta)\} \end{aligned} \quad (4.7)$$

极大化 (4.7) 式就是求模型 (4.6) 参数的极大似然估计。极大化对数似然的过程如下：

1. 重参数化 $\nu^2 = \tau^2/\sigma^2$ ，则 $V = R(\phi) + \nu^2 \mathbf{I}$;
2. 给定 V ，对数似然函数 (4.7) 在

$$\begin{aligned} \hat{\beta}(V) &= (D^\top V^{-1} D)^{-1} D^\top V^{-1} \mathbf{y} \\ \hat{\sigma}^2(V) &= n^{-1} \{\mathbf{y} - D\hat{\beta}(V)\}^\top V^{-1} \{\mathbf{y} - D\hat{\beta}(V)\} \end{aligned} \quad (4.8)$$

取得极大值；

3. 将 (4.8) 式代入对数似然函数 (4.7) 式，可获得一个简化的对数似然

$$L_0(\nu^2, \phi) = -0.5\{n \log(2\pi) + n \log \hat{\sigma}^2(V) + \log |V| + n\} \quad (4.9)$$

关于参数 ν^2, ϕ 极大化 (4.9) 式，获得 ν^2, ϕ 的估计值，然后将其回代 (4.8) 式，获得 $\hat{\beta}$ 和 $\hat{\sigma}^2$ 。

上述优化过程可能依赖协方差函数族，如在使用 Matérn 协方差函数的时候，形状参数 κ (shape parameter) 常常很难识别。因此，让 κ 分别取 0.5, 1.5, 2.5，使得空间过程 \mathcal{S} 覆盖到不同自由度的均方可微性^[43]。

原则上，极大似然估计的可变性可以通过观察对数似然曲面来分析，但是，通常地，似然曲面的维数不允许直接观察。在这种情形下，另一个基于似然的想法是剖面似然 (profile likelihood)。一般地，假定我们有一个模型，其含有参数 (α, ϕ) ，似然表示为 $L(\alpha, \phi)$ 。则关于 α 的剖面对数似然 (profile log-likelihood) 定义为

$$L_p(\alpha) = L(\alpha, \hat{\psi}(\alpha)) = \max_{\psi} (L(\alpha, \psi)) \quad (4.10)$$

即我们考虑似然随 α 的变化情况，对每一个 α （保持 α 不变），我们指定 ψ 的值使得对数似然取得最大值。剖面似然就是让我们可以观察到关于 α 的似然曲面，显然，其维数比完全似然面要低，与只有一个参数的对数似然一样，它也可以用来计算单个参数的置信区间。现在，我们注意到简化的对数似然 (4.9) 其实可以看作模型 (4.6) 关于 (ν^2, ϕ) 的剖面对数似然^[40]。

4.3 参数估计的算法

贝叶斯方法构造马尔科夫链，需要很多次反复迭代，收敛速度慢，求解模型 (3.4) 需要花费很多时间，极大似然和重要性采样相结合的方法出现了，称之为蒙特卡罗极大似然法，简称 MCML。1994 年 Charles J. Geyer 首先从理论证明 MCML 方法的收敛性及相关似然估计的正态性，为后续算法的开发、改进以及应用提供了理论支持^[14]。2002 年 Hao Zhang 在做模型的估计和预测的时候提出蒙特卡罗期望极大梯度 (Monte Carlo EM Gradient) 算法^[15]。2016 年 Fatemeh Hosseini 在 MCEMG 的基础上提出近似蒙特卡罗期望极大梯度 (Approximate Monte Carlo EM Gradient) 算法^[44]。2004 年 Ole F Christensen 将 MCML 方法用于 SGLMM 模型^[12]，2016 年 Peter J. Diggle 和 Emanuele Giorgi 将 MCML 方法应用于分析西非流行病调查数据。

在依据似然函数做统计推断的情况下，必须解决高维积分的问题，解决高维积分的两条路：采用拉普拉斯近似方法近似积分，蒙特卡罗方法求积分

4.3.1 拉普拉斯近似算法

主要参考文献^[45]，空间广义线性混合效应模型的结构表述如下

$$\begin{aligned} \mathbf{Y}(\mathbf{x})|S(\mathbf{x}) &\sim f(\cdot; \boldsymbol{\mu}(\mathbf{x}), \psi) \\ g(\boldsymbol{\mu}(\mathbf{x})) &= D\boldsymbol{\beta} + S(\mathbf{x}) = D\boldsymbol{\beta} + \sigma R(\mathbf{x}; \phi) + \tau z \\ S(\mathbf{x}) &\sim N(\mathbf{0}, \Sigma) \end{aligned} \quad (4.11)$$

SGLMM 模型假定在给定高斯空间过程 $S(\mathbf{x})$ 的条件下, Y_1, Y_2, \dots, Y_n 是独立的, 并且服从分布 $f(\cdot; \boldsymbol{\mu}(\mathbf{x}), \psi)$ 。此分布有两个参数集, 其一来自与联系函数 g 关联的线性预测 $\boldsymbol{\mu}(\mathbf{x})$, 其二是来自密度或概率分布函数 f 的发散参数 (precision or dispersion parameter) ψ , 可以看作是似然函数中的附加参数。空间过程分解为空间相关 $R(\mathbf{x}; \phi)$ 和独立过程 Z , 二者分别被参数 σ 和 τ 归一化而具有单位方差。

线性预测包含一组固定效应 $D\boldsymbol{\beta}$, 空间相关的随机效应 $R(\mathbf{x}; \phi)$, 不相关的随机效应 $\tau z \sim N(\mathbf{0}, \tau^2 \mathbf{I})$ 。 D 是观测协变量得到的数据矩阵, $\boldsymbol{\beta}$ 是 $p \times 1$ 维的回归参数向量。

$R(\mathbf{x}; \phi)$ 是具有单位方差的高斯随机场 (Gaussian Random Field, 简称 GRF), 其自相关函数为 $\rho(u, \phi)$, 这里 u 表示一对空间位置之间的距离, ϕ 是刻画空间相关性的参数。自相关函数 $\rho(u, \phi) (\in \mathbb{R}^d)$ 是 d 维空间到一维空间的映射函数, 特别地, 假定这里的空间过程的自相关函数仅仅依赖成对点之间的欧氏距离, 即 $u = \|x_i - x_j\|$ 。常见的自相关函数有指数族、梅隆族和球族。线性预测的随机效应部分协方差矩阵 $\Sigma = \sigma^2 R(\mathbf{x}; \phi) + \tau^2 \mathbf{I}$ 。

估计 SGLMM 模型 (4.11) 的参数 $\boldsymbol{\theta} = (\boldsymbol{\beta}, \sigma^2, \tau^2, \phi, \psi)$ 需要极大化边际似然函数

$$L(\boldsymbol{\theta}; \mathbf{y}) = \int_{\mathbb{R}^n} [\mathbf{Y}(\mathbf{x})|S(\mathbf{x})][S(\mathbf{x})]dS(\mathbf{x}) \quad (4.12)$$

一般地, 边际似然函数包含两个分布的乘积和随机效应的积分, 并且这个积分无法显式的表示, 第一个分布是观测变量 \mathbf{Y} 的抽样分布, 第二个分布典型的假设是多元高斯分布。一个特殊的情况是 \mathbf{Y} 也假设为高斯分布, 这时积分有显式表达式。注意到参数 τ 和 ψ 是不可识别的。

边际似然函数 (4.12) 卷入的数值积分是充满挑战的, 因为积分的维数 n 是观测值的数目, 所以像二次、高斯-埃尔米特或适应性高斯-埃尔米特数值积分方式都是不可用的。1986 年 Luke Tierney 和 Joseph B. Kadane 提出拉普拉斯近似方法^[22], 它在纵向数据分析中被大量采用^[46]。总之, 对空间广义线性混合效应模型而言, 拉普拉斯近似还可以继续采用, 想法是近似边际似然函数中的高维 ($n > 3$) 积分, 获得一个易于处理的表达式, 有了积分的显式表达式, 就可以用数值的方法求边际似然函数的极大值。拉普拉斯方法即用如下方式近似 (4.12) 中的积分

$$I = \int_{\mathbb{R}^n} \exp\{Q(\mathbf{s})\} d\mathbf{s} \approx (2\pi)^{n/2} | -Q''(\hat{\mathbf{s}}) |^{-1/2} \exp\{Q(\hat{\mathbf{s}})\} \quad (4.13)$$

其中, $Q(\mathbf{s})$ 为已知的 n 元函数, $\hat{\mathbf{s}}$ 是其极大值点, $Q''(\hat{\mathbf{s}})$ 是黑塞矩阵。

这种近似方法可以用于一般的广义线性混合效应模型的似然推断, 特别地, 空间广义线性混合效应模型也一样。假定条件分布 f 可以表示成如下单参数指数族的形式

$$f(\mathbf{y}; \boldsymbol{\beta}) = \exp\{\mathbf{y}^\top (D\boldsymbol{\beta} + S(\mathbf{x})) - \mathbf{1}^\top b(D\boldsymbol{\beta} + S(\mathbf{x})) + \mathbf{1}^\top c(\mathbf{y})\} \quad (4.14)$$

其中 b 是特定的函数。常用的分布有泊松分布和二项伯努利分布。多元高斯密度函数

$$f(S(\mathbf{x}); \Sigma) = (2\pi)^{-n/2} |\Sigma|^{-1/2} \exp\left\{-\frac{1}{2} S(\mathbf{x})^\top \Sigma^{-1} S(\mathbf{x})\right\} \quad (4.15)$$

$$= \exp\left\{-\frac{n}{2} \log(2\pi) - \frac{1}{2} \log |\Sigma| - \frac{1}{2} S(\mathbf{x})^\top \Sigma^{-1} S(\mathbf{x})\right\} \quad (4.16)$$

将边际似然函数 (4.12) 写成适合使用拉普拉斯近似的格式

$$L(\boldsymbol{\theta}; \mathbf{y}) = \int_{\mathbb{R}^n} \exp\{Q(S(\mathbf{x}))\} dS(\mathbf{x}) \quad (4.17)$$

其中

$$\begin{aligned} Q(S(\mathbf{x})) = & \mathbf{y}^\top (D\boldsymbol{\beta} + S(\mathbf{x})) - \mathbf{1}^\top b(D\boldsymbol{\beta} + S(\mathbf{x})) + \mathbf{1}^\top c(\mathbf{y}) \\ & - \frac{n}{2} \log(2\pi) - \frac{1}{2} \log |\Sigma| - \frac{1}{2} S(\mathbf{x})^\top \Sigma^{-1} S(\mathbf{x}) \end{aligned} \quad (4.18)$$

方程 (4.18) 凸显了采纳拉普拉斯近似方法拟合空间广义线性混合效应模型的方便性, 可以把 (4.18) 当成两部分来看, 前一部分是广义线性模型下样本的对数似然的和的形式, 后一部分是多元高斯分布的对数似然, 表示潜变量 (latent variable, 又称随机效应, random effect)。要使用 (4.13) 式, 我们需要函数 $Q(S(\mathbf{x}))$ 的极大值点 $\hat{\mathbf{s}}$, 这里我们采用牛顿-拉夫森算法 (Newton-Raphson, 简称 NR) 寻找 n 元函数的极大值点, NR 算法需要重复计算

$$\mathbf{s}_{i+1} = \mathbf{s}_i - Q''(\mathbf{s}_i)^{-1} Q'(\mathbf{s}_i) \quad (4.19)$$

直到收敛 $\hat{\mathbf{s}}$ 。在这个阶段 (内迭代过程), 我们将所有的参数 $\boldsymbol{\theta}$ 都当作已知的。 Q 函数的一阶和二阶导数如下

$$Q'(\mathbf{s}) = \{\mathbf{y} - b'(D\boldsymbol{\beta} + \mathbf{s})\}^\top - \mathbf{s}^\top \Sigma^{-1} \quad (4.20)$$

$$Q''(\mathbf{s}) = -\text{diag}\{b''(D\boldsymbol{\beta} + \mathbf{s})\} - \Sigma^{-1} \quad (4.21)$$

用拉普拉斯方法近似的对数似然如下

$$\begin{aligned} \ell(\boldsymbol{\theta}; \mathbf{y}) = & \frac{n}{2} \log(2\pi) - \frac{1}{2} \log |-\text{diag}\{b''(D\boldsymbol{\beta} + \mathbf{s})\} - \Sigma^{-1}| \\ & + \mathbf{y}^\top (D\boldsymbol{\beta} + \hat{\mathbf{s}}) - \mathbf{1}^\top b(D\boldsymbol{\beta} + \hat{\mathbf{s}}) + \mathbf{1}^\top c(\mathbf{y}) \\ & - \frac{n}{2} \log(2\pi) - \frac{1}{2} \log |\Sigma| - \frac{1}{2} \hat{\mathbf{s}}^\top \Sigma^{-1} \hat{\mathbf{s}} \end{aligned} \quad (4.22)$$

现在，我们需要极大化近似对数似然 (4.22) 式，此时是求模型参数，可谓之外迭代过程，常用的算法是 Broyden-Fletcher-Goldfarb-Shanno (BFGS) 算法，它内置在 R 函数 `optim()` 中。方便起见，模型参数记为 $\boldsymbol{\theta} = (\boldsymbol{\beta}, \log(\sigma^2), \log(\tau^2), \log(\phi), \log(\psi))$ ，且 $\hat{\boldsymbol{\theta}}$ 表示 $\boldsymbol{\theta}$ 的最大似然估计，则 $\hat{\boldsymbol{\theta}}$ 的渐进分布为

$$\hat{\boldsymbol{\theta}} \sim N(\boldsymbol{\theta}, \mathbf{I}_o^{-1}(\hat{\boldsymbol{\theta}}))$$

其中 $\mathbf{I}_o(\hat{\boldsymbol{\theta}})$ 观察到的样本信息矩阵，注意到在空间广义线性混合效应模型下，我们不能计算 Fisher 信息阵，因为对数似然函数没有显式表达式，只有数值迭代获得在 $\hat{\boldsymbol{\theta}}$ 处的观测信息矩阵。通常，这类渐进近似对协方差参数的估计效果不好，而我们感兴趣的恰恰是协方差参数 σ^2, τ^2, ϕ ，在数据集不太大的情形下，我们使用剖面似然 (Profile likelihood) 方法计算协方差参数的置信区间。剖面似然的详细实现过程见 **bbmle** 包^[47]。下面给出计算的细节步骤：

1. 选择模型参数 $\boldsymbol{\theta}$ 的初始值 $\boldsymbol{\theta}_i$;
2. 计算协方差矩阵 Σ 及其逆 Σ^{-1} ;
3. 通过如下步骤极大，估计 $\hat{\mathbf{s}}$:
 - (a) 为 \mathbf{s} 选择初始值;
 - (b) 按 (4.20) 式计算 $Q'(\mathbf{s})$ ，按 (4.21) 式计算 $Q''(\mathbf{s})$ ，其中导数的计算可以参见黄湘云的文章^[48];
 - (c) 解线性方程组 $Q''(\mathbf{s})\mathbf{s}^* = Q'(\mathbf{s})$;
 - (d) 更新 $\mathbf{s} = \mathbf{s} + \mathbf{s}^*$;
 - (e) 迭代直到收敛以获得 $\hat{\mathbf{s}}$ 。
4. 用 $\hat{\mathbf{s}}$ 替换 $S(\mathbf{x})$ ，在 (4.18) 式中计算 $Q(\hat{\mathbf{s}})$;
5. 用 (4.13) 式计算积分的近似值，以获得边际似然 (4.22) 式的值;
6. 用 BFGS 算法获得下一个值 $\boldsymbol{\theta}_{i+1}$;
7. 重复上述过程直到收敛，获得参数的估计值 $\hat{\boldsymbol{\theta}}$ 。

