

# 医薬品開発の意思決定における

## Bayesian Posterior Probabilityの適用例

### ～ Random-Walk Metropolis vs. No-U-Turn Sampler ～

○作井 将、清水 康平、舟尾 暢男

武田薬品工業株式会社 日本開発センター 生物統計室

#### Using Bayesian Posterior Probability for Go/No-Go Decision Making in Clinical Development

Sho Sakui, Kohei Shimizu, Nobuo Funao

Takeda Pharmaceutical Company, Ltd.

## 要旨

2つの薬剤効果を比較する並行群間/クロスオーバーデザインにて Bayesian Posterior Probability の算出と意思決定への適用例を紹介するとともに、mcmc プロシジャの random-walk Metropolis と No-U-Turn Sampler の結果を比較する。

キーワード：意思決定 (Go/No-Go Decision)、Bayesian Posterior Probability、並行群間デザイン、クロスオーバーデザイン、mcmc プロシジャ、No-U-Turn Sampler (NUTS)

## 1 はじめに

医薬品開発では様々な場面で意思決定を行う機会があるが、重要なものとしては早期臨床試験終了後に次相以降の臨床試験実施の可否を検討することが挙げられる。特に、第2相試験や第3相試験（後期臨床試験）での失敗は、多くの患者さんに対して有用性が乏しい化合物を曝露したことへの倫理的問題が生じ、臨床試験に携わっていただいた医療機関の方々の多大な努力が無駄になり、医薬品開発コストの高騰や他の新薬候補の開発の遅れにも繋がることから、後期臨床試験への移行に関する意思決定（Go/No-Go Decision）が非常に重要となることは本稿で詳述するまでもない。この問題を解消する方法として、Go/No-Go Decision を行うためのデータを早期臨床試験にて取得することが挙げられる。そのため例えば、後期第2相試験の前に、有効性評価を主目的とした Proof-of-Concept (PoC) 試験を実施することや、第1相試験において有効性評価項目に関するデータを収集することが行われる。ただ、これらを実施した場合でも、従来の頻度論にてデータ解析を行ってしまうと、被験者数が少ないことも遠因となり、当該試験のデータの記述に終始してしまうことが多く、Go/No-Go Decision に繋がる材料が得られにくい。一方、FDA (2010) や Walley et. al. (2015) では Bayesian Posterior Probability による意思決定方法が提案されており、後者では、PoC 試験終了後に有効性評価項目に関して「被験薬の効果がプラセ

ポに対して閾値以上の差がある」確率を Bayesian Posterior Probability として算出し、後期臨床試験への移行に関する Go/No-Go Decision の材料とすることを提案している。

本稿では、2つの薬剤の有効性に関する効果を比較することを想定し、PoC 試験において良く用いられる並行群間デザインに加え、第1相試験においてしばしば適用されるクロスオーバーデザインについて Bayesian Posterior Probability を算出する例を紹介する。それぞれのデザインにおいて、まず解析的又は漸近的に Bayesian Posterior Probability を算出する SAS マクロを紹介した後、次に Markov Chain Monte Carlo (MCMC) にて同様の計算を行うための mcmc プロシジャの記述例を紹介する。mcmc プロシジャにて解析を行う際、random ステートメントによるコーディングを行うことがあるが、本ステートメントを用いると乱数列の収束やサンプリング効率が一般的に落ちる。そこで、比較的新しいサンプリング手法である No-U-Turn Sampler (NUTS) にて解析を行う例を紹介し、mcmc プロシジャにてデフォルトで用いられているアルゴリズム (random-walk Metropolis) との結果の違いも併せて考察することにする。

## 2 解析的/漸近的アプローチによる Bayesian Posterior Probability の算出例

早期臨床試験において2つの薬剤 (1: 被験薬、2: 対照薬) の有効性に関する効果を比較することを考える。治験デザインは並行群間試験又は2×2クロスオーバー試験を想定し、有効性に関する応答に関して「対照薬に対する被験薬の平均値の差又は薬剤間の効果の差が2以上」となれば被験薬に関する臨床的な効果があると判定し、後期臨床試験へ移行することとする。実際の手順としては、例えば治験終了時に後期臨床試験への移行に関する Go/No-Go Decision のために Bayesian Posterior Probability を算出し、本確率が「あらかじめ設定した閾値 (例えば60%、70%、80%)」を上回った場合は後期臨床試験へ移行するものとする。頻度論に対するベイズ解析の利点として、事前情報を事前分布として設定し、解析に加味することが出来ることが挙げられるが、本稿ではこの点については議論せず、無情報事前分布を用いることとする。

### 並行群間デザイン

有効性に関する応答として、被験薬  $X_{1i} \sim N(\mu_1, \sigma_1^2)$ 、対照薬  $X_{2j} \sim N(\mu_2, \sigma_2^2)$ ,  $i = 1, \dots, n_1$ ,  $j = 1, \dots, n_2$ 、薬剤の平均値の差  $\delta = \mu_1 - \mu_2$  に興味があり、平均値の差  $\delta$  の事後分布 ( $\sigma_1^2 = \sigma_2^2 = \sigma^2$  だが  $\sigma^2$  は未知) を求めることを考える。事前分布として無情報事前分布  $\sigma^{-(n_1+n_2+2)}$  を設定すると、 $\delta = \mu_1 - \mu_2$  の事後分布は以下の Shifted & Scaled  $t$  分布に従う (計算過程は繁樹 (1985) を参照)。

- $T = \frac{\delta - (\bar{x}_1 - \bar{x}_2)}{\sqrt{(n_1^{-1} + n_2^{-1})(s_1^2 + s_2^2)/v}}$  は自由度  $v = n_1 + n_2 - 2$  の  $t$  分布に従う
- $\bar{x}_k = n_k^{-1} \sum_{i=1}^{n_k} x_{ki}$ ,  $s_k^2 = \sum_{i=1}^{n_k} (x_{ki} - \bar{x}_k)^2$ ,  $k = 1, 2$

数値例として、各薬剤の例数は20、応答に関する各薬剤の平均値はそれぞれ3と0、標準偏差はそれぞれ4と5、「平均値の差が2以上となる確率」を算出すると、Bayesian Posterior Probability は75.54%となる。

```
%macro PROB1(_DELTA=2, _N1=20, _N2=20, _MU1=3, _MU2=0, _SD1=4, _SD2=5, _LOWER="Y");
data PROB1;
  T=(&_DELTA. -(&_MU1. -&_MU2.))/
  sqrt( (1/&_N1. +1/&_N2.) * ((&_N1. -1)*&_SD1. **2 + (&_N2. -1)*&_SD2. **2) / (&_N1. +&_N2. -2) );
  if (&_LOWER. = "Y") then PROB= cdf("t", T, &_N1. +&_N2. -2);
  else
    PROB=1-cdf("t", T, &_N1. +&_N2. -2);
%mend PROB1;
```

```
run ;
proc print noobs ; run ;
%mend ;
%PROB1(LOWER="N"); * 引数 LOWER="Y" にて分布の下側確率を算出する ;
```

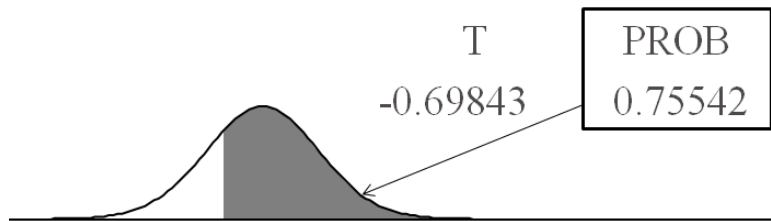


図 2-1 %PROB1 の実行結果と Bayesian Posterior Probability のイメージ

### 2×2 クロスオーバーデザイン

有効性に関する応答を  $Y_{ijk}$  とし、以下の 2×2 クロスオーバーデザイン (AB/BA デザイン) を想定する。

表 2-1 2×2 クロスオーバーデザインの例

	時期 1	時期 2
順序 1 ( $n_1 = 20$ ) 被験薬(1) → 対照薬(2)	$\bar{y}_{11\cdot} = 6$	$\bar{y}_{12\cdot} = 3$
順序 2 ( $n_2 = 20$ ) 対照薬(2) → 被験薬(1)	$\bar{y}_{21\cdot} = 2$	$\bar{y}_{22\cdot} = 5$

