

# SAS-WinBUGSを用いたベイズモデル分析

○畑山 知慶 本多 隆之 矢田 真城 林 行和

株式会社ACRONET 開発本部 データサイエンス第1部 統計解析・DMSグループ

An Application of MCMC method using SAS  
and WinBUGS for Bayesian data analysis

Tomoyoshi Hatayama Takayuki Honda Shinjou Yada Yukikazu Hayashi  
Data Sciences Dept. I , Clinical Development, ACRONET Corporation

## 発表構成

- はじめに(背景)
- SAS-WinBUGSによるベイズ分析
  - データ解析
  - モンテカルロシミュレーション
- おわりに

## はじめに(背景)

### 近年の医薬統計

Adaptive design  
Dose response  
中間解析



ベイズ統計学の  
使用機会の増加

背景その1

ベイズ統計の市民権向上  
試験の柔軟性の要求

背景その2

技術的にも適用が容易に

## 技術的背景 事後分布導出の困難

ベイズ統計学では推測の興味は事後分布にあるが

### ベイズの定理

$$\pi(\theta | D) = \frac{\pi(D | \theta)\pi(\theta)}{\int \pi(D | \theta)\pi(\theta)d\theta}$$

$\theta$ : パラメータ  
 $D$ : データ

パラメータ  
が高次元



解析的には導出不能  
通常の数値積分でも導出不能

計算の諸問題により、実問題への適用が難しかった

## 技術的背景 MCMC法による事後分布の導出

解析的には困難な事後分布の導出

$$\pi(\theta | D) = \frac{\pi(D | \theta)\pi(\theta)}{\int \pi(D | \theta)\pi(\theta)d\theta}$$



**MCMC法**を用いて計算

.....  
実問題へのベイズの適用が容易に！

## 技術的背景 MCMC法による事後分布の導出

### MCMC法

数値積分ではなく、マルコフ連鎖サンプリング  
を応用した**モンテカルロ法**

1950年代 核物理の分野で誕生

1980年代 ベイズの分野での応用

2000年代 WinBUGSなどのソフトで身近に



現在、MCMCが身近になり、よりベイズも適用が容易に

## MCMC法を行うには

### ➤ フリーソフト **WinBUGS**

- MCMCによるベイズ分析を行うソフト
- 最も広く用いられ、使用実績に優れる
- 臨床試験の評価にも用いられた Smith(2006)

### ➤ フリーソフト R

- MCMCパッケージが利用可能
- WinBUGSを制御し、MCMCを行わせるpackage (R2WinBUGS) も配布されており、広く用いられている

### ➤ SAS V9.2評価版のMCMCプロシジャ

- MH法も使用可能
- V9.1.3時点では有力なものはない

## WinBUGS 特徴



- ギブス・サンプラーによるMCMCを行うソフト
- 使用実績は随一
- プログラムは不要、統計モデルを記述するのみ



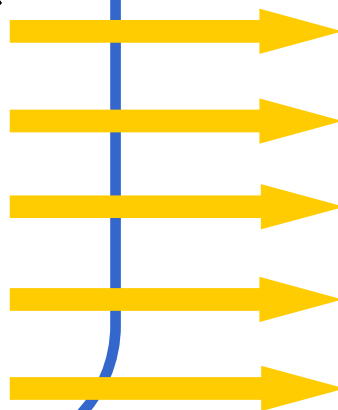
- データハンドリングが出来ない
- IF(条件分岐)、Loop(反復)の仕組み自体が無い
- ボタン・タブで操作を行うため、再現性が無い



# SASからWinBUGSを制御し、部品の様に扱う

SASからWinBUGSを制御し、MCMC計算を簡単に行うプログラムを開発した

- モデル化
- MCMCサンプリング
- データハンドリング
- 分岐(IF)
- 反復(loop)
- グラフ化
- 要約統計



- モデル化
- MCMCサンプリング**
- データハンドリング
- 分岐(IF)
- 反復(loop)
- グラフ化
- 要約統計

# WinBUGSをSASの一部として扱う 手順

## SAS

①

データハンドリング



③

WinBUGS形式の  
データセット  
(text file)を作成

データ受け渡し



MCMCによる  
結果を返す



⑥

解析結果の加工  
推測、グラフ化

## WinBUGS

②

モデリング



④

MCMC  
サンプリング



⑤

計算結果(text file)  
を出力

## 発表構成

- はじめに(背景)
- SAS-WinBUGSによるベイズ分析
  - ・データ解析
  - ・モンテカルロシミュレーション
- おわりに

# SAS-WinBUGSによるベイズ分析 -データ解析-

## Example 用量反応曲線のNDLMによる推定

$\bar{Y}_j$  投与群jの用量反応(群平均)

$j = 1$  : Placebo

$j = 2 \sim 8$  : 被験薬群1~7

用量反応(群平均)の観測値

投与群	Placebo	Dose1	Dose2	Dose3	Dose4	Dose5	Dose6	Dose7
$\bar{Y}_j$	0.8	0.2	0.5	1.0	5.0	4.0	5.5	4.5

NDLMを用いて観測値に基づく推測(グラフ化)

# SAS-WinBUGSによるベイズ分析 -データ解析-

## Example 用量反応曲線のNDLMによる推定

NDLM(Normal Dynamic Linear Model)

- ベイズモデル
- 状態空間モデルの特殊な場合(正規性の仮定)
- 用量反応曲線に単調性の仮定が不要
- 医薬分野での使用例もあり(Adaptive Design)

# SAS-WinBUGSによるベイズ分析 -データ解析-

## 用量反応曲線のモデル化

### 観測方程式

$$\bar{Y}_j = \theta_j + v_j \quad v_j \sim N(0, \sigma^2)$$

### システム方程式

$$\begin{aligned} \theta_j &= \theta_{j-1} + \delta_{j-1} + \omega_j & \omega_j &\sim N(0, W \sigma^2) \\ \delta_j &= \delta_{j-1} + \varepsilon_j & \varepsilon_j &\sim N(0, W \sigma^2) \end{aligned}$$

### 事前分布

$$\begin{aligned} \theta_0 &\sim N(0, 10000) & \delta_0 &\sim N(0, 10000) \\ W &\sim U(0.01, 100) & \sigma^2 &\sim IG(0.1, 0.1) \end{aligned}$$

3つの部分  
から構成

## SAS-WinBUGSによるベイズ分析 -データ解析-

用量反応曲線モデル データが観測される

観測方程式

$$\bar{Y}_j = \theta_j + v_j$$

システム方程式の更新

$$\pi(\theta | \bar{Y}) = \frac{\pi(\bar{Y} | \theta)\pi(\theta)}{\int \pi(\bar{Y} | \theta)\pi(\theta)d\theta}$$

システム方程式

$$\theta_j = \theta_{j-1} + \delta_{j-1}$$

$$\delta_j = \delta_{j-1} + \varepsilon_j$$

$$\varepsilon_j \sim N(0, W\sigma^2)$$

この部分が  
WinBUGS3つの部分  
から構成更新

事前分布

$$\theta_0 \sim N(0, 10000)$$

$$W \sim U(0.01, 100)$$

$$\delta_0 \sim N(0, 10000)$$

$$\sigma^2 \sim IG(0.1, 0.1)$$

# SAS-WinBUGSを用いて計算

## SAS

①

データハンドリング



③

WinBUGS形式の  
データセット  
(text file)を作成

データ受け渡し



⑥

解析結果の加工  
推測、グラフ化MCMCによる  
結果を返す

## WinBUGS

②

モデリング



④

MCMC  
サンプリング

⑤

計算結果(text file)  
を出力



# BUGS言語による統計モデルの記述

The image shows a SAS window titled "SAS - [NDLM.sas]" with a menu bar (File, Edit, View, Tools, Run, Solution, Window, Help) and a toolbar. The script content is as follows:

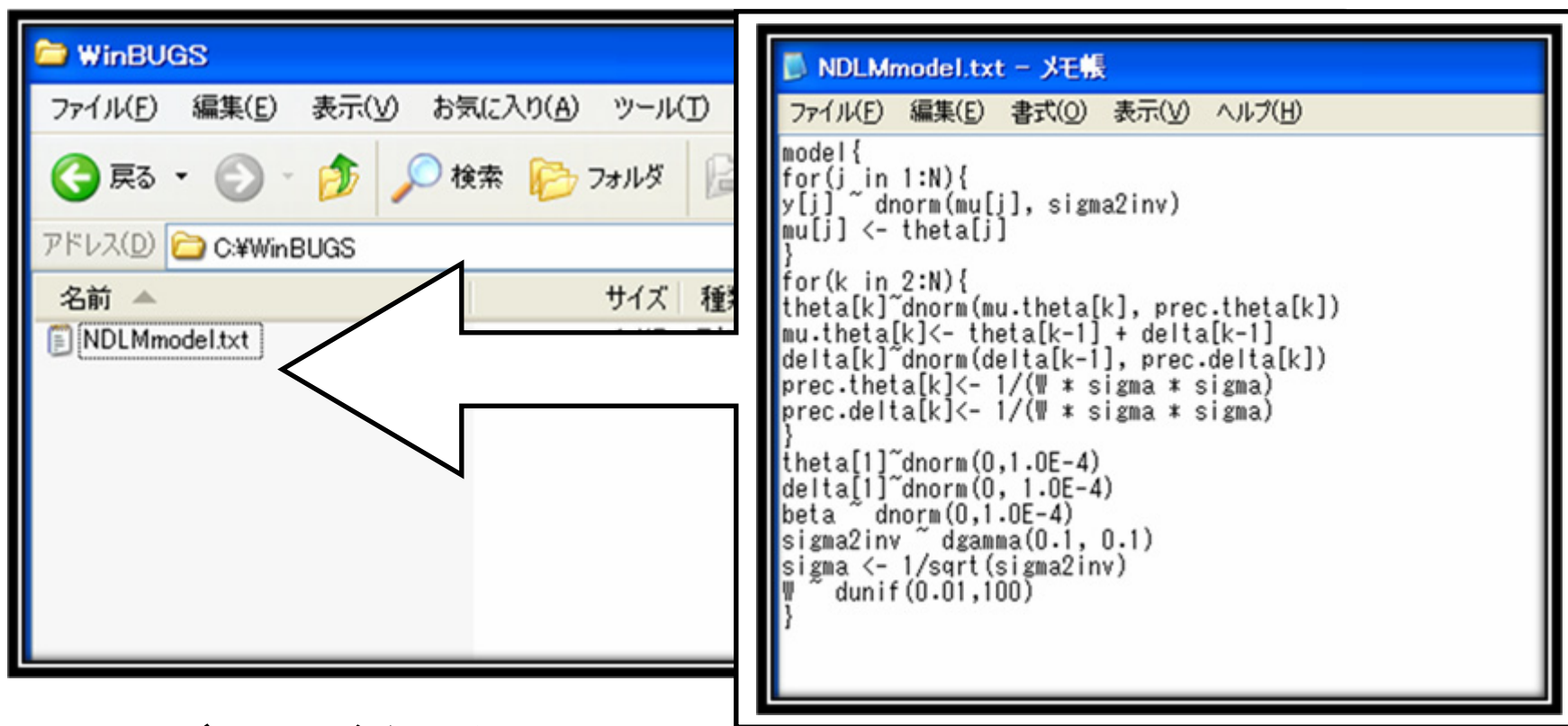
```
21 *モデルの格納;  
22 data model;  
23   input model $80.;  
24   cards;  
25 model{  
26   for(j in 1:N){  
27     y[j] ~ dnorm(mu[j], sigma2inv)  
28     mu[j] <- theta[j]  
29   }  
30   for(k in 2:N){  
31     theta[k]~dnorm(mu.theta[k], prec.theta[k])  
32     mu.theta[k]<- theta[k-1] + delta[k-1]  
33     delta[k]~dnorm(delta[k-1], prec.delta[k])  
34     prec.theta[k]<- 1/(W * sigma * sigma)  
35     prec.delta[k]<- 1/(W * sigma * sigma)  
36   }  
37   theta[1]~dnorm(0,1.OE-4)  
38   delta[1]~dnorm(0, 1.OE-4)  
39   beta ~ dnorm(0,1.OE-4)  
40   sigma2inv ~ dgamma(0.1, 0.1)  
41   sigma <- 1/sqrt(sigma2inv)  
42   W ~ dunif(0.01,100)  
43 }  
44 ;  
45 run;  
46 *モデルの保存;  
47 data _null_;  
48   set model;  
49   file "C:\%WinBUGS%\NDLMmodel.txt";  
50   put model;  
51 run;
```