牛顿-拉夫森算法收敛速度是很快的，但是必须提供一个很好的初值。合理的初值对于快速收敛到近似似然函数 $\ell(\boldsymbol{\theta}; \mathbf{y})$ 的极大值点很重要。我们给出一个简单的策略指定外迭代中的初值 $\boldsymbol{\theta}_0$ ，首先拟合一个简单的广义线性模型，获得回归系数 $\boldsymbol{\beta}$ 的初值，基于这些值计算线性预测值 $\hat{\boldsymbol{\mu}}$ ；然后计算残差 $\hat{\mathbf{r}} = (\hat{\boldsymbol{\mu}} - \mathbf{y})$ ， $\hat{\mathbf{r}}$ 的方差作为 σ^2 的初值，如果 SGLMM 带有块金效应（非空间相关的随机效应），就用 σ^2 的初值的一定比例（如 10%）作为 τ^2 的初值；最后， ϕ 的初值选择两个距离最大的观测点之间的距离的 10%，比较保险的办法是选择不同的 ϕ 作为初值。

4.3.2 蒙特卡罗极大似然算法

模型 (4.2) 中参数 $\boldsymbol{\beta}$ 和 $\boldsymbol{\theta}' = (\sigma^2, \phi, \tau^2)$ 的似然函数是通过对 T_i 内的随机效应积分获得的。用大写 D 表示 $n \times p$ 的数据矩阵， $\mathbf{y}' = (y_1, y_2, \dots, y_n)$ 表示各空间位置 x_i 处响应变量的观测值，对应模型 (4.2) 中的 $Y_i \sim \text{Binomial}(m_i, p_i)$ ， $T = (T_1, T_2, \dots, T_n)$ 的边际分布是 $N(D\boldsymbol{\beta}, \Sigma(\boldsymbol{\theta}))$ ，其中，协方差矩阵 $\Sigma(\boldsymbol{\theta})$ 的对角元是 $\sigma^2 + \tau^2$ ，非对角元是 $\sigma^2 \rho(u_{ij})$ ， u_{ij} 是位置 x_i 与 x_j 之间的距离。在给定 $T' = t' = (t_1, t_2, \dots, t_n)$ 下， $\mathbf{Y}' = (Y_1, \dots, Y_n)$ 的条件分布是独立二项概率分布函数的乘积 $f(\mathbf{y}|\mathbf{t}) = \prod_{i=1}^n f(y_i|t_i)$ ，因此， $\boldsymbol{\beta}$ 和 $\boldsymbol{\theta}$ 的似然函数可以写成

$$L(\boldsymbol{\beta}, \boldsymbol{\theta}) = f(\mathbf{y}; \boldsymbol{\beta}, \boldsymbol{\theta}) = \int_{\mathbb{R}^n} N(\mathbf{t}; D\boldsymbol{\beta}, \Sigma(\boldsymbol{\theta})) f(\mathbf{y}|\mathbf{t}) d\mathbf{t} \quad (4.23)$$

其中， $N(\cdot; \boldsymbol{\mu}, \Sigma)$ 表示均值为 $\boldsymbol{\mu}$ ，协方差矩阵为 Σ 的多元高斯分布的密度函数。1994 年 Charles J. Geyer 在给定 $\mathbf{Y} = \mathbf{y}$ 的情况下，使用 T 的条件分布 $f(T|\mathbf{Y} = \mathbf{y})$ 模拟近似方程 (4.23) 中的高维积分^[14]，则似然函数 $L(\boldsymbol{\beta}, \boldsymbol{\theta})$ 可以重写为

$$\begin{aligned} L(\boldsymbol{\beta}, \boldsymbol{\theta}) &= \int_{\mathbb{R}^n} \frac{N(\mathbf{t}; D\boldsymbol{\beta}, \Sigma(\boldsymbol{\theta})) f(\mathbf{y}|\mathbf{t})}{N(\mathbf{t}; D\boldsymbol{\beta}_0, \Sigma(\boldsymbol{\theta}_0)) f(\mathbf{y}|\mathbf{t})} f(\mathbf{y}, \mathbf{t}) d\mathbf{t} \\ &\propto \int_{\mathbb{R}^n} \frac{N(\mathbf{t}; D\boldsymbol{\beta}, \Sigma(\boldsymbol{\theta}))}{N(\mathbf{t}; D\boldsymbol{\beta}_0, \Sigma(\boldsymbol{\theta}_0))} f(\mathbf{t}|\mathbf{y}) d\mathbf{t} \\ &= E_{T|\mathbf{y}} \left[\frac{N(\mathbf{t}; D\boldsymbol{\beta}, \Sigma(\boldsymbol{\theta}))}{N(\mathbf{t}; D\boldsymbol{\beta}_0, \Sigma(\boldsymbol{\theta}_0))} \right] \end{aligned} \quad (4.24)$$

其中， $\boldsymbol{\beta}_0, \boldsymbol{\theta}_0$ 作为迭代初始值预先给定，则 \mathbf{Y} 和 T 的联合分布可以表示成 $f(\mathbf{y}, \mathbf{t}) = N(\mathbf{t}; D\boldsymbol{\beta}_0, \Sigma(\boldsymbol{\theta}_0)) f(\mathbf{y}|\mathbf{t})$ 。再通过蒙特卡罗方法，用求和代替积分，去近似期望，做法是从条件分布 $f(T|\mathbf{Y} = \mathbf{y}; \boldsymbol{\beta}_0, \boldsymbol{\theta}_0)$ 抽取 m 个样本 $t_{(i)}$ ，那么，可以用方程 (4.25) 近似方程 (4.24)

$$L_m(\boldsymbol{\beta}, \boldsymbol{\theta}) = \frac{1}{m} \sum_{i=1}^m \frac{N(t_{(i)}; D\boldsymbol{\beta}, \Sigma(\boldsymbol{\theta}))}{N(t_{(i)}; D\boldsymbol{\beta}_0, \Sigma(\boldsymbol{\theta}_0))} \quad (4.25)$$

这样做的依据是不管样本序列 $t_{(i)}$ 是否相关， $L_m(\boldsymbol{\beta}, \boldsymbol{\theta})$ 都是 $L(\boldsymbol{\beta}, \boldsymbol{\theta})$ 的一致估计 (con-

sistent estimator)。最优的 β_0, θ_0 是 β, θ 的极大似然估计，即

$$\max_{\beta, \theta} L_m(\beta, \theta) \rightarrow 1, m \rightarrow \infty$$

既然给定的初始值 β_0, θ_0 必定与真实的极大似然估计值不同，第 m 步迭代获得的似然函数值 $L_m(\hat{\beta}_m, \hat{\theta}_m)$ 与 1 的距离可以用来刻画蒙特卡罗近似的准确度。实际操作中，用 $\hat{\beta}_m$ 和 $\hat{\theta}_m$ 表示最大化 $L_m(\beta, \theta)$ 获得的 MCML 估计，重复迭代 $\beta_0 = \hat{\beta}_m$ 和 $\theta_0 = \hat{\theta}_m$ 直到收敛。求蒙特卡罗近似的对数似然 $l_m(\beta, \theta) = \log L_m(\beta, \theta)$ 的极值，可以使用 **PrevMap** 包，迭代 $L_m(\beta, \theta)$ 的过程中，可以选择 BFGS 算法。MCML 估计 θ_m 的标准误差 (standard errors) 取似然函数 $l_m(\beta, \theta)$ 的负黑塞矩阵的逆的对角线元素的平方根。迭代次数足够多时 (m 充分大，一般取到 10000 及以上)，此时蒙特卡罗误差可忽略，即方程 (4.25) 近似 (4.24) 的误差可忽略。

4.3.3 贝叶斯 MCMC 算法

在贝叶斯框架里， β, θ 的后验分布由贝叶斯定理和 β, θ 的联合先验分布确定。假定 β, θ 的先验分布如下：

$$\theta \sim g(\cdot), \quad \beta | \sigma^2 \sim N(\cdot; \xi, \sigma^2 \Omega)$$

其中 $g(\cdot)$ 可以是 θ 的任意分布， ξ 和 Ω 分别是 β 的高斯先验的均值向量和协方差矩阵。 β, θ 和 T 的后验分布是

$$\pi(\beta, \theta, t | y) \propto g(\theta) N(\beta; \xi, \sigma^2 \Omega) N(t; D\beta, \Sigma(\theta)) f(y | t) \quad (4.26)$$

PrevMap 包内的函数 `binomial.logistic.Bayes` 可以从上述后验分布中抽得样本，这个抽样的过程使用了 MCMC 算法， θ, β 和 T 迭代的过程如下：

1. 初始化 β, θ 和 T
2. 对协方差 $\Sigma(\theta)$ 中的参数做如下变换^[49]

$$(\tilde{\theta}_1, \tilde{\theta}_2, \tilde{\theta}_3) = (\log \sigma, \log(\sigma^2 / \phi^{2\kappa}), \log \tau^2)$$

使用随机游走 Metropolis-Hastings 轮流更新上述三个参数，在第 i 次迭代时，候选高斯分布的标准差 h 是 $h_i = h_{i-1} + c_1 i^{-c_2} (\alpha_i - 0.45)$ ，其中， $c_1 > 0$ 和 $c_2 \in (0, 1]$ 是预先给定的常数， α_i 是第 i 次迭代时的接受概率，0.45 是一元高斯分布的最优接受概率。

3. 使用吉布斯步骤更新 β ，所需条件分布 $\beta | \theta, T$ 是高斯分布，均值 $\tilde{\xi}$ ，协方差矩阵 $\sigma^2 \tilde{\Omega}$ ，且与 y 不相关，

$$\begin{aligned}\tilde{\xi} &= \tilde{\Omega}(\Omega^{-1}\xi + D'R(\theta)^{-1}T) \\ \sigma^2\tilde{\Omega} &= \sigma^2(\Omega^{-1} + D'R(\theta)^{-1}D)^{-1}\end{aligned}$$

其中, $\sigma^2 R(\theta) = \Sigma(\theta)$ 。

4. 使用汉密尔顿蒙特卡罗算法^[50]更新条件分布 $T|\beta, \theta, y$, 用 $H(t, u)$ 表示汉密尔顿函数

$$H(t, u) = u'u/2 - \log f(t|y, \beta, \theta)$$

其中, $u \in \mathbb{R}^2$, $f(t|y, \beta, \theta)$ 表示给定 β , θ 和 y 下, T 的条件分布。根据汉密尔顿方程, 函数 $H(u, t)$ 的偏导决定 u, t 随时间 v 的变化过程,

$$\begin{aligned}\frac{dt_i}{dv} &= \frac{\partial H}{\partial u_i} \\ \frac{du_i}{dv} &= -\frac{\partial H}{\partial t_i}\end{aligned}$$

其中, $i = 1, \dots, n$, 上述动态汉密尔顿微分方程根据 leapfrog 方法离散^[50], 然后求解离散后的方程组获得近似解。

4.3.4 低秩近似算法

低秩近似算法分两部分来阐述, 第一部分讲空间高斯过程的近似, 第二部分将该近似方法应用于 SGLMM 模型。

4.3.4.1 空间高斯过程的近似

空间高斯过程 $\mathcal{S}, \mathcal{S} = \{S(x), x \in \mathbb{R}^2\}$, 对任意给定一组空间位置 $x_1, x_2, \dots, x_n, \forall x_i \in \mathbb{R}^2$, 随机变量 $S(x_i), i = 1, 2, \dots, n$ 的联合分布 $\mathcal{S} = \{S(x_1), S(x_2), \dots, S(x_n)\}$ 是多元高斯分布, 其由均值 $\mu(x) = E[S(x)]$ 和协方差 $G_{ij} = \gamma(x_i, x_j) = \text{Cov}\{S(x_i), S(x_j)\}$ 完全确定, 即 $\mathcal{S} \sim \text{MVN}(\mu_S, G)$ 。

低秩近似算法使用奇异值分解协方差矩阵 G ^[40], 将协方差矩阵 G 分解, 也意味着将空间高斯过程 \mathcal{S} 分解, 分解过程如下

$$\mathcal{S} = AZ \tag{4.27}$$

其中, $A = U\Lambda^{1/2}$, 对角矩阵 Λ 包含 G 的所有特征值, U 是特征值对应的特征向量。将特征值按从大到小的顺序排列, 取 A 的前 $m(< n)$ 列, 即可获得 \mathcal{S} 的近似 \mathcal{S}^* ,

$$\mathcal{S}^* = A_m Z \quad (4.28)$$

现在, Z 只包含 m 个相互独立的标准正态随机变量。方程 (4.28) 可以表示成

$$\mathcal{S}^* = \sum_{j=1}^m Z_j f_j(x_i), i = 1, 2, \dots, n \quad (4.29)$$

不难看出, 方程(4.29)不仅是 \mathcal{S} 的低秩近似, 还可用作空间高斯过程 \mathcal{S} 的定义。更一般地, 空间连续的随机过程 $S(x)$ 都可以表示成函数 $f_j(x)$ 和随机系数 A_j 的线性组合。

$$S(x) = \sum_{j=1}^m A_j f_j(x), \forall x \in \mathbb{R}^2 \quad (4.30)$$

若 A_j 服从零均值, 协方差为 $\text{Cov}(A_j, A_k) = \gamma_{jk}$ 的多元高斯分布, 则 \mathcal{S} 均值为 0, 协方差为

$$\text{Cov}(S(x), S(x')) = \sum_{j=1}^m \sum_{k=1}^m \gamma_{jk} f_j(x) f_k(x') \quad (4.31)$$

一般情况下, 协方差结构 (4.31) 不是平稳的, 其中, $f_k(\cdot)$ 来自一组正交基

$$\int f_j(x) f_k(x) dx = \begin{cases} 1, & i \neq j \\ 0, & i = j \end{cases}$$

随机系数 A_j 满足相互独立。

4.3.4.2 低秩近似算法

为方便叙述起见, 低秩近似算法以模型 (4.32) 为描述对象, 它是模型 (4.1) 的特殊形式, 主要区别是模型 (4.32) 中, 联系函数 $g(\mu) = \log\left(\frac{\mu}{1-\mu}\right)$

$$\log\left\{\frac{p_i}{1-p_i}\right\} = T_i = d(x_i)' \beta + S(x_i) + Z_i \quad (4.32)$$

模型 (4.32) 中的高斯过程 $\mathcal{S} = S(x)$ 可以表示成高斯噪声的卷积形式

$$S(x) = \int_{\mathbb{R}^2} K(\|x - t\|; \phi, \kappa) dB(t) \quad (4.33)$$

其中, B 表示布朗运动, $\|\cdot\|$ 表示欧氏距离, $K(\cdot)$ 表示自相关函数, 其形如

$$K(u; \phi, \kappa) = \frac{\Gamma(\kappa + 1)^{1/2} \kappa^{(\kappa+1)/4} u^{(\kappa-1)/2}}{\pi^{1/2} \Gamma((\kappa + 1)/2) \Gamma(\kappa)^{1/2} (2\kappa^{1/2} \phi)^{(\kappa+1)/2}} \mathcal{K}_\kappa(u/\phi), u > 0 \quad (4.34)$$

将方程 (4.33) 离散化，且让 r 充分大，以获得低秩近似^[51]

$$S(x) \approx \sum_{i=1}^r K(\|x - \tilde{x}_i\|; \phi, \kappa) U_i \quad (4.35)$$

式(4.35)中， $(\tilde{x}_1, \dots, \tilde{x}_r)$ 表示空间网格的格点， U_i 是独立同分布的高斯变量，均值为 0，方差 σ^2 。特别地，尺度参数 ϕ 越大时，空间曲面越平缓，如图 3.2 所示，在格点数 r 比较少时也能得到不错的近似效果。此外，空间格点数 r 与样本量 n 是独立的，因此，低秩近似算法在样本量比较大时，计算效率还比较高。

注意到平稳空间高斯过程 $S(x)$ 经过方程 (4.35) 的近似已不再平稳。通过乘以

$$\frac{1}{n} \sum_{i=1}^n \sum_{j=1}^m K(\|\tilde{x}_j - \tilde{x}_i\|; \phi, \kappa)^2$$

