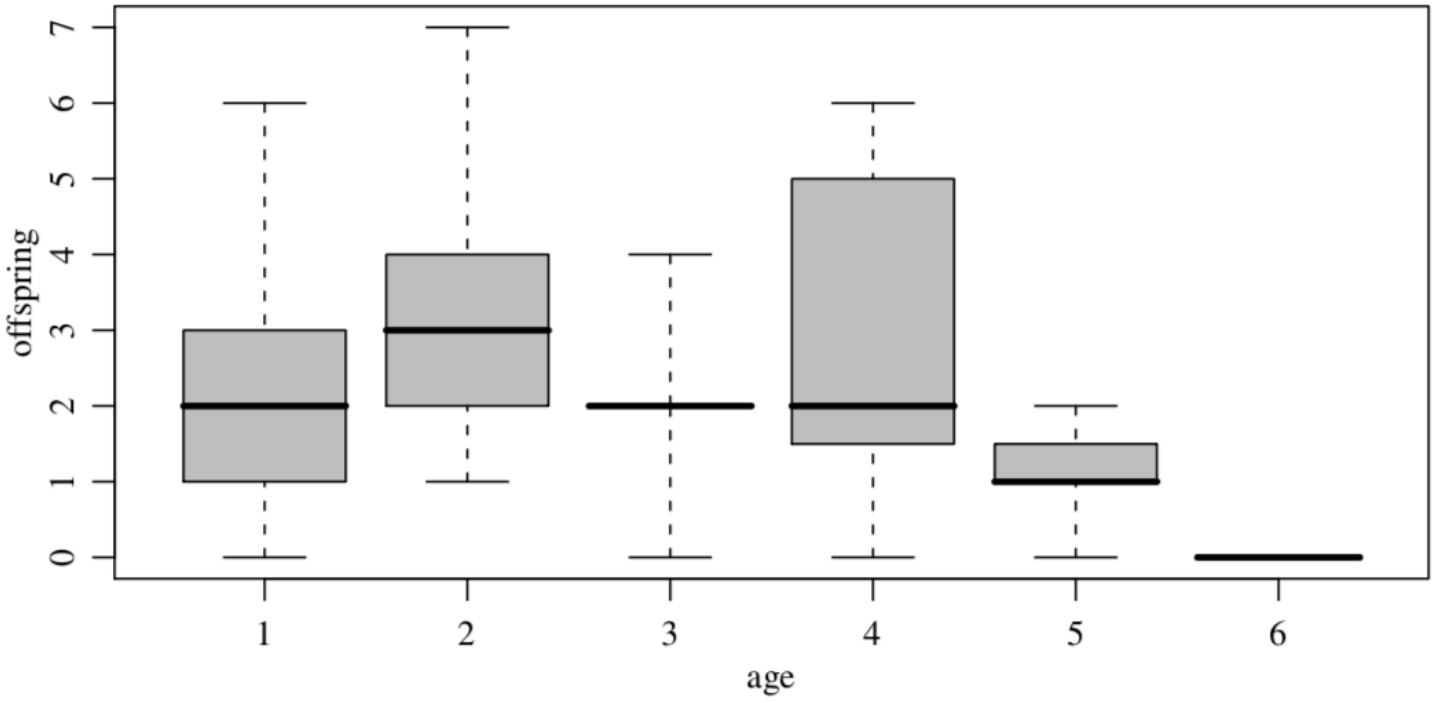
第五章 Metropolis-Hastings 算法, HMC 算法与 SMC 算法

引言

- 有些 Bayesian 模型不存在共轭的先验分布，参数的完全条件分布也很难进行抽样 (无法使用 Gibbs 抽样)
- 更通用的 MCMC 方法 — Metropolis-Hastings 算法，几乎适用估计任何先验分布下的 Bayesian 模型
- 52 只雌雀在一个夏天的繁殖数据



如何用一个概率模型拟合该数据以预测雌雀各年龄繁殖后代数的期望？

A Bayesian Poisson Regression Model

- Notation.
  - 响应变量 (response) y_i: 雌雀 i 繁殖的后代数，$y_i \in \{0, 1, 2, \dots\}$
  - 解释变量 x_i: 雌雀 i 的年龄

- 考虑如下的泊松模型:

$$y_i \mid x_i \sim Po(\theta(x_i)), \quad i = 1, \dots, n. \tag{1}$$

  - 根据 boxplot，如果假设 $\theta(x_i) = \beta_1 + \beta_2 x_i + \beta_3 x_i^2$，那么估计的系数 $\boldsymbol{\beta} = (\beta_1, \beta_2, \beta_3)$ 可能使 $\theta(x_i) < 0$
  - 一种解决方法是假设

$$\log \theta(x_i) = \beta_1 + \beta_2 x_i + \beta_3 x_i^2 \tag{2}$$

A Bayesian Poisson Regression Model

- Bayesian 分析的优点：在小样本下可以更准确地量化参数的不确定性 (估计方差、置信区间等)

- 为 Bayesian 泊松回归模型(1)-(2)的参数 β 设定如下的先验分布:

$$\beta \sim N_3(\mathbf{0}, 100 I_3) \tag{3}$$

- 此时 β 的后验分布不是多元正态分布，各分量的完全条件分布也不是常见的分布 (无法使用 Gibbs 抽样), 考虑使用 Metropolis 方法构建马尔可夫链获得 β 后验分布的样本

Metropolis 算法

- Metropolis 算法通过持续地在参数空间随机“游走”寻找后验分布 $p(\boldsymbol{\theta} \mid \boldsymbol{y})$ 概率密度较高的区域
  - 假设当前时刻马尔可夫链得到的样本为 $\boldsymbol{\theta}^{(t)}$，在 $\boldsymbol{\theta}^{(t)}$ 附近随机产生一点，如果该点对应的 $p(\boldsymbol{\theta} \mid \boldsymbol{y})$ 的值高于 $p(\boldsymbol{\theta}^{(t)} \mid \boldsymbol{y})$，就让马尔可夫链移动到该点；反之以一定概率决定是否沿概率密度较低的方向移动
- 使用 Metropolis 算法需要先选取一个对称的 proposal 分布 $g(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)})$，$g(\cdot \mid \cdot)$ 满足 $g(\boldsymbol{\theta}_a \mid \boldsymbol{\theta}_b) = g(\boldsymbol{\theta}_b \mid \boldsymbol{\theta}_a)$，常用的 proposal 分布有:
  - $g(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)}) \sim \boldsymbol{U}\left(\boldsymbol{\theta}^{(t)} - \boldsymbol{\delta}, \boldsymbol{\theta}^{(t)} + \boldsymbol{\delta}\right)$
  - $g(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)}) \sim N_p(\boldsymbol{\theta}^{(t)}, \mathrm{diag}(\boldsymbol{\delta}))$

Metropolis 算法

- 选定 proposal 分布 $g(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)})$ 后，Metropolis 算法如下产生样本 $\boldsymbol{\theta}^{(t+1)}$:

  1. 抽样 $\boldsymbol{\theta}^{\star} \sim g(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)})$
  2. 计算接受比率

$$r_t = \frac{p(\boldsymbol{\theta}^{\star} \mid \boldsymbol{y})}{p(\boldsymbol{\theta}^{(t)} \mid \boldsymbol{y})} = \frac{p(\boldsymbol{y} \mid \boldsymbol{\theta}^{\star}) p(\boldsymbol{\theta}^{\star})}{p(\boldsymbol{y} \mid \boldsymbol{\theta}^{(t)}) p(\boldsymbol{\theta}^{(t)})}$$

  3. 令

$$\boldsymbol{\theta}^{(t+1)} = \begin{cases} \boldsymbol{\theta}^{\star} & \text{概率 } \min(r_t, 1) \\ \boldsymbol{\theta}^{(t)} & \text{概率 } 1 - \min(r_t, 1) \end{cases}$$

Metropolis 算法的收敛性

- 一条遍历的马尔可夫链会收敛到唯一的平稳分布，如果分布 $\pi(\boldsymbol{x})$ 满足

$$\pi(\boldsymbol{x}) = \int p(\boldsymbol{x} \mid \boldsymbol{x}')\pi(\boldsymbol{x}')d\boldsymbol{x}', \quad \forall \boldsymbol{x}, \boldsymbol{x}' \tag{4}$$