解析するモデル式は  $Y_{ijk} = \mu + \pi_j + \tau_{d[i,j]} + \lambda_{d[i,j-1]} + s_{ik} + \varepsilon_{ijk}$  なる分散分析モデルとする。また、Within-subjects residual SS を SSE、Between-subjects residual SS を SSP と略記する。

- $\mu$  : 切片項
- $\pi_j$  : 時期  $j$  の効果 ( $j = 1, 2$ )
- $\tau_{d[i,j]}$  : 順序  $i$ 、時期  $j$  の薬剤効果 ( $i = 1, 2$ ; 薬剤  $d[i,j] = 1, 2$ )
- $\lambda_{d[i,j-1]}$  : 薬剤  $d[i,j-1]$  の持ち越し効果 ( $d[i,0] = 0$  とする)
- $s_{ik}$  : 順序  $i$  の被験者  $k$  の効果 ( $i = 1, 2; k = 1, \dots, n_i$ )、平均 0、分散  $\sigma_s^2$  の確率変数
- $\varepsilon_{ijk}$  : 誤差項 (平均 0、分散  $\sigma^2$ )
- SSE :  $\sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^{n_i} y_{ijk}^2 - \sum_{i=1}^2 \sum_{k=1}^{n_i} y_{i\cdot k}^2 / 2 - \sum_{i=1}^2 \sum_{j=1}^2 y_{ij\cdot}^2 / n_i + \sum_{i=1}^2 y_{i\cdot\cdot}^2 / (2n_i)$
- SSP :  $\sum_{i=1}^2 \sum_{k=1}^{n_i} y_{i\cdot k}^2 / 2 - \sum_{i=1}^2 y_{i\cdot\cdot}^2 / (2n_i)$   
( $y_{ij\cdot} = \sum_{k=1}^{n_i} y_{ijk}$ ,  $y_{i\cdot k} = \sum_{j=1}^2 y_{ijk}$ ,  $y_{i\cdot\cdot} = \sum_{j=1}^2 y_{ij\cdot}$ ,  $\bar{y}_{ij\cdot} = n_i^{-1} \sum_{k=1}^{n_i} y_{ijk}$ )

ここで、 $T = \tau_1 = -\tau_2$ ,  $R = \lambda_1 = -\lambda_2$  なる制約をおくと (すなわち、 $\tau_1 - \tau_2 = 2T$ ,  $\lambda_1 - \lambda_2 = 2R$ )、確率変数  $T$  と  $R$  の推定値はそれぞれ  $\hat{T} = (\bar{y}_{11\cdot} - \bar{y}_{12\cdot} - \bar{y}_{21\cdot} + \bar{y}_{22\cdot})/4$ ,  $\hat{R} = (\bar{y}_{11\cdot} + \bar{y}_{12\cdot} - \bar{y}_{21\cdot} - \bar{y}_{22\cdot})/2$  となる。Bayesian Posterior Probability を算出するためには  $T|y$  の分布に興味をもつことになるが、この計算は結構大変なので、Grieve (1994) の漸近式を用いる。 $T$  と  $R$  の無情報同時事前分布を  $(\sigma^2(2\sigma_s^2 + \sigma^2))^{-1}$  の定数倍とし、 $m = (n_1 + n_2)/n_1 n_2$  とおくと、各事後分布は以下の分布に従う。

- $R|y$  :  $\frac{R - \hat{R}}{\sqrt{mSSP / 2(n_1 + n_2 - 2)}}$  は自由度  $n_1 + n_2 - 2$  の  $t$  分布に従う

- $T|R,y$  :  $\frac{T-\hat{T}-R/2}{\sqrt{mSSE/8(n_1+n_2-2)}}$  は自由度  $n_1 + n_2 - 2$  の  $t$  分布に従う
- $T|y$  :  $\frac{T-\hat{T}-\hat{R}/2}{\sqrt{mb_0/8b_1}}$  は自由度  $b_1$  の  $t$  分布に従う  $\left(b_1 = \frac{(SSE+SSP)^2(n_1+n_2-6)}{SSE^2+SSP^2} + 4, b_0 = \frac{(b_1-2)(SSE+SSP)}{n_1+n_2-4}\right)$

数値例として、各順序の例数は 20、各順序・各時期の応答に関する平均値は上表、SSE と SSP はそれぞれ 250 と 480、「薬剤の平均値の差が 2 以上となる確率」を  $T|R,y$  及び  $T|y$  の分布から算出すると、いずれも Bayesian Posterior Probability は 50%となる。ただし、実際の場面では、 $T|R,y$  と  $T|y$  のどちらの分布を使用するかは判断が難しく、結果も大きく異なる場合があるので注意が必要である。

```
%macro PROB2(_DELTA=2, _N1=20, _N2=20, _Y11=6, _Y12=3, _Y21=2, _Y22=5, _SSE=250, _SSP=480,
             _LOWER="Y");
data PROB2 ;
  M =(&_N1. +&_N2.)/(&_N1.*&_N2.) ;
  R =(&_Y11. +&_Y12. -&_Y21. -&_Y22.)/2 ;
  T =(&_Y11. -&_Y12. -&_Y21. +&_Y22.)/4 ;
  T1=(&_DELTA. -R)/sqrt((M*&_SSP.)/(2*(&_N1. +&_N2. -2))) ;
  T2=(&_DELTA. -T-R/2)/sqrt((M*&_SSE.)/(8*(&_N1. +&_N2. -2))) ;
  B1=(&_N1. +&_N2. -6)*(&_SSE. +&_SSP.)**2/(&_SSE.**2+&_SSP.**2)+4 ;
  B0=(B1-2)*(&_SSE. +&_SSP.)/(&_N1. +&_N2. -4) ;
  T3=(&_DELTA. -T-R/2)/sqrt((M*B0)/(8*B1)) ;

  if (&_LOWER="Y") then do ;
    PROB1=cdf("t", T1, &_N1. +&_N2. -2) ;
    PROB2=cdf("t", T2, &_N1. +&_N2. -2) ;
    PROB3=cdf("t", T3, B1) ;
  end ;
  else do ;
    PROB1=1-cdf("t", T1, &_N1. +&_N2. -2) ;
    PROB2=1-cdf("t", T2, &_N1. +&_N2. -2) ;
    PROB3=1-cdf("t", T3, B1) ;
  end ;
run ;
proc print noobs ; run ;
%mend ;
%PROB2(_LOWER="N"); * 引数 _LOWER="Y" にて分布の下側確率を算出する ;
```

M	R	T	T1	T2	B1	B0	T3	PROB1	PROB2	PROB3
0.1	1	1.5	1.25831	0	65.8593	1294.93	0	0.10798	<u>0.5</u>	<u>0.5</u>

### 3 Markov Chain Monte Carlo による Bayesian Posterior Probability の算出例

前項では、並行群間デザインと 2x2 クロスオーバーデザインにおいて Bayesian Posterior Probability を算出する例を挙げたが、並行群間デザインの場合は共変量の追加や経時データへの拡張、クロスオーバーデザインの場合は高次のデザインへの拡張を行う等、デザインをより複雑にすると解析的/漸近的に事後分布を求めることが困難となる。また、2x2 クロスオーバーデザインにて Grieve (1994) の漸近式を用いた場合、 $T|R,y$  と  $T|y$  の分布の選択が難しい。そこで、先程と同様の設定で、今度は mcmc プロシジャによる Bayesian Posterior Probability の算出を試みる。なお、2x2 クロスオーバーデザインにおける mcmc プロシジャのモデルには持ち越し効果の代わりに順序効果を指定した。