来调整 σ^2 。事实上，调整后的 σ^2 会更接近于高斯过程 $S(x)$ 的实际方差。

低秩近似的关键是对高斯过程 \mathcal{S} 的协方差矩阵 $\Sigma(\theta)$ 做降维分解，这对 $\Sigma(\theta)$ 的逆和行列式运算是非常重要的，在计算之前，将 $K(\theta)$ 表示为 $n \times r$ 的核矩阵，它是由自相关函数（或称核函数）决定的空间距离矩阵，协方差矩阵 $\Sigma(\theta) = \sigma^2 K(\theta) K(\theta)' + \tau^2 I_n$ ， I_n 是 $n \times n$ 的单位矩阵。根据 Woodbury 公式可得

$$\Sigma(\theta)^{-1} = \sigma^2 \nu^{-2} (I_n - \nu^{-2} K(\theta) (\nu^{-2} K(\theta)' K(\theta) + I_r)^{-1} K(\theta)')$$

其中， $\nu^2 = \tau^2 / \sigma^2$ ，求 n 阶方阵 $\Sigma(\theta)$ 的逆变成求 r 阶方阵的逆，从而达到了降维的目的。下面再根据 Sylvester 行列式定理计算 $\Sigma(\theta)$ 的行列式 $|\Sigma(\theta)|$

$$\begin{aligned} |\Sigma(\theta)| &= |\sigma^2 K(\theta) K(\theta)' + \tau^2 I_n| \\ &= \tau^{2n} |\nu^{-2} K(\theta)' K(\theta) + I_r| \end{aligned}$$

类似地，行列式运算的维数从 $n \times n$ 降到了 $r \times r$ 。

4.4 贝叶斯 STAN-MCMC 算法

介绍先验分布^[52]

4.4.1 算法提出的背景和意义

贝叶斯 MCMC 算法是一个计算密集型的算法，高效的实现对理论和应用都有非常重要的意义。因此，早在 1989 年剑桥大学研究人员开发出了 Windows 上的应用程序 WinBUGS，并被广泛使用。随着个人电脑的普及、Linux 和 MacOS 系统的蓬勃发展，只能运行于 Windows 系统上的 WinBUGS 逐渐落后于时代，并在 2008 年宣布停止开

发。随后, OpenBUGS 以开源的开发方式重现了 WinBUGS 的功能, 还可跨平台运行, 弥补了 WinBUGS 的不足, 而后又出现了同类的开源软件 JAGS。无论是 OpenBUGS 还是 JAGS 都无法适应当代计算机硬件的迅猛发展, 它们的设计由于历史局限性, 已经无法满足在兼容性、扩展性和高效性方面的要求。因此, 哥伦比亚大学的统计系以开源的方式开发了新一代贝叶斯推断子程序库 Stan, 它与前辈们最明显的最直观的不同在于, 它不是一个像 WinBUGS/OpenBUGS/JAGS 那样的软件有菜单窗口, Stan 是一种概率编程语言^[29], 可以替代 BUGS (Bayesian inference Using Gibbs Sampling)^[53] 作为 MCMC 算法的高效实现。相比较于同类软件, Stan 的优势有: 底层完全基于 C++ 实现; 拥有活跃和快速发展的社区; 支持在 CPU/GPU 上大规模并行计算; 独立于系统和硬件平台; 提供多种编程语言的接口, 如 PyStan、RStan 等等。在大数据的背景下, 拥有数千台服务器的企业越来越多, 计算机资源达到前所未有的规模, 这为 Stan 的广泛应用打下了基础, 我们基于 R 语言编程接口实现求解 SGLMM 模型的贝叶斯 MCMC 算法 (以下简称 STAN-MCMC 算法)。

4.4.2 STAN-MCMC 算法实现过程

为了与本章第 4.3.3 节提出的贝叶斯 MCMC 算法比较, 我们基于它编写了程序。目前, 我与 Paul Bürkner 一起开发了 **brms** 包^[54], 主要工作是修复程序调用和文档书写错误, 特别是与求解 SGLMM 模型相关的 `gp` 函数, 相关细节见 **brms** 的 Github 开发仓库。

在 SGLMM 模型下, STAN-MCMC 算法, 先从条件分布 $[S|\theta, \beta, Y]$ 抽样, 然后是 $[\theta|S]$, 最后是 $[\beta|S, Y]$, 具体步骤如下:

1. 选择初始值 θ, β, S , 如 β 的初始值来自正态分布, θ 的初始值来自对数正态分布;
2. 更新参数向量 θ : (i) 从指定的先验分布中均匀抽取新的 θ' ; (ii) 以概率 $\Delta(\theta, \theta') = \min \left\{ \frac{p(S|\theta')}{p(S|\theta)}, 1 \right\}$, 否则不改变 θ ;
3. 更新高斯过程 S 的取值: (i) 抽取新的值 S'_i , 向量 S 的第 i 值来自一元条件高斯密度 $p(S'_i|S_{-i}, \theta)$, S'_{-i} 表示移除 S 中的第 i 个值; (ii) 以概率 $\Delta(S_i, S'_i) = \min \left\{ \frac{p(y_i|S'_i, \beta)}{p(y_i|S_i, \beta)}, 1 \right\}$ 接受 S'_i , 否则不改变 S_i ; (iii) 重复 (i) 和 (ii) $\forall i = 1, 2, \dots, n$;
4. 更新模型系数 β : 从条件密度 $p(\beta'|S)$ 以概率

$$\Delta = \min \left\{ \frac{\prod_{j=1}^n p(y_j|s_j, \beta') p(\beta|\beta')}{\prod_{j=1}^n p(y_j|s_j, \beta) p(\beta'|\beta)}, 1 \right\}$$

接受 β' , 否则不改变 β 。

5. 重复步骤 2, 3, 4 既定的次数。获得参数 β, θ 的迭代序列。

即到达平稳的初始位置, 然后根据后续的平稳序列采样, 获得各参数的后验分布的样本及对应估计值。程序实现的主要步骤 (以 R 语言接口 **rstan** 和 **brms** 为例说明):

1. 安装 C++ 编译工具，如果在 Windows 平台上，就从 R 官网下载安装 RTools，它包含一套完整的 C++ 开发工具。然后添加 gcc/g++ 编译器的路径到系统环境变量。如果在 Linux 系统上，这些工具都是自带的，环境变量也不用配置，减少了很多麻烦。其它配置细节见 Stan 开发官网¹。
2. 在 R 软件控制台，安装 **rstan** 和 **brms** 包以及相关依赖包。
3. 加载 **rstan** 和 **brms** 包，配置参数如下：

```
# 加载程序包
library(rstan)
library(brms)
# 以并行方式运行STAN-MCMC算法，指定 CPU 的核心数
options(mc.cores = parallel::detectCores())
# 将编译后的模型写入磁盘，可防止重新编译
rstan_options(auto_write = TRUE)
```

4. 调用 brms 包的 brm 函数

```
fit.binomial <- brm(formula = y | trials(units.m) ~
  0 + intercept + x1 + x2 + gp(d1, d2),
  data = sim_binom_data,
  prior = set_prior("normal(0,10)", class = "b"),
  chains = 4, thin = 5, iter = 15000, family = binomial()
)
```

brm 函数可设置的参数有几十个，下面仅列出部分

1. *formula*：设置 SGLMM 模型的结构，其中波浪号左侧是响应变量，trials 表示在每个采样点抽取的样本量；波浪号右侧 0 + intercept 表示截距项，x1 和 x2 表示协变量，gp(d1, d2) 表示采样坐标为 (d1, d2) 自相关函数为幂指数族的平稳高斯过程
2. *data*：SGLMM 模型拟合的数据 sim_binom_data
3. *prior*：设置 SGLMM 模型参数的先验分布
4. *chains*：指定同时生成马尔科夫链的数目
5. *iter*：STAN-MCMC 算法总迭代次数
6. *thin*：burn-in 位置之后，每隔 thin 的间距就采一个样本
7. *family*：指定响应变量服从的分布，如二项分布，泊松分布等

¹<https://github.com/stan-dev/rstan/wiki>

4.4.3 求解模型的 R 包

R 语言作为免费自由的统计计算和绘图环境,因其更新快,社区庞大,扩展包更是超过了 13000 个,提供了大量前沿统计方法的代码实现。如 **spBayes** 包使用贝叶斯 MCMC 算法估计 SGLMM 模型的参数^[55]; **coda** 包诊断马尔科夫链的平稳性^[56]; **MCMCvis** 包分析和可视化贝叶斯 MCMC 算法的输出,提取模型参数,转化 JAGS、Stan 和 BUGS 软件的输出结果到 R 对象,以利后续分析; **geoR** 包在空间线性混合效应模型上实现了贝叶斯 MCMC 算法^[11]; **geoRglm** 包在 **geoR** 包的基础上将模型范围扩展到 SGLMM 模型^[57]; **glmmBUGS** 包提供了 WinBUGS、OpenBUGS 和 JAGS 软件的统一接口^[58]; **gstat** 包迁移自 S 语言,提供了克里金插值方法^[39]。目前, R 语言社区提供的求解 SGLMM 模型的 R 包和功能实现,见表 4.1。

表 4.1: 求解空间广义线性混合效应模型的 R 包功能比较: 加号 + 表示可用, 减号 - 表示不可用, 星号 * 标记的只在空间线性混合效应模型下可用

	PrevMap	geoR	geoRglm	geostatsp	geoBayes	spBayes
二项模型	+	-	+	+	+	+
似然推断	+	-	+	-	-	-
贝叶斯推断	+	-	+	+	+	+
块金效应	+	-	+	+	+	-
低秩近似	+	-	-	-	-	+
分层模型	+	-	-	+	-	-
非线性预测	+	+	+	-	+	+
多元预测	+	+	+	-	+	+
各向异性	-	+	+	+	-	-
非梅隆核	-	+	+	-	+	+

5 数值模拟

空间广义线性混合效应模型在广义线性混合效应模型基础上添加了与空间位置相关的随机效应,这种随机效应在文献中常称为空间效应^[3]。它与采样点的位置、数量都有关系,其中采样点的位置决定空间过程的协方差结构,而采样点的数量决定空间效应的维度,从而导致空间广义线性混合效应模型比普通的广义线性混合效应模型复杂。作为过渡,我们在第 5.1.1 和 5.1.2 节模拟了一维和二维平稳高斯过程。第 5.2 节模拟 SGLMM 模型,分两个小节展开叙述,第 5.2.1 小节模拟响应变量服从二项分布的情形,第 5.2.2 小节模拟响应变量服从泊松分布的情形,在这两个小节里,比较了蒙特卡罗极大似然算法,低秩近似算法,贝叶斯马尔科夫链蒙特卡罗算法和贝叶斯 STAN-MCMC 算法的性能差异。

模拟空间广义线性混合效应模型 (5.1) 分三节进行,第一节模拟空间高斯过程 $S(x)$,第二节模拟响应变量 Y 服从二项分布,第三节模拟响应变量 Y 服从正态分布。空间高斯过程在模型 (5.1) 中作为随机效应存在,不同于一般随机效应的地方在于它与样本的空间位置有关,一般地,假定 $S(x)$ 服从 N 元高斯分布 $N(\mu_S, G)$, $x = (d_1, d_2) \in \mathbb{R}^2$, $\mu_S = \mathbf{0}_{N \times 1}$, $G_{(ij)} = \text{Cov}(S(x_i), S(x_j)) = \sigma^2 * \rho(u_{ij})$, $S(x)$ 的相关函数 $\rho(u_{ij}) = \exp(-u_{ij}/\phi)$, $u_{ij} \equiv \|x_i - x_j\|_2$, 其中 ϕ 和 σ^2 都是未知待估参数。可见采样点的位置 (d_1, d_2) 和相关函数 $\rho(u)$ 一起决定空间高斯过程的形式,并且 $S(x)$ 的维度就是采样点的数目 N 。这样通常导致空间效应的维度比协变量的数目大很多,模型 (5.1) 的估计问题也比一般的广义线性混合效应模型困难。

$$g(\mu_i) = d(x_i)' \beta + S(x_i) + Z_i \quad (5.1)$$

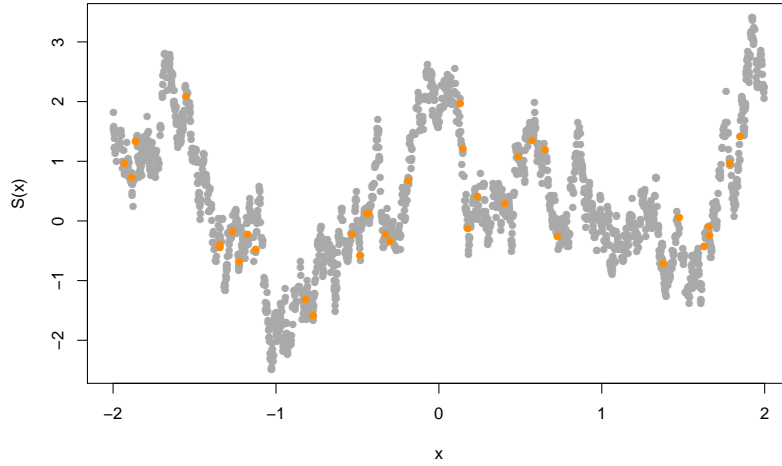
5.1 平稳空间高斯过程

5.1.1 一维平稳空间高斯过程

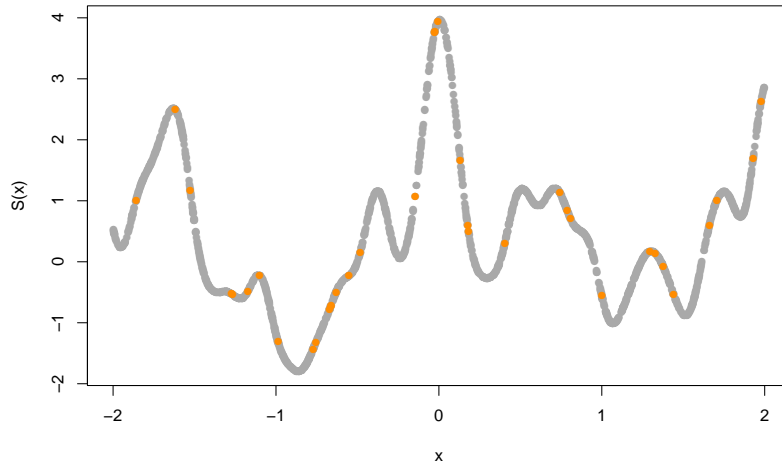
一维情形下,平稳高斯过程 $S(x)$ 的协方差函数采用幂指数型,见公式 (5.3),当 $\kappa = 1$ 时,为指数型,见公式 (5.3)。分 $\kappa = 1$ 和 $\kappa = 1$,模拟两个一维平稳空间高斯过程,协方差参数均为 $\sigma^2 = 1$, $\phi = 0.15$,均值向量都是 $\mathbf{0}$,在 $[-2, 2]$ 的区间上,产生 2000 个服从均匀分布的随机数,由这些随机数的未知和协方差函数公式 (5.2) 或 (5.3) 计算得到 2000 维的高斯分布的协方差矩阵 G ,为保证协方差矩阵的正定性,在矩阵对角线上添加扰动 1×10^{-12} ,然后即可根据 Cholesky 分解该对称正定矩阵,得到下三角块 L ,使得 $G = L \times L^\top$,再产生 2000 个服从标准正态分布的随机向量 η ,而 $L\eta$ 即为所需的服从平稳高斯过程的一组随机数。

$$\text{Cov}(S(x_i), S(x_j)) = \sigma^2 \exp \left\{ -\frac{|x_i - x_j|}{\phi} \right\} \quad (5.2)$$

$$\text{Cov}(S(x_i), S(x_j)) = \sigma^2 \exp \left\{ -\left(\frac{|x_i - x_j|}{\phi} \right)^\kappa \right\}, 0 < \kappa \leq 2 \quad (5.3)$$