  那么 $\pi(\boldsymbol{x})$ 是马尔可夫链的平稳分布

- (4)成立的一个充分条件是

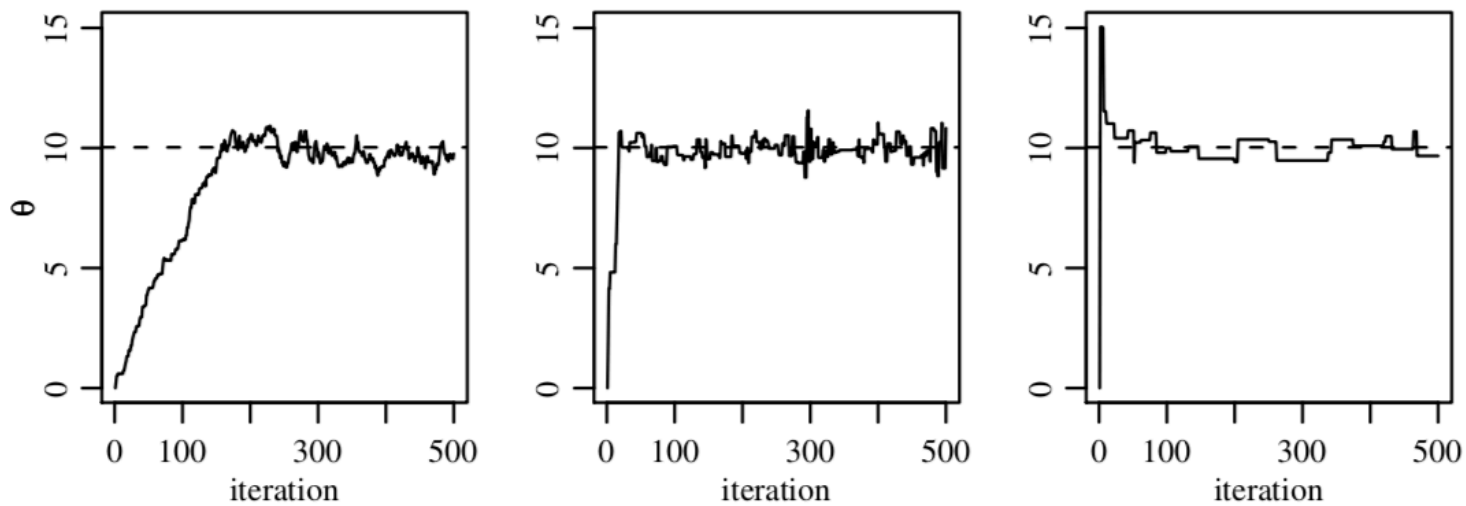
$$p(\boldsymbol{x}' \mid \boldsymbol{x})\pi(\boldsymbol{x}) = p(\boldsymbol{x} \mid \boldsymbol{x}')\pi(\boldsymbol{x}') \quad \forall \boldsymbol{x}, \boldsymbol{x}' \tag{5}$$

  称(5)为 detail balance 条件

- 证明 Metropolis 算法产生的马尔可夫链的平稳分布为参数的后验分布 $p(\boldsymbol{\theta} \mid \boldsymbol{y})$

Metropolis 算法的收敛性

- 如果参数 θ 是一个 p 维连续向量，将 proposal 分布 $g(\theta \mid \theta^{(t)})$ 选为 $N_p(\theta^{(t)}, \text{diag}(\delta))$ 可以得到遍历的马尔可夫链
- 在 Metropolis 算法中，接受比率并不是越高越好
  - 如果 proposal 分布 $g(\theta \mid \theta^{(t)})$ 中选取的 $\|\delta\|$ 很小，候选样本很容易被接受。但会导致马尔可夫链移动非常缓慢，样本之间的相关性较高，样本均值对后验期望的近似精度下降 (复习 ESS)
  - 如果 proposal 分布选取的 $\|\delta\|$ 很大，候选样本很容易被拒绝，导致马尔可夫链在一段时间被 “困在” 某点不动，样本间相关性较高
  - 实践中一般将马尔可夫链的接受比率控制在 20% 到 50% 之间 (Hoff, 2009)



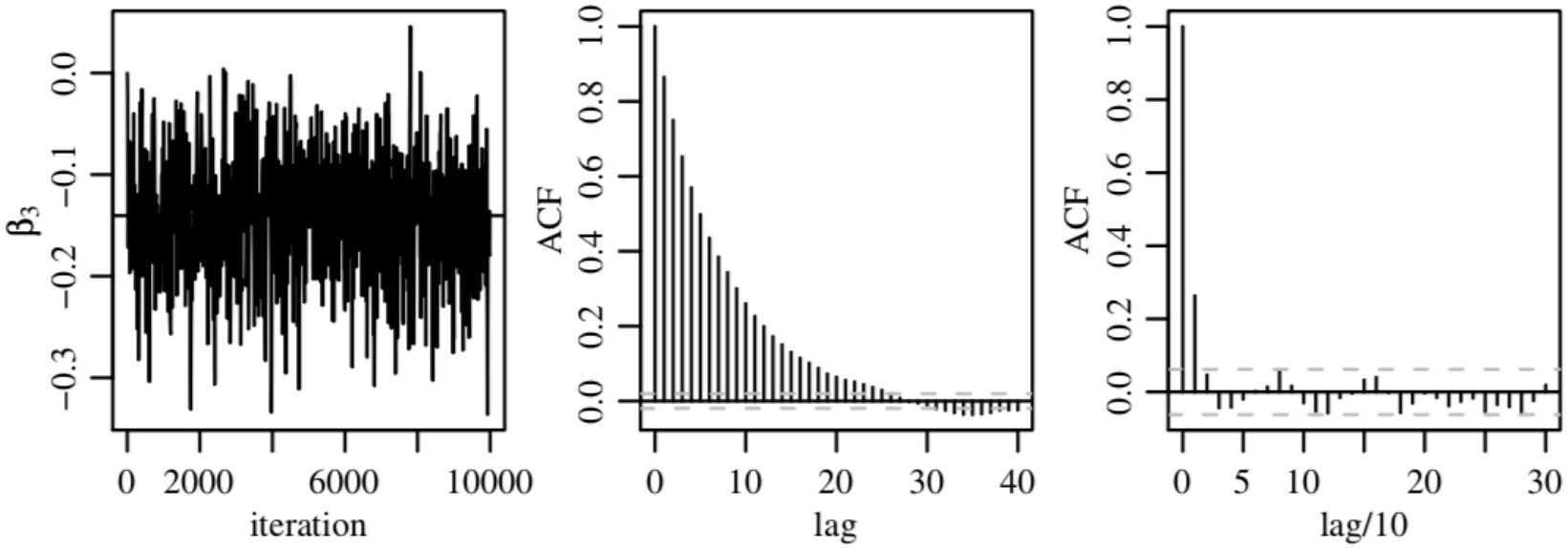
Bayesian 泊松回归模型的 Metropolis 算法

使用 Metropolis 算法估计 Bayesian 泊松回归模型(1)-(3)的后验分布 $p(\boldsymbol{\beta} \mid \{(x_i, y_i) : i = 1, \ldots, n\})$

- 令 $\boldsymbol{x}_i = (1, x_i, x_i^2)^\top$，矩阵 $X = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^\top$
- 将 proposal 分布 $g(\boldsymbol{\beta} \mid \boldsymbol{\beta}^{(t)})$ 取为 $N(\boldsymbol{\beta}^{(t)}, \hat{\sigma}^2 (X^\top X)^{-1})$
  - $\hat{\sigma}^2$ 是 $\{\log(y_1 + 1/2), \ldots, \log(y_n + 1/2)\}$ 的样本方差
- 从 proposal 分布产生一个候选样本 $\boldsymbol{\beta}^\star$，此时 Metropolis 算法的接受比率为：

$$
\begin{aligned}
r &= \frac{p(\boldsymbol{\beta}^\star \mid X, \boldsymbol{y})}{p(\boldsymbol{\beta}^{(t)} \mid X, \boldsymbol{y})} \\
&= \frac{N_3(\boldsymbol{\beta}^\star \mid \mathbf{0}, 100 I_3) \prod_{i=1}^n \mathrm{Po}\left(y_i \mid \exp(\boldsymbol{x}_i^\top \boldsymbol{\beta}^\star)\right)}{N_3(\boldsymbol{\beta}^{(t)} \mid \mathbf{0}, 100 I_3) \prod_{i=1}^n \mathrm{Po}\left(y_i \mid \exp(\boldsymbol{x}_i^\top \boldsymbol{\beta}^{(t)})\right)}
\end{aligned}
$$

Bayesian 泊松回归模型的 Metropolis 算法



- 上述 Metropolis 算法在初值 $\boldsymbol{\beta}^{(0)} = \mathbf{0}$ 下产生的马尔可夫链的接受比率为 43%
- 从该马尔可夫链上每 10 步取一个样本组成一条 thinned Markov chain，新的马尔可夫链上样本间的相关性很小，ESS=726