Annotations on the right side of the script, connected by brackets to specific lines:

- 観測方程式** (Observation Equation): Points to lines 27-28.
- システム方程式** (System Equation): Points to lines 31-33.
- 事前分布** (Prior Distribution): Points to lines 37-42.
- 統計モデルを保存しておく** (Save the statistical model): Points to lines 47-50.

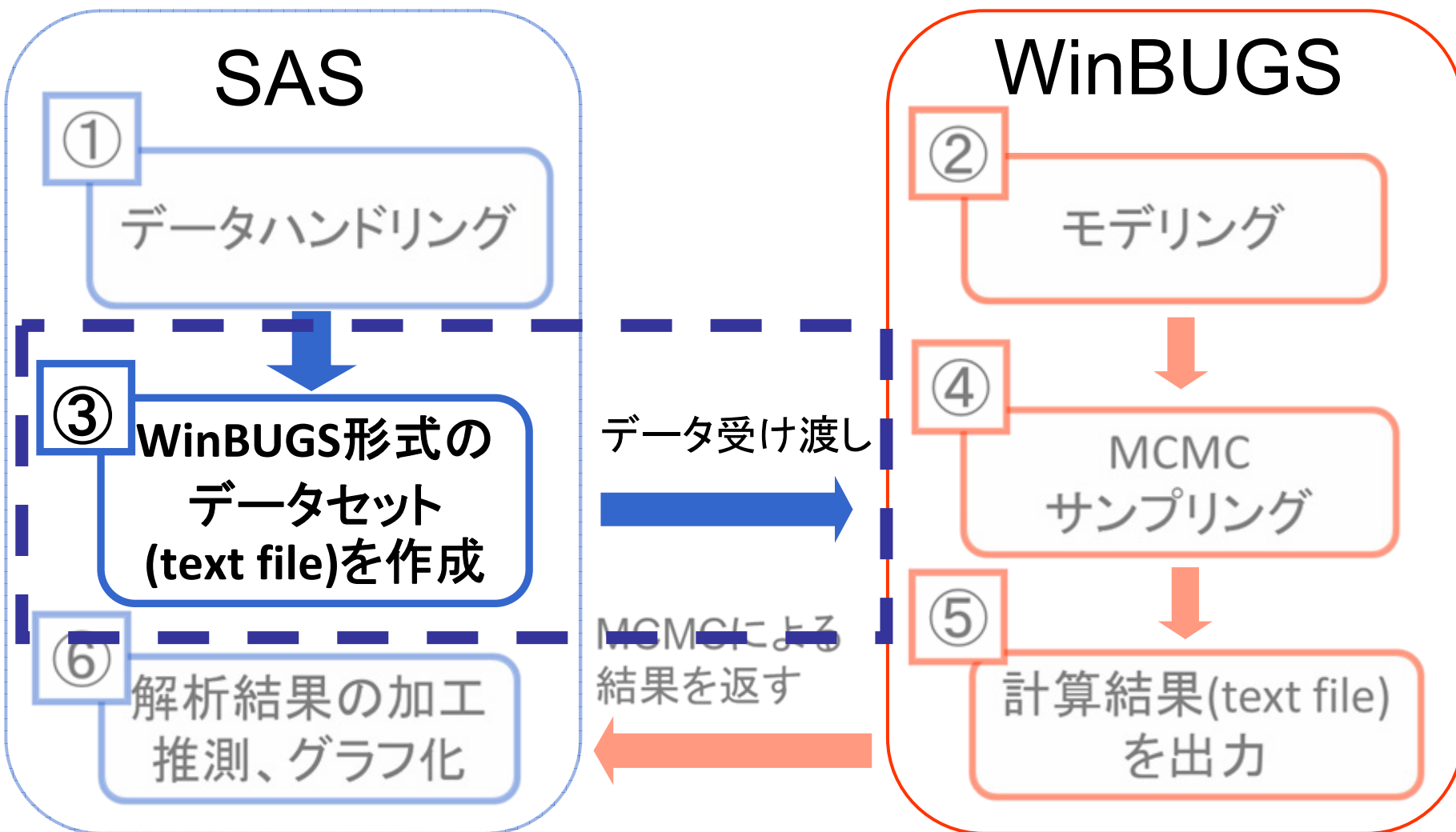
## BUGS言語による統計モデルの記述

任意のフォルダを用意し、ファイルを保存



統計モデルが保存された

# SAS-WinBUGSを用いて計算



# SASデータセットをWinBUGSデータセットに変換

## SASマクロ%DTRANSを用いる


`%dtrans(data=data, outfile=C:¥WinBUGS¥Data.txt, var=y);`

SASデータセット名

WinBUGSデータのファイル名

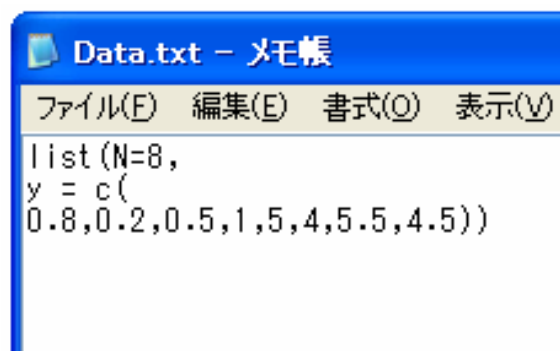
変換する変数名

WinBUGSデータに変換したい



	dose	y
1	1	0.8
2	2	0.2
3	3	0.5
4	4	1
5	5	5
6	6	4
7	7	5.5
8	8	4.5

SASデータセット

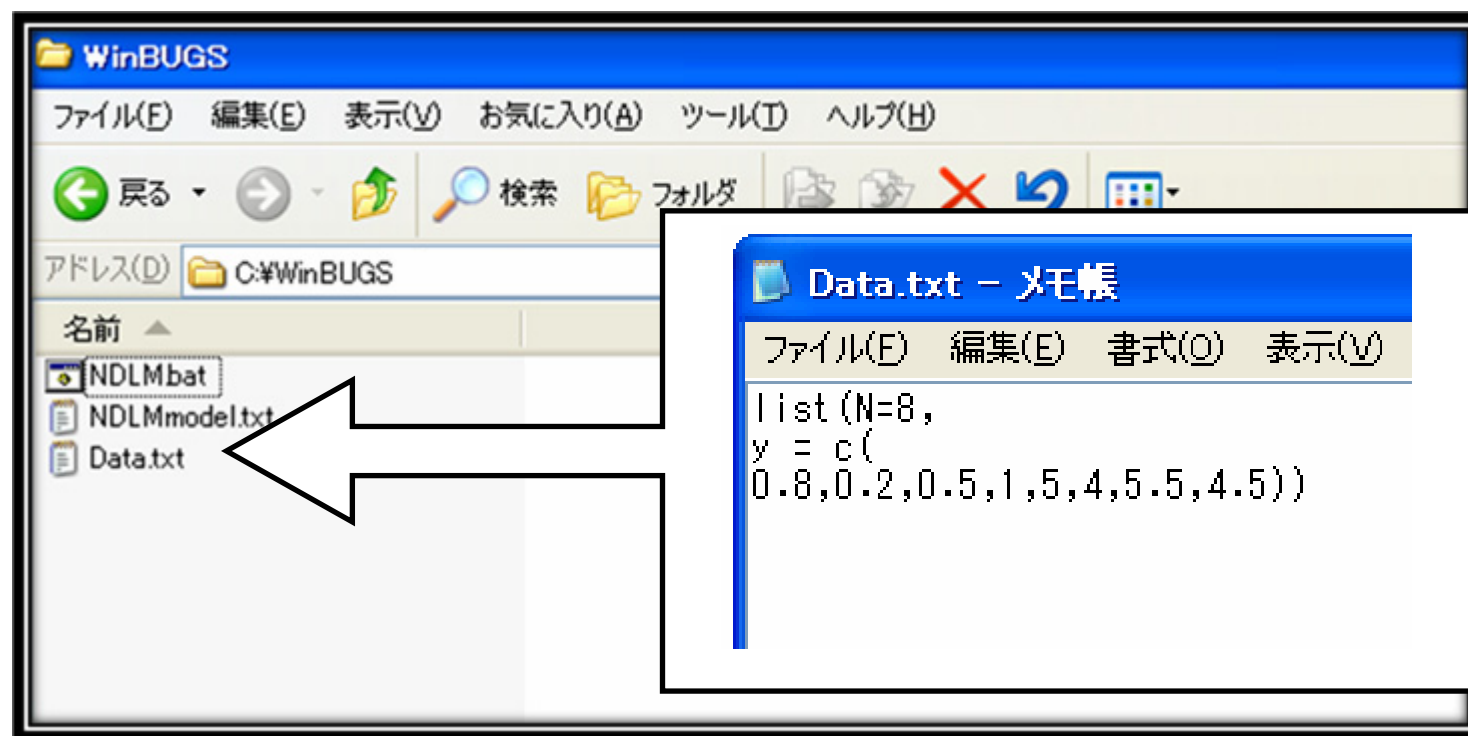


```
list(N=8,  
y = c(  
0.8,0.2,0.5,1,5,4,5.5,4.5))
```