(a) 平稳空间高斯过程 $S(x)$ 的协方差函数是指数型，均值向量为 $\mathbf{0}$ ，协方差参数 $\sigma^2 = 1$ ， $\phi = 0.15$



(b) 平稳空间高斯过程 $S(x)$ 的协方差函数是幂二次指数型，均值向量为 $\mathbf{0}$ ，协方差参数 $\sigma^2 = 1$ ， $\phi = 0.15$ ， $\kappa = 2$

图 5.1: 模拟一维平稳空间高斯过程，协方差函数分别为指数型 (5.2) 和幂二次指数型 (5.3)，均值为 $\mathbf{0}$ ，协方差参数 $\sigma^2 = 1$ ， $\phi = 0.15$ ，横坐标表示采样的位置，纵坐标是目标值 $S(x)$ ，图中 2000 个灰色点表示服从相应随机过程的随机数，橘黄色点是从中随机选择的 36 个点。

根据定理 2.4，指数型协方差函数的平稳高斯过程在原点连续但是不可微，而幂二次指数型协方差函数在原点无穷可微，可微性越好图像上表现越光滑，对比图 5.1 的两

个子图, 可以看出, 在协方差参数 $\sigma^2 = 1$, $\phi = 0.15$ 相同的情况下, κ 越大越光滑。

5.1.2 二维平稳空间高斯过程

二维情形下, 在规则平面上模拟平稳高斯过程 $\mathcal{S} = S(x), x \in \mathbb{R}^2$, 其均值向量为零向量 $\mathbf{0}$, 协方差函数为指数型 (见公式 (5.2)), 协方差参数 $\phi = 1, \sigma^2 = 1$ 。在单位平面区域为 $[0, 1] \times [0, 1]$ 模拟服从上述二维空间平稳高斯过程, 不妨将此区域划分为 6×6 的小网格, 而每个格点作为采样的位置, 共计 36 个采样点, 在这些采样点上的观察值即为目标值 $S(x)$ 。

类似本章第 5.1.1 节模拟一维平稳空间过程的步骤, 首先根据采样点位置坐标和协方差函数 (5.3) 计算得目标空间过程的 \mathcal{S} 协方差矩阵 G , 然后使用 R 包 **MASS** 提供的 `mvrnorm` 函数产生多元正态分布随机数, 与 5.1.1 节不同的是这里采用特征值分解, 即 $G = L\Lambda L^\top$, 与 Cholesky 分解相比, 特征值分解更稳定些, 但是 Cholesky 分解更快, Stan 即采用此法, 后续过程与一维模拟一致。模拟获得的随机数用图 5.2 表示, 格点上的值即为平稳空间高斯过程在该点的取值 (为方便显示, 已四舍五入保留两位小数)。

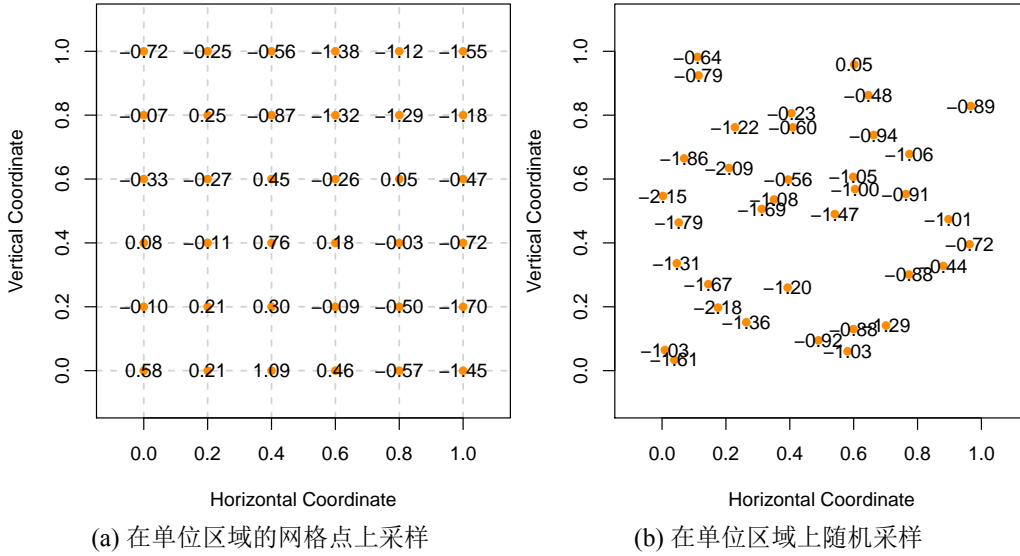


图 5.2: 模拟二维平稳空间高斯过程, 自相关函数为指数形式, 水平方向为横坐标, 垂直方向为纵坐标, 图中的橘黄色点是采样的位置, 其上的数字是目标值 $S(x)$

同 5.1.1 节, 二维平稳空间高斯过程 $S(x)$ 的协方差函数也可以为更一般的梅隆型, 如公式 (5.4) 所示。

$$\rho(u) = \sigma^2 \{2^{\kappa-1} \Gamma(\kappa)\}^{-1} (u/\phi)^\kappa \mathcal{K}_\kappa(u/\phi) \quad (5.4)$$

且在区域 $[0, 1] \times [0, 1]$ 上也可以随机采点, 如图 5.2 的右子图所示。

模拟平稳空间高斯过程的其它 R 包实现方法: Paulo J. Ribeiro Jr. 和 Peter J. Diggle 开发了 **geoR** 包^[11], 提供的 `grf` 函数除了实现 Cholesky 分解, 还实现了奇异值分解, 特

征值分解协方差矩阵的算法。当采样点不太多时，Cholesky 分解已经足够好，下面的第 5.2 节对平稳空间高斯过程的数值模拟即采用此法，当采样点很多，为了加快模拟的速度，可以选用 Martin Schlather 等开发的 **RandomFields** 包^[59]，内置的 **GaussRF** 函数实现了用高斯马尔科夫随机场近似平稳空间高斯过程的算法，此外，Håvard Rue 等 (2009 年)^[13] 也实现了从平稳高斯过程到高斯马尔科夫随机场的近似算法，开发了更为高效的 INLA 程序库^[60]，其内置的近似程序得到了广泛的应用^[61;62;63]。

5.2 空间广义线性混合效应模型

5.2.1 响应变量服从二项分布

响应变量服从二项分布 $Y_i \sim \text{Binomal}(m_i, p(x_i))$ ，在位置 x_i 处，以概率 $p(x_i)$ 重复抽取了 m_i 个样本，总样本数 $M = \sum_{i=1}^N m_i$ ， N 是采样点的个数，二项空间广义线性混合效应模型为 (5.5)，联系函数为 $g(\mu_i) = \log\{\frac{p(x_i)}{1-p(x_i)}\}$ ， $S(x)$ 是均值为 $\mathbf{0}$ ，协方差函数为 $\text{Cov}(S(x_i), S(x_j)) = \sigma^2 \{2^{\kappa-1} \Gamma(\kappa)\}^{-1} (u/\phi)^\kappa K_\kappa(u/\phi)$ ， $\kappa = 0.5$ 的平稳空间高斯过程。

$$g(\mu_i) = \log\left\{\frac{p(x_i)}{1-p(x_i)}\right\} = \alpha + S(x_i) \quad (5.5)$$

固定效应参数 $\alpha = 0$ ，协方差参数记为 $\boldsymbol{\theta} = (\sigma^2, \phi) = (0.5, 0.2)$ ，采样点数目为 $N = 64$ ，每个采样点抽取的样本数 $m_i = 4, i = 1, 2, \dots, 64$ 。首先模拟平稳空间高斯过程 $S(x)$ ，在单位区域 $[0, 1] \times [0, 1]$ 划分为 8×8 的网格，格点选为采样位置，用 **geoR** 包提供的 **grf** 函数产生协方差参数为 $\boldsymbol{\theta} = (\sigma^2, \phi) = (0.5, 0.2)$ 的平稳空间高斯过程，由公式 (5.5) 可知 $p(x_i) = \exp(S(x_i)) / (1 + \exp(S(x_i)))$ ，即每个格点处二项分布的概率值，然后依此概率，由 **rbinom** 函数产生服从二项分布的观察值 Y_i ， Y_i 的取值范围为 $0, 1, 2, 3, 4$ ，模拟的数据集可以用图 5.3 直观表示。

基于 Langevin-Hastings 采样器实现的马尔科夫链蒙特卡罗算法，参数 α 的先验分布选均值为 0，方差为 1 的标准正态分布，参数 ϕ 的先验分布选期望为 0.2 的指数分布，参数 σ^2 的先验分布是非中心的逆卡方分布，其非中心化参数为 0.5，自由度为 5，各参数的先验选择来自 Ole F. Christensen 和 Peter J. Ribeiro Jr. (2002 年)^[57]。Langevin-Hastings 算法运行 11000 次迭代，前 1000 次迭代用作热身 (warm-up)，后 10 万次迭代里间隔 100 采样。

$$\alpha \sim \mathcal{N}(0, 1) \quad (5.6)$$

$$\phi \sim \text{Exp}(0.2) \quad (5.7)$$

$$\sigma^2 \sim \text{Inv-}\chi^2(5, 0.5) \quad (5.8)$$

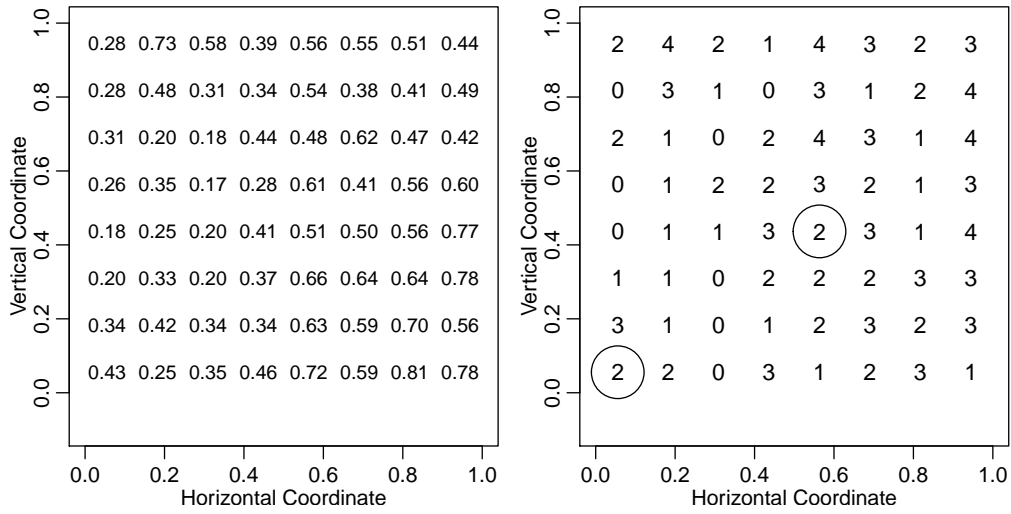


图 5.3: 左图表示二维规则平面上的平稳空间高斯过程, 格点是采样点的位置, 其上的数字是 $p(x)$ 的值, 已经四舍五入保留两位小数, 右图表示观察值 Y 随空间位置的变化, 格点上的值即为观察值 Y , 图中的两个圈分别是第 1 个 (左下) 和第 29 个 (右上) 采样点

表 5.1: Langevin-Hastings 算法与 NUTS 算法的数值模拟比较, 模型参数 α, ϕ, σ^2 的估计值 (后验分布的均值)、方差 (后验分布的方差) 和 5 个分位点 (后验分布的 5 个分位点), 采样点数目分别考虑了 $N = 36, 64, 81, 100$ 的情况, 括号内为相应参数的真值

	mean	var	2.5%	25%	50%	75%	97.5%	N
α	-0.354(0)	0.079	-0.938	-0.524	-0.361	-0.173	0.215	36
ϕ	0.121(0.2)	0.006	0.005	0.055	0.110	0.180	0.285	36
σ^2	0.683(0.5)	0.147	0.215	0.408	0.596	0.850	1.667	36
α	0.003(0)	0.089	-0.596	-0.169	0.013	0.179	0.609	64
ϕ	0.194(0.2)	0.004	0.070	0.145	0.195	0.250	0.295	64
σ^2	0.656(0.5)	0.096	0.254	0.449	0.592	0.781	1.453	64
β	-0.155(0)	0.044	-0.565	-0.284	-0.156	-0.03	0.273	81
ϕ	0.116(0.2)	0.006	0.005	0.055	0.105	0.17	0.280	81
σ^2	0.468(0.5)	0.057	0.180	0.311	0.414	0.56	1.129	81

参数 α, ϕ, σ^2 的贝叶斯估计没有显式表达式, 通常用后验分布的均值作为参数的估计, 估计的精度或者说好坏用后验分布的方差衡量, 而均方误差在参数估计取后验均值时, 就是后验方差, 所以表 5.1 不再提供估计的均方误差值, 而是提供了 5 个后验分布的分位点, 在 95% 的置信水平下, 样本分位点 0.025 和 0.975 的值组成了置信区间的上下界。64 个采样点处 $p(x_i), i = 1, \dots, 64$ 的后验分布的均值、方差、标准差和 5 个分位点见附表 7.1

5.2.2 响应变量服从泊松分布

响应变量 Y 服从泊松分布, 即 $Y_i \sim \text{Poisson}(\lambda(x_i))$, 泊松空间广义线性混合效应模型 (5.9)

	mean	var	2.5%	25%	50%	75%	97.5%
σ^2	0.883	0.083	0.446	0.681	0.833	1.036	1.610

6 数据分析

loaloe 和 rongelap 两个真实数据集分别来自 R 包 **geoRglm** 和 **PrevMap**，且被 Peter J. Diggle 等 (2007 年)^[6] 和 Ole F Christensen (2004 年)^[12] 分析过，第 6.2 节和第 6.3 节给出分析这两个数据集的模型和结果。其中，响应变量服从二项分布的空间广义线性混合效应模型拟合数据集 loaloe，而响应变量服从泊松分布的空间广义线性混合效应模型拟合数据集 rongelap。

6.1 空间线性混合效应模型

Walter W. Stroup 和 P. Stephen Baenziger (1994 年)^[64] 采用完全随机的区组设计研究小麦产量与品种等因素的关系，在 4 块肥力不同的地里都随机种植了 56 种不同的小麦，实验记录了小麦产量、品种、位置以及土地肥力等数据，José Pinheiro 和 Douglas Bates (2000 年)^[65] 将该数据集命名为 Wheat2，整理后放在 **nlme** 包里。这里利用该真实的农业生产数据构建带空间效应的线性混合效应模型，展示诊断和添加空间效应的过程。

图 6.1 按土壤肥力不同分块展示每种小麦的产量，图中暗示数据中有 block 效应，块之间也呈现异方差性，为了更好的表达这种依赖效应，可以基于经纬度信息添加与空间相关的结构 (spatial correlation structures)，相应的线性模型

$$y_{ij} = \tau_i + \epsilon_{ij}, \quad \epsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{\Lambda}) \quad (6.1)$$

其中， y_{ij} 表示第 i 种小麦在第 j 块试验田里的产量， $i = 1, \dots, 56$ ， $j = 1, \dots, 4$ 。 τ_i 表示第 i 种小麦的平均产量， ϵ_{ij} 是随机误差，假定服从均值为 0，协差阵为 $\sigma^2 \mathbf{\Lambda}$ 的正态分布

探索 $\mathbf{\Lambda}$ 的结构，先假定模型 (6.1) 的随机误差是独立且方差齐性的，即 $\mathbf{\Lambda} = \mathbf{I}$ 。用如下一行命令拟合模型

```
m1 <- gls(yield ~ variety - 1, Wheat2)
```