Metropolis-Hastings 算法

- Gibbs 抽样和 Metropolis 算法是更一般的 Metropolis-Hastings (M-H) 算法的两个特例
- M-H 算法与 Metropolis 算法很相似，但是 M-H 算法允许任何形式的 proposal 分布，不一定是对称的条件分布
- 目标分布是 $p(\boldsymbol{\theta} \mid \boldsymbol{y})$，选取 proposal 分布 $\tilde{g}(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)})$ 后，M-H 算法如下产生样本 $\boldsymbol{\theta}^{(t+1)}$：
  1. 抽样 $\boldsymbol{\theta}^{\star} \sim \tilde{g}(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)})$
  2. 计算接受比率

$$r_t = \frac{p(\boldsymbol{\theta}^{\star} \mid \boldsymbol{y})}{p(\boldsymbol{\theta}^{(t)} \mid \boldsymbol{y})} \cdot \frac{\tilde{g}(\boldsymbol{\theta}^{(t)} \mid \boldsymbol{\theta}^{\star})}{\tilde{g}(\boldsymbol{\theta}^{\star} \mid \boldsymbol{\theta}^{(t)})} = \frac{p(\boldsymbol{y} \mid \boldsymbol{\theta}^{\star}) p(\boldsymbol{\theta}^{\star}) \tilde{g}(\boldsymbol{\theta}^{(t)} \mid \boldsymbol{\theta}^{\star})}{p(\boldsymbol{y} \mid \boldsymbol{\theta}^{(t)}) p(\boldsymbol{\theta}^{(t)}) \tilde{g}(\boldsymbol{\theta}^{\star} \mid \boldsymbol{\theta}^{(t)})}.$$

  3. 令

$$\boldsymbol{\theta}^{(t+1)} = \begin{cases} \boldsymbol{\theta}^{\star} & \text{概率 } \min(r_t, 1) \\ \boldsymbol{\theta}^{(t)} & \text{概率 } 1 - \min(r_t, 1) \end{cases}$$

Metropolis-Hastings 算法

- M-H 算法的适用范围更广，因为对称的 proposal 分布有时并不合理
- 与 Metropolis 算法类似，可以用 detail balance 条件证明 M-H 算法产生的马尔可夫链收敛到 $p(\boldsymbol{\theta} \mid \boldsymbol{y})$
- 如果 M-H 算法的目标分布 $\pi(\boldsymbol{\theta})$ 可以分解为如下两部分：

$$\pi(\boldsymbol{\theta}) \propto \alpha(\boldsymbol{\theta}) g(\boldsymbol{\theta})$$

其中 $g(\boldsymbol{\theta})$ 占主导地位且对应一个容易抽样的分布，可将 proposal 分布选为 $g(\boldsymbol{\theta})$，此时 M-H 算法的接受比率简化为

$$r_t = \frac{\pi(\boldsymbol{\theta}^\star)}{\pi(\boldsymbol{\theta}^{(t)})} \cdot \frac{g(\boldsymbol{\theta}^{(t)})}{g(\boldsymbol{\theta}^\star)} = \frac{\alpha(\boldsymbol{\theta}^\star)}{\alpha(\boldsymbol{\theta}^{(t)})}$$

Gibbs 抽样是 M-H 算法的特例

以一个简单的二元分布 $\pi(u, v)$ 的抽样为例

- Gibbs 抽样:
  - 抽样 $u^{(t+1)} \sim \pi(u \mid v^{(t)})$
  - 抽样 $v^{(t+1)} \sim \pi(v \mid u^{(t+1)})$

- M-H 算法: 选取 proposal 分布 $g_u(u \mid u', v')$ 和 $g_v(v \mid u', v')$
  1. 更新 U:
     1. 抽样 $u^\star \sim g_u(u \mid u^{(t)}, v^{(t)})$
     2. 计算接受比率 $ru_t = \dfrac{\pi(u^\star, v^{(t)})}{\pi(u^{(t)}, v^{(t)})} \cdot \dfrac{g_u(u^{(t)} \mid u^\star, v^{(t)})}{g_u(u^\star \mid u^{(t)}, v^{(t)})}$
     3. 令 $u^{(t+1)} = \begin{cases} u^\star & \text{概率 } \min(ru_t, 1) \\ u^{(t)} & \text{概率 } 1 - \min(ru_t, 1) \end{cases}$
  2. 更新 V:

     ...

  - 如果 $g_u(u \mid u', v') = \pi(u \mid v')$, 接受比率 $ru_t \equiv 1$

Hamiltonian Monte Carlo 方法

考虑如下一个简单的 Bayesian 模型：

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i,\ \varepsilon_i \overset{iid}{\sim} N(0, \sigma^2),\ \ i = 1, \ldots, n.$$

参数的先验分布为

$$\begin{aligned}
\beta_0 &\sim N(0, 10^2) \\
\beta_1 &\sim N(0, 10^2) \\
\sigma &\sim p(\sigma) = \frac{2}{\pi(1+\sigma^2)} \cdot \mathbf{1}(\sigma > 0) \quad \text{(half-Cauchy)}
\end{aligned}$$

分别用 M-H 算法和 HMC 算法估计参数的后验分布，考察两种方法生成的马尔可夫链在 β_0 和 β_1 参数空间中的动态移动情况： ▸ M-H ▸ HMC

Hamiltonian Monte Carlo 方法

- M-H 算法通过随机游走的方式探索参数空间的效率较低，HMC 方法是一种基于梯度的抽样方法，可以使样本每步移动的幅度更大，且能有效减少样本间的相关性
- HMC 方法借鉴了物理中 Hamiltonian dynamics 的想法：假设小球在一个无摩擦的环境下运动
  - $\boldsymbol{\theta}_t \in \mathbb{R}^d$：小球在 t 时刻的位置
  - $U(\boldsymbol{\theta}_t)$：小球在 t 时刻的势能
  - $\boldsymbol{q}_t \in \mathbb{R}^d$：小球在 t 时刻的动量 $(\boldsymbol{q}_t = m \cdot \boldsymbol{v}_t)$
  - $K(\boldsymbol{q}_t)$：小球在 t 时刻的动能，$K(\boldsymbol{q}_t) = \boldsymbol{q}_t^\top \boldsymbol{q}_t/(2m)$

  定义函数 $H(\boldsymbol{\theta}, \boldsymbol{q}) = U(\boldsymbol{\theta}) + K(\boldsymbol{q})$，小球的运动由 Hamilton 方程组描述：

$$\begin{aligned} \frac{d\theta_j}{dt} &= \frac{\partial H}{\partial q_j} \\ \frac{dq_j}{dt} &= -\frac{\partial H}{\partial \theta_j} \end{aligned} \tag{6}$$

  - 由(6)可得 $\frac{dH}{dt} = 0$，即 $H(\boldsymbol{\theta}_t, \boldsymbol{q}_t)$ 不随时间变化

Hamiltonian Monte Carlo 方法

在 HMC 中

- 上述 θ 对应要抽样的变量
- 抽样的目标分布为 $\pi(\boldsymbol{\theta})$, 令 $U(\boldsymbol{\theta}) = -\log \pi(\boldsymbol{\theta})$
- HMC 为每一个分量 θ_j 引入一个辅助的 "动量" 变量 q_j, 一般令

$$\boldsymbol{q} \sim N_d(\mathbf{0}, M), \ M = \mathrm{diag}(m_1, \ldots, m_d)$$

$$K(\boldsymbol{q}) = -\log p(\boldsymbol{q}) = \boldsymbol{q}^\top M^{-1} \boldsymbol{q}/2 = \sum_{j=1}^{d} \frac{q_j^2}{2m_j}$$

- 根据方程组(6)

$$\begin{aligned} \frac{d\theta_j}{dt} &= \frac{q_j}{m_j} \\ \frac{dq_j}{dt} &= \frac{\partial \log \pi(\boldsymbol{\theta})}{\partial \theta_j} \end{aligned} \tag{7}$$

HMC 在每步迭代中使用 Metropolis 方法更新 $(\boldsymbol{\theta}, \boldsymbol{q})$, 候选样本从(7)中产生，这种方式产生的候选样本可以与当前样本距离较远，且能保证较高的接受概率

Hamiltonian Monte Carlo 方法

为模拟 $(\boldsymbol{\theta}, \boldsymbol{q})$ 的运动，以时间间隔 ϵ 对方程组(7)进行离散化近似: 从 $t = 0$ 开始，依次计算 $t = \epsilon, 2\epsilon, \ldots$ 时的 $\boldsymbol{\theta}(t)$ 和 $\boldsymbol{q}(t)$

- Euler 方法

$$q_j(t+\epsilon) = q_j(t) + \epsilon \frac{dq_j(t)}{dt} = q_j(t) - \epsilon \frac{\partial U(\boldsymbol{\theta}(t))}{\partial \theta_j}$$

$$\theta_j(t+\epsilon) = \theta_j(t) + \epsilon \frac{d\theta_j(t)}{dt} = \theta_j(t) + \epsilon \frac{q_j(t)}{m_j}$$