## 並行群間デザイン

まず、各薬剤の例数は20、応答に関する各薬剤の平均値はそれぞれ3と0、標準偏差はそれぞれ4と5と想定し、乱数にてデータを生成した後、glm プロシジャにて頻度論の枠組みで解析を行い、各薬剤の平均値とその95%信頼区間及び平均値の差の点推定値とその95%信頼区間を算出する。

```
data MYDATA ;
  do GROUP=1 to 2 ;
    do ID=1 to 20 ;
      Y=3*(2-GROUP)+(3+GROUP)*normal(7777) ; output ;
    end ;
  end ;
run ;

proc glm data=MYDATA ;
  class GROUP ;
  model Y = GROUP / solution ss3 ;
  lsmeans GROUP / cl pdiff ;
run ;
```

GLM プロシジャ			
Y の最小 2			
GROUP	乗平均	95% 信頼限界	
1	2.725909	0.677707	4.774111
2	-0.219506	-2.267708	1.828696
効果 GROUP に対する最小 2 乗平均			
		LSMean(i)-LSMean(j) の	
i	j	平均の差	95% 信頼限界
1	2	2.945414	0.048819 5.842010

次に、mcmc プロシジャにより解析を行った後、「平均値の差が2以上となる確率」を算出すると、**Bayesian Posterior Probability** は75.95%となる。各薬剤の事後平均とその95%確信区間及び平均値の差に関する事後平均とその95%確信区間は頻度論の結果と同様であるが、Autocorrelation Time や Efficiency の結果を見ると、サンプリングの効率が良くなく、自己相関が残っていることが伺える。

```
proc mcmc data=MYDATA outpost=OUT seed=777 nmc=10000 monitor=(_parms_ DIFF)
  statistics(alpha=0.05) ;
  parm mu1 0 mu2 0 ;
  parm sig2 1 ;
  prior mu: ~ general(0) ;
  prior sig2 ~ general(-log(sig2), lower=0) ;
  diff = mu1 - mu2 ;
  if (GROUP=1) then mu = mu1 ;
  else mu = mu2 ;
  model Y ~ normal(mu, var=sig2) ;
run ;

proc format ;
  value diff_f low < 2 = "Diff < 2"
                2 <= high = "Diff >= 2" ;
run ;

proc freq data=OUT ;
  tables diff / nocum ;
  format diff diff_f. ;
run ;
```

The MCMC Procedure						FREQ プロシジャ		
Posterior Summaries and Intervals						diff 度数 パーセント		
Parameter	N	Mean	Standard Deviation	95% HPD Interval				
mu1	10000	2.7678	0.9947	0.8838	4.7515	Diff < 2	2405	24.05
mu2	10000	-0.2749	1.0316	-2.3385	1.7203	Diff >= 2	7595	<u>75.95</u>
sig2	10000	21.6602	5.1779	12.6395	31.5817			
diff	10000	3.0428	1.4397	0.4024	6.0471			

Effective Sample Sizes			
Autocorrelation			
Parameter	ESS	Time	Efficiency
mu1	1361.9	<u>7.3427</u>	<u>0.1362</u>
mu2	1529.0	<u>6.5402</u>	<u>0.1529</u>
sig2	1323.4	<u>7.5563</u>	<u>0.1323</u>
diff	1417.9	<u>7.0528</u>	<u>0.1418</u>

そこで、mcmc プロシジャにおける Burn-in、thinning 及びサンプリングの数を 5 倍 (nbi=5000、thin=5、nmc=50000) にして再実行する。結果として効率はある程度上がるが、計算時間が増えてしまう。

```
proc mcmc data=MYDATA outpost=OUT seed=777 nbi=5000 thin=5 nmc=50000 monitor=(_parms_ DIFF)
  statistics(alpha=0.05) ;
```

(以降は前プログラムと同一の命令)

Effective Sample Sizes			
Autocorrelation			
Parameter	ESS	Time	Efficiency
mu1	5572.2	<u>1.7946</u>	<u>0.5572</u>
mu2	6005.7	<u>1.6651</u>	<u>0.6006</u>
sig2	5909.9	<u>1.6921</u>	<u>0.5910</u>
diff	6085.1	<u>1.6434</u>	<u>0.6085</u>

参考までに、上記 2 つのプログラムの診断プロットを次頁に示す。

**【参考】**前掲のプログラムでは、薬剤を表すカテゴリ変数 GROUP の値によって条件分岐を行うことで、model ステートメントで使用するパラメータ mu の値を指定していた。切片項を表すパラメータと random ステートメントを併用することで、カテゴリ変数 GROUP の値による条件分岐を行う必要がなくなりコーディングが簡便となる (ただし、サンプリングの効率が落ちる)。次項の 2 × 2 クロスオーバーデザインにてこのコーディング方針を適用する。

```
proc mcmc data=MYDATA outpost=OUT seed=777 nbi=5000 thin=5 nmc=50000 statistics(alpha=0.05) ;
  parm int 0 ;
  parm sig2 1 ;
  prior int ~ general(0) ;
  prior sig2 ~ general(-log(sig2), lower=0) ;
  random mu ~ general(0) subject=GROUP monitor=(mu) init=0 zero=last ;

  model Y ~ normal(int+mu, var=sig2) ;
run ;
```

Effective Sample Sizes			
Autocorrelation			
Parameter	ESS	Time	Efficiency
int	2988.7	<u>3.3459</u>	<u>0.2989</u>
sig2	4802.6	<u>2.0822</u>	<u>0.4803</u>
mu_1	2806.7	<u>3.5629</u>	<u>0.2807</u>