拟合残差的样本变差是探索误差中的空间相关性的主要工具，Variogram 方法可以获得 gls 类中的样本变差，即

```
Variogram(m1, form = ~latitude + longitude)
```

图 6.2 显示样本变差随空间距离有明显的增长趋势，可见空间相关的结构明显，且块金效应大约是 0.2，样本变差为 1 对应的空间距离是 31 左右。用局部多项式回归方法拟合散点，获得平滑曲线，也可以用局部加权回归散点平滑法来确定初值^[66]。

我们使用球形结构拟合这组数据中的空间结构，位置参数 ϕ 和块金效应的初始值可以从图 6.2 获取，其中 ϕ 是样本变差等于 1 时对应的空间距离。拟合模型 (6.1)

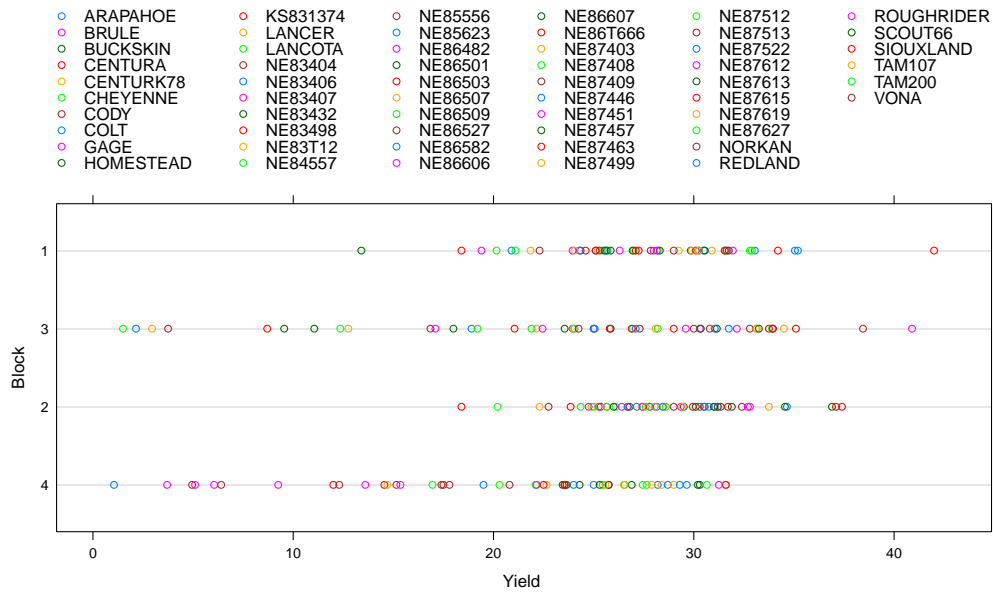


图 6.1: 小麦产量与土壤肥力的关系，图中纵轴表示试验田的 4 种类型，且土壤肥力强弱顺序是 $1 > 2 > 3 > 4$ ，横轴表示小麦产量，每块试验田都种植了 56 种小麦，图中分别以不同的颜色标识，图上方是小麦类型的编号

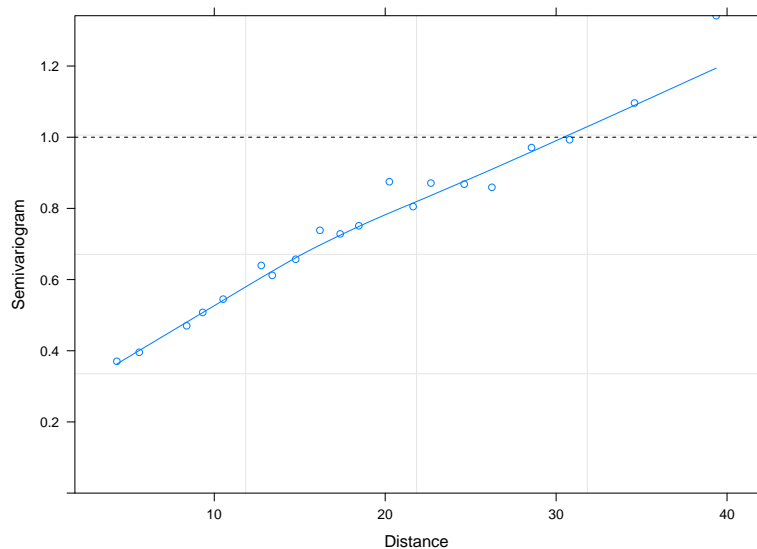


图 6.2: 样本变差散点图及局部多项式拟合的变化趋势，横坐标是小麦之间的欧氏距离，纵坐标是样本变差

```
m2 <- update(m1, corr = corSpher(
  c(31, 0.2), form = ~latitude + longitude, nugget = TRUE
))
```

用限制极大似然法做广义最小二乘拟合，空间自相关函数采用球型和二次有理型函数。似然函数对初始值很敏感，去掉样本变差中距离比较远的

表 6.1: 小麦模型比较

	自相 关函 数	$\hat{\phi}(\phi_0)$	$\hat{\tau}^2(\tau_0^2)$	$\hat{\sigma}^2(\sigma_0^2)$	log- REML
模型 I	球型	$1.515 \times 10^5(31)$	$5.471 \times 10^{-5}(0.2)$	466.785	- 533.418
模型 II	二次 有理 型	13.461(13)	0.193(0.2)	8.847	- 532.639
模型 III	球型	27.457(28)	0.209(0.2)	7.410	- 533.931

说明两件事，其一选择合适的自相关函数可以取得好的拟合效果，其二算法对初值很敏感，选择合适的初值很重要。

标准化残差拟合图

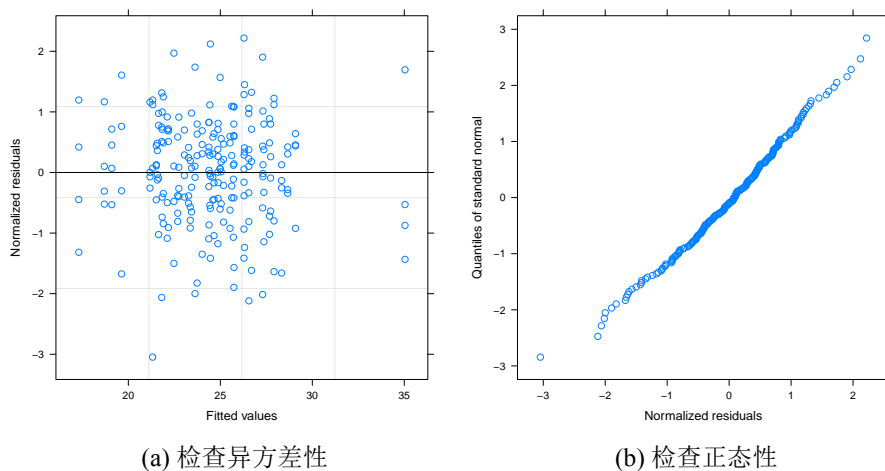


图 6.3: 模型诊断

6.2 喀麦隆及周边地区盘尾丝虫病的空间分布

盘尾丝虫病是由一种可致盲的热带疾病，非洲盘尾丝虫病控制项目 APOC (African Programme for Onchocerciasis Control) 搜集了 $N = 168$ 个村庄的 $M = 21938$ 个血液样本，每个村庄抽取的样本量为 $m_i = \text{NO_EXAM}$ ，其中感染了的 NO_INF 人，在该村庄（坐标 x_i ）观察到的感染比例 $p(x_i) = \text{NO_INF}/\text{NO_EXAM}$ ，在村庄 1 公里的范围内添加了周围环境的指标，有从美国地质调查局获得的海拔信息 ELEVATION (<https://www.usgs.gov/>) 和卫星在 1999 年至 2001 年间测得的植被绿色度数据 NDVI (<http://free.vgt.vito.be>)，NDVI 分四个指标，分别是所有 NDVI 的平均值 MEAN9901，最大值 MAX9901，最小值 MIN9901，标准差 STDEV9901。样本采集的区域如图 6.4 所示

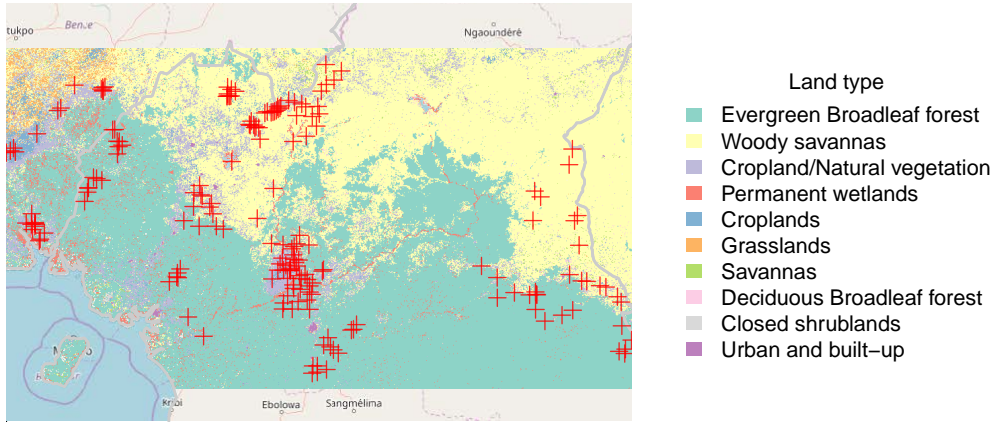


图 6.4: 红色加号标注样本所在的村庄

为了分析 loaloa 数据集，我们建立响应变量 Y 服从二项分布的空间广义线性混合效应模型

$$\log \left\{ \frac{p(x_i)}{1 - p(x_i)} \right\} = \beta_0 + \beta_1 \times \text{ELEVATION}_i + \beta_2 \times \text{MEAN9901}_i + \beta_3 \times \text{MAX9901}_i + \beta_4 \times \text{MIN9901}_i + \beta_5 \times \text{STDEV9901}_i + S(x_i)$$

其中， β_0 是截距， $\beta_1, \beta_2, \beta_3, \beta_4, \beta_5$ 是各指标的系数， $Y_i \sim \text{Binomial}(m_i, p(x_i))$ ，平稳空间高斯过程 $\mathcal{S} = S(x), x = (\text{LONGITUDE}, \text{LATITUDE}) \in \mathbb{R}^2$ 的均值为 0，自协方差函数为

$$\text{Cov}(S(x_i), S(x_j)) = \sigma^2 \left\{ 2^{\kappa-1} \Gamma(\kappa) \right\}^{-1} (u_{ij}/\phi)^\kappa K_\kappa(u_{ij}/\phi), \kappa = 2$$

在实际数据分析中，选择一组合适的初始值可以缩短各算法迭代的过程。我们分两步获取 $\beta = (\beta_0, \beta_1, \beta_2, \beta_3, \beta_4, \beta_5)$ 和 $\theta = (\sigma^2, \phi)$ 的初始值。第一步，离散自协方差函数中的 κ ，再调用 **PrevMap** 包中的 `shape.matern` 函数选择一个 κ ；第二步，在去掉

空间效应 $S(x)$ 的情况下, 以广义线性模型拟合数据得到 β 的初始估计值。然后分别使用贝叶斯 MCMC 算法和贝叶斯 STAN-MCMC 算法估计参数 $\beta = (\beta_0, \beta_1, \beta_2, \beta_3, \beta_4, \beta_5)$ 和 $\theta = (\sigma^2, \phi)$, 结果如表 6.3 和表 6.4 所示。

表 6.2: MCML 算法估计模型的参数

参数	估计	标准差	标准误
β_0	-11.120	1.447	8.268e-03
β_1	-0.001	3.155e-04	1.023e-05
β_2	13.513	2.223	4.877e-02
β_3	1.454	2.013	3.094e-02
β_4	-0.576	1.315	1.016e-02
β_5	11.216	5.181	5.321e-02
σ^2	1.171	0.272	1.300e-03
ϕ	0.486	0.353	2.344e-03

通过表 6.2 的 P 值, 可以看出 $\beta_0, \beta_1, \beta_2$ 是显著的, 分别对应模型的截距项, 海拔 (ELEVATION) 和 NDVI 的平均值 (MEAN9901), 在这组数据中, 刻画 NDVI 指标使用平均值比较能体现村庄周围植被的整体绿色度, 而最大值 MAX9901, 最小值 MIN9901, 标准差 STDEV9901 与影响不显著。

表 6.3: 贝叶斯 MCMC 算法估计模型的参数

参数	估计	标准差	均方误差
β_0	-11.562	0.681	5.563e-03
β_1	-8.721e-04	1.415e-04	1.155e-06
β_2	8.426	2.064	1.686e-02
β_3	4.857	1.334	1.090e-02
β_4	0.233	1.367	1.116e-02
β_5	11.087	3.565	2.911e-02
σ^2	1.096	0.170	1.390e-03
ϕ	3.675	0.582	4.754e-03

表 6.4: 贝叶斯 STAN-MCMC 算法估计模型的参数

参数	估计	标准差	均方误差
β_0	-12.878	1.460	3.515e-03

参数	估计	标准差	均方误差
β_1	-6.596e-04	3.188e-04	7.733e-07
β_2	9.647	2.136	5.222e-03
β_3	4.380	1.993	5.026e-03
β_4	-0.102	1.159	2.822e-03
β_5	11.216	5.539	1.343e-02
σ^2	1.027	0.09	2.171e-04
ϕ	1.157e-02	1.292e-03	3.244e-06

比较表 6.3 和表 6.4，发现贝叶斯 STAN-MCMC 算法比贝叶斯 MCMC 算法稍好一点，多半的参数估计的标准差要小一些，均方误差普遍要小一点。

6.3 朗格拉普岛放射性残留物的空间分布

Peter J. Diggle 等 (1998 年)^[3] 提出蒙特卡罗极大似然方法估计不带块金效应的响应变量服从泊松分布的空间广义混合效应模型 (见模型 (6.2)) 的参数，分析了朗格拉普岛上核污染浓度的空间分布，后来，Ole F Christensen (2004 年)^[12] 发现该核污染数据集中存在不能被泊松分布解释的残差，因此在 Peter J. Diggle 等 (1998 年)^[3] 的基础上添加了块金效应，见模型 (6.3)，同样使用蒙特卡罗极大似然方法估计了模型中的参数。

朗格拉普岛位于南太平洋上，是马绍尔群岛的一部分，二战后，美国在该岛上进行了核武器测试，核爆炸后产生的放射性尘埃笼罩了全岛，目前该岛仍然不适合人类居住，只有经批准的科学研究人员才能登岛，核污染残留浓度是其中一项^[67]。rongelap 数据集包含伽马射线在 $N = 157$ 站点不同时间间隔的放射量，伽马射线在时间间隔 units.m 内放射的粒子数目为 y ，d1 和 d2 分别为测量坐标。建立 Poisson-SGLMM 模型

$$\log\{\lambda(x_i)\} = \beta + S(x_i) \quad (6.2)$$

$$\log\{\lambda(x_i)\} = \beta + S(x_i) + Z_i \quad (6.3)$$

其中， β_0 是截距，响应变量 $Y_i \sim \text{Poisson}(\lambda(x_i))$ ，平稳空间高斯过程的自协方差函数为 $\text{Cov}(S(x_i), S(x_j)) = \sigma^2 \exp(-\|x_i - x_j\|_2 / \phi)$ ， $\text{Var}(Z_i) = \tau^2, i = 1, \dots, N = 157$

蒙特卡罗极大似然算法迭代的初值 $\beta_0 = 6.2, \sigma_0^2 = 2.40, \phi_0 = 340, \tau_{rel}^2 = 2.074$ ，模拟次数为 30000 次，前 10000 次迭代视为预热阶段 (warm-up) 而不用参数的估计，其后每隔 20 次迭代采一个样本点，存储各模型参数的迭代值，其间模拟条件分布 $[S|y]$ 产生蒙特卡罗马尔科夫链使用了 Langevin-Hastings 算法^[68]。