- 改进的 Euler 方法

$$q_j(t+\epsilon) = q_j(t) - \epsilon \frac{\partial U(\boldsymbol{\theta}(t))}{\partial \theta_j}$$

$$\theta_j(t+\epsilon) = \theta_j(t) + \epsilon \frac{q_j(t+\epsilon)}{m_j}$$

在更新 θ_j 时代入新的 q_j 的值

Hamiltonian Monte Carlo 方法

- Leapfrog 方法

$$
\begin{aligned}
q_j(t+\epsilon/2) &= q_j(t) - \frac{\epsilon}{2} \cdot \frac{\partial U(\boldsymbol{\theta}(t))}{\partial \theta_j} \\
\theta_j(t+\epsilon) &= \theta_j(t) + \epsilon \frac{q_j(t+\epsilon/2)}{m_j} \\
q_j(t+\epsilon) &= q_j(t+\epsilon/2) - \frac{\epsilon}{2} \cdot \frac{\partial U(\boldsymbol{\theta}(t+\epsilon))}{\partial \theta_j}
\end{aligned} \tag{8}
$$

  每步更新时，先对动量变量 q_j 更新半步，代入新的 q_j 再对位置变量 θ_j 更新一整步，代入新的 θ_j 后再对 q_j 更新剩下的半步

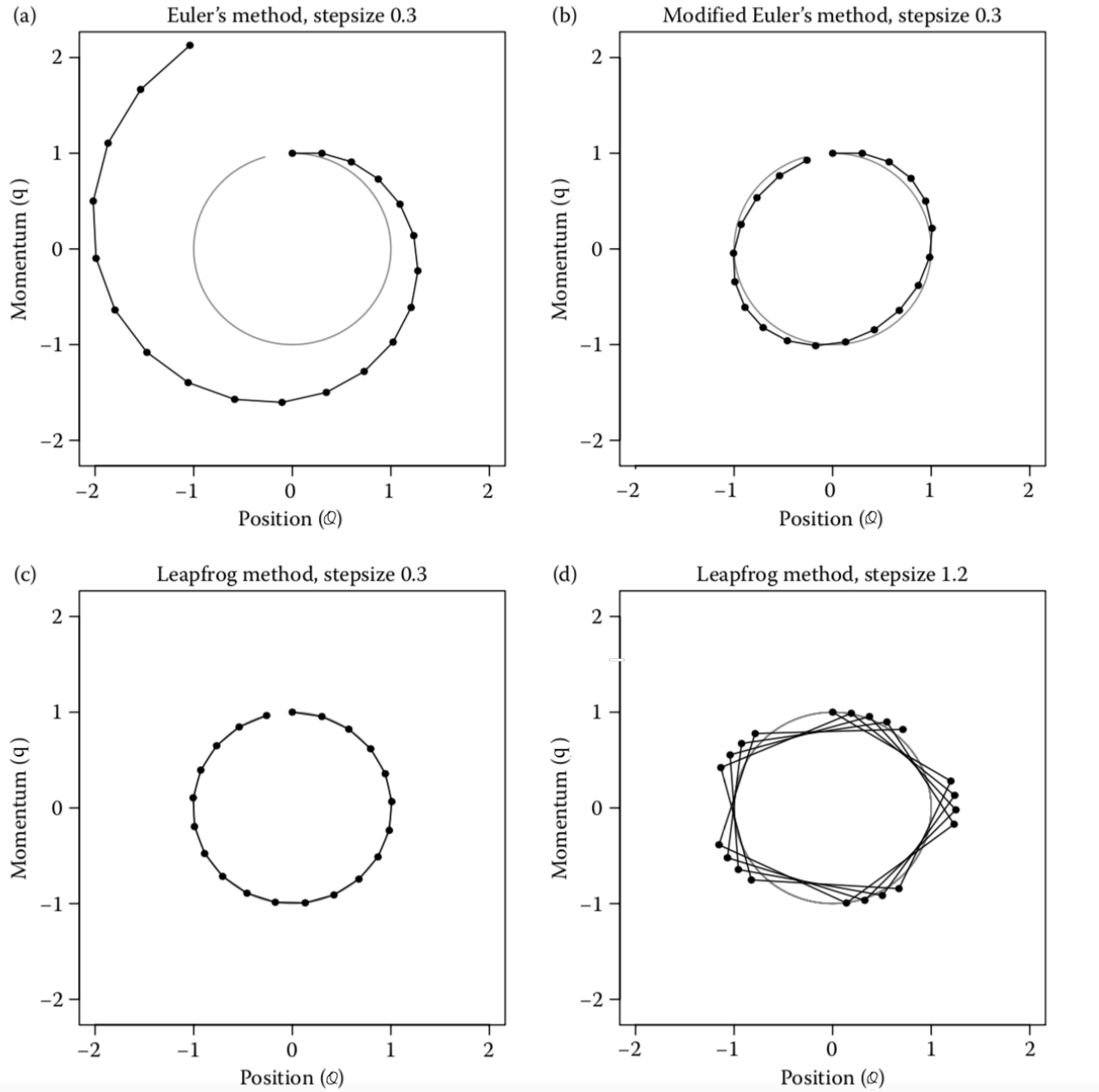
- 一维 Hamiltonian dynamics 例子:

  $H(\theta, q) = U(\theta) + K(q), \quad U(\theta) = \frac{\theta^2}{2}, \quad K(q) = \frac{q^2}{2}$

  根据 Hamilton 方程组(6), $\frac{d\theta}{dt} = q, \quad \frac{dq}{dt} = -\theta$

  该微分方程组存在解析解: $\theta(t) = r\cos(a+t), \quad q(t) = -r\sin(a+t)$

Hamiltonian Monte Carlo 方法



Hamiltonian Monte Carlo 方法

- Euler 方法及改进的 Euler 方法的局部误差都是 $O(\epsilon^2)$, 全局误差 $O(\epsilon)$; leapfrog 方法的局部误差为 $O(\epsilon^3)$, 全局误差 $O(\epsilon^2)$
- 在 HMC 算法中，目标分布 $\pi(\boldsymbol{\theta})$ 的归一化常数可以未知，因此估计 Bayesian 模型时，可以令 $U(\boldsymbol{\theta}) = -\log\left[L(\boldsymbol{y} \mid \boldsymbol{\theta})p(\boldsymbol{\theta})\right]$
- 给定初始值 $\boldsymbol{\theta}^{(0)}$, 假设已获得当前样本 $\boldsymbol{\theta}^{(t)}$, HMC 算法如下产生新的样本 $\boldsymbol{\theta}^{(t+1)}$:
  1. 抽样 $\boldsymbol{q} \sim N_d(\mathbf{0}, M)$
  2. 从 $(\boldsymbol{\theta}^{(t)}, \boldsymbol{q})$ 出发，使用 leapfrog 方法按 Hamilton 方程组(7)移动 L 步，每步时间间隔 ϵ, 得到候选样本 $(\boldsymbol{\theta}^\star, \boldsymbol{q}^\star)$
  3. 计算接受比率

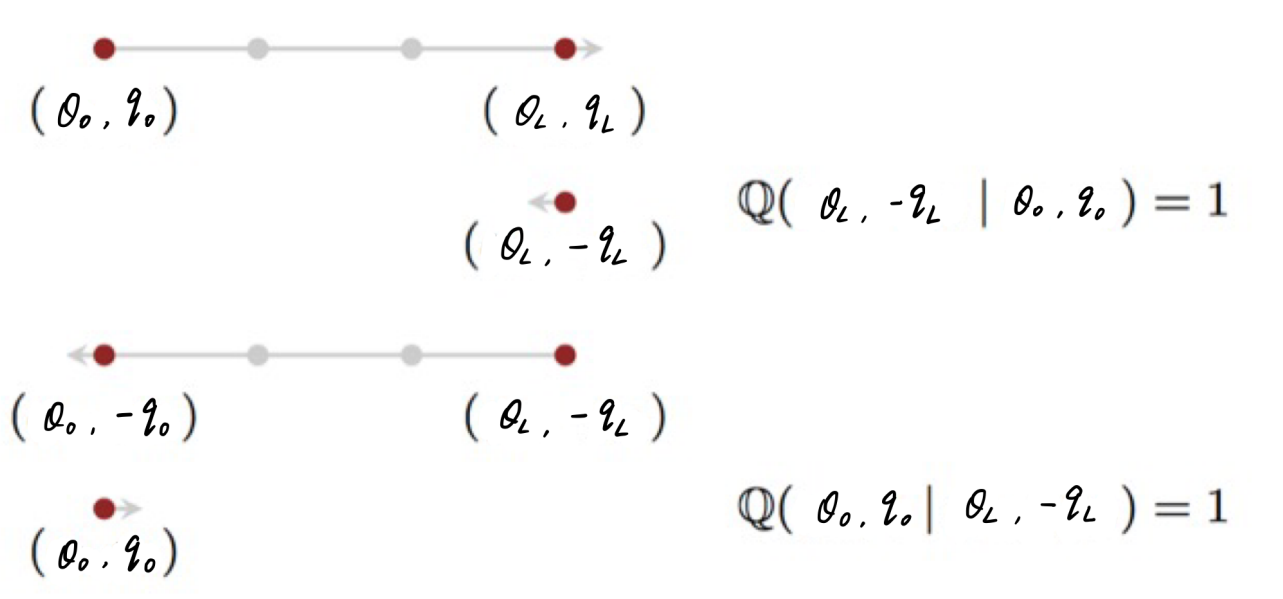
$$r = \exp\left[H(\boldsymbol{\theta}^{(t)}, \boldsymbol{q}) - H(\boldsymbol{\theta}^\star, \boldsymbol{q}^\star)\right] = \exp\left[U(\boldsymbol{\theta}^{(t)}) + K(\boldsymbol{q}) - U(\boldsymbol{\theta}^\star) - K(\boldsymbol{q}^\star)\right] \tag{9}$$