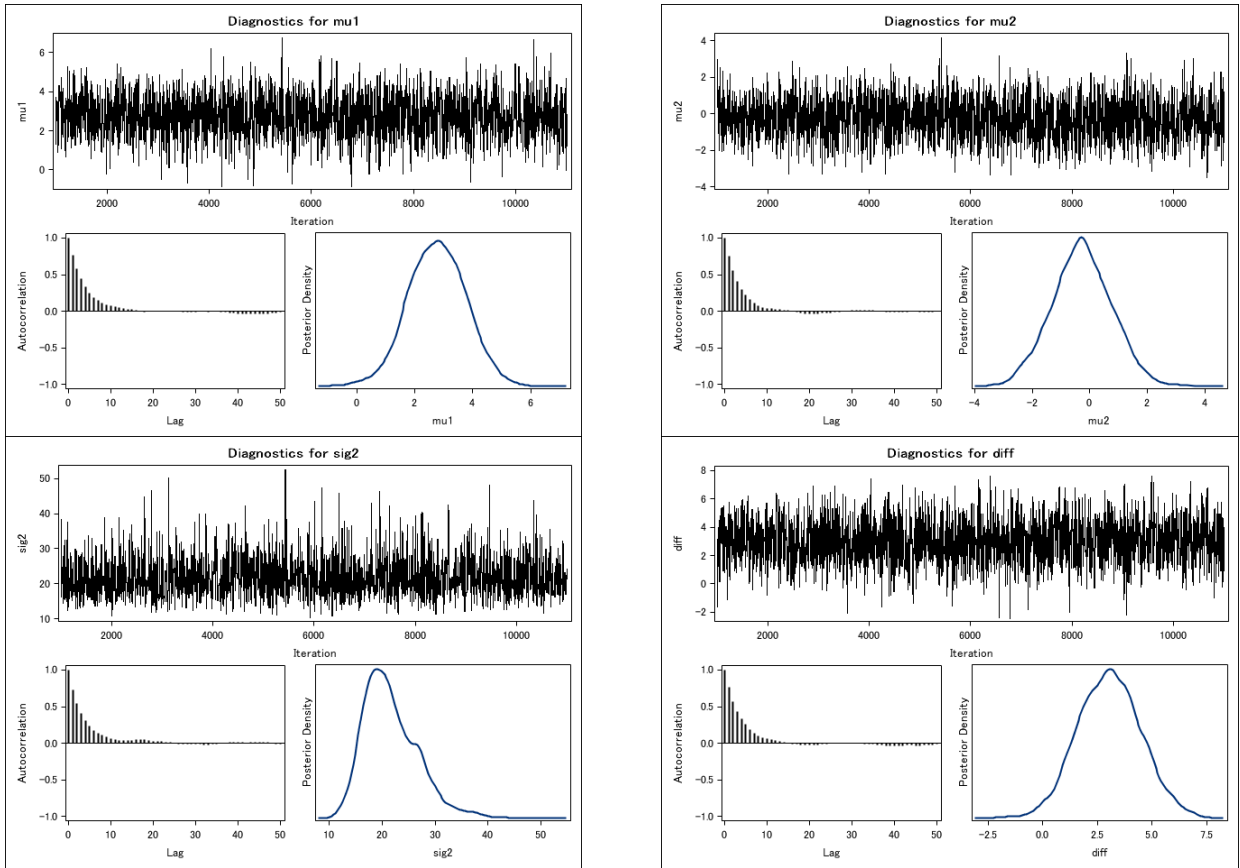


図 3-1 mcmc プロシジャの実行結果 ( $n_{\text{bi}}=1000$ 、 $\text{thin}=1$ 、 $n_{\text{mc}}=10000$ )

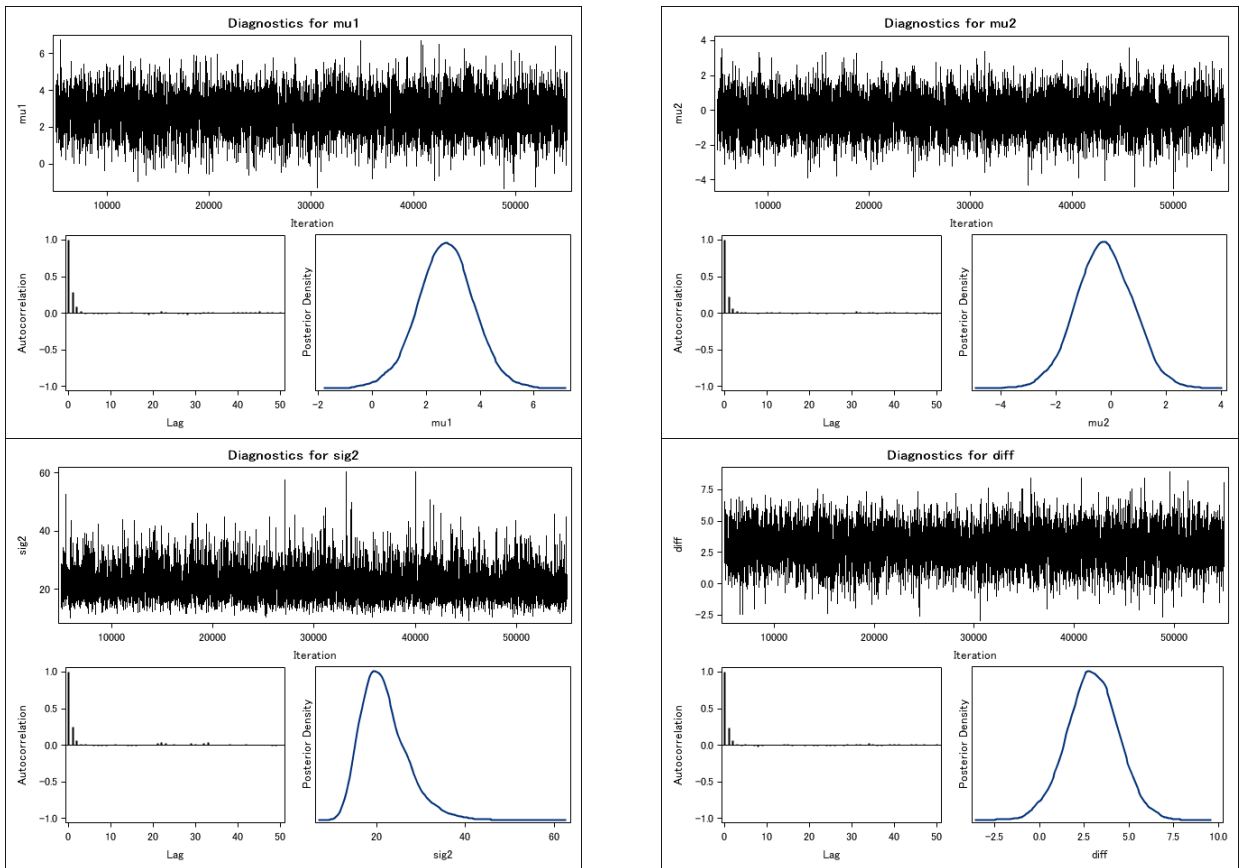


図 3-2 mcmc プロシジャの実行結果 ( $n_{\text{bi}}=5000$ 、 $\text{thin}=5$ 、 $n_{\text{mc}}=50000$ )

## 2×2 クロスオーバーデザイン

まず、各順序の例数及び各セルの応答に関する平均値は表 2-1 と同様、各セルの応答に関する標準偏差は全て 3 と想定し、乱数にてデータを生成した後、glm プロシジャにて頻度論の枠組みで解析を行い、薬剤効果の値の差の点推定値とその 95%信頼区間を算出する。

```
data MYDATA ;
  GROUP=1 ;
  do ID=1 to 20 ;
    PERIOD=1; TREAT=1; Y=6+3*normal(7777) ; output ;
    PERIOD=2; TREAT=2; Y=3+3*normal(7777) ; output ;
  end ;
  GROUP=2 ;
  do ID=21 to 40 ;
    PERIOD=1; TREAT=2; Y=2+3*normal(7777) ; output ;
    PERIOD=2; TREAT=1; Y=5+3*normal(7777) ; output ;
  end ;
run ;

proc glm data=MYDATA ;
  class ID GROUP PERIOD TREAT ;
  model Y = GROUP ID(GROUP) PERIOD TREAT / solution ss3 ;
  test h=GROUP e=ID(GROUP) ;
  lsmeans TREAT / cl pdiff ;
run ;
```

GLM プロシジャ				
効果 TREAT に対する最小 2 乗平均				
		LSMean(i)-LSMean(j) の		
i	j	平均の差	95% 信頼限界	
1	2	2.834587	1.657083	4.012091

次に、mcmc プロシジャにより解析を行った後、「薬剤効果の差が 2 以上となる確率」を算出すると、Bayesian Posterior Probability は 91.16%となる。薬剤効果の値の差の点推定値とその 95%確信区間は頻度論の結果と同様である。しかし、mcmc プロシジャにおける Burn-in、thinning 及びサンプリングの数をそれぞれ 5000、5、50000 にして実行したにも関わらず、Autocorrelation Time や Efficiency の結果を見ると、サンプリングの効率が良くなく、自己相関が残っていることが伺える。この原因としては、モデルに複数の random ステートメントが含まれていることが挙げられる。

```
proc mcmc data=MYDATA outpost=OUT seed=777 nbi=5000 thin=5 nmc=50000 statistics(alpha=0.05) ;
  parms residual 1 sig2 1 ;
  prior residual ~ igamma(0.001, scale=0.001) ;
  prior sig2 ~ igamma(0.001, scale=0.001) ;
  random seq ~ normal(0, var=1000) subject=GROUP monitor=(seq) zero=last ;
  random tau ~ normal(0, var=1000) subject=TREAT monitor=(tau) zero=last ;
  random pi ~ normal(0, var=1000) subject=PERIOD monitor=(pi) ;
  random s ~ normal(0, var=sig2) subject=ID ;

  model y ~ normal(seq + tau + pi + s, var=residual) ;
run ;

proc freq data=OUT ;
  tables tau_1 / nocum ;
  format tau_1 diff_f. ;
run ;
```