WinBUGSデータセット

# WinBUGSを制御するバッチコードの作成

## WinBUGSデータが保存される



# SAS-WinBUGSを用いて計算

## SAS

①

データハンドリング



③

WinBUGS形式の  
データセット  
(text file)を作成

データ受け渡し



⑥

解析結果の加工  
推測、グラフ化MCMCによる  
結果を返す

## WinBUGS

②

モデリング



④

MCMC  
サンプリング

⑤

計算結果(text file)  
を出力



## WinBUGSを制御するバッチコードの作成

以下の手順をWinBUGSに行わせたい

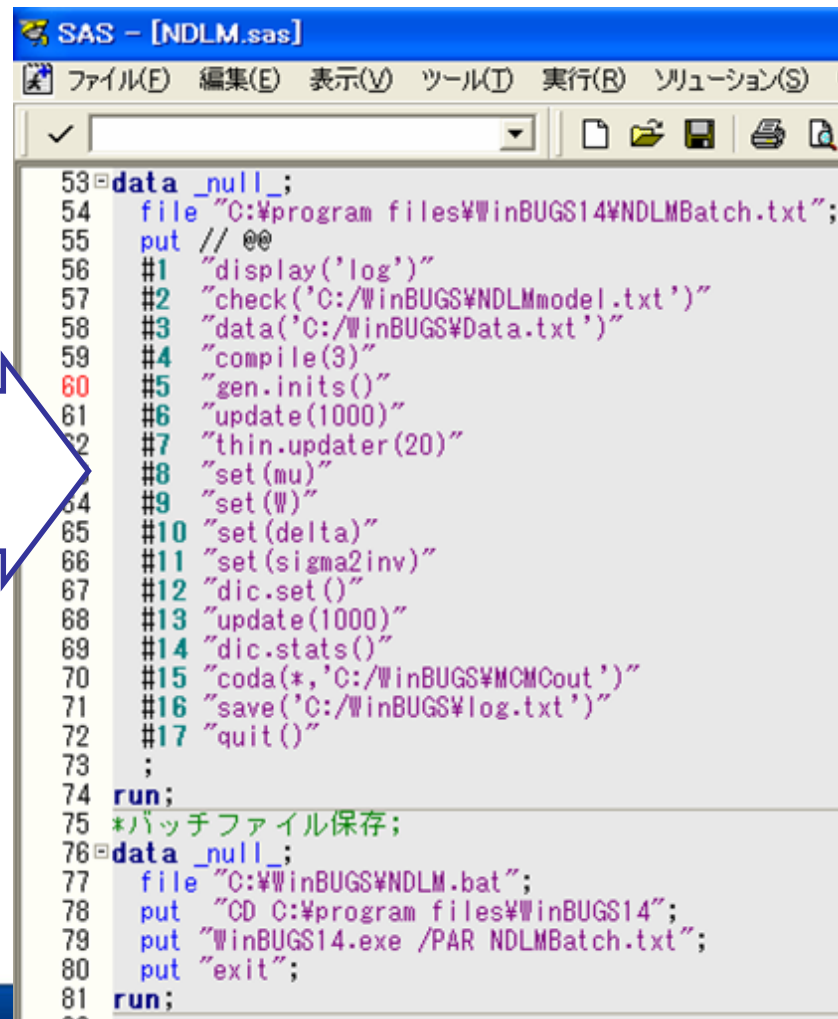
- ①統計モデルを読み込む
- ②データを読み込む
- ③マルコフ連鎖の本数を決定
- ④Burn inのサンプル数を決定
- ⑤マルコフ連鎖からのサンプリング間隔を決定
- ⑥推定するパラメータを指定
- ⑦マルコフ連鎖からサンプリングするサンプル数を決定
- ⑧マルコフ連鎖からの確率標本を保存

# WinBUGSを制御するバッチコードの作成

WinBUGSはスクリプトで実行することが出来る

データの読み込み  
モデルのコンパイル  
計算のための設定  
計算結果の保存先 etc.

全ての手順を  
バッチファイルで保存



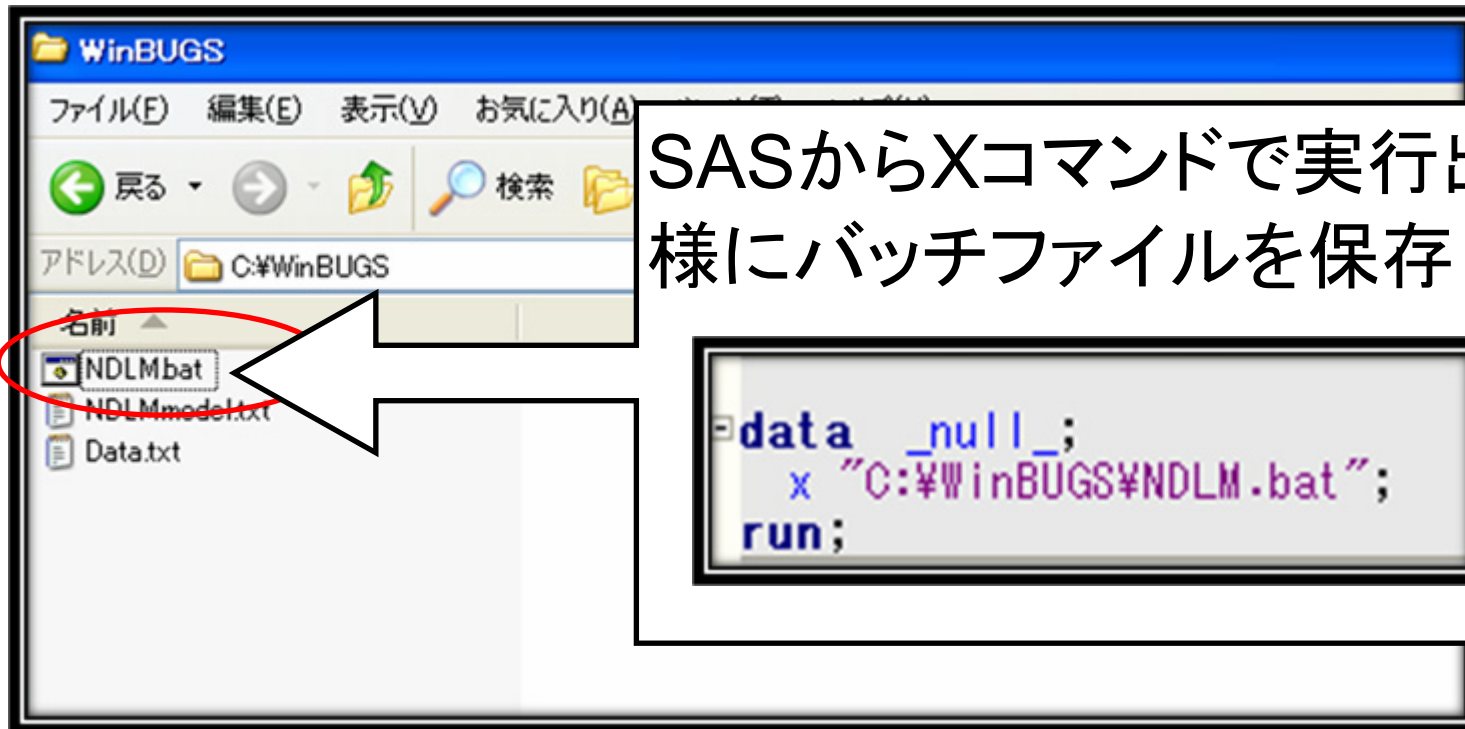
```
SAS - [NDLM.sas]
ファイル(F) 編集(E) 表示(V) ツール(T) 実行(R) ソリューション(S)

53 data _null_;
54   file "C:\program files\WinBUGS14\NDLMBatch.txt";
55   put // @@
56   #1 "display('log')"
57   #2 "check('C:/WinBUGS\NDLMmodel.txt')"
58   #3 "data('C:/WinBUGS\Data.txt')"
59   #4 "compile(3)"
60   #5 "gen.inits()"
61   #6 "update(1000)"
62   #7 "thin.updater(20)"
63   #8 "set(mu)"
64   #9 "set(W)"
65   #10 "set(delta)"
66   #11 "set(sigma2inv)"
67   #12 "dic.set()"
68   #13 "update(1000)"
69   #14 "dic.stats()"
70   #15 "coda(*,'C:/WinBUGS\MCMCout')"
71   #16 "save('C:/WinBUGS\log.txt')"
72   #17 "quit()"
73 ;
74 run;
75 *バッチファイル保存;
76 data _null_;
77   file "C:\WinBUGS\NDLM.bat";
78   put "CD C:\program files\WinBUGS14";
79   put "WinBUGS14.exe /PAR NDLMBatch.txt";
80   put "exit";
81 run;
```



## WinBUGSを制御するバッチコードの作成

SASからバッチファイルを実行することで、WinBUGSが行う全ての処理をSASから制御可能



SASからXコマンドで実行出来る様にバッチファイルを保存

```
data _null_;  
  x "C:\WinBUGS\NDLM.bat";  
run;
```