令

$$\boldsymbol{\theta}^{(t+1)} = \begin{cases} \boldsymbol{\theta}^\star & \text{概率 } \min(1, r) \\ \boldsymbol{\theta}^{(t)} & \text{概率 } 1 - \min(1, r). \end{cases}$$

Hamiltonian Monte Carlo 方法

- 视频 ▸ HMC 展示了 HMC 生成马尔可夫链的动态过程
- HMC 每步迭代使用的 "proposal 分布" 对应 L 步 leapfrog, 这一过程是确定的，即 $Q(\boldsymbol{\theta}_L, \boldsymbol{q}_L \mid \boldsymbol{\theta}_0, \boldsymbol{q}_0) = 1$，如果在 L 步 leapfrog 结束后将动量变量 $\boldsymbol{q}$ 反向，由于系统的运动轨迹是可逆的，再运行 L 步 leapfrog, $(\boldsymbol{\theta}, \boldsymbol{q})$ 将 "原路" 返回到初始位置，将动量变量反向即得 $(\boldsymbol{\theta}_0, \boldsymbol{q}_0)$



此时的 "proposal 分布" $Q()$ 是对称的. 但在实践中可以省略对 $\boldsymbol{q}_L$ 的反向操作

Hamiltonian Monte Carlo 方法

- 由于 HMC 产生候选样本的机制是对称的，在选取的接受比率(9)下，HMC 生成的马尔可夫链满足 detail balance 条件，且 $(\boldsymbol{\theta}, \boldsymbol{q})$ 在平稳分布下的联合概率密度函数为 $\exp\left[-H(\boldsymbol{\theta}, \boldsymbol{q})\right]$，因此边际马尔可夫链 $\{\boldsymbol{\theta}_t\}$ 会收敛到目标分布 $\pi(\boldsymbol{\theta})$

- 如果 HMC 产生的马尔可夫链不会陷入到局部区域，那么一般是遍历的. 如果 leapfrog 轨迹具有某种周期性，经过 L 步 leapfrog 产生的候选样本 $(\boldsymbol{\theta}^\star, \boldsymbol{q}^\star)$ 可能与原点 $(\boldsymbol{\theta}^{(t)}, \boldsymbol{q}^{(t)})$ 几乎重合，此时生成的马尔可夫链 $\{\boldsymbol{\theta}_t\}$ 不具有遍历性

  - 解决办法之一: 每次产生候选样本时，在较小的范围内随机选取 ϵ 和 L
  - Hoffman and Gelman (2014) 提出了 no U-turn sampler (NUTS)

Hamiltonian Monte Carlo 方法

- HMC 有三处可调参数: (i) $\boldsymbol{q}$ 的协方差矩阵 M (ii) leapfrog 的时间间隔 ϵ (iii) leapfrog 的步数 L
  - 一般选取 $M = I$，在满足一定假设条件下，HMC 最优接受比率约为 65% (Neal et al., 2011)
  - 如果马尔可夫链的整体接受比率较低，可能是 leapfrog 每步 "跳跃" 太大，可以减小 ϵ, 增加 L; 相反如果接受比率过高，可以增加 ϵ, 减小 L
- HMC 每步迭代都先抽一个新的 $\boldsymbol{q}$ 是为了改变函数 $H(\boldsymbol{\theta}, \boldsymbol{q})$ 的值, 因为当 leapfrog 时间步长 ϵ 足够小时，$H(\boldsymbol{\theta}, \boldsymbol{q})$ 的值几乎不变，如果每步迭代不对 $\boldsymbol{q}$ 重新抽样，则 $H(\boldsymbol{\theta}^{(t)}, \boldsymbol{q}^{(t)})$ 的值几乎不随 t 变化，而 $U(\boldsymbol{\theta}^{(t)})$ 和 $K(\boldsymbol{q}^{(t)})$ 一般都非负，此时 $U(\boldsymbol{\theta}^{(t)})$ 总是无法超过初始值 $H(\boldsymbol{\theta}^{(0)}, \boldsymbol{q}^{(0)})$

使用 HMC 估计 Bayesian 混合效应模型

- R package `lme4` 提供了一项研究睡眠不足与反应时间关系的公开数据 sleepstudy (Belenky et al., 2003)
- 该数据记录了 18 个受试者在前 10 天睡眠不足的情况下每天的反应时间 (ms)

```
library(lme4)
str(sleepstudy)
'data.frame':^^I180 obs. of  3 variables:
$ Reaction: num  250 259 251 321 357 ...
$ Days    : num  0 1 2 3 4 5 6 7 8 9 ...
$ Subject : Factor w/ 18 levels "308","309","310",..: 1 1 1 1 1
    1 1 1 1 1 ...
```

使用 HMC 估计 Bayesian 混合效应模型

- 当数据涉及分组结构且每个组内有多个观察值时，比较适合用混合效应模型 (mixed effects model)来分析
  - 区分数据在**不同组间的变化**和**同一组内的变化**
  - 捕捉解释变量对结果的影响如何随分组不同而改变
  - 描述同一组内观察值之间的相关性

- Notation
  - y_{ij}：受试者 j 第 i 个反应时间的观察值
  - D_{ij}：y_{ij} 对应的睡眠不足的天数

- 考虑到每个受试者初始的反应时间及睡眠不足对反应时间的影响都可能因人而异，建立如下的混合效应模型:

$$y_{ij} = \mu_0 + \gamma_{0j} + (\mu_1 + \gamma_{1j}) D_{ij} + \varepsilon_{ij},\ \varepsilon_{ij} \overset{iid}{\sim} N(0, \sigma_e^2),\ i = 1, \ldots, n_j \quad (10)$$

$$\begin{pmatrix} \gamma_{0j} \\ \gamma_{1j} \end{pmatrix} \overset{iid}{\sim} N_2 \left(\mathbf{0}, \Sigma = \begin{pmatrix} \sigma_0^2 & \sigma_{01} \\ \sigma_{01} & \sigma_1^2 \end{pmatrix} \right),\ j = 1, \ldots, J \quad (11)$$

使用 HMC 估计 Bayesian 混合效应模型

(10)描述的是**同一组内 (within-group) 的数据分布**

$$\boldsymbol{y}_j = \begin{pmatrix} y_{1j} \\ \vdots \\ y_{n_j,j} \end{pmatrix}, \boldsymbol{X}_j = \begin{pmatrix} 1 & D_{1j} \\ \vdots & \vdots \\ 1 & D_{n_j,j} \end{pmatrix}, \boldsymbol{\mu} = \begin{pmatrix} \mu_0 \\ \mu_1 \end{pmatrix}, \boldsymbol{\gamma}_j = \begin{pmatrix} \gamma_{0j} \\ \gamma_{1j} \end{pmatrix}$$

- 受试者 j 所有观察到的反应时间 $\boldsymbol{y}_j$ 服从条件分布：

$$\boldsymbol{y}_j \sim N_{n_j}\left(\boldsymbol{X}_j\boldsymbol{\mu} + \boldsymbol{X}_j\boldsymbol{\gamma}_j, \sigma_e^2 I_{n_j}\right) \tag{12}$$

- 称 $\boldsymbol{\mu} = (\mu_0, \mu_1)^\top$ 为固定效应 (fixed effects)系数，它们是未知的常数
  - μ_0：实验开始时受试者的平均反应时间
  - μ_1：反应时间 (y) 随睡眠不足天数 (D) 增加的平均增长速率

使用 HMC 估计 Bayesian 混合效应模型

(11)描述的是**不同组间 (between-group)** 回归系数 γ_j 的分布

- 称 $\gamma_1, \ldots, \gamma_J$ 为随机效应 (random-effects)参数，它们是随机向量
- (11)不是 $\gamma_1, \ldots, \gamma_J$ 的先验分布，而是模型的一部分，其中的协方差矩阵 Σ 也是待估计的参数
- (11)起到了不同组间信息共享的作用，它使得从样本较小的组估计的 γ_j 更稳定
- 如果将条件分布(12)中的随机向量 γ_j 积分掉，可得 $\boldsymbol{y}_j$ 的边际分布:

$$\boldsymbol{y}_j \sim N_{n_j}\left(\boldsymbol{X}_j\boldsymbol{\mu}, \sigma_e^2 I_{n_j} + \boldsymbol{X}_j \Sigma \boldsymbol{X}_j^\top\right)$$