The MCMC Procedure						FREQ プロシジャ		
Posterior Summaries and Intervals						diff 度数 パーセント		
Parameter	N	Mean	Standard Deviation	95% HPD Interval				
residual	10000	8.0493	1.9739	4.5591	11.9088	Diff < 2	884	8.84
sig2	10000	2.0339	1.7767	0.00362	5.3782	Diff >= 2	9116	<u>91.16</u>
seq_1	10000	0.5742	0.7792	-0.9330	2.0854			
tau_1	10000	2.8447	0.6341	1.6619	4.1611			
pi_1	10000	2.0109	0.7208	0.5419	3.3974			
pi_2	10000	2.6627	0.7194	1.1777	3.9717			
Effective Sample Sizes								
Parameter	ESS	Autocorrelation		Time	Efficiency			
residual	245.0			<u>40.8182</u>	<u>0.0245</u>			
sig2	153.8			<u>65.0369</u>	<u>0.0154</u>			
seq_1	1470.4			<u>6.8010</u>	<u>0.1470</u>			
tau_1	2486.9			<u>4.0210</u>	<u>0.2487</u>			
pi_1	1430.4			<u>6.9912</u>	<u>0.1430</u>			
pi_2	1482.3			<u>6.7462</u>	<u>0.1482</u>			

#### 4 No-U-Turn Sampler による Bayesian Posterior Probability の算出例

Hamiltonian Monte Carlo Method (HMC) は、解析力学等で用いられる Hamiltonian の概念を利用し、Leap-frog Method と呼ばれる手法で乱数を生成する。アルゴリズムの詳細は豊田 (2015) が詳しいため本稿では詳述を避けるが、HMC は従来の手法と比較して収束が速く、乱数間の相関が低いため、通常の MCMC (例えば `mcmc` プロシジャでデフォルトとして用いられる `random-walk Metropolis`) よりもサンプリング数が数十分の一で済む場合があるとされている。また、No-U-Turn Sampler (NUTS) は HMC の変法で、HMC の計算で必要となる各パラメータを自動で調整しつつサンプリングを行うため、より実践的な手法である。NUTS は `mcmc` プロシジャの他、Stan という確率的プログラミング言語で実装されている。Stan の実装例としては例えば R のパッケージのひとつである `RStan` があるが、`RStan` では解析を実行する度に、R や Stan のコードを C++ に変換してコンパイルを行った後、C++ 上でサンプリングを行うため、簡単なモデルでも実行時間がかかる。一方、`mcmc` プロシジャではアルゴリズムを `random-walk Metropolis` から NUTS に変更しても実行環境や動作は大きく変わらないため、実行時間はそれほど増えず、動作が軽い印象がある。本項では、先程紹介した `mcmc` プロシジャを NUTS で再実行し、効率等の観点から通常の MCMC (`random-walk Metropolis`) と比較を試みる。

以下、前項の `mcmc` プロシジャのプログラムについて、アルゴリズムを NUTS に変更して再実行した結果を紹介するが、アルゴリズムの変更方法は `mcmc` プロシジャの 1 行目に「`alg=nuts`」を追加するのみである。

```
proc mcmc data=MYDATA outpost=OUT seed=777 ... statistics(alpha=0.05) alg=nuts ;
```

#### 並行群間デザイン

前項の `mcmc` プロシジャのプログラムを再実行するが、その際、Burn-in 及び thinning の数をそれぞれ 5000 と 5、アルゴリズムを「デフォルト (`random-walk Metropolis`)」と「NUTS」の両方、NUTS の場合はサンプリング数を「50000」「5000」「500」の 3 通り、計 4 通りの実行結果を以下に示す。まず、事後分布に関する要約統計量はいずれも同様の結果となった。

パラメータ	Posterior Summaries and Intervals					
alg= <u>normal</u> (デフォルト) nbi=5000 thin=5 nmc= <u>50000</u>	Parameter	N	Mean	Standard Deviation	95% HPD Interval	
	mu1	10000	2.7263	1.0324	0.7039	4.8101
	mu2	10000	-0.2373	1.0241	-2.3522	1.6881
	sig2	10000	21.7136	5.2970	12.9774	32.6929
	diff	10000	2.9636	1.4399	0.0918	5.7872
alg= <u>nuts</u> nbi=5000 thin=5 nmc= <u>50000</u>	Parameter	N	Mean	Standard Deviation	95% HPD Interval	
	mu1	10000	2.7142	1.0440	0.7138	4.8019
	mu2	10000	-0.2074	1.0590	-2.2917	1.7970
	sig2	10000	21.6331	5.3229	12.2837	31.9301
	diff	10000	2.9215	1.4892	-0.0224	5.8308
alg= <u>nuts</u> nbi=5000 thin=5 nmc= <u>5000</u>	Parameter	N	Mean	Standard Deviation	95% HPD Interval	
	mu1	1000	2.7291	1.0202	0.9364	5.0202
	mu2	1000	-0.2457	1.0474	-2.3428	1.7765
	sig2	1000	21.7164	5.4621	12.4431	32.1279
	diff	1000	2.9748	1.4814	0.1426	6.0151
alg= <u>nuts</u> nbi=5000 thin=5 nmc= <u>500</u>	Parameter	N	Mean	Standard Deviation	95% HPD Interval	
	mu1	100	2.6088	1.0192	0.4052	4.4017
	mu2	100	-0.2433	0.9066	-1.7168	1.4576
	sig2	100	22.3600	5.5752	13.8750	34.6725
	diff	100	2.8521	1.3400	-0.1079	5.2955

次に、効率（Effective Sample Size）に関する結果は、アルゴリズムを NUTS に変更するだけでサンプリングの効率や自己相関が改善した。また、前表の結果も踏まえると、アルゴリズムを NUTS にした場合はサンプリング数を 1/10 程度に減らしても推定精度がある程度維持されていることが分かる。

パラメータ	効率 (Effective Sample Size)			
alg= <u>normal</u> (デフォルト) nbi=5000 thin=5 nmc= <u>50000</u>	Parameter	ESS	Autocorrelation	
			Time	Efficiency
	mu1	5572.2	1.7946	0.5572
	mu2	6005.7	1.6651	0.6006
	sig2	5909.9	1.6921	0.5910
alg= <u>nuts</u> nbi=5000 thin=5 nmc= <u>50000</u>	Parameter	ESS	Autocorrelation	
			Time	Efficiency
	mu1	9641.5	1.0372	0.9642
	mu2	10000.0	1.0000	1.0000
	sig2	5277.5	1.8948	0.5278
alg= <u>nuts</u> nbi=5000 thin=5 nmc= <u>5000</u>	Parameter	ESS	Autocorrelation	
			Time	Efficiency
	mu1	936.5	1.0678	0.9365
	mu2	1066.6	0.9375	1.0666
	sig2	379.9	2.6320	0.3799
alg= <u>nuts</u> nbi=5000 thin=5 nmc= <u>500</u>	Parameter	ESS	Autocorrelation	
			Time	Efficiency
	mu1	91.9	1.0881	0.9191
	mu2	78.0	1.2818	0.7801
	sig2	33.5	2.9817	0.3354
	diff	107.0	0.9349	1.0696

参考までに、Geweke's Convergence Diagnosticの結果と、2~3番目のプログラムの診断プロットを以下に示す（なお、1番目のプログラムの診断プロットは図3-2に掲載済み）。