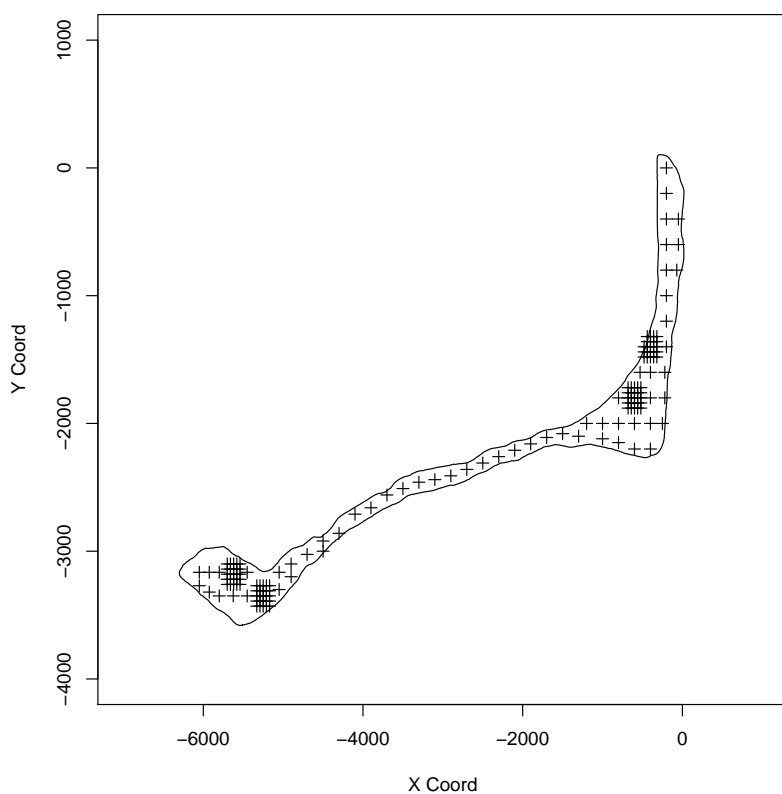


图 6.5: 朗格拉普岛上采样点的位置，加号 + 表示采样的位置

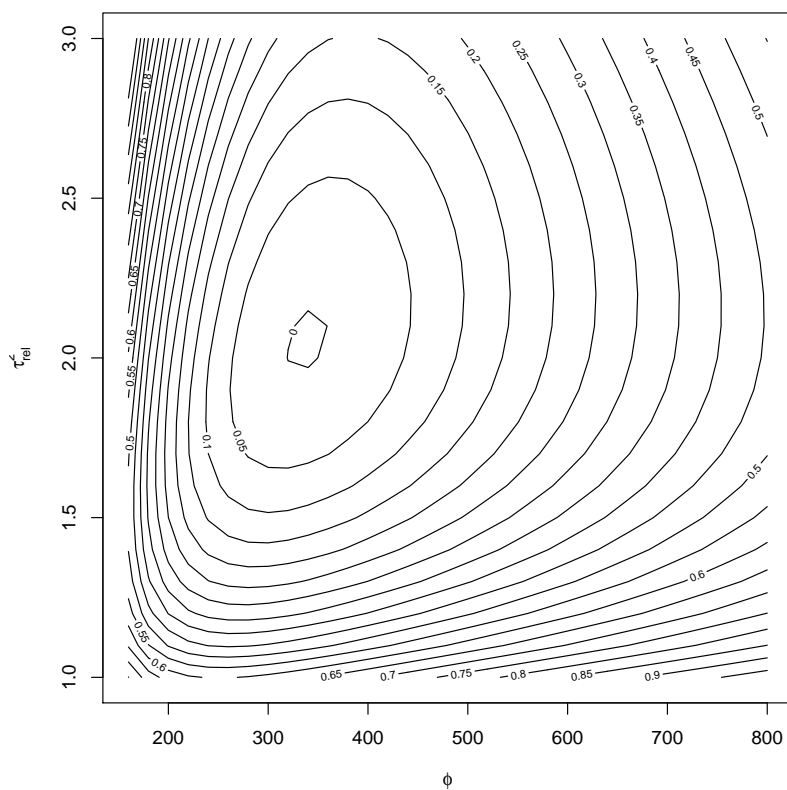


图 6.6: 关于 ϕ 和相对块金效应 $\tau_{rel}^2 = \tau^2/\sigma^2$ 的剖面似然函数曲面，平稳空间高斯过程的协方差函数选用指数型

表 6.5: 蒙特卡罗极大似然估计模型 (6.3) 的参数, 其中块金效应的估计值 $\hat{\tau}^2 = \hat{\sigma}^2 \times \hat{\tau}_{rel}^2 = 4.929$

参数	$\hat{\beta}$	$\hat{\sigma}^2$	$\hat{\phi}$	$\hat{\tau}_{rel}^2$	$\log L_m$
初始值	6.200	2.400	340.000	2.074	-
估计值	6.190	2.401	338.126	2.053	2.892×10^{-3}

6.4 冈比亚儿童疟疾流行强度的空间分布

5 岁以下儿童在非洲冈比亚的 65 个村庄里调查疟疾感染情况, 依冈比亚狭长的地带和横穿东西全境的冈比亚河, 将其看作东、中和西三个部分, 东部是河流上游, 西部是河流下游相应地, 村庄分为 5 个区域, 西部两个, 南岸和北岸各一个, 中游一个在南岸, 东部两个, 也是南岸和北岸各有一个, 村庄的位置在图中的黑点标记。

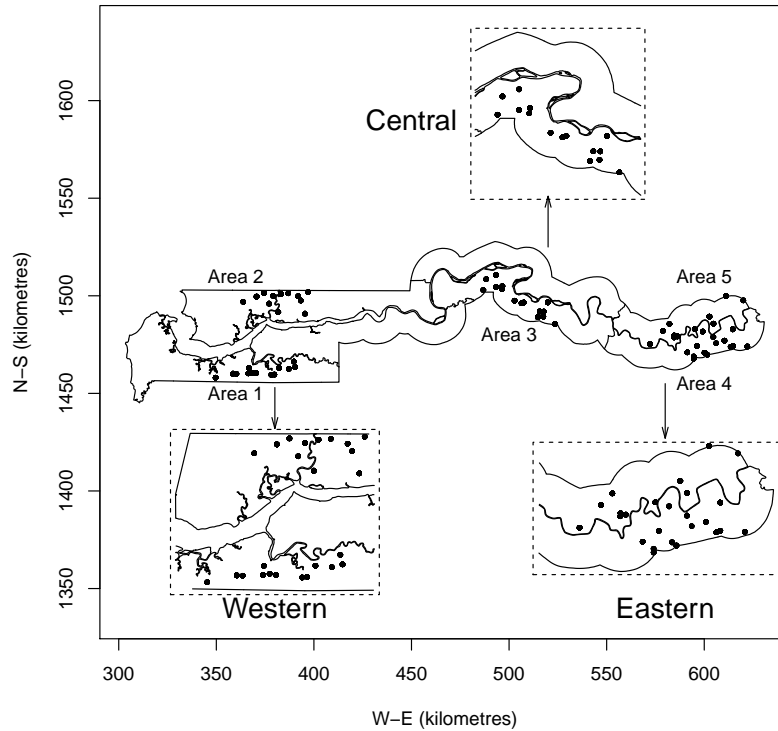


图 6.7: 冈比亚 65 个村庄的空间位置, 黑点标记, 插入的图是响应区域的放大版, 由箭头指向

$$\log\{p_{ij}/(1 - p_{ij})\} = \alpha + \beta^T z_{ij} + S(x_i) \quad (6.4)$$

表 6.6: 不带块金效应的模型 (6.4) 参数估计和 95% 的置信区间

参数	2.5% 分位点	97.5% 分位点	均值 (mean)	中位数 (median)
α	-2.966473	2.624348	-0.131214	-0.077961
$\beta_1(\text{age})$	0.000455	0.000933	0.000689	0.000685
$\beta_2(\text{untreated})$	-0.673143	-0.042011	-0.357825	-0.359426
$\beta_3(\text{treated})$	-0.753803	0.088418	-0.32954	-0.325853
$\beta_4(\text{greenness})$	-0.085675	0.047924	-0.020068	-0.020834
$\beta_5(\text{PHC})$	-0.787913	0.129883	-0.344846	-0.349915
$\beta_6(\text{area 2})$	-1.14419	0.51023	-0.324665	-0.331634
$\beta_7(\text{area 3})$	-1.40862	0.558616	-0.5321	-0.559229
$\beta_8(\text{area 4})$	-0.109472	2.425342	1.049441	1.016969
$\beta_9(\text{area 5})$	0.164828	2.606357	1.309553	1.325129
σ^2	0.311756	1.050227	0.585592	0.553477
ϕ	0.915789	10.20069	2.522294	1.422975
κ	0.079522	2.784646	1.084108	0.937436

表 6.6 来自 Peter Diggle 等 (2002)^[5] 的结果

$$\log\{p_{ij}/(1 - p_{ij})\} = \alpha + \beta^\top z_{ij} + U_i + S(x_i) \quad (6.5)$$

表 6.7: 带块金效应的模型 (6.5) 参数估计和 95% 的置信区间

参数	2.5% 分位点	97.5% 分位点	均值 (mean)	中位数 (median)
α	-4.232073	1.114734	-1.664353	-1.696228
$\beta_1(\text{age})$	0.000442	0.000918	0.000677	0.000676
$\beta_2(\text{untreated})$	-0.684407	-0.083811	-0.383750	-0.385772
$\beta_3(\text{treated})$	-0.778149	0.054543	-0.355655	-0.355632
$\beta_4(\text{greenness})$	-0.039706	0.071505	-0.018833	0.020079
$\beta_5(\text{PHC})$	-0.791741	0.180737	-0.324738	-0.322760
ν^2	0.000002	0.515847	0.117876	0.018630
σ^2	0.240826	1.662284	0.793031	0.740790
ϕ	1.242164	53.351207	11.653717	7.032258
κ	0.150735	1.955524	0.935064	0.830548

将投影坐标系 (UTM) 转化为地理坐标系 (WGS84)

7 总结展望

本文研究的重点是估计空间广义线性混合效应模型参数的算法，包括蒙特卡罗最大似然算法（简称 MCML 算法）、低秩近似算法（简称 Low-Rank 算法）、贝叶斯 MCMC 算法和贝叶斯 STAN-MCMC 算法。在相同设置下，比较了它们的性能，结论是 MCML 算法比 Low-Rank 算法收敛要慢很多，但是准确度比较它高，Low-Rank 可以增加采样点的数目快速接近 MCML 算法的精度，代价是运行时间会显著增加；基于 Stan 实现的贝叶斯 STAN-MCMC 算法比贝叶斯 MCMC 算法快很多，但是需要较多的计算机资源才能体现它的优势。目前，得益于计算机硬件的快速发展，无论是 CPU 还是 GPU，甚至是谷歌专为机器学习量身打造的 TPU 也都进入量产了，分布式的多机并行算法一定是大规模数据集和复杂模型的出路，所以将可并行的传统算法用 Stan 重写是有实际应用价值的。

近年来，基于近似贝叶斯推断的算法也受到越来越多的关注，尤其是 INLA 算法 (Integrated Nested Laplace Approximations)，因其高效的计算性能，快速发展的 INLA 社区¹，相关软件 R-INLA 的广泛使用和日益成熟的理论。将 INLA 算法和蒙特卡罗方法结合是值得研究的方向，Stan 程序库在 GPU 上的并行已经列入开发日程²，可以预见，将估计模型参数的算法基于 Stan 实现是快速提高计算效率的捷径。

¹<http://www.r-inla.org/>

²<https://github.com/stan-dev/stan/wiki/Longer-Term-To-Do-List>

参考文献

- [1] Cressie N A C. Statistics for spatial data[M]. Rev. ed. London: John Wiley and Sons Inc., 1993: 27–104
- [2] Krige D G. A statistical approach to some basic mine valuation problems on the witwatersrand[J]. Journal of the Chemical, Metallurgical and Mining Society of South Africa, 1951, 52: 119–139.
- [3] Diggle P J, Tawn J A, Moyeed R A. Model-based geostatistics[J]. Journal of the Royal Statistical Society, Series C, 1998, 47(3): 299–350.
- [4] Thomson M C, Connor S J, D'Alessandro U, et al. Predicting malaria infection in gambian children from satellite data and bed net use surveys: the importance of spatial correlation in the interpretation of results[J]. The American Journal of Tropical Medicine and Hygiene, 1999, 61(1): 2–8.
- [5] Diggle P, Moyeed R, Rowlingson B, et al. Childhood malaria in the gambia: a case-study in model-based geostatistics[J]. Journal of the Royal Statistical Society, Series C, 2002, 51(4): 493–506.
- [6] Diggle P J, Thomson M C, Christensen O F, et al. Spatial modelling and the prediction of loa loa risk: decision making under uncertainty[J]. Annals of Tropical Medicine and Parasitology, 2007, 101(6): 499–509.
- [7] Schlüter D K, Ndeffombah M L, Takougang I, et al. Using community-level prevalence of loa loa infection to predict the proportion of highly-infected individuals: Statistical modelling to support lymphatic filariasis and onchocerciasis elimination programs[J]. Plos Neglected Tropical Diseases, 2016, 10(12): 1–15.
- [8] Takougang I, Meremikwu M, Wandji S, et al. Rapid assessment method for prevalence and intensity of loa loa infection.[J]. Bulletin of the World Health Organization, 2002, 80(11): 852–858.
- [9] Boussinesq M, Gardon J, Kamgno J, et al. Relationships between the prevalence and intensity of loa loa infection in the central province of cameroon.[J]. Annals of Tropical Medicine and Parasitology, 2001, 95(5): 495–507.
- [10] Gardon J, Gardon-Wendel N, Demanga-Ngangue, et al. Serious reactions after mass treatment of onchocerciasis with ivermectin in an area endemic for loa loa infection.[J]. Lancet, 1997, 350(9070): 18–22.
- [11] Ribeiro Jr. P J, Diggle P J. geoR: A package for geostatistical analysis[J]. R News, 2001, 1(2): 14–18.
- [12] Christensen O F. Monte carlo maximum likelihood in model-based geostatistics[J]. Journal of Computational and Graphical Statistics, 2004, 13(3): 702–718.

- [13] Rue H, Martino S, Chopin N. Approximate Bayesian inference for latent Gaussian models using integrated nested Laplace approximations (with discussion).[J]. Journal of the Royal Statistical Society, Series B, 2009, 71: 319–392.
- [14] Geyer C J. On the convergence of monte carlo maximum likelihood calculations[J]. Journal of the Royal Statistical Society, Series B, 1994, 56(1): 261–274.
- [15] Zhang H. On estimation and prediction for spatial generalized linear mixed models[J]. Biometrics, 2002, 58(1): 129–36.
- [16] Diggle P J, Giorgi E. Model-based geostatistics for prevalence mapping in low-resource settings[J]. Journal of the American Statistical Association, 2016, 111(515): 1096–1120.
- [17] McCullagh P, Nelder J. Generalized linear models[M]. Second ed. London: Chapman and Hall/CRC, 1989: 28–32
- [18] 王松桂, 史建红, 尹素菊, 等. 线性模型引论[M]. 北京: 科学出版社, 2004: 78–90
- [19] Bartlett M S. An introduction to stochastic process with special reference to methods and applications[M]. First ed. Cambridge: Cambridge University Press, 1955: 215–221
- [20] Abramowitz M, Stegun I A. Handbook of mathematical functions[M]. Tenth ed. New York: National Bureau of Standards, 1972: 374–375
- [21] Campbell J B. On temme’s algorithm for the modified bessel function of the third kind [J]. ACM Transactions on Mathematical Software, 1980, 6(4): 581–586.
- [22] Tierney L, Kadane J B. Accurate approximations for posterior moments and marginal densities[J]. Journal of the American Statistical Association, 1986, 81(393): 82–86.
- [23] Rasmussen C E, Williams C K I. Gaussian processes for machine learning[M]. Cambridge, Massachusetts: MIT Press, 2006: 8–11
- [24] 黄湘云. 随机数生成及其在统计模拟中的应用[EB/OL]. (2017-05-26)[2018-08-12]. <https://cosx.org/2017/05/random-number-generation>.
- [25] Hoffman M D, Gelman A. The No-U-Turn sampler: Adaptively setting path lengths in hamiltonian monte carlo[J]. Journal of Machine Learning Research, 2014, 15: 1593–1623.
- [26] Rubin D B. Estimation in parallel randomized experiments[J]. Journal of Educational Statistics, 1981, 6(4): 377–401.
- [27] Alderman D L, Powers D E. The effects of special preparation on sat-verbal scores[J]. American Educational Research Journal, 1980, 17(2): 239–251.
- [28] Gelman A, Carlin J B, Stern H S, et al. Bayesian data analysis[M]. Second ed. London: Chapman and Hall/CRC, 2003: 138–144
- [29] Carpenter B, Gelman A, Hoffman M, et al. Stan: A probabilistic programming language [J]. Journal of Statistical Software, 2017, 76(1): 1–32.