通过给组特有 (group-specific) 系数 γ_j 加入随机性(11)，混合效应模型实现了描述组内观察值之间的相关性

使用 HMC 估计 Bayesian 混合效应模型

为了更好地描述模型参数的不确定性，使用 Bayesian 方法估计上述混合效应模型(10) - (11)

- 为保证先验分布包含参数的真实值，可以为 μ_0, μ_1 选取正态先验分布，为 σ_e^2 和 Σ 分别选取 inverse-gamma 和 inverse-Wishart 分布
- `rstan` 在估计 Bayesian 模型时，推荐让协方差矩阵服从 LKJ prior (Lewandowski et al., 2009)，它使 HMC 算法运行得更高效，同时保证协方差矩阵的后验样本是对称正定的
  - LKJ prior 一般加在相关系数矩阵的 Cholesky 分解矩阵上，对协方差矩阵做分解

    $$\Sigma = \begin{pmatrix} \sigma_0 & 0 \\ 0 & \sigma_1 \end{pmatrix} \Omega \begin{pmatrix} \sigma_0 & 0 \\ 0 & \sigma_1 \end{pmatrix}$$

    相关系数矩阵 Ω 存在 Cholesky 分解 $\Omega = LL^\top$，其中 L 是下三角矩阵，令 L 服从 LKJ prior
  - 根据上述分解，对 γ_j 重新参数化，令

    $$\gamma_j = \begin{pmatrix} \sigma_0 & 0 \\ 0 & \sigma_1 \end{pmatrix} L\eta_j,\ \eta_j \sim N_2(\mathbf{0}, I_2),\ j = 1, \ldots, J$$

使用 HMC 估计 Bayesian 混合效应模型

使用 rstan 估计参数的后验分布. 首先，需要把数据组织成一个 list object:

```
d_stan = list(Subject = as.numeric(factor(sleepstudy$Subject,
labels=1:length(unique(sleepstudy$Subject)))),
Days = sleepstudy$Days,
RT = sleepstudy$Reaction/1000,
N = nrow(sleepstudy),
J = length(unique(sleepstudy$Subject)) )
```

使用 HMC 估计 Bayesian 混合效应模型

然后打开一个文本文档，输入以下几部分程序，保存为 “.stan” 文件，比如 “sleep_model.stan”

- 在该文档中首先对数据进行变量声明

```
data {
int<lower=1> N;                  // number of observations
real RT[N];                      // reaction time

// predictor (days of sleep deprivation)
int<lower=0,upper=9> Days[N];

// grouping factor
int<lower=1> J;                          // number of subjects
int<lower=1,upper=J> Subject[N];  // subject id
}
```

使用 HMC 估计 Bayesian 混合效应模型

- 其次列出待估计的参数

```
parameters {
vector[2] mu;                        // fixed-effects parameters
real<lower=0> sigma_e;                 // residual std
// random effects standard deviations
vector<lower=0>[2] sigma_gam;
// declare L to be the Cholesky factor of a 2x2 correlation
// matrix
cholesky_factor_corr[2] L;
matrix[2,J] eta;                       // random effect matrix
}

transformed parameters {
// this transform random effects so that they have the corr
// matrix specified by the correlation matrix above
matrix[2,J] gamma;
gamma = diag_pre_multiply(sigma_gam, L) * eta;
}
```

使用 HMC 估计 Bayesian 混合效应模型

- 模型设定

```
model {
real m_RT; // conditional mean of y

// priors
// LKJ prior for the Cholesky factor of correlation matrix
L ~ lkj_corr_cholesky(1.5);
to_vector(eta) ~ normal(0,1); // elementwise prior
// prior for residual standard deviation
sigma_e ~ normal(0,5);
mu[1] ~ normal(0.3, 0.5);  // prior for fixed-effect intercept
mu[2] ~ normal(0.2, 2);    // prior for fixed-effect slope

// likelihood
for (i in 1:N){
m_RT = mu[1] + gamma[1,Subject[i]] + (mu[2]+gamma[2,Subject[i
    ]])*Days[i];
RT[i] ~ normal(m_RT, sigma_e);
}
}
```

使用 HMC 估计 Bayesian 混合效应模型

- 最后在文档中加入以下代码储存随机效应的相关系数矩阵 Ω 的后验样本

```
generated quantities {
matrix[2, 2] Omega;
Omega = L * L';   // so that it returns the correlation matrix
}
```

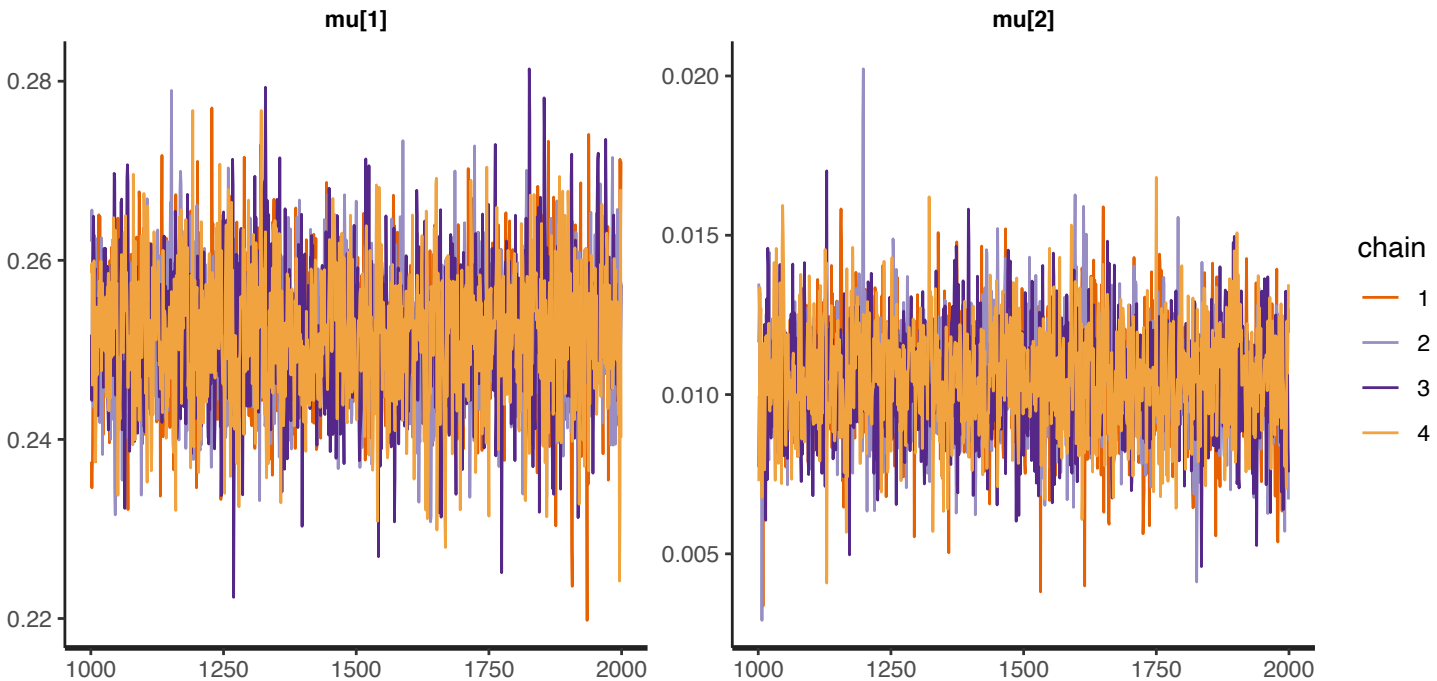
在 R 中将工作目录设为 sleep_model.stan 文件所在的文件夹，然后调用 stan 函数估计上述模型

```
library(rstan)
# indicate stan to use multiple cores if available
options(mc.cores = parallel::detectCores())
sleep_model <- stan(file = "sleep_model.stan", data = d_stan,
iter = 2000, chains = 4)
```

使用 HMC 估计 Bayesian 混合效应模型

检查参数后验样本的移动轨迹以判断模型的收敛性：

```
traceplot(sleep_model, pars = c("mu"), inc_warmup = FALSE)
```