パラメータ	Geweke's Convergence Diagnostic		
alg= <u>normal</u> (デフォルト) nbi=5000 thin=5 nmc= <u>50000</u>	Parameter	z	Pr >  z
	mu1	0.7724	0.4399
	mu2	-1.5570	0.1195
	sig2	1.7239	0.0847
	diff	1.7669	0.0772
alg= <u>nuts</u> nbi=5000 thin=5 nmc= <u>50000</u>	Parameter	z	Pr >  z
	mu1	0.6110	0.5412
	mu2	-0.7086	0.4786
	sig2	0.2948	0.7682
	diff	0.9859	0.3242
alg= <u>nuts</u> nbi=5000 thin=5 nmc= <u>5000</u>	Parameter	z	Pr >  z
	mu1	-1.0353	0.3005
	mu2	-0.2225	0.8239
	sig2	0.8094	0.4183
	diff	-0.5526	0.5805
alg= <u>nuts</u> nbi=5000 thin=5 nmc= <u>500</u>	Parameter	z	Pr >  z
	mu1	0.4566	0.6479
	mu2	1.2622	0.2069
	sig2	-0.7638	0.4450
	diff	-0.0709	0.9435

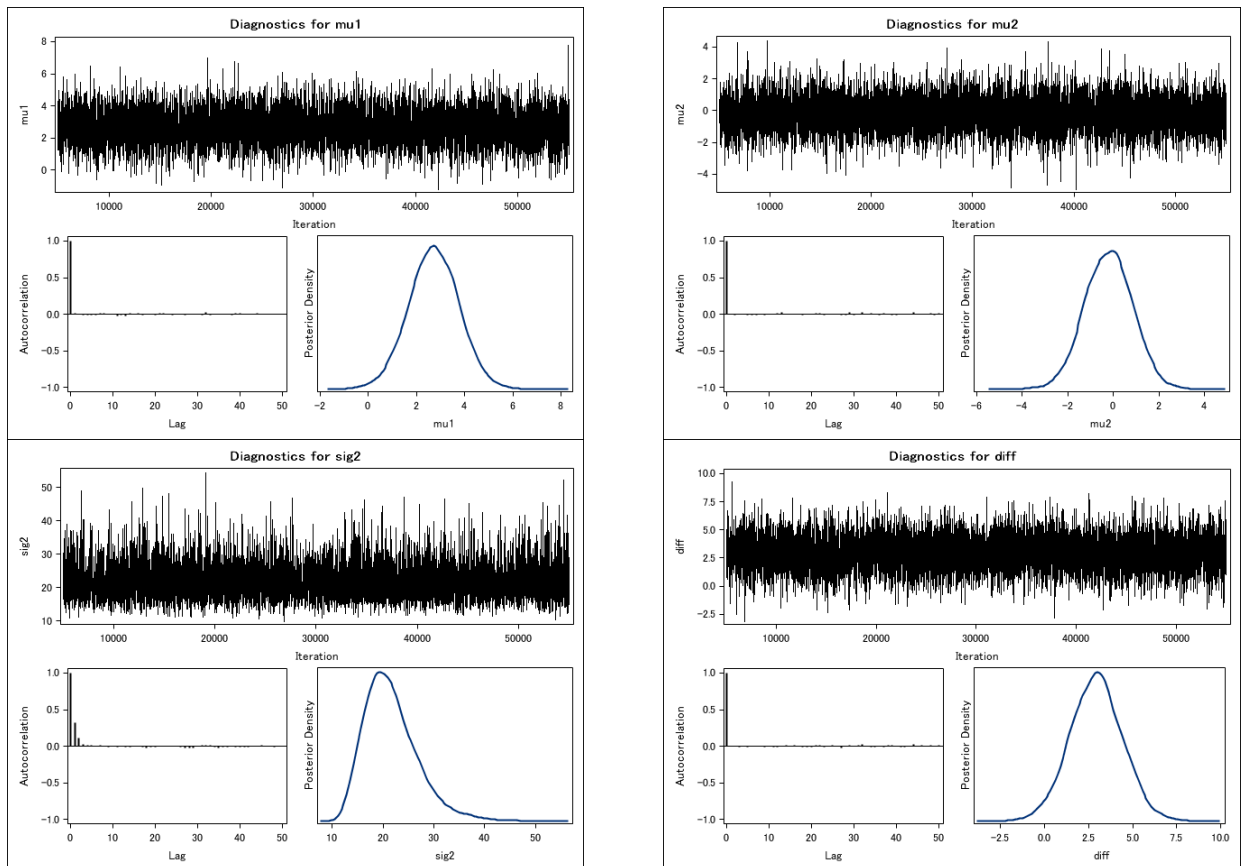


図 4-1 mcmc プロシジャの実行結果 (nbi=5000、thin=5、nmc=50000、alg=nuts)

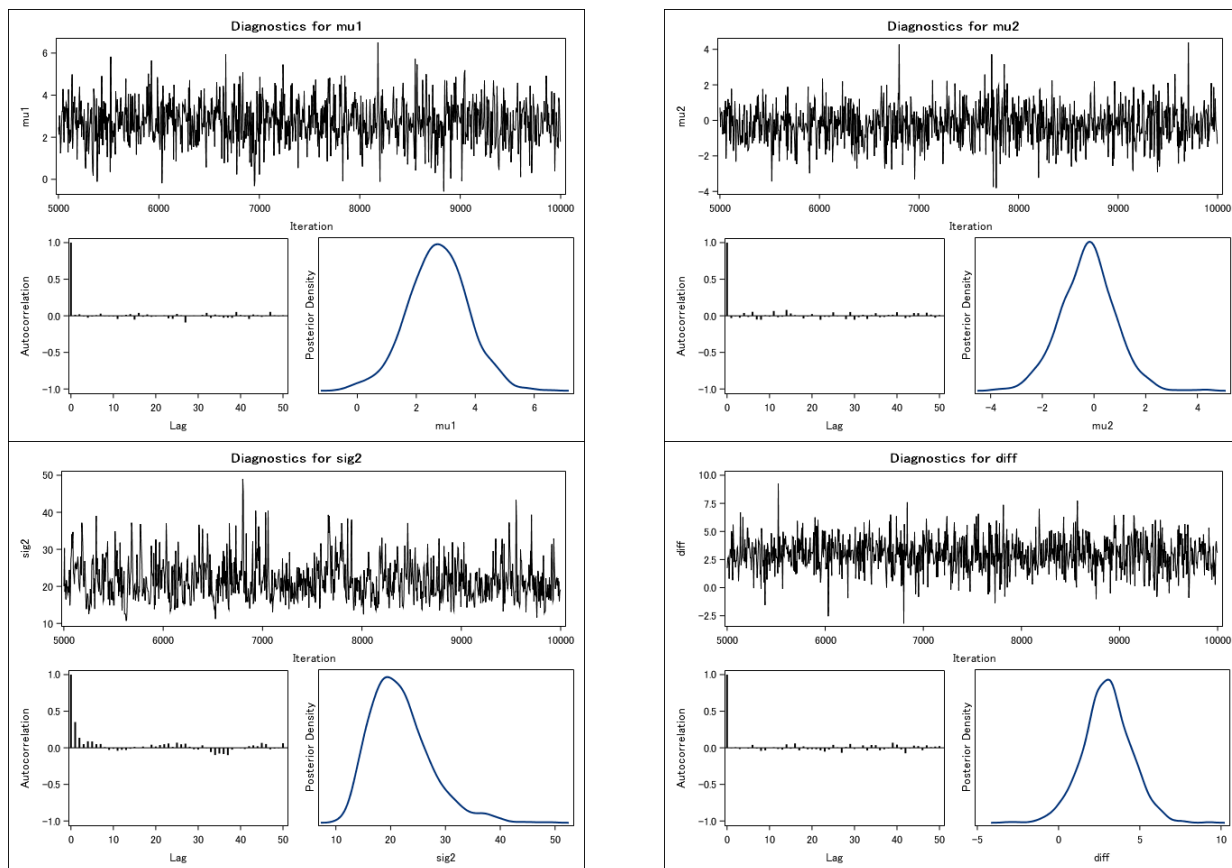


図 4-2 mcmc プロシジャの実行結果 (nbi=5000、thin=5、nmc=5000、alg=nuts)