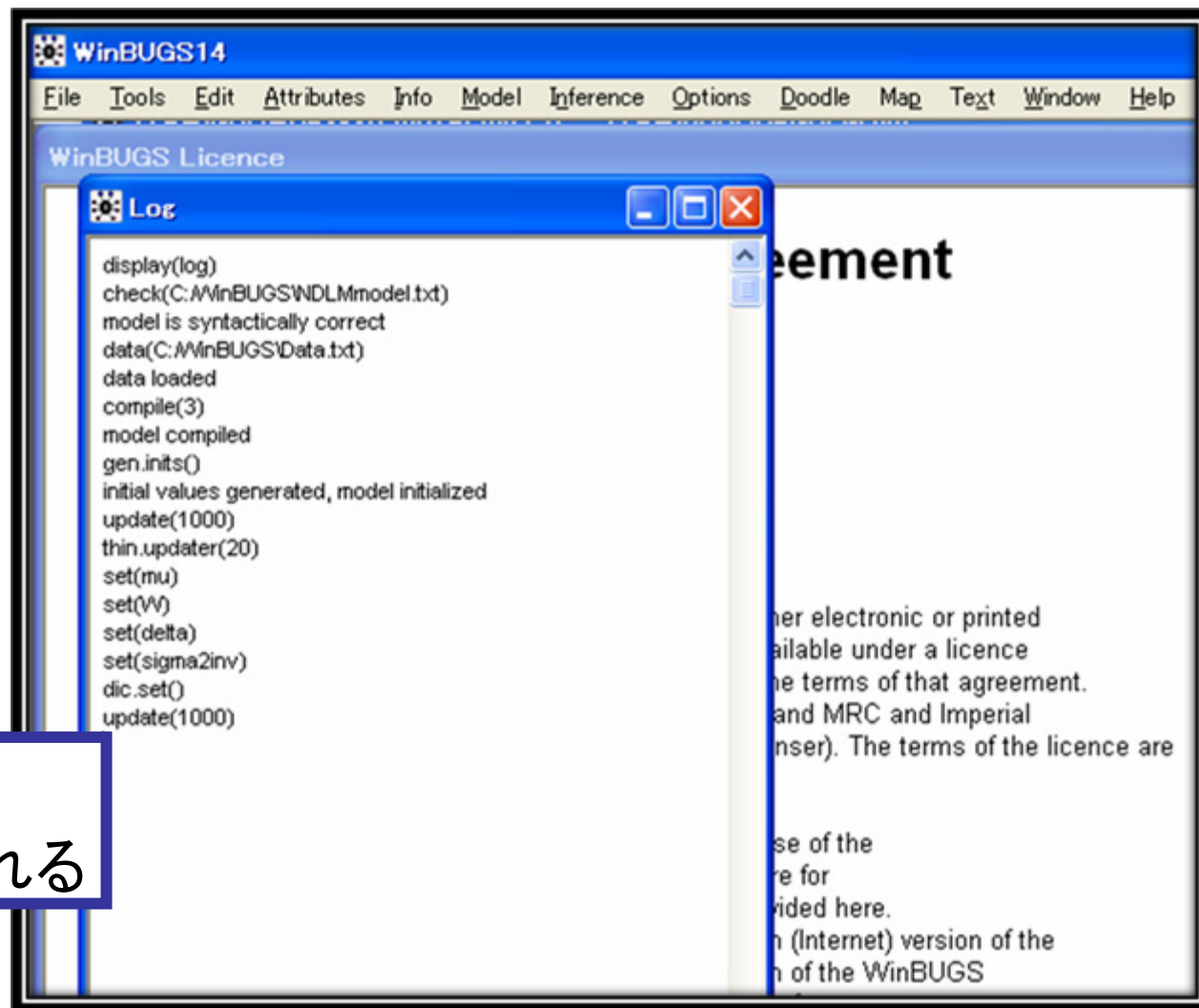
# MCMCサンプリングの実行

## バッチファイル実行

```
data _null_;  
  x "C:\WinBUGS\NDLM.bat";  
run;
```



WinBUGSが起動し、  
一連の処理が実行される



# MCMCサンプリング結果(確率標本)の保存

The screenshot displays the WinBUGS interface. On the left, the 'File' menu is open, showing a list of files in the 'C:\WinBUGS' directory. A red box highlights the files 'MCMCout1.txt', 'MCMCout2.txt', 'MCMCout3.txt', and 'MCMCoutIndex.txt'. A large black arrow points from this box towards the right. On the right, two windows show the contents of these files. The 'MCMCoutIndex.txt' window displays a table of parameters and their corresponding MCMC iterations. The 'MCMCout1.txt' window displays a list of MCMC samples.

**WinBUGS File List:**

- NDLMBat
- NDLMmodel.txt
- Data.txt
- log.txt
- MCMCout1.txt**
- MCMCout2.txt**
- MCMCout3.txt**
- MCMCoutIndex.txt**

**MCMCoutIndex.txt - メモ帳**

パラメータ	1000	2000	3000	4000	5000	6000	7000	8000	9000	10000	11000	12000	13000	14000	15000	16000	17000	18000
W	1																	
delta[1]		1001	2000															
delta[2]			2001	3000														
delta[3]				3001	4000													
delta[4]					4001	5000												
delta[5]						5001	6000											
delta[6]							6001	7000										
delta[7]								7001	8000									
delta[8]									8001	9000								
mu[1]										9001	10000							
mu[2]											10001	11000						
mu[3]												11001	12000					
mu[4]													12001	13000				
mu[5]														13001	14000			
mu[6]															14001	15000		
mu[7]																15001	16000	
mu[8]																	16001	17000
sigma2inv																		17001 18000

**MCMCout1.txt - メモ帳**

サンプル番号	値
1001	0.8129
1002	5.901
1003	50.92
1004	83.18
1005	12.79
1006	20.56
1007	15.98
1008	59.36
1009	6.709
1010	10.34
1011	28.79
1012	8.264
1013	2.998
1014	0.6706
1015	40.66
1016	23.56
1017	8.341
1018	51.37
1019	10.46
1020	2.618
1021	0.1953
1022	0.09126
1023	3.84
1024	29.87
1025	9.073

Indexファイルとモンテカルロ標本ファイルが保存される

# SAS-WinBUGSを用いて計算

## SAS

①

データハンドリング



③

WinBUGS形式の  
データセット  
(text file)を作成

データ受け渡し



⑥

解析結果の加工  
推測、グラフ化MCMCによる  
結果を返す

## WinBUGS

②

モデリング



④

MCMC  
サンプリング

⑤

計算結果(text file)  
を出力

# WinBUGSからMCMCの計算結果をSASに返す

## モンテカルロ標本のレコードとパラメータを紐付ける

MCMCoutIndex.txt - メモ帳			
ファイル(F)	編集(E)	書式(O)	表示(V)
W	1	1000	
delta[1]		1001	2000
delta[2]		2001	3000
delta[3]		3001	4000
delta[4]		4001	5000
delta[5]		5001	6000
delta[6]		6001	7000
delta[7]		7001	8000
delta[8]		8001	9000
mu[1]	9001	10000	
mu[2]	10001	11000	
mu[3]	11001	12000	
mu[4]	12001	13000	
mu[5]	13001	14000	
mu[6]	14001	15000	
mu[7]	15001	16000	
mu[8]	16001	17000	
sigma2inv		17001	18000

MCMCout1.txt - メモ帳	
ファイル(F)	編集(E)
1001	0.8129
1002	5.901
1003	50.92
1004	89.18
1005	12.79
1006	20.56
1007	15.98
1008	59.36
1009	6.709
1010	10.34
1011	28.79
1012	8.264
1013	2.998
1014	0.6706
1015	40.66
1016	23.56
1017	8.341
1018	51.37
1019	10.46
1020	2.618
1021	0.1953
1022	0.09126
1023	3.84
1024	29.87
1025	9.073
1026	31.19
1027	80.88
1028	19.05



パラメータINDEX(WinBUGS出力)

モンテカルロ標本(WinBUGS出力)

# WinBUGSからMCMCの計算結果をSASに返す

## モンテカルロ標本のレコードとパラメータを紐付ける

MCMCoutIndex.txt - メモ帳

ファイル(F)	編集(E)	書式(O)	表示(V)	ヘルプ(H)
W	1	1000		
delta[1]	1001	2000		
delta[2]	2001	3000		
delta[3]	3001	4000		
mu[2]	10001	11000		
mu[3]	11001	12000		
mu[4]	12001	13000		
mu[5]	13001	14000		
mu[6]	14001	15000		
mu[7]	15001	16000		
mu[8]	16001	17000		
sigma2inv	17001	18000		

MCMCout1.txt - メモ帳

ファイル(F)	編集(E)	書式(O)	表示(V)	ヘルプ(H)
1001	0.6			
1002	5.9			
1003	50			
1004	89			
	12			
	20			
	15			
	50			
	28			
	8.2			
	2.9			
1014	0.6			
1015	40			
1016	23			
1017	8.3			
1018	51			
1019	10.46			
1020	2.618			
1021	0.1953			
1022	0.09126			
1023	3.84			
1024	29.87			
1025	9.073			
1026	31.19			
1027	80.88			
1028	19.05			
1029	19.05			

最初の1000行は  
「W」のモンテカルロ標本

次の1000行「delta[1]」  
のモンテカルロ標本

パラメータINDEX(WinBUGS出力)

モンテカルロ標本(WinBUGS出力)



# WinBUGSからMCMCの計算結果をSASに返す

## SASマクロ%**CODATRANS**を用いる

<b>%codatrans(</b>	
<b>codapath=C:¥WinBUGS,</b>	WinBUGS出力ファイルの保存場所
<b>prefix=mcmcout ,</b>	ファイル名のprefix
<b>outcoda=coda,</b>	モンテカルロ標本のSASデータセット名
<b>outstats=stat,</b>	モンテカルロ標本の 要約統計量データセット名
<b>outnums=3,</b>	マルコフチェーンの数
<b>outgraph=Y);</b>	事後分布のグラフを出力

# WinBUGSからMCMCの計算結果をSASに返す

マクロ引数: **outcoda=coda**

モンテカルロ標本をSASデータセット化

VIEWTABLE: Work.Coda

	number	w	delta_1	delta_2	delta_3	delta_4	delta_5	delta_6	delta_7	delta_8	mu_1	mu_2	mu_3
1	1001	0.8129	-0.4978	-0.3816	0.8263	1.559	-0.5785	0.4429	2.186	1.015	2.34	1.26	0.5337
2	1002	5.901	0.4036	0.004171							0.9441	-0.02201	0.6612
3	1003	50.92	1.938	4.638							0.4768	0.1587	0.6892
4	1004	83.18	-7.067	-0.9522							0.8554	0.4955	0.705
5	1005	12.79	-0.3146	-0.09374							0.5969	-0.4451	0.5789
6	1006	20.56	-0.7494	0.2539	1.161	2.09	0.7264	-0.1552	-0.787	-0.1274	1.046	0.1316	0.7951
7	1007	15.98	-0.156	0.2087	1.051	0.7619	-1.097	0.5844	-0.173	0.06441	1.332	0.5328	-0.006877
8	1008	59.36	0.8956	2.396	1.863	2.48	2.366	3.475	1.656	0.4958	1.146	0.004825	0.6047
9	1009	6.709	-2.103	0.06867	-0.2643	1.422	0.9293	0.1561	-0.1995	-0.513	1.336	0.04096	1.011
10	1010	10.34	0.345	2.378	0.7441	3.183	0.1838	0.2782	1.117	2.678	1.161	0.2685	0.3562
11	1011	28.79	1.064	-2.107	-3.388	-0.8365	0.8605	-0.04649	-2.499	-2.969	0.5851	0.1492	0.2799
12	1012	8.264	1.165	1.18	1.577	2.425	1.164	1.295	1.155	2.958	0.731	0.04572	0.3866
13	1013	2.998	0.5043	-0.2285	1.264	0.9211	-1.68	-0.8235	-0.2605	-1.973	-0.5193	1.561	-0.4777
14	1014	0.6706	-2.68	-2.948	-0.177	2.706	0.4849	1.488	2.525	2.679	2.735	-0.7499	0.5146
15	1015	40.66	0.8446	1.287	2.203	0.3125	-1.649	-1.555	-2.596	-0.5446	0.4836	0.4023	0.5209
16	1016	23.56	1.5	0.881	3.056	2.046	2.539	1.12	-1.422	-4.23	1.03	0.09807	0.8117
17	1017	8.341	0.2025	0.7596	1.528	2.017	0.464	-0.946	-1.351	-1.456	0.9969	0.2831	0.4142
18	1018	51.37	-1.122	0.9344	-0.008039	1.006	-0.6672	-0.6759	-0.9335	-0.3696	0.9125	0.2616	0.9156
19	1019	10.46	0.1752	0.05642	-1.498	-0.5997	0.5691	0.7636	1.842	5.211	1.646	0.6908	0.6246
20	1020	2.618	0.8432	1.416	0.5596	0.9903	-0.731	0.2322	0.6909	0.7695	2.452	2.397	-0.3486
21	1021	0.1953	0.3093	0.2487	0.9681	0.7882	0.7215	0.7345	0.7367	1.019	0.5159	0.6261	1.387
22	1022	0.09126	0.5452	0.3657	0.4406	0.4932	0.8736	0.9983	1.012	1.366	0.3412	1.264	1.424