使用 HMC 估计 Bayesian 混合效应模型

print 函数可以总结参数估计的结果：

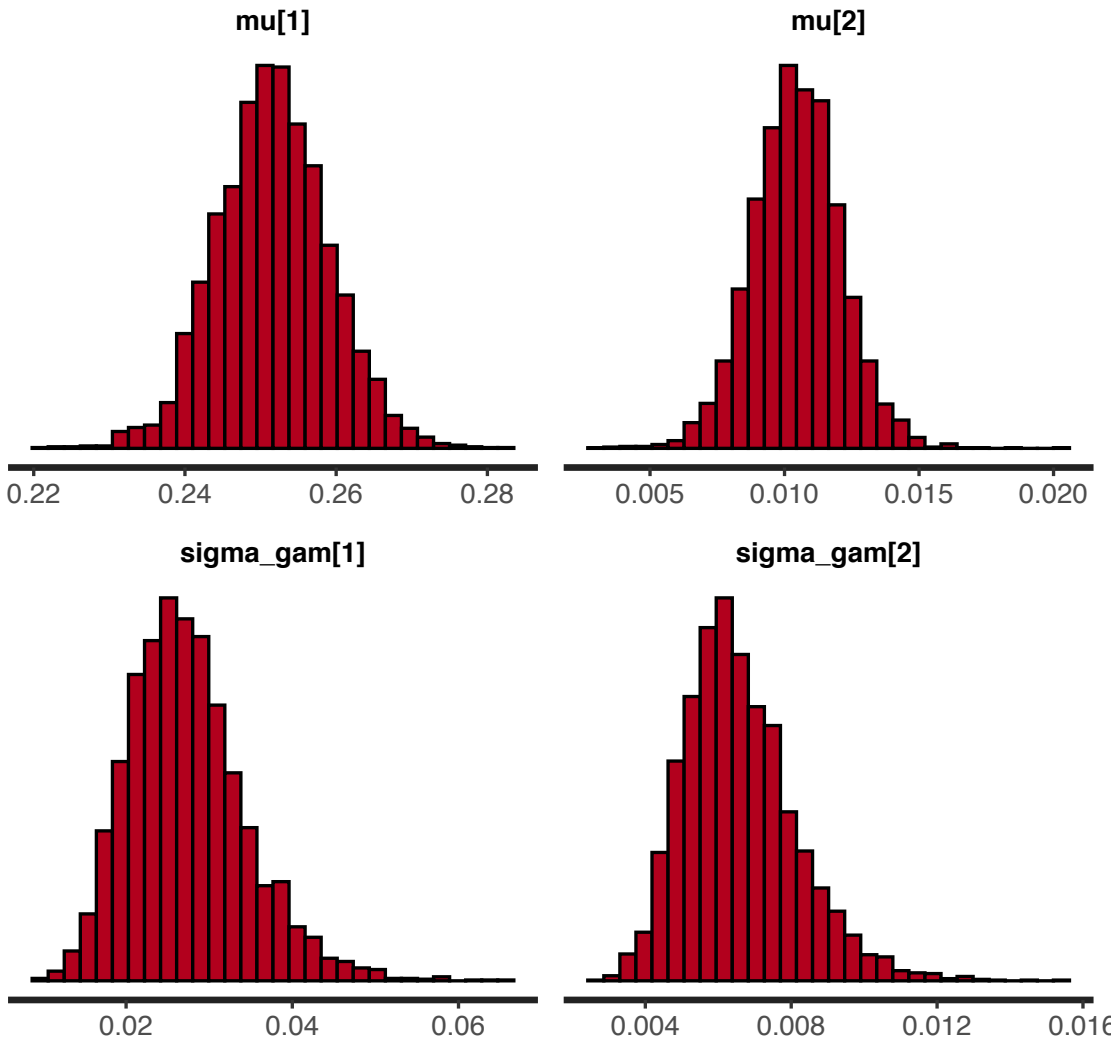
```
print(sleep_model, pars = c("mu"), probs = c(0.025, 0.975),
      digits = 3)
Inference for Stan model: sleep_model.
4 chains, each with iter=2000; warmup=1000; thin=1;
post-warmup draws per chain=1000, total post-warmup draws=4000.

       mean se_mean    sd  2.5% 97.5% n_eff  Rhat
mu[1] 0.252       0 0.007 0.237 0.266  2082 1.000
mu[2] 0.010       0 0.002 0.007 0.014  2496 1.001
```

使用 HMC 估计 Bayesian 混合效应模型

plot 函数可以展示参数的后验分布：

```
plot(sleep_model,plotfun = "hist",pars = c("mu", "sigma_gam"))
```



使用 HMC 估计 Bayesian 混合效应模型

再检查一下随机效应的相关系数矩阵的估计：

```
print(sleep_model, pars = c("Omega"), digits = 3)
Inference for Stan model: sleep_model.
4 chains, each with iter=2000; warmup=1000; thin=1;
post-warmup draws per chain=1000, total post-warmup draws=4000.

              mean se_mean   sd   2.5%    25%   50%  75% 97.5% n_eff  Rhat
Omega[1,1]   1.000     NaN 0.000  1.00  1.000 1.000 1.00 1.000   NaN   NaN
Omega[1,2]   0.082   0.008 0.288 -0.46 -0.125 0.075 0.29 0.641  1319 1.003
Omega[2,1]   0.082   0.008 0.288 -0.46 -0.125 0.075 0.29 0.641  1319 1.003
Omega[2,2]   1.000   0.000 0.000  1.00  1.000 1.000 1.00 1.000  4045 0.999
```

最后与混合效应模型(11)-(10)的 restricted MLE (REML) 结果做比较

- 在混合效应模型中，方差部分的 MLE 是有偏的，REML 考虑了估计固定效应损失的自由度，得到的是无偏估计
- R package `lme4` 的 `lmer` 函数可以给出线性混合效应模型的 REML

使用 HMC 估计 Bayesian 混合效应模型

```
fm1 = lmer(Reaction/1000 ~ Days + (Days | Subject), sleepstudy)
summary(fm1)
Linear mixed model fit by REML ['lmerMod']
Formula: Reaction/1000 ~ Days + (Days | Subject)
Data: sleepstudy

REML criterion at convergence: -715.5

Scaled residuals:
Min      1Q  Median      3Q     Max
-3.9536 -0.4634  0.0231  0.4633  5.1793

Random effects:
Groups   Name        Variance  Std.Dev. Corr
Subject  (Intercept) 6.119e-04 0.024737
Days        3.508e-05 0.005923 0.07
Residual              6.549e-04 0.025592
Number of obs: 180, groups:  Subject, 18

Fixed effects:
Estimate Std. Error t value
(Intercept) 0.251405   0.006824  36.843
Days        0.010467   0.001546   6.771
```

Sequential Monte Carlo 方法

- SMC 是估计状态空间模型的常用方法
  - 状态空间模型 (state space model) 是一个时间序列模型，由状态方程和观测方程组成，描述一列可观测的变量和不可观测的状态变量之间的动态关系

- 线性高斯模型

$$\begin{cases} X_t &= AX_{t-1} + U\epsilon_t,\ \epsilon_t \sim N_p(\mathbf{0}, I_p) \\ Y_t &= BX_t + V\eta_t,\ \eta_t \sim N_q(\mathbf{0}, I_q) \end{cases}$$

  - $\{Y_t\}_{t=1}^T$ 是可观测的随机向量，其观察值为 $\{\boldsymbol{y}_t\}_{t=1}^T$
  - $\{X_t\}_{t=1}^T$ 是不可观测的状态变量，其数目随时间 t 增加
  - A, B, U, V 是已知的常数矩阵
  - 目标：估计给定 $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_t$ 下 X_t 的条件分布，$t = 1, \ldots, T$

Kalman filter

- 在初始时刻 $t = 0$，假设 $X_0 \sim N(\boldsymbol{\mu}_0, \Sigma_0)$
- 对 $t = 1, 2, \ldots, T$
  - 给定 $X_{t-1} \mid \boldsymbol{y}_1, \ldots, \boldsymbol{y}_{t-1} \sim N(\boldsymbol{\mu}_{t-1}, \Sigma_{t-1})$
  - 则 $(X_t, Y_t) \mid \boldsymbol{y}_1, \ldots, \boldsymbol{y}_{t-1} \sim N(\boldsymbol{\theta}^{(t)}, \Omega^{(t)})$, 其中

$$\boldsymbol{\theta}^{(t)} = \begin{pmatrix} \boldsymbol{\theta}_X^{(t)} \\ \boldsymbol{\theta}_Y^{(t)} \end{pmatrix} = \begin{pmatrix} A\boldsymbol{\mu}_{t-1} \\ BA\boldsymbol{\mu}_{t-1} \end{pmatrix}$$

$$\Omega^{(t)} = \begin{pmatrix} \Omega_{XX}^{(t)} & \Omega_{XY}^{(t)} \\ \Omega_{YX}^{(t)} & \Omega_{YY}^{(t)} \end{pmatrix} = \begin{pmatrix} A\Sigma_{t-1}A^\top + UU^\top, & (A\Sigma_{t-1}A^\top + UU^\top)B^\top \\ B(A\Sigma_{t-1}A^\top + UU^\top), & B(A\Sigma_{t-1}A^\top + UU^\top)B^\top + VV^\top \end{pmatrix}$$