-
- [30] Gelman A, Carlin J B, Stern H S, et al. Bayesian data analysis[M]. Third ed. Boca Raton, Florida: Chapman and Hall/CRC, 2013: 284–285
- [31] Simon N, Friedman J, Hastie T, et al. Regularization paths for cox’s proportional hazards model via coordinate descent[J]. Journal of Statistical Software, 2011, 39(5): 1–13.
- [32] Saldana D F, Feng Y. SIS: An R package for sure independence screening in ultrahigh dimensional statistical models[J]. Journal of Statistical Software, 2016, 83(2): 1–25.
- [33] Nelder J A, Wedderburn R W M. Generalized linear models[J]. Journal of the Royal Statistical Society, Series A, 1972, 135(3): 370–384.
- [34] 陈希孺. 广义线性模型的拟似然法[M]. 合肥: 中国科学技术大学出版社, 2011: 002–004
- [35] Bates D, Mächler M, Bolker B, et al. Fitting linear mixed-effects models using lme4[J]. Journal of Statistical Software, 2015, 67(1): 1–48.
- [36] Yang J, Benyamin B, Mcevoy B P, et al. Common snps explain a large proportion of heritability for human height[J]. Nature Genetics, 2010, 42(7): 565–569.
- [37] Hadfield J D. MCMC methods for multi-response generalized linear mixed models: The MCMCglmm R package[J]. Journal of Statistical Software, 2010, 33(2): 1–22.
- [38] Pebesma E J. Multivariable geostatistics in S: the gstat package[J]. Computers and Geosciences, 2004, 30(7): 683–691.
- [39] Gräler B, Pebesma E, Heuvelink G. Spatio-temporal interpolation using gstat[J]. The R Journal, 2016, 8(1): 204–218.
- [40] Diggle P J, Ribeiro Jr. P J. Model-based geostatistics[M]. New York, NY: Springer-Verlag, 2007: 112–114
- [41] Natarajan R, McCulloch C E. A note on the existence of the posterior distribution for a class of mixed models for binomial responses[J]. Biometrika, 1995, 82(3): 639–643.
- [42] Kass R E, Wasserman L. The selection of prior distributions by formal rules[J]. Journal of the American Statistical Association, 1996, 91(435): 1343–1370.
- [43] Warnes J J, Ripley B D. Problems with likelihood estimation of covariance functions of spatial gaussian processes[J]. Biometrika, 1987, 74(3): 640–642.
- [44] Hosseini F. A new algorithm for estimating the parameters of the spatial generalized linear mixed models[J]. Environmental and Ecological Statistics, 2016, 23(2): 205–217.
- [45] Bonat W H, Ribeiro Jr. P J. Practical likelihood analysis for spatial generalized linear mixed models[J]. Environmetrics, 2016, 27(2): 83–89.
- [46] Diggle P J, Heagerty P, Liang K Y, et al. Analysis of longitudinal data[M]. Second ed. New York: Oxford University Press, 2002
- [47] Bolker B, R Development Core Team. bbmle: Tools for general maximum likelihood estimation[EB/OL]. 2017. <https://CRAN.R-project.org/package=bbmle>.

- [48] 黄湘云. R 语言做符号计算[EB/OL]. (2016-07-08)[2018-08-12]. <https://cosx.org/2016/07/r-symbol-calculate>.
- [49] Christensen O F, Roberts G O, Sköld M. Robust markov chain monte carlo methods for spatial generalized linear mixed models[J]. *Journal of Computational and Graphical Statistics*, 2006, 15(1): 1–17.
- [50] Brooks S, Gelman A, Jones G, et al. Handbook of markov chain monte carlo[M]. Boca Raton, Florida: Chapman and Hall/CRC, 2011: 113–162
- [51] Giorgi E, Diggle P J. PrevMap: An R package for prevalence mapping[J]. *Journal of Statistical Software*, 2017, 78(8): 1–29.
- [52] SAS Institute Inc. SAS/STAT® 14.3 user’s guide[M]. Cary, NC: SAS Institute Inc., 2017: 127–159
- [53] Lunn D, Spiegelhalter D, Thomas A, et al. The BUGS project: Evolution, critique and future directions[J]. *Statistics in Medicine*, 2009, 28(25): 3049–3067.
- [54] Bürkner P. brms: An R package for bayesian multilevel models using Stan[J]. *Journal of Statistical Software*, 2017, 80(1): 1–28.
- [55] Finley A O, Banerjee S, E.Gelfand A. spBayes for large univariate and multivariate point-referenced spatio-temporal data models[J]. *Journal of Statistical Software*, 2015, 63(13): 1–28.
- [56] Plummer M, Best N, Cowles K, et al. CODA: Convergence diagnosis and output analysis for MCMC[J]. *R News*, 2006, 6(1): 7–11.
- [57] Christensen O, Ribeiro Jr. P. geoRglm: A package for generalised linear spatial models [J]. *R News*, 2002, 2(2): 26–28.
- [58] Brown P E, Zhou L. MCMC for generalized linear mixed models with glmmBUGS[J]. *R Journal*, 2010, 2(1): 13–17.
- [59] Schlather M, Malinowski A, Menck P J, et al. Analysis, simulation and prediction of multivariate random fields with package RandomFields[J]. *Journal of Statistical Software*, 2015, 63(8): 1–25.
- [60] Lindgren F, Rue H. Bayesian spatial modelling with R-INLA[J]. *Journal of Statistical Software*, 2015, 63(19): 1–25.
- [61] Gómez-Rubio V, Rue H. Markov chain monte carlo with the integrated nested laplace approximation[J]. *Statistics and Computing*, 2018, 28(5): 1033–1051.
- [62] Marta Blangiardo M C. Spatial and spatio-temporal bayesian models with R-INLA[M]. Chichester, UK: John Wiley and Sons Inc., 2015
- [63] Xiaofeng Wang R Y, Faraway J. Bayesian regression with INLA[M]. Boca Raton, Florida: Chapman and Hall/CRC, 2018
- [64] Stroup W W, Baenziger P S. Removing spatial variation from yield trials: a comparison

- of methods[J]. Crop Science, 1994, 34(1): 63–66.
- [65] Pinheiro J C, Bates D M. Mixed-effects models in S and S-PLUS[M]. New York, NY: Springer-Verlag, 2000: 260–266
- [66] 谢益辉. 用局部加权回归散点平滑法观察二维变量之间的关系[EB/OL]. (2008-11-26)[2018-08-12]. <https://cosx.org/2008/11/lowess-to-explore-bivariate-correlation-by-yihui>.
- [67] Bordner A S, Crosswell D A, Katz A O, et al. Measurement of background gamma radiation in the northern Marshall islands[J]. Proceedings of the National Academy of Sciences, 2016, 113(25): 6833–6838.
- [68] Papaspiliopoulos O. Non-centered parameterisations for data augmentation and hierarchical models with applications to inference for lèvy-based stochastic volatility models [D]. Lancaster: Lancaster University, 2003.

致 谢

三年时间说短不短，说长不长，但是对我却是意义重大的三年，无论是学习还是生活，学校对我的影响都是终生难忘的。首先，我要感谢父母一如既往的默默支持，没有他们就没有我的今天，虽然远隔千山万里，也照顾不到我的学习和生活，但只要想到不管我做怎样的决定，他们都会全力支持，我很感动。

最后我要感谢统计之都，特别是创始人谢益辉，除了使用他开发的工具打造毕业论文模板，使得论文排版工作量直接降低了几个量级，一年多以来，还一直对我的问题有问必答。三年来，帮助过我的老师，同学，同事，朋友太多，他们当中很多都直接或间接地帮助了我的毕业论文，人生最大的幸运莫过于结识你们。

作者简介

黄湘云，男（1992-），2015年毕业于中国矿业大学（北京），获理学学位；2018年毕业于中国矿业大学（北京），攻读硕士学位，专业为统计学，研究方向为数据分析与统计计算。

主要获奖

1. 2015-2016 年度获研究生优秀学生一等奖学金
2. 2016-2017 年度获研究生优秀学生奖学金

附录

表格

表 7.1: Langevin-Hastings 算法: 模型 (5.5) 中 64 个采样点处概率 $p(x_i) = \exp[S(x_i)] / \{1 + \exp[S(x_i)]\}$ 的后验分布的均值 (mean), 方差 (variance), 标准差 (standard deviation) 和 5 个分位点, 样本量为 1000

	mean	var	sd	2.5%	25%	50%	75%	97.5%
$p(x_1)$	0.476	0.019	0.138	0.231	0.368	0.467	0.573	0.759
$p(x_2)$	0.423	0.017	0.129	0.190	0.331	0.417	0.510	0.695
$p(x_3)$	0.313	0.015	0.122	0.106	0.221	0.309	0.389	0.582
$p(x_4)$	0.470	0.020	0.141	0.204	0.372	0.466	0.564	0.755
$p(x_5)$	0.431	0.018	0.133	0.181	0.338	0.425	0.527	0.686
$p(x_6)$	0.516	0.017	0.131	0.256	0.429	0.517	0.612	0.766
$p(x_7)$	0.580	0.017	0.132	0.326	0.485	0.583	0.669	0.831
$p(x_8)$	0.483	0.019	0.138	0.220	0.386	0.484	0.578	0.736
$p(x_9)$	0.487	0.020	0.141	0.235	0.383	0.482	0.584	0.772
$p(x_{10})$	0.333	0.014	0.117	0.121	0.251	0.326	0.411	0.583
$p(x_{11})$	0.262	0.013	0.112	0.083	0.175	0.248	0.334	0.499
$p(x_{12})$	0.367	0.016	0.126	0.150	0.279	0.358	0.446	0.627
$p(x_{13})$	0.491	0.017	0.129	0.248	0.403	0.487	0.579	0.742
$p(x_{14})$	0.585	0.016	0.127	0.343	0.493	0.589	0.673	0.826
$p(x_{15})$	0.573	0.016	0.125	0.320	0.491	0.576	0.660	0.811
$p(x_{16})$	0.610	0.016	0.127	0.347	0.526	0.612	0.701	0.843
$p(x_{17})$	0.336	0.016	0.127	0.130	0.241	0.323	0.415	0.605
$p(x_{18})$	0.299	0.013	0.114	0.108	0.217	0.292	0.368	0.566
$p(x_{19})$	0.269	0.012	0.109	0.088	0.190	0.258	0.337	0.502
$p(x_{20})$	0.429	0.016	0.128	0.192	0.336	0.428	0.520	0.687
$p(x_{21})$	0.504	0.015	0.124	0.270	0.417	0.499	0.583	0.761
$p(x_{22})$	0.550	0.015	0.121	0.308	0.469	0.556	0.633	0.785
$p(x_{23})$	0.617	0.015	0.123	0.360	0.538	0.622	0.705	0.842
$p(x_{24})$	0.646	0.015	0.124	0.380	0.563	0.660	0.732	0.868
$p(x_{25})$	0.246	0.012	0.111	0.066	0.166	0.237	0.312	0.483
$p(x_{26})$	0.287	0.013	0.113	0.097	0.206	0.275	0.356	0.541
$p(x_{27})$	0.341	0.014	0.118	0.121	0.261	0.332	0.416	0.586

	mean	var	sd	2.5%	25%	50%	75%	97.5%
$p(x_{28})$	0.525	0.016	0.128	0.298	0.427	0.524	0.609	0.772
$p(x_{29})$	0.540	0.016	0.128	0.295	0.446	0.542	0.631	0.783
$p(x_{30})$	0.583	0.015	0.123	0.348	0.496	0.583	0.671	0.813
$p(x_{31})$	0.517	0.017	0.130	0.251	0.432	0.525	0.606	0.756
$p(x_{32})$	0.689	0.014	0.117	0.437	0.612	0.693	0.775	0.898
$p(x_{33})$	0.260	0.012	0.111	0.075	0.178	0.251	0.328	0.494
$p(x_{34})$	0.304	0.014	0.119	0.101	0.218	0.292	0.371	0.577
$p(x_{35})$	0.394	0.016	0.125	0.171	0.308	0.389	0.472	0.669
$p(x_{36})$	0.497	0.017	0.130	0.249	0.412	0.495	0.587	0.746
$p(x_{37})$	0.604	0.017	0.131	0.346	0.518	0.606	0.700	0.844
$p(x_{38})$	0.546	0.016	0.126	0.298	0.459	0.548	0.636	0.774
$p(x_{39})$	0.494	0.017	0.129	0.242	0.404	0.498	0.582	0.735
$p(x_{40})$	0.639	0.015	0.123	0.394	0.559	0.647	0.724	0.864
$p(x_{41})$	0.380	0.017	0.132	0.154	0.284	0.369	0.466	0.669
$p(x_{42})$	0.339	0.015	0.122	0.128	0.257	0.331	0.416	0.595
$p(x_{43})$	0.318	0.014	0.118	0.111	0.234	0.311	0.398	0.552
$p(x_{44})$	0.479	0.016	0.127	0.247	0.387	0.473	0.566	0.745
$p(x_{45})$	0.655	0.015	0.123	0.415	0.568	0.659	0.746	0.880
$p(x_{46})$	0.601	0.015	0.123	0.354	0.519	0.607	0.689	0.839
$p(x_{47})$	0.524	0.017	0.129	0.275	0.437	0.525	0.612	0.768
$p(x_{48})$	0.696	0.014	0.118	0.440	0.620	0.704	0.783	0.901
$p(x_{49})$	0.353	0.015	0.124	0.122	0.266	0.348	0.433	0.615
$p(x_{50})$	0.493	0.017	0.132	0.254	0.402	0.492	0.584	0.760
$p(x_{51})$	0.379	0.014	0.120	0.160	0.293	0.374	0.460	0.619
$p(x_{52})$	0.378	0.016	0.128	0.135	0.282	0.376	0.470	0.620
$p(x_{53})$	0.591	0.015	0.124	0.345	0.512	0.591	0.678	0.834
$p(x_{54})$	0.521	0.017	0.132	0.244	0.432	0.531	0.613	0.773
$p(x_{55})$	0.566	0.016	0.125	0.305	0.480	0.573	0.654	0.789
$p(x_{56})$	0.703	0.014	0.120	0.449	0.622	0.711	0.794	0.900
$p(x_{57})$	0.494	0.019	0.137	0.235	0.400	0.495	0.587	0.762
$p(x_{58})$	0.606	0.019	0.139	0.328	0.515	0.608	0.702	0.857
$p(x_{59})$	0.482	0.018	0.133	0.236	0.387	0.478	0.575	0.725
$p(x_{60})$	0.443	0.017	0.130	0.195	0.349	0.447	0.536	0.695
$p(x_{61})$	0.636	0.015	0.124	0.390	0.547	0.638	0.723	0.867

	mean	var	sd	2.5%	25%	50%	75%	97.5%
$p(x_{62})$	0.620	0.015	0.122	0.371	0.537	0.623	0.708	0.843
$p(x_{63})$	0.568	0.016	0.128	0.321	0.470	0.571	0.660	0.802
$p(x_{64})$	0.633	0.017	0.131	0.354	0.545	0.644	0.729	0.861