パラメータ毎に変数を作成



# WinBUGSからMCMCの計算結果をSASに返す

マクロ引数: **outcoda=coda**

モンテカルロ標本をSASデータセット化

VIEWTABLE: Work.Coda

	number	w	delta_1	delta_2	delta_3	delta_4	delta_5	delta_6	delta_7	delta_8	mu_1	mu_2	mu_3
1	1001	0.8129	-0.4978	-0.3816	0.8263	1.559	-0.5785	0.4429	2.186	1.015	2.34	1.26	0.5337
2	1002	5.901	0.4036	0.004171	1.427	1.016	0.192	0.847	0.8589	1.492	0.9441	-0.02201	0.6612
3	1003	50.92	1.938	4.638	5.398	3.666	-0.7563	-1.306	-0.07741	-4.174	0.4768	0.1587	0.6892
4	1004	83.18	-7.067	-0.9522	0.2444	0.5754	3.26	2	-0.2767	4.102	0.8554	0.4955	0.705
5	1005	12.70	-0.2146	-0.00274	0.6802	2.562	0.8167	-1.622	-0.02260	-0.4600	0.5060	-0.4451	0.5790

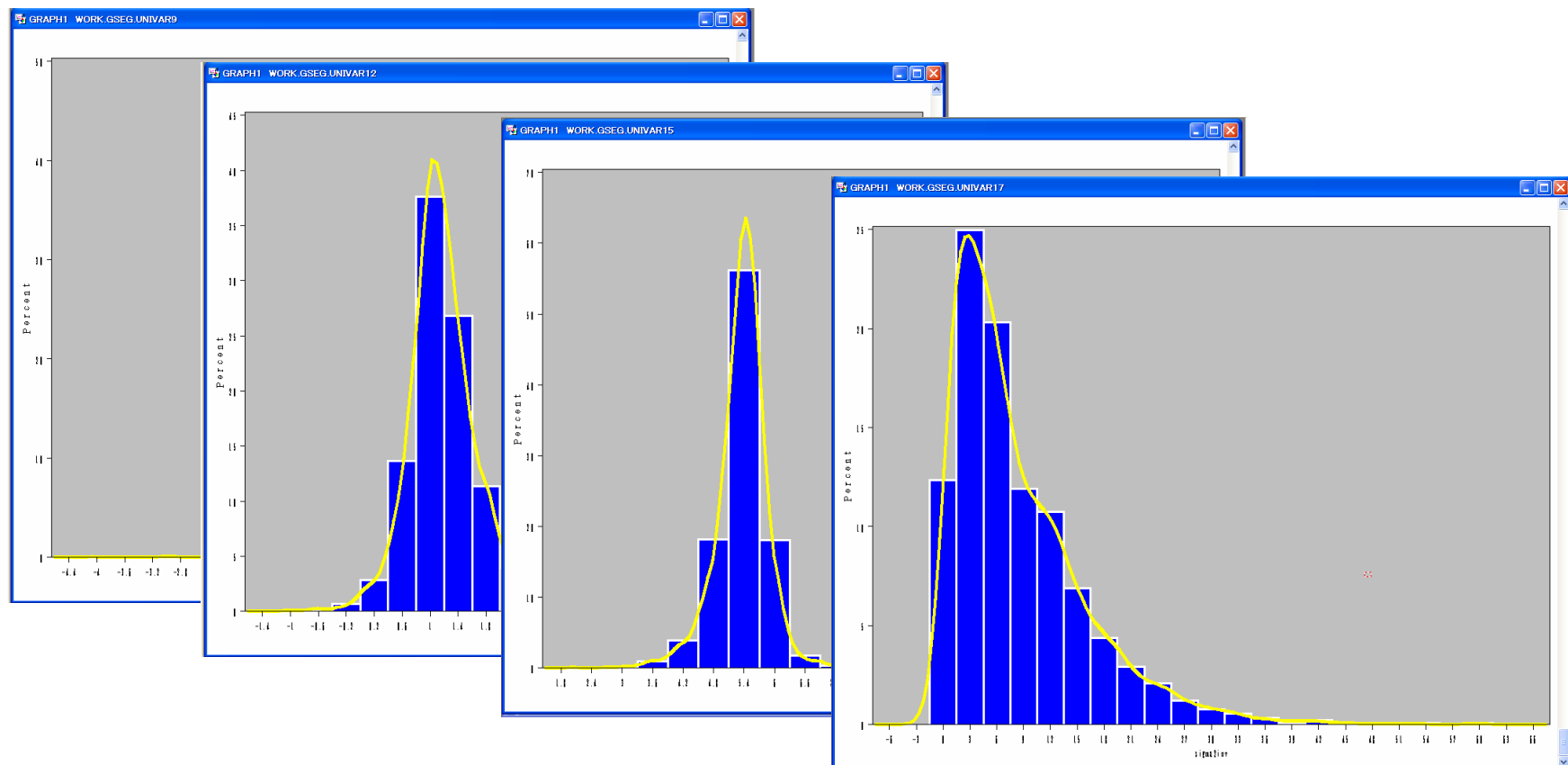
グラフ化すると事後分布のグラフが得られ、  
Proc.UNIVARIATEで事後分布の要約統計量が得られる

16	1016	23.56	1.5	0.881	3.056	2.046	2.539	1.12	-1.422	-4.23	1.03	0.09807	0.8117
17	1017	8.341	0.2025	0.7596	1.528	2.017	0.464	-0.946	-1.351	-1.456	0.9969	0.2831	0.4142
18	1018	51.37	-1.122	0.9344	-0.008039	1.006	-0.6672	-0.6759	-0.9335	-0.3696	0.9125	0.2616	0.9156
19	1019	10.46	0.1752	0.05642	-1.498	-0.5997	0.5691	0.7636	1.842	5.211	1.646	0.6908	0.6246
20	1020	2.618	0.8432	1.416	0.5596	0.9903	-0.731	0.2322	0.6909	0.7695	2.452	2.397	-0.3486
21	1021	0.1953	0.3093	0.2487	0.9681	0.7882	0.7215	0.7345	0.7367	1.019	0.5159	0.6261	1.387
22	1022	0.09126	0.5452	0.3657	0.4406	0.4932	0.8736	0.9983	1.012	1.366	0.3412	1.264	1.424