  - 此时 $X_t \mid \boldsymbol{y}_1, \ldots, \boldsymbol{y}_t \sim N(\boldsymbol{\mu}_t, \Sigma_t)$, 其中

$$\boldsymbol{\mu}_t = \boldsymbol{\theta}_X^{(t)} + \Omega_{XY}^{(t)} \left(\Omega_{YY}^{(t)}\right)^{-1} (\boldsymbol{y}_t - \boldsymbol{\theta}_Y^{(t)})$$

$$\Sigma_t = \Omega_{XX}^{(t)} - \Omega_{XY}^{(t)} \left(\Omega_{YY}^{(t)}\right)^{-1} \Omega_{YX}^{(t)}$$

状态空间模型

一般的状态空间模型可写为以下形式：

$$\begin{cases} X_t \sim g_t(x \mid x_0, \ldots, x_{t-1}) & \text{状态方程} \\ Y_t \sim f_t(y \mid x_0, \ldots, x_t) & \text{观测方程} \end{cases} \tag{13}$$

其中每一时刻 t 的状态分布 g_t 和观测分布 f_t 是已知的，如何根据观察到的 $y_1, \ldots, y_t$，对未知的状态变量 X_t 进行实时估计？

- filtering: 估计条件期望 $E(X_t \mid y_1, \ldots, y_t)$, $t = 1, 2, \ldots$
- smoothing: 估计未来时刻 $T > t$ 下的条件期望 $E(X_t \mid y_1, \ldots, y_T)$

$$E(X_t \mid y_1, \ldots, y_t) = \frac{\int \cdots \int x_t g_0(x_0) \prod_{s=1}^{t} \left(g_s(x_s \mid x_{0:s-1}) f_s(y_s \mid x_{0:s})\right) dx_0 \cdots dx_t}{\int \cdots \int g_0(x_0) \prod_{s=1}^{t} \left(g_s(x_s \mid x_{0:s-1}) f_s(y_s \mid x_{0:s})\right) dx_0 \cdots dx_t}$$

- 随着 t 增加，高维积分很难有解析形式
- 数值积分方法的计算量过大
- SMC 方法可以避免高维积分

Importance Sampling 方法

- Importance sampling 是 SMC 方法的基础
- 如何估计目标分布 $\pi(x)$ 的期望 $\mu = E_\pi(X) = \int x\pi(x)dx$？该积分没有显式表达式且很难从 $\pi(x)$ 抽样
- Importance sampling 的基本想法：先从一个 proposal 分布 $q(x)$ 抽样，然后通过修正样本权重来近似目标期望 μ
  1. 从 proposal 分布产生样本 $x_1, x_2, \ldots, x_N$
  2. 计算每个样本的权重：

$$w_j \propto \pi(x_j)/q(x_j),\ j = 1, \ldots, N$$

  3. μ 的估计量为

$$\hat{\mu} = \frac{\sum_{j=1}^{N} w_j x_j}{\sum_{j=1}^{N} w_j}$$

- 计算样本权重 w_j 时，允许 $\pi(x)$ 存在未知的归一化常数
- $\hat{\mu}$ 是 μ 的一致 (consistent) 估计量

Sequential Importance Sampling

- 使用 importance sampling 估计状态空间模型的条件期望 $E(X_t \mid y_1, \ldots, y_t)$, 需要为状态变量选取一个 proposal 分布:

$$q(x_{0:t}) = q_0(x_0) q_1(x_1 \mid x_0) \cdots q_t(x_t \mid x_{0:t-1})$$

- 状态变量 $X_{0:t}$ 的目标分布为

$$\pi(x_{0:t}) = p(x_{0:t} \mid y_{1:t}) \propto g_0(x_0) \prod_{s=1}^{t} g_s(x_s \mid x_{0:s-1}) f_s(y_s \mid x_{0:s})$$

- 来自 proposal 分布的样本 $x_{0:t}$ 的权重为

$$\begin{aligned} w(x_{0:t}) &= \frac{\pi(x_{0:t})}{q(x_{0:t})} = \frac{g_0(x_0) \prod_{s=1}^{t} g_s(x_s \mid x_{0:s-1}) f_s(y_s \mid x_{0:s})}{q_0(x_0) \prod_{s=1}^{t} q_s(x_s \mid x_{0:s-1})} \\ &= w_{t-1}(x_{0:t-1}) \frac{g_t(x_t \mid x_{0:t-1}) f_t(y_t \mid x_{0:t})}{q_t(x_t \mid x_{0:t-1})} \end{aligned} \tag{14}$$

Sequential Importance Sampling

- SIS 方法

  1. 从 proposal 分布抽大量样本 $x_{0:t}^{(j)}$, $j = 1, \ldots, N$, 每个样本也被称为粒子(particle)
  2. 按照(14)计算每个粒子 $x_{0:t}^{(j)}$ 的权重 $w_t^{(j)}$,
  3. $\mu = E(X_t \mid y_1, \ldots, y_t)$ 可如下估计:

$$\hat{\mu} = \sum_{j=1}^{N} x_t^{(j)} w_t^{(j)} / \sum_{j=1}^{N} w_t^{(j)} \tag{15}$$

- SIS 方法的一个缺陷: 随着时间 t 增加, 粒子的权重 $\{w_t^{(j)}\}$ 会变得越来越不均匀, 使用过多权重很小的粒子计算(15)是一种浪费

重抽样 (resampling)

- 为解决粒子权重过度偏斜的问题，人们设计了一个重抽样步骤：
  1. 给每个粒子 $x_{0:t}^{(j)}$ 分配一个概率 $\alpha_t^{(j)}$，$j = 1, \ldots, N$ 且 $\sum_{j=1}^{N} \alpha_t^{(j)} = 1$
  2. 对 $j = 1, \ldots, N$
     - ★ 从集合 $\{x_{0:t}^{(i)} : i = 1, \ldots, N\}$ 中按概率 $\{\alpha_t^{(i)} : i = 1, \ldots, N\}$ 随机抽一个样本 $x_{0:t}^{\star(j)}$
     - ★ 如果 $x_{0:t}^{\star(j)} = x_{0:t}^{(k)}$，给 $x_{0:t}^{\star(j)}$ 赋予新权重 $w_t^{\star(j)} = w_t^{(k)} / \alpha_t^{(k)}$
  3. 输出新的带权样本集 $\{(x_{0:t}^{\star(j)}, w_t^{\star(j)}) : j = 1, \ldots, N\}$

- Gordon et al. (1993) 使用粒子归一化的权重 $\{\alpha_t^{(j)} = w_t^{(j)} / \sum_{j=1}^{N} w_t^{(j)}\}$ 作为重抽样的概率，但是当粒子的权重极度偏斜时，重抽样会造成粒子多样性的退化

- Liu (2008) 从保护粒子多样性的角度给出如下形式的重抽样概率：

$$\alpha_t^{(j)} \propto \left[w_t^{(j)}\right]^{\alpha}, \ \alpha > 0, \ j = 1, \ldots, N$$

Sequential Monte Carlo 方法

在 SIS 方法中加入重抽样的算法被称为 SMC 算法：

- $t = 0$ 时，抽样 $x_0^{(j)} \sim q_0(x)$，并令 $w_0^{(j)} = g_0(x_0^{(j)})/q_0(x_0^{(j)})$，$j = 1, \ldots, N$
- 对 $t = 1, \ldots, T$
  1. 抽样：$\tilde{x}_t^{(j)} \sim q_t(x \mid x_{0:t-1}^{(j)})$，并令 $\tilde{x}_{0:t}^{(j)} = (x_{0:t-1}^{(j)}, \tilde{x}_t^{(j)})$，$j = 1, \ldots, N$
  2. 更新权重：令 $\tilde{w}_t^{(j)} = w_{t-1}^{(j)} u_t^{(j)}$，其中

$$u_t^{(j)} = \frac{g_t(\tilde{x}_t^{(j)} \mid x_{0:t-1}^{(j)}) f_t(y_t \mid \tilde{x}_{0:t}^{(j)})}{q_t(\tilde{x}_t^{(j)} \mid x_{0:t-1}^{(j)})}, \quad j = 1, \ldots, N$$

  3. 推断：计算目标期望 $E(h(x_{0:t}) \mid y_{1:t})$ 的估计量

$$\frac{\sum_{j=1}^{N} \tilde{w}_t^{(j)} h(\tilde{x}_{0:t}^{(j)})}{\sum_{j=1}^{N} \tilde{w}_t^{(j)}}$$

  4. 重抽样：按照权重 $\{\alpha_t^{(j)} : j = 1, \ldots, N\}$ 对粒子集合 $\{\tilde{x}_{0:t}^{(j)} : j = 1, \ldots, N\}$ 进行重抽样，得到一组新的带权粒子集 $\{(x_{0:t}^{(j)}, w_t^{(j)}) : j = 1, \ldots, N\}$