数据模拟

7.0.1 模拟平稳高斯过程

STAN 代码模拟高斯过程，自协方差函数见方程 ??

```
// Sample from a Gaussian process using exponentiated covariance function.
// Fixed kernel hyperparameters: phi=0.15, sigma=sqrt(1)
```

```
data {
  int<lower=1> N;
  real<lower=0> phi;
  real<lower=0> sigma;
}
transformed data {
  vector[N] zeros;
  zeros = rep_vector(0, N);
}
model {}
generated quantities {
  real x[N];
  vector[N] f;
  for (n in 1:N)
    x[n] = uniform_rng(-2,2);
  {
    matrix[N, N] cov;
    matrix[N, N] L_cov;

    // cov = cov_exp_quad(x, sigma, phi);

    for (i in 1:(N - 1)) {
      cov[i, i] = square(sigma);
```

```

    for (j in (i + 1):N) {
        cov[i, j] = square(sigma) * exp(- fabs(x[i] - x[j]) * inv(phi));
        cov[j, i] = cov[i, j];
    }
}
cov[N, N] = square(sigma);

for (n in 1:N)
    cov[n, n] = cov[n, n] + 1e-12;

L_cov = cholesky_decompose(cov);
f = multi_normal_cholesky_rng(zeros, L_cov);
}
}

```

7.0.2 模拟空间广义线性混合效应模型

模拟空间广义线性混合效应模型的函数：`generate_sim_data` 函数可生成响应变量服从泊松分布或二项分布，平稳高斯过程的自相关函数为二次幂指数型或梅隆型的模型

```

generate_sim_data <- function(N = 49, intercept = -1.0,
                              slope1 = 1.0, slope2 = 0.5,
                              lscale = 1, sdgp = 1,
                              cov.model = "exp_quad", type = "binomal") {

  # set.seed(2018)
  ## 单位区域上采样
  d <- expand.grid(
    d1 = seq(0, 1, l = sqrt(N)),
    d2 = seq(0, 1, l = sqrt(N))
  )
  D <- as.matrix(dist(d)) # 计算采样点之间的欧氏距离
  switch (cov.model,
    matern = {
      phi = lscale
      corr_m = geoR::matern(D, phi = phi, kappa = 2) # 固定的 kappa = 2
      m = sdgp^2 * corr_m
    },

```

```

    exp_quad = {
      phi <- 2 * lscale^2
      m <- sdgp^2 * exp(-D^2 / phi) # 多元高斯分布的协方差矩阵
    }
  )
  # powered.exponential (or stable)
  # rho(h) = exp[-(h/phi)^kappa] if 0 < kappa <= 2 此处 kappa 固定为 2
  S <- MASS::mvrnorm(1, rep(0, N), m) # 产生服从多元高斯分布的随机数
  # 模拟两个固定效应
  x1 <- rnorm(N, 0, 1)
  x2 <- rnorm(N, 0, 4)
  switch(type,
    binomal = {
      units.m <- rep(100, N) # N 个 100
      pred <- intercept + slope1 * x1 + slope2 * x2 + S
      mu <- exp(pred) / (1 + exp(pred))
      y <- rbinom(N, size = 100, prob = mu) # 每个采样点抽取100个样本
      data.frame(d, y, units.m, x1, x2)
    },
    poisson = {
      pred <- intercept + slope1 * x1 + slope2 * x2 + S
      y <- rpois(100, lambda = exp(pred)) # lambda 是泊松分布的期望
      # Y ~ Possion(lambda) g(u) = ln(u) u = lambda = exp(g(u))
      data.frame(d, y, x1, x2)
    }
  )
}

```

HMC 算法

```

# 加载程序包
library(rstan)
library(brms)
# 以并行方式运行STAN-MCMC算法，指定CPU的核心数
options(mc.cores = parallel::detectCores())
# 将编译后的模型写入磁盘，可防止重新编译
rstan_options(auto_write = TRUE)

```

```

theme_set(theme_default())
prior <- c(
  set_prior("normal(0,10)", class = "b"), # 均值0 标准差 10 的先验
  set_prior("lognormal(0,1)", class = "lscale"),
  set_prior("lognormal(0,1)", class = "sdgp")
)
sim_binom_data <- generate_sim_data(type = "binomal")
benchmark.binomal <- microbenchmark::microbenchmark({
  fit.binomal <- brm(y | trials(units.m) ~ 0 + intercept + x1 + x2 + gp(d1, d2),
    sim_binom_data,
    prior = prior,
    chains = 4, thin = 5, iter = 15000, warmup = 5000,
    algorithm = "sampling", family = binomial()
  )
}, times = 10L)
summary(fit.binomal)

sim_poisson_data <- generate_sim_data(type = "poisson")
benchmark.poisson <- microbenchmark::microbenchmark({
  fit.poisson <- brm(y ~ 0 + intercept + x1 + x2 + gp(d1, d2),
    sim_poisson_data,
    prior = prior,
    chains = 4, thin = 5, iter = 15000, warmup = 5000,
    algorithm = "sampling", family = poisson()
  )
}, times = 10L)
summary(fit.poisson)
plot(fit.poisson)

knitr::include_graphics(path = c(
  "figures/compiling.png", "figures/sampling.png"
))

## 32 Intel(R) Xeon(R) CPU E5-2680 v2 @ 2.80GHz
cat(system("cat /proc/cpuinfo | grep name | cut -f2 -d: | uniq -c", intern = TRUE), sep = '\n')
#>      32 Intel(R) Xeon(R) CPU E5-2680 v2 @ 2.80GHz

```


PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
2001	xiangyun	20	0	1535992	1.3g	6000	R	99.7	27.9	0:14.31	cc1plus
9	root	20	0	0	0	0	S	0.3	0.0	0:00.71	rcu_sched
1306	xiangyun	20	0	683880	78672	5844	S	0.3	1.6	0:15.82	rsession
1	root	20	0	128032	2604	1192	S	0.0	0.1	0:01.51	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.01	kthreadd
3	root	20	0	0	0	0	S	0.0	0.0	0:00.21	ksoftirq+
5	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kworker/+
7	root	rt	0	0	0	0	S	0.0	0.0	0:00.00	migratio+

(a) 编译

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
2053	xiangyun	20	0	859904	423088	3520	R	81.1	8.6	0:14.31	cc1plus
2056	xiangyun	20	0	893900	456396	2936	R	77.1	9.3	0:15.82	rsession
2055	xiangyun	20	0	893900	456396	2936	R	72.8	9.3	0:15.82	rsession
2054	xiangyun	20	0	859904	413896	3516	R	68.4	8.4	0:14.31	cc1plus
1306	xiangyun	20	0	683880	78676	5844	S	0.7	1.6	0:15.82	rsession
1940	xiangyun	20	0	161840	2188	1552	R	0.3	0.0	0:00.21	ksoftirq+
1	root	20	0	128032	2604	1192	S	0.0	0.1	0:01.51	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.01	kthreadd

(b) 采样

图 7.1: 基于 Stan 编程的模型编译和采样过程

```
parallel::detectCores()
```

```
#> [1] 32
```

setting	value
version	R version 3.5.0 (2017-01-27)
os	Ubuntu 14.04.5 LTS
system	x86_64, linux-gnu
ui	X11
language	(EN)
collate	en_US.UTF-8
ctype	en_US.UTF-8
tz	UTC
date	2018-10-13

package	version	source
abind	1.4-7	R-Forge (R 3.5.0)
assertthat	0.2.0	CRAN (R 3.5.0)
backports	1.1.2	CRAN (R 3.5.0)
base64enc	0.1-3	CRAN (R 3.5.0)
bayesplot	1.6.0	CRAN (R 3.5.0)
BH	1.66.0-1	CRAN (R 3.5.0)
bibtex	0.4.2	CRAN (R 3.5.0)
bindr	0.1.1	CRAN (R 3.5.0)
bindrepp	0.2.2	CRAN (R 3.5.0)
bitops	1.0-6	CRAN (R 3.5.0)
bookdown	0.7.21	Github (rstudio/bookdown@d1f44e5)
bridgesampling	0.5-2	CRAN (R 3.5.0)
brms	2.5.0	CRAN (R 3.5.0)

package	version	source
codetools	0.2-15	CRAN (R 3.5.0)
colorspace	1.4-0	R-Forge (R 3.5.0)
colourpicker	1.0	CRAN (R 3.5.0)
crayon	1.3.4	CRAN (R 3.5.0)
crosstalk	1.0.0	CRAN (R 3.5.0)
curl	3.2	CRAN (R 3.5.0)
DBI	1.0.0	CRAN (R 3.5.0)
desc	1.2.0	CRAN (R 3.5.0)
digest	0.6.18	CRAN (R 3.5.0)
doRNG	1.7.1	CRAN (R 3.5.0)
doSNOW	1.0.17	R-Forge (R 3.5.0)
dplyr	0.7.6	CRAN (R 3.5.0)
DT	0.4	CRAN (R 3.5.0)
dygraphs	1.1.1.6	CRAN (R 3.5.0)
e1071	1.7-0	CRAN (R 3.5.0)
evaluate	0.12	CRAN (R 3.5.0)
fansi	0.4.0	CRAN (R 3.5.0)
foreach	1.5.1	R-Forge (R 3.5.0)
formatR	1.5	CRAN (R 3.5.0)
gdtools	0.1.7	CRAN (R 3.5.0)
ggplot2	3.0.0.9000	Github (tidyverse/ggplot2@3e1e6e4)
ggribes	0.5.1	CRAN (R 3.5.0)
glue	1.3.0	CRAN (R 3.5.0)
gridExtra	2.3	CRAN (R 3.5.0)
gtable	0.2.0	CRAN (R 3.5.0)
gtools	3.8.1	CRAN (R 3.5.0)
highr	0.7	CRAN (R 3.5.0)
hms	0.4.2.9001	Github (tidyverse/hms@979286f)
htmldeps	0.1.1	Github (rstudio/htmldeps@c1023e0)
htmltools	0.3.6	CRAN (R 3.5.0)
htmlwidgets	1.3	CRAN (R 3.5.0)
httpuv	1.4.5	CRAN (R 3.5.0)
httr	1.3.1	CRAN (R 3.5.0)
igraph	1.2.2	CRAN (R 3.5.0)

package	version	source
inline	0.3.15	CRAN (R 3.5.0)
iterators	1.0.11	R-Forge (R 3.5.0)
jsonlite	1.5	CRAN (R 3.5.0)
kableExtra	0.9.0	CRAN (R 3.5.0)
knitr	1.20	CRAN (R 3.5.0)
labeling	0.3	CRAN (R 3.5.0)
later	0.7.5	CRAN (R 3.5.0)
lattice	0.20-35	CRAN (R 3.5.0)
lazyeval	0.2.1	CRAN (R 3.5.0)
lme4	1.1-18-1	CRAN (R 3.5.0)
loo	2.0.0	CRAN (R 3.5.0)
magrittr	1.5	CRAN (R 3.5.0)
markdown	0.8	CRAN (R 3.5.0)
MASS	7.3-49	CRAN (R 3.5.0)
Matrix	1.2-14	CRAN (R 3.5.0)
matrixStats	0.54.0	CRAN (R 3.5.0)
mcmc	0.9-5	CRAN (R 3.5.0)
mgcv	1.8-23	CRAN (R 3.5.0)
mime	0.6	CRAN (R 3.5.0)
miniUI	0.1.1.1	CRAN (R 3.5.0)
minqa	1.2.4	CRAN (R 3.5.0)
munsell	0.5.0	CRAN (R 3.5.0)
mvtnorm	1.0-8	CRAN (R 3.5.0)
nleqslv	3.3.2	CRAN (R 3.5.0)
nlme	3.1-137	CRAN (R 3.5.0)
nloptr	1.2.1	CRAN (R 3.5.0)
openssl	1.0.2	CRAN (R 3.5.0)
packrat	0.4.9-3	CRAN (R 3.5.0)
pillar	1.3.0	CRAN (R 3.5.0)
pkgconfig	2.0.2	CRAN (R 3.5.0)
pkgmaker	0.27	CRAN (R 3.5.0)
PKI	0.1-5.1	CRAN (R 3.5.0)
plogr	0.2.0	CRAN (R 3.5.0)
plyr	1.8.4	CRAN (R 3.5.0)

package	version	source
promises	1.0.1	CRAN (R 3.5.0)
purrr	0.2.5	CRAN (R 3.5.0)
R6	2.3.0	CRAN (R 3.5.0)
RColorBrewer	1.1-2	CRAN (R 3.5.0)
Rcpp	0.12.19	CRAN (R 3.5.0)
RcppEigen	0.3.3.4.0	CRAN (R 3.5.0)
RCurl	1.95-4.11	CRAN (R 3.5.0)
readr	1.2.0	Github (tidyverse/readr@156dc31)
registry	0.5	CRAN (R 3.5.0)
reshape2	1.4.3	CRAN (R 3.5.0)
RJSONIO	1.3-0	CRAN (R 3.5.0)
rlang	0.2.2	CRAN (R 3.5.0)
rmarkdown	1.10.14	Github (rstudio/rmarkdown@dcc7d37)
rngtools	1.3.1	CRAN (R 3.5.0)
rprojroot	1.3-2	CRAN (R 3.5.0)
rsconnect	0.8.8	CRAN (R 3.5.0)
rstan	2.17.4	CRAN (R 3.5.0)
rstanarm	2.17.4	CRAN (R 3.5.0)
rstansim	0.1.1	CRAN (R 3.5.0)
stantools	1.5.1	CRAN (R 3.5.0)
rstudioapi	0.8	CRAN (R 3.5.0)
rvest	0.3.2	CRAN (R 3.5.0)
scales	1.0.0	CRAN (R 3.5.0)
selectr	0.4-1	CRAN (R 3.5.0)
sessioninfo	1.1.0	CRAN (R 3.5.0)
sf	0.6-3	CRAN (R 3.5.0)
shiny	1.1.0	CRAN (R 3.5.0)
shinyjs	1.0	CRAN (R 3.5.0)
shinytan	2.5.0	CRAN (R 3.5.0)
shinythemes	1.1.1	CRAN (R 3.5.0)
snow	0.4-3	CRAN (R 3.5.0)
sourcetools	0.1.7	CRAN (R 3.5.0)
spData	0.2.9.4	CRAN (R 3.5.0)
StanHeaders	2.18.0	CRAN (R 3.5.0)

package	version	source
stringi	1.2.4	CRAN (R 3.5.0)
stringr	1.3.1	CRAN (R 3.5.0)
survival	2.41-3	CRAN (R 3.5.0)
svglite	1.2.1	CRAN (R 3.5.0)
threejs	0.3.1	CRAN (R 3.5.0)
tibble	1.4.2	CRAN (R 3.5.0)
tidyselect	0.2.5	CRAN (R 3.5.0)
tinytex	0.8	CRAN (R 3.5.0)
units	0.6-1	CRAN (R 3.5.0)
utf8	1.1.4	CRAN (R 3.5.0)
viridisLite	0.3.0	CRAN (R 3.5.0)
withr	2.1.2	CRAN (R 3.5.0)
xfun	0.3	CRAN (R 3.5.0)
xml2	1.2.0	CRAN (R 3.5.0)
xtable	1.8-3	CRAN (R 3.5.0)
xts	0.11-1	CRAN (R 3.5.0)
yaml	2.2.0	CRAN (R 3.5.0)
zoo	1.8-4	R-Forge (R 3.5.0)