# 事後分布のグラフと要約統計量

マクロ引数: **outgraph=Y**

各パラメータの事後分布のグラフを出力



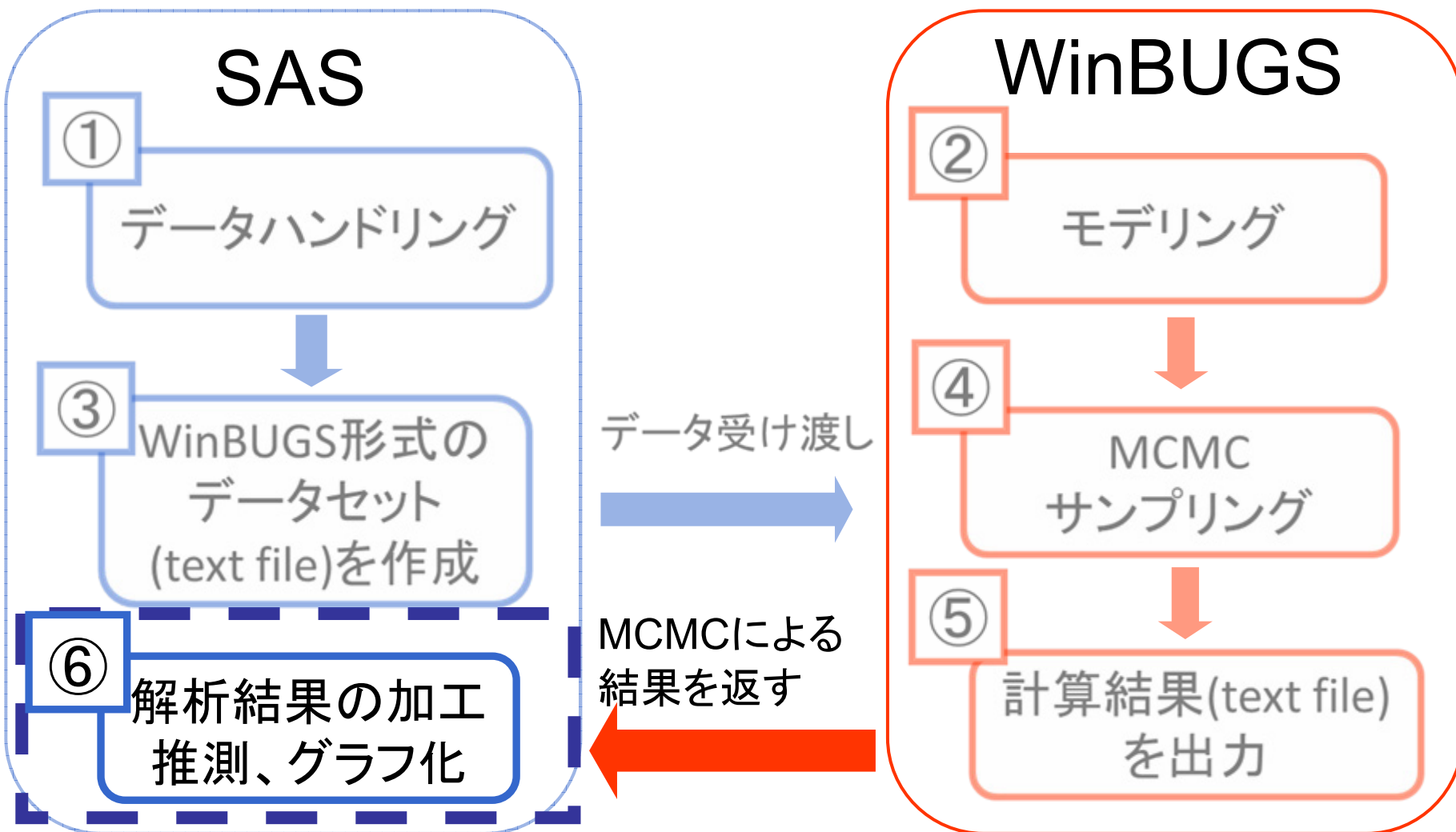
# 事後分布のグラフと要約統計量

**outstats=stat**

各パラメータの事後分布の要約統計量が算出される

VIEWTABLE: Work.Stat														
	parameter	prm_grp	prm_seq	非欠損値数	平均	標準偏差	歪度	分散	尖度	最大値	中央値	最小値	10,000 パーセント点	90,000 パーセント
1	W	w	1	3000	26.63708114	24.930282067	1.10313302	621.51896394	0.3467789804	99.93	18.25	0.02041	2.1365	65.73
2	delta[1]	delta	1	3000	-0.0605741	1.5780603826	-0.0577204	2.4902745712	5.2651096041	11.46	-0.06721	-12.05	-1.876	1.692
3	delta[2]	delta	2	3000	0.375608218	1.3135699507	0.12012019	1.7254660155	3.9762815768	10.19	0.3925	-6.826	-1.1635	1.8665
4	delta[3]	delta	3	3000	0.893037103	1.2864807354	-0.0875959	1.6550326826	2.2982085407	7.951	0.8974	-6.048	-0.5834	2.3955
5	delta[4]	delta	4	3000	1.664046959	1.2983624929	0.10190963	1.6857451631	2.1514752598	8.375	1.6085	-6.697	0.199	3.219
6	delta[5]	delta	5	3000	0.563245414	1.3279683376	0.20267103	1.7634999058	6.7209786227	15.09	0.5831	-7.298	-0.9575	2.04
7	delta[6]	delta	6	3000	0.566029353	1.3314326337	-0.1367976	1.7727128581	2.7665595527	7.602	0.5562	-9.556	-1.0075	2.082
8	delta[7]	delta	7	3000	-0.16210578	1.58556413	-0.4309007	2.5140136104	5.7990972119	7.602	-0.1351	-16.47	-1.9965	1.6335
9	delta[8]	delta	8	3000	-0.20656803	2.4696494323	-0.0374644	6.0991683183	3.1817268086	17.81	-0.1718	-14.31	-3.07	2.5875
10	mu[1]	mu	1	3000	0.731519339	0.5548388908	-0.8417992	0.3078461947	6.8118860482	4.449	0.7628	-4.062	0.14485	1.293
11	mu[2]	mu	2	3000	0.252234799	0.4961424499	-0.1773653	0.2461573306	7.416356079	3.001	0.23525	-5.03	-0.26305	0.81495
12	mu[3]	mu	3	3000	0.526916099	0.4983875894	0.27259134	0.2483901893	4.7451164958	4.451	0.5124	-2.779	-0.01624	1.0765
13	mu[4]	mu	4	3000	1.208953486	0.5331263368	1.23312418	0.284223691	9.3064522439	7.718	1.145	-1.019	0.6605	1.8475
14	mu[5]	mu	5	3000	4.676099167	0.5715701419	-1.1001139	0.3266924271	2.7637747851	6.792	4.775	0.8475	3.9325	5.27
15	mu[6]	mu	6	3000	4.133931767	0.4786931605	0.04757807	0.229147142	2.9523977202	6.774	4.106	0.9503	3.608	4.72
16	mu[7]	mu	7	3000	5.366895333	0.5415760395	0.96349092	0.2933046066	24.26059374	13.91	5.399	1.986	4.7625	5.894
17	mu[8]	mu	8	3000	4.5829778	0.5791824063	0.79869325	0.3354522597	7.9043618734	8.918	4.5565	0.1524	4.004	5.166
18	sigma2inv	sigma2inv	1	3000	8.447894677	7.5173891195	1.74068392	56.511139174	4.4925582882	60.09	6.195	0.08833	1.347	18.3

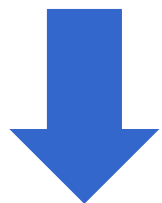
# SAS-WinBUGSを用いて計算



# 解析結果の加工

## 統計解析の目的によって処理の内容は異なる

今回の例では、Dose response curveに興味がある

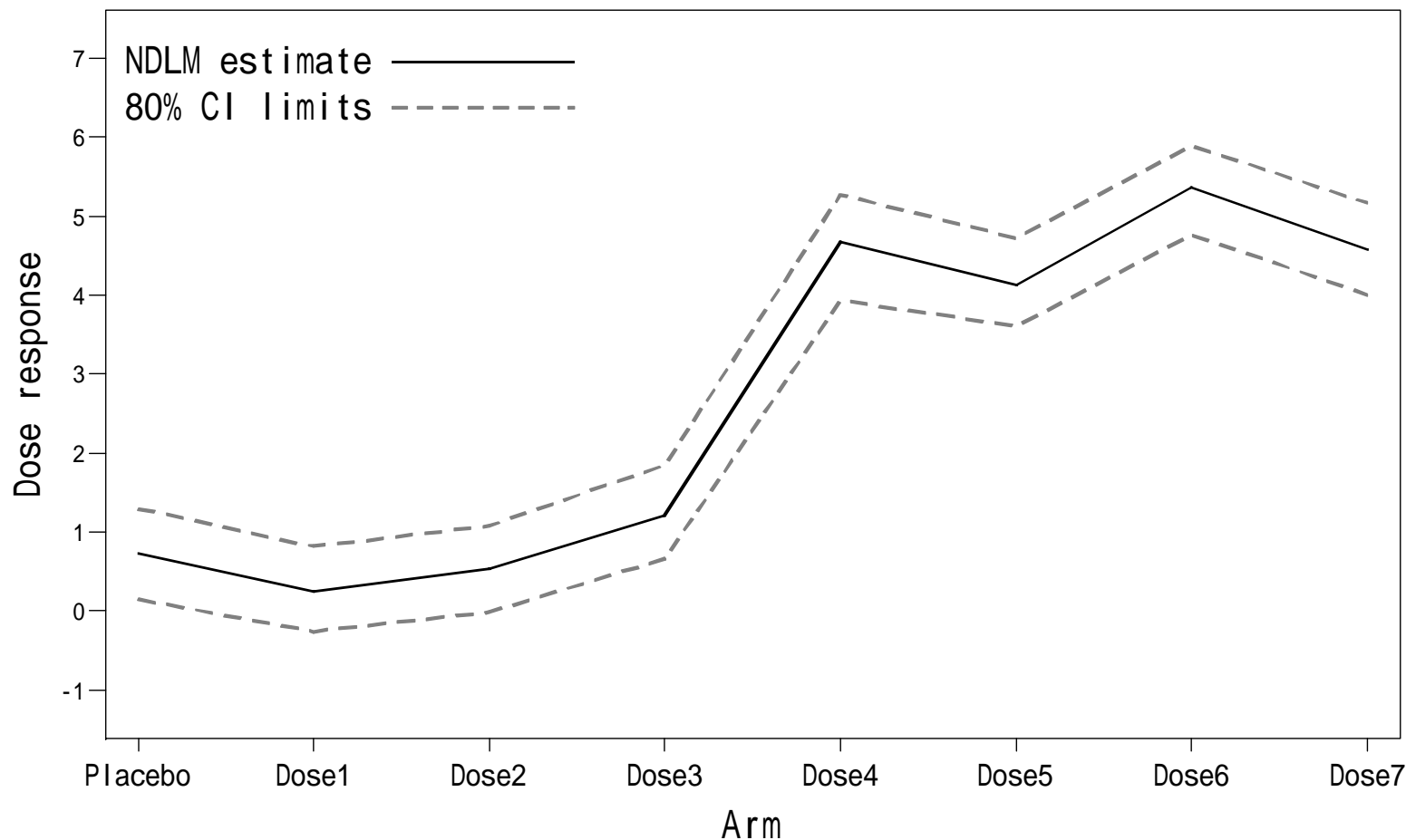


要約統計量データセットから  
平均・パーセント点をグラフ化

VIEWTABLE: Work.Stat														
	parameter	prm_grp	prm_seq	非欠損値数	平均	標準偏差	歪度	分散	尖度	最大値	中央値	最小値	10.0000 パーセント点	90.0000 パーセント
1	W	w	1	3000	26.63708114	24.930282067	1.10313302	621.51896394	0.3467789804	99.93	18.25	0.02041	2.1365	65.73
2	delta[1]	delta	1	3000	-0.0605741	1.5780603826	-0.0577204	2.4902745712	5.2651096041	11.46	-0.06721	-12.05	-1.876	1.692
3	delta[2]	delta	2	3000	0.375608218	1.3135699507	0.12012019	1.7254660155	3.9762815768	10.19	0.3925	-6.826	-1.1635	1.8665
4	delta[3]	delta	3	3000	0.893037103	1.2864807354	-0.0875959	1.6550326826	2.2982085407	7.951	0.8974	-6.048	-0.5834	2.3955
5	delta[4]	delta	4	3000	1.664046959	1.2983624929	0.10190963	1.6857451631	2.1514752598	8.375	1.6085	-6.697	0.199	3.219
6	delta[5]	delta	5	3000	0.563245414	1.3279683376	0.20267103	1.7634999058	6.7209786227	15.09	0.5831	-7.298	-0.9575	2.04
7	delta[6]	delta	6	3000	0.566029353	1.3314326337	-0.1367976	1.7727128581	2.7665595527	7.602	0.5562	-9.556	-1.0075	2.082
8	delta[7]	delta	7	3000	-0.16210578	1.58556413	-0.4309007	2.5140136104	5.7990972119	7.602	-0.1351	-16.47	-1.9965	1.6335
9	delta[8]	delta	8	3000	-0.20656803	2.4696494323	-0.0374644	6.0991683183	3.1817268086	17.81	-0.1718	-14.31	-3.07	2.5875
10	mu[1]	mu	1	3000	0.731519339	0.5548388908	-0.8417992	0.3078461947	6.8118860482	4.449	0.7628	-4.062	0.14485	1.293
11	mu[2]	mu	2	3000	0.252234799	0.4961424499	-0.1773653	0.2461573306	7.416356079	3.001	0.23525	-5.03	-0.26305	0.81495
12	mu[3]	mu	3	3000	0.526916099	0.4983875894	0.27259134	0.2483901893	4.7451164958	4.451	0.5124	-2.779	-0.01624	1.0765
13	mu[4]	mu	4	3000	1.208953486	0.5331263368	1.23312418	0.284223691	9.3064522439	7.718	1.145	-1.019	0.6605	1.8475
14	mu[5]	mu	5	3000	4.676099167	0.5715701419	-1.1001139	0.3266924271	2.7637747851	6.792	4.775	0.8475	3.9325	5.27
15	mu[6]	mu	6	3000	4.133931767	0.4786931605	0.04757807	0.229147142	2.9523977202	6.774	4.106	0.9503	3.608	4.72
16	mu[7]	mu	7	3000	5.366895333	0.5415760395	0.96349092	0.2933046066	24.26059374	13.91	5.399	1.986	4.7625	5.894
17	mu[8]	mu	8	3000	4.5829778	0.5791824063	0.79869325	0.3354522597	7.9043618734	8.918	4.5565	0.1524	4.004	5.166
18	sigma2inv	sigma2inv	1	3000	8.447894677	7.5173891195	1.74068392	56.511139174	4.4925582882	60.09	6.195	0.08833	1.347	18.3