## 2×2 クロスオーバーデザイン

前項の mcmc プロシジャのプログラムを再実行するが、その際、Burn-in 及び thinning の数をそれぞれ 10000 と 10、アルゴリズムを「デフォルト (random-walk Metropolis)」と「NUTS」の両方、NUTS の場合はサンプリング数を「100000」「10000」の 2 通り、計 3 通りの実行結果を以下に示す。まず、事後分布に関する要約統計量はいずれも同様の結果となった。

パラメータ	Posterior Summaries and Intervals				
alg= <u>normal</u> (デフォルト) nbi=10000 thin=10 nmc= <u>100000</u>	Parameter	N	Mean	Standard Deviation	95% HPD Interval
	residual	10000	8.0683	1.9779	4.5044 11.9148
	sig2	10000	2.0432	1.8148	0.00182 5.4304
	seq_1	10000	0.5564	0.7843	-0.9888 2.0990
	tau_1	10000	2.8352	0.6388	1.6269 4.1330
	pi_1	10000	2.0284	0.7149	0.5868 3.4007
	pi_2	10000	2.6823	0.7229	1.2121 4.0507
	alg= <u>nuts</u> nbi=10000 thin=10 nmc= <u>100000</u>	Parameter	N	Mean	Standard Deviation
residual		10000	7.9362	1.9161	4.4868 11.6854
sig2		10000	2.1671	1.7581	0.0305 5.4274
seq_1		10000	0.5342	0.7850	-0.9584 2.1152
tau_1		10000	2.8416	0.6290	1.5967 4.0575
pi_1		10000	2.0477	0.7023	0.6432 3.3944
pi_2		10000	2.7046	0.7082	1.3497 4.1421

パラメータ	Posterior Summaries and Intervals					
alg= <u>nuts</u> nbi=10000 thin=10 nmc= <u>10000</u>	Parameter	N	Mean	Standard Deviation	95% HPD Interval	
	residual	1000	7.7681	1.8430	4.4485	11.1698
	sig2	1000	2.3257	1.7355	0.0405	5.4617
	seq_1	1000	0.5792	0.8020	-0.8966	2.2719
	tau_1	1000	2.8499	0.6397	1.5788	4.0941
	pi_1	1000	2.0556	0.7144	0.5593	3.3252
	pi_2	1000	2.7097	0.7136	1.3508	4.1274

次に、効率 (Effective Sample Size) に関する結果は、アルゴリズムを NUTS に変更するだけでサンプリングの効率や自己相関が改善したが、並行群間デザインの場合と比較すると、改善の度合いは小さく、特にばらつきに関するパラメータについては満足出来る結果とはいえない。原因としては、やはりモデルに複数の random ステートメントが含まれていることが挙げられる。この状況下では、アルゴリズムを NUTS にしたとしてもサンプリング数の減少まで期待することは出来ないようである。

パラメータ	効率 (Effective Sample Size)			
alg= <u>normal</u> (デフォルト) nbi=10000 thin=10 nmc= <u>100000</u>	Autocorrelation			
	Parameter	ESS	Time	Efficiency
	residual	395.1	25.3128	0.0395
	sig2	271.7	36.8032	0.0272
	seq_1	2772.2	3.6073	0.2772
	tau_1	4808.7	2.0796	0.4809
	pi_1	2693.1	3.7132	0.2693
pi_2	2844.0	3.5162	0.2844	
alg= <u>nuts</u> nbi=10000 thin=10 nmc= <u>100000</u>	Autocorrelation			
	Parameter	ESS	Time	Efficiency
	residual	1169.3	8.5525	0.1169
	sig2	700.3	14.2789	0.0700
	seq_1	9401.5	1.0637	0.9402
	tau_1	8551.2	1.1694	0.8551
	pi_1	9519.5	1.0505	0.9520
pi_2	9452.7	1.0579	0.9453	
alg= <u>nuts</u> nbi=10000 thin=10 nmc= <u>10000</u>	Autocorrelation			
	Parameter	ESS	Time	Efficiency
	residual	160.1	6.2452	0.1601
	sig2	82.4	12.1287	0.0824
	seq_1	859.6	1.1634	0.8596
	tau_1	704.1	1.4203	0.7041
	pi_1	577.8	1.7307	0.5778
pi_2	659.0	1.5175	0.6590	

参考までに、全ての Geweke's Convergence Diagnostic の結果と、1~2 番目のプログラムの診断プロットを以下に示す。

パラメータ	Geweke's Convergence Diagnostic		
alg= <u>normal</u> (デフォルト) nbi=10000 thin=10 nmc= <u>100000</u>	Parameter	z	Pr >  z
	residual	-0.6721	0.5015
	sig2	-0.0806	0.9358
	seq_1	1.5806	0.1140
	tau_1	-0.1310	0.8957
	pi_1	-1.4010	0.1612
	pi_2	-1.2311	0.2183

パラメータ	Geweke's Convergence Diagnostic		
alg= <u>nuts</u> nbi=10000 thin=10 nmc= <u>100000</u>	Parameter	z	Pr >  z
	residual	-0.7958	0.4261
	sig2	0.5213	0.6021
	seq_1	1.3532	0.1760
	tau_1	0.2820	0.7779
	pi_1	0.6765	0.4987
	pi_2	0.0095	0.9924
alg= <u>nuts</u> nbi=10000 thin=10 nmc= <u>10000</u>	Parameter	z	Pr >  z
	residual	-0.3277	0.7431
	sig2	0.9485	0.3429
	seq_1	0.1327	0.8944
	tau_1	-0.4700	0.6384
	pi_1	0.6774	0.4981
	pi_2	0.1971	0.8437