## 解析結果の加工

### 事後分布の平均と信用区間をグラフ化



## 発表構成

- はじめに(背景)
- SAS-WinBUGSによるベイズ分析
  - ・データ解析
  - ・モンテカルロシミュレーション
- おわりに

## モンテカルロシミュレーション

解析的にはアプローチ困難な問題

- 複雑な仮定 (様々なif)
- 複雑な数式



モンテカルロシミュレーション

- 計算機による大量の反復計算
- 容易に近似解が得られる



# WinBUGSによるベイズモデルのシミュレーション

WinBUGSに出来ないこと

- データハンドリングが出来ない
- IF(条件分岐)、Loop(反復)の仕組み自体が無い

シミュレーションには大量反復計算が必要

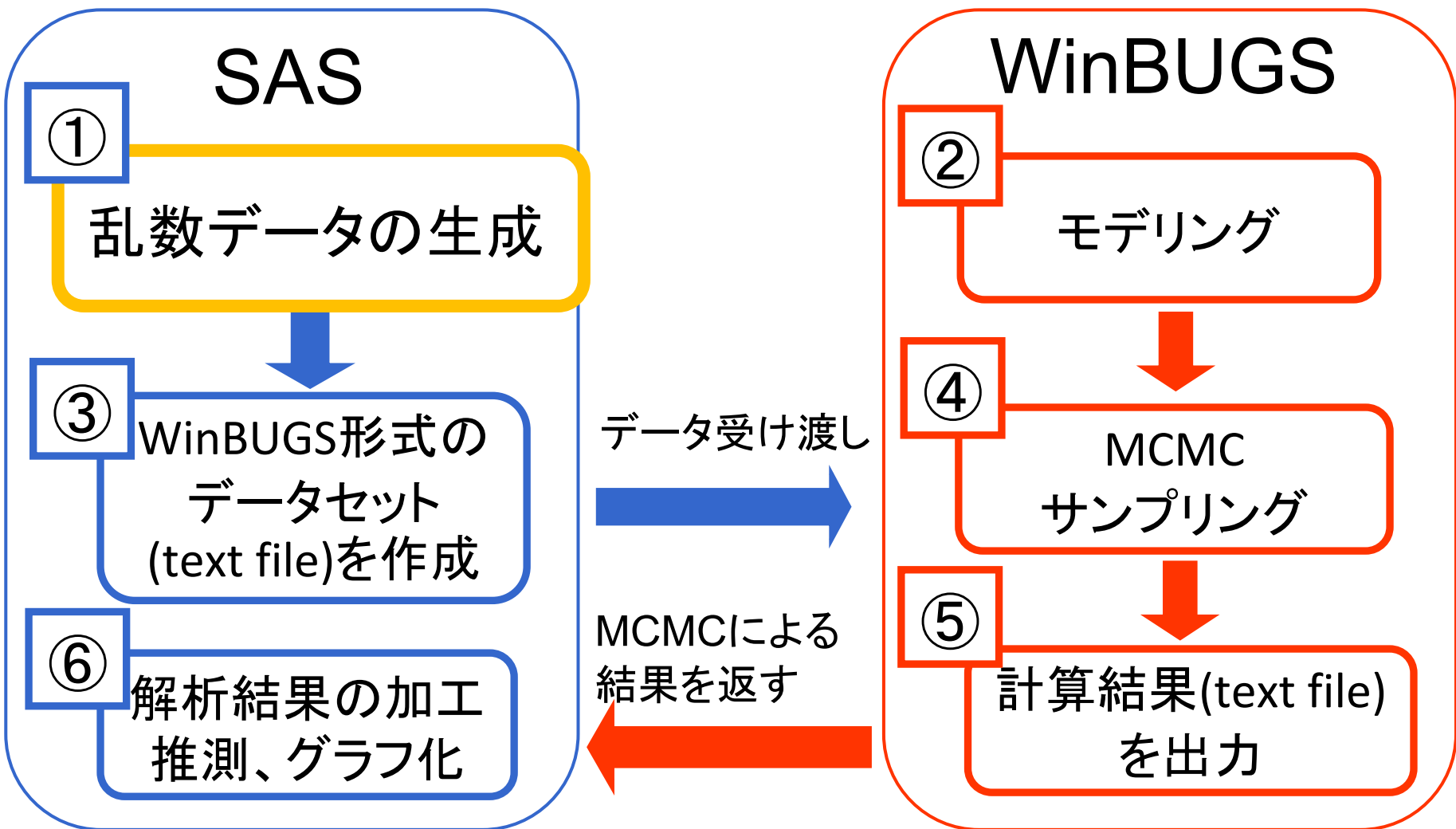


処理全体をSASで制御し、反復実行



モンテカルロシミュレーションの実行

## SASを用いて解析処理を反復実行



# SAS-WinBUGSによるモンテカルロシミュレーション

## Example 用量反応曲線のNDLMによる推定

$\bar{Y}_j$  投与群jの用量反応(群平均)

$j = 1$  : Placebo

$j = 2 \sim 8$  : 被験薬群1~7

観測方程式

$$\bar{Y}_j = \theta_j + v_j$$

$$v_j \sim N(0, \sigma^2)$$

システム方程式

$$\theta_j = \theta_{j-1} + \delta_{j-1} + \omega_j$$

$$\omega_j \sim N(0, W \sigma^2)$$

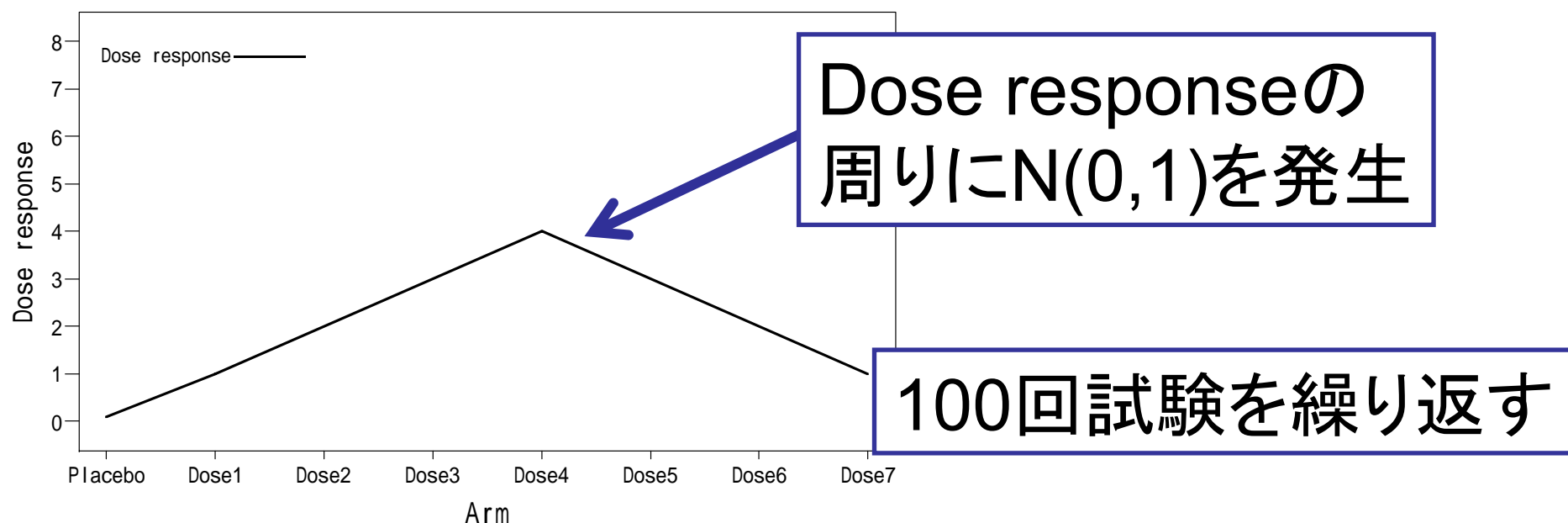
$$\delta_j = \delta_{j-1} + \varepsilon_j$$