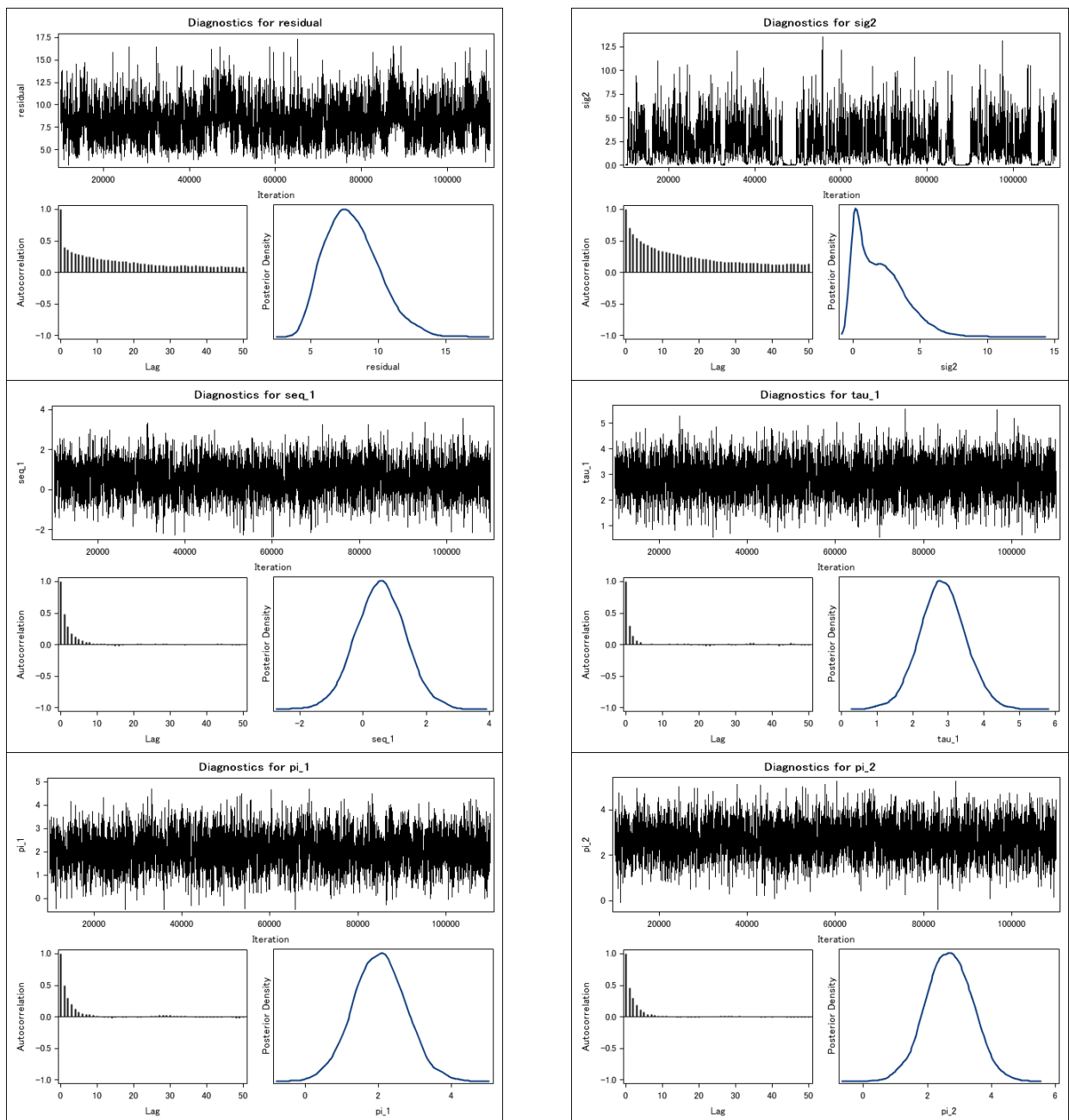


図 4-3 mcmc プロシジャの実行結果 (nbi=10000、thin=10、nmc=100000、alg=normal)

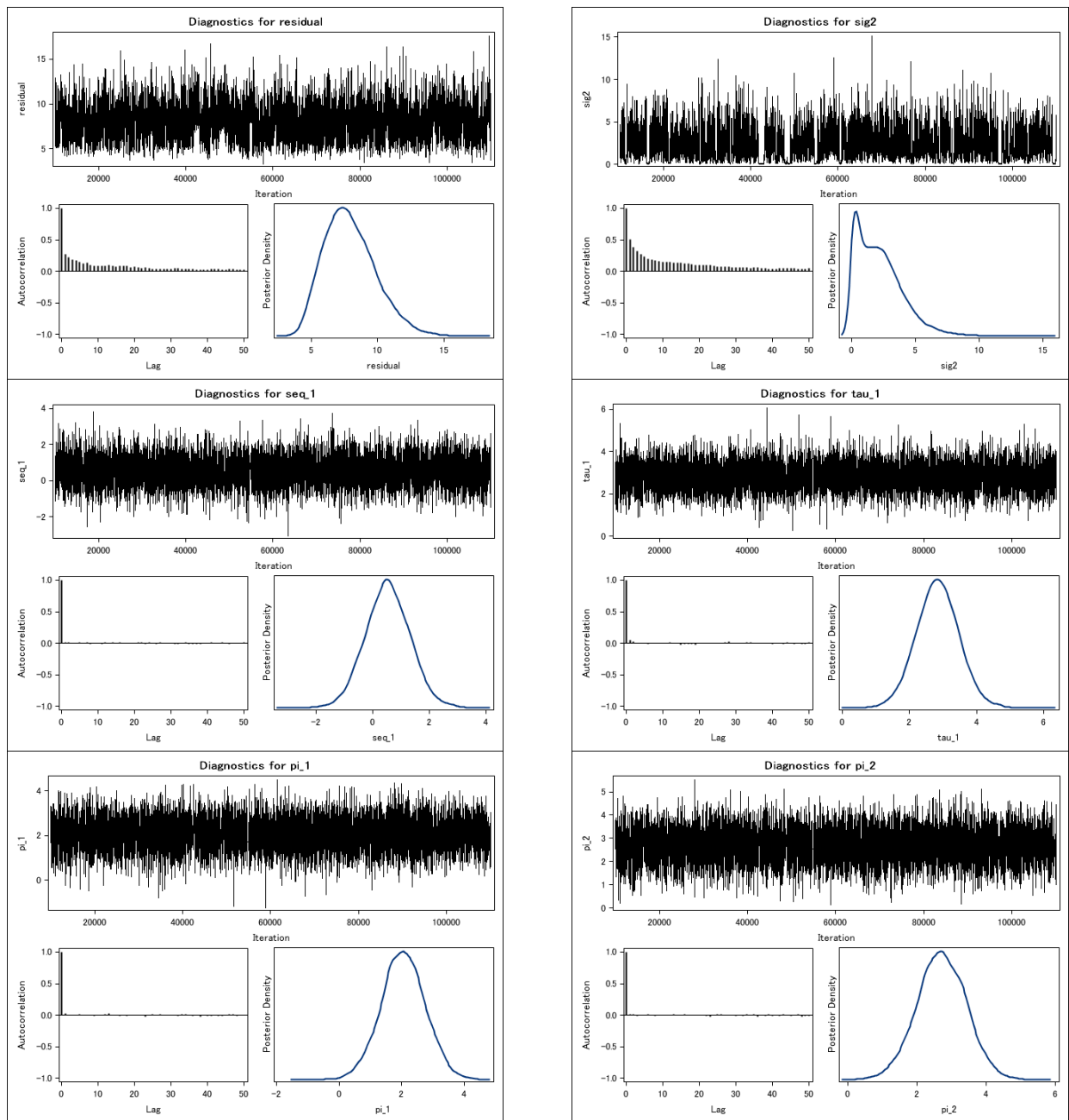


図 4-4 mcmc プロシジャの実行結果 (nbi=10000、thin=10、nmc=100000、alg=nuts)

## 5 まとめ

臨床試験終了後に Bayesian Posterior Probability を算出することで、後期臨床試験への移行に関する意思決定 (Go/No-Go Decision) のための材料が得られることを示した。早期臨床試験においては並行群間デザインに加え、クロスオーバーデザインもしばしば適用されるが、本稿ではいずれの場合においても本確率を計算することが出来る。また、mcmc プロシジャを適用して Markov Chain Monte Carlo (MCMC) による解析が可能となることも紹介したが、解析対象となるモデルや仮定が変わっても、mcmc プロシジャであれば柔軟に対応できることも申し添えておく。ただし、解析するモデルによっては乱数列の収束やサンプリング効率が良くない場合があり、特に random ステートメントを含むモデルを定義した場合は、乱数列の収束やサンプリング効率が一般的に落ちる。このような場合は NUTS によるサンプリングにより乱数列の収束状況や効率をある程度改善出来る場合があることを示した。

## 参考文献

- FDA (2010) "Guidance for the use of bayesian statistics in medical device clinical trials."  
<https://www.fda.gov/MedicalDevices/ucm071072.htm>
- Grieve AP (1994) "Bayesian analyses of two-treatment crossover studies," *Statistical Methods in Medical Research*; 3, 407-429.
- Jones B, Kenward MG (2014) "Design and Analysis of Cross-Over Trials, Third Edition," Chapman & Hall/CRC.
- SAS Institute Inc. (2015) "SAS/STAT(R) 14.1 User's Guide."
- Walley RJ. et. al. (2015) "Advantages of a wholly Bayesian approach to assessing efficacy in early drug development: a case study," *Pharmaceutical Statistics*, 14: 205-215.
- 繁樹 算男 (1985) "ベイズ統計入門" 東京大学出版会
- 豊田 秀樹 (2015) "基礎からのベイズ統計学" 朝倉書店
- 松浦 健太郎 他 (2016) "Stan と R でベイズ統計モデリング" 共立出版

以上