$$\varepsilon_j \sim N(0, W \sigma^2)$$

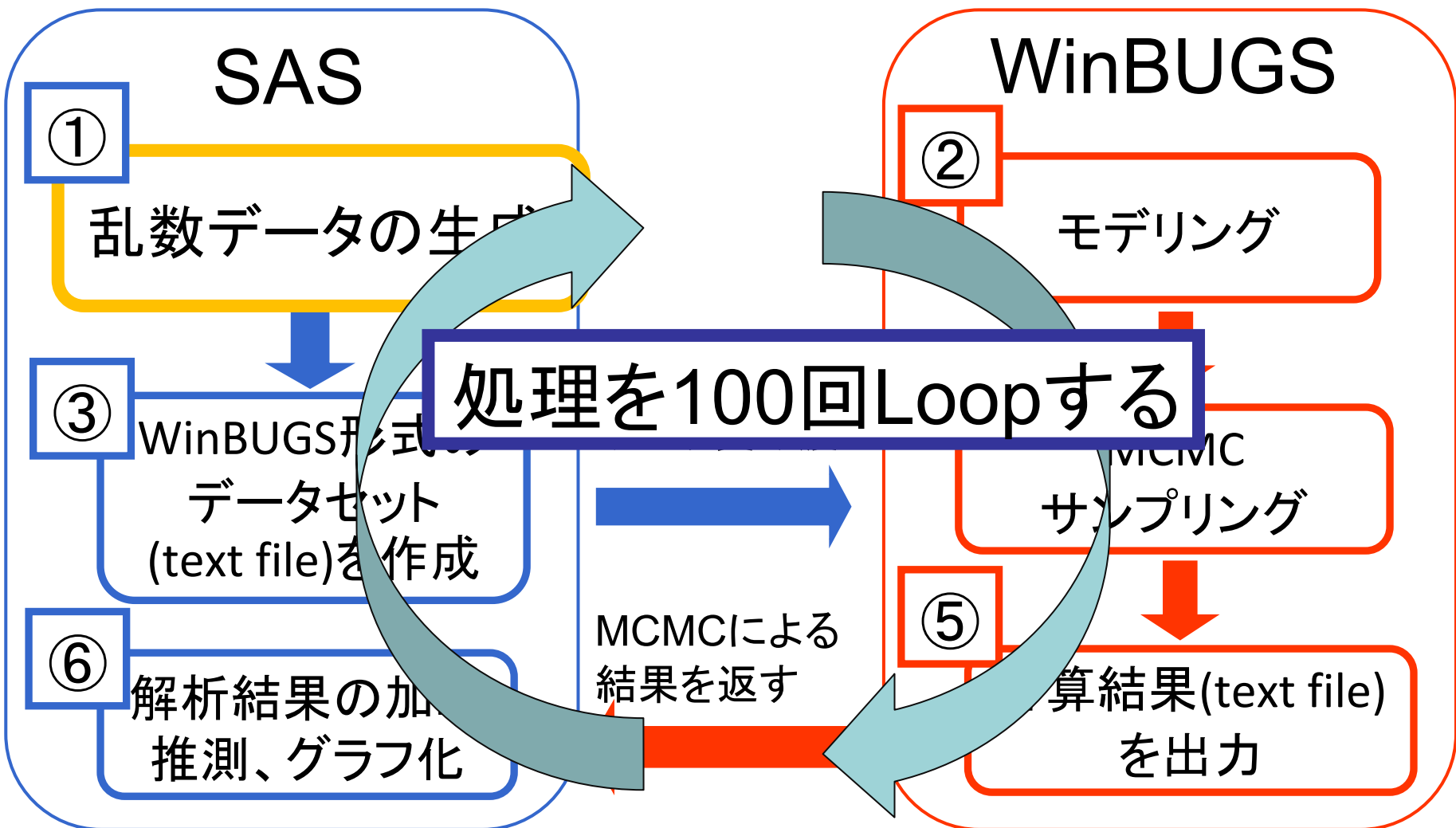
# SAS-WinBUGSによるモンテカルロシミュレーション

## Example 用量反応曲線のNDLMによる推定

### 乱数による解析データの発生



## SASを用いて解析処理を反復実行



# SAS-WinBUGSによるモンテカルロシミュレーション

## シミュレーションを100回実行すると

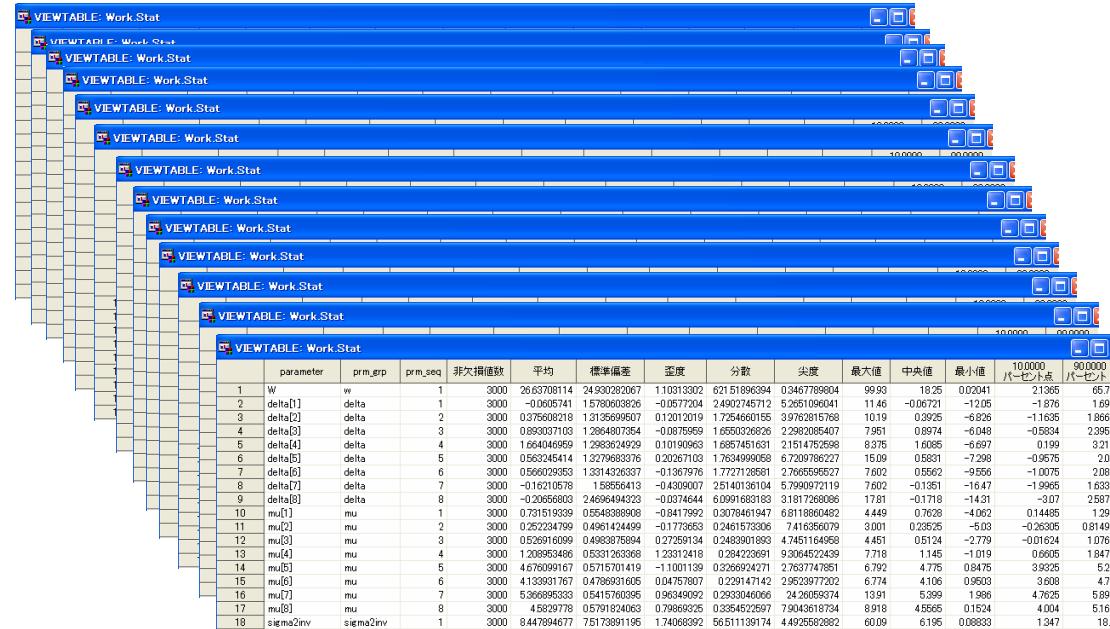
### 要約統計量データセット

VIEWTABLE: Work.Stat														
	parameter	prm_grp	prm_seq	非欠損値数	平均	標準偏差	歪度	分数	尖度	最大値	中央値	最小値	100000 パーセント点	900000 パーセント点
1	W	w	1	3000	26.63708114	24.930282067	1.10313302	621.51096394	0.3467789804	99.93	18.25	0.02841	21.365	65.73
2	delta[1]	delta	1	3000	-0.0605741	1.5780603826	-0.0577204	2.4902745712	5.2651096041	11.46	-0.06721	-12.05	-1.876	1.692
3	delta[2]	delta	2	3000	0.375608218	1.3135699507	0.12012019	1.7254660155	3.9762815768	10.19	0.3925	-6.826	-1.1635	1.8665
4	delta[3]	delta	3	3000	0.893037103	1.2864807354	-0.0875959	1.6550326826	2.2982085407	7.951	0.8974	-6.048	-0.5834	2.3955
5	delta[4]	delta	4	3000	1.664046959	1.2983624929	0.10190963	1.6857451631	2.1514752598	8.375	1.6085	-6.697	0.199	3.219
6	delta[5]	delta	5	3000	0.563245414	1.3279683376	0.20267103	1.7634990058	6.7209786227	15.09	0.5831	-7.298	-0.9575	2.04
7	delta[6]	delta	6	3000	0.566029353	1.3014526337	-0.1367976	1.7727128581	2.7665595627	7.602	0.5562	-9.556	-1.0075	2.082
8	delta[7]	delta	7	3000	-0.16210576	1.58555413	-0.4009007	2.5140138104	5.7989972119	7.602	-0.1351	-16.47	-1.9965	1.6355
9	delta[8]	delta	8	3000	-0.20656803	2.4696494323	-0.0374644	6.0091683193	3.1817268086	17.51	-0.1719	-14.31	-3.07	2.5875
10	mu[1]	mu	1	3000	0.731519339	0.5543388908	-0.8477992	0.3078461947	6.8118860482	4.449	0.7628	-4.062	0.14485	1.293
11	mu[2]	mu	2	3000	0.252234799	0.4961424499	-0.1773653	0.2461573306	7.416356079	3.001	0.23525	-5.03	-0.26305	0.81495
12	mu[3]	mu	3	3000	0.526916099	0.4983875894	0.27259134	0.2483901893	4.7451164958	4.451	0.5124	-2.779	-0.01624	1.0765
13	mu[4]	mu	4	3000	1.208953486	0.5331263368	1.23312418	0.284223691	9.3064522439	7.718	1.145	-1.019	0.6605	1.8475
14	mu[5]	mu	5	3000	4.676099167	0.5715701419	-1.1001139	0.3266924271	2.7637747851	6.792	4.775	0.8475	3.9325	5.27
15	mu[6]	mu	6	3000	4.133931767	0.4788931605	0.0475807	0.229147142	2.9523977202	6.774	4.106	0.9503	3.609	4.72
16	mu[7]	mu	7	3000	5.366895333	0.5415760395	0.96349092	0.2933040606	24.26055974	13.91	5.399	1.986	4.7625	5.894
17	mu[8]	mu	8	3000	4.5829778	0.5791824063	0.79869325	0.3354522697	7.9043618734	8.918	4.5565	0.1524	4.004	5.166
18	sigma2inv	sigma2inv	1	3000	8.447894677	7.5173891195	1.74068392	56.511139174	4.4925582882	60.09	6.195	0.08833	1.347	18.3

# SAS-WinBUGSによるモンテカルロシミュレーション

## シミュレーションを100回実行すると

### 要約統計量データセット



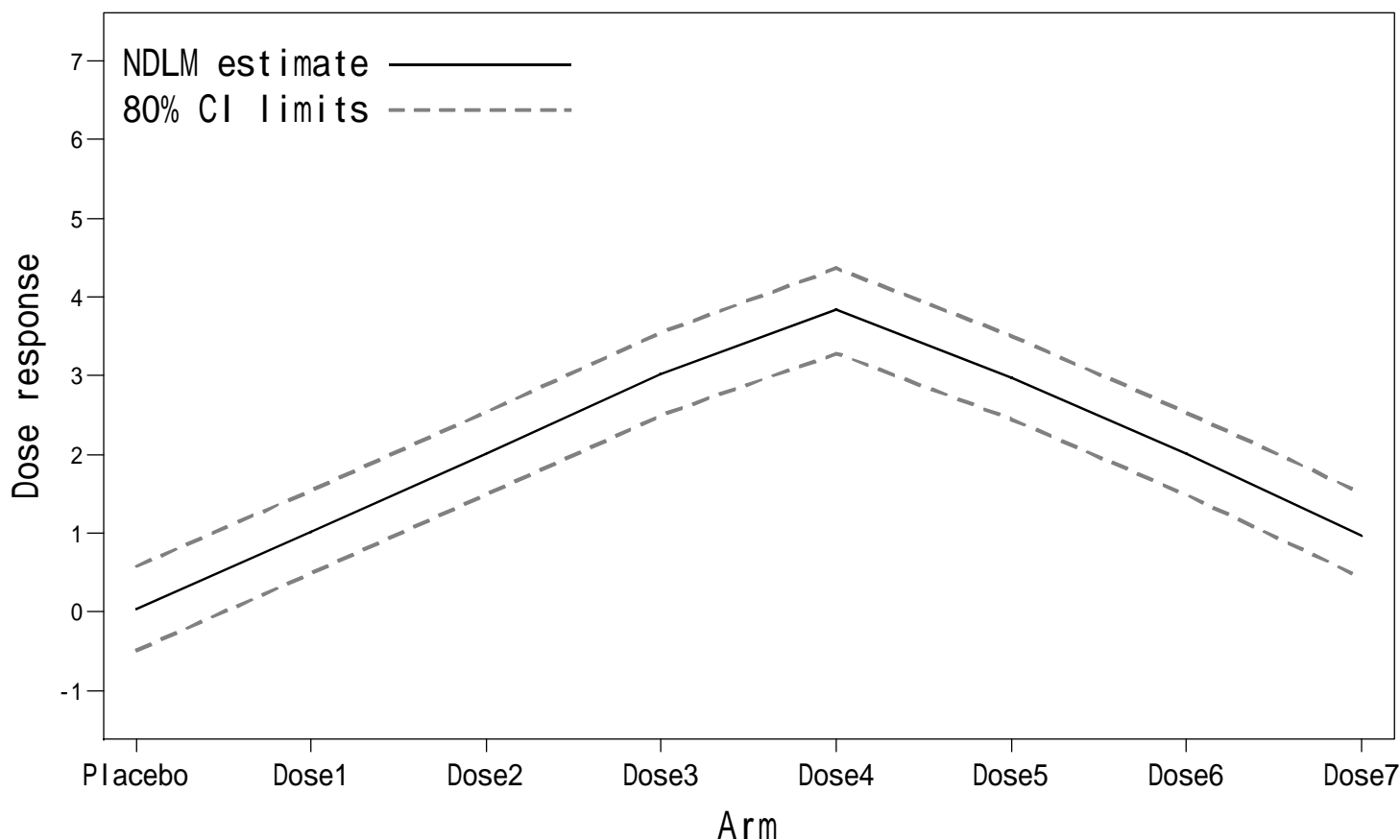
	parameter	prm_grp	prm_seq	非欠損値数	平均	標準偏差	歪度	分数	尖度	最大値	中央値	最小値	10,000パーセンタイル	90,000パーセンタイル
1	W	w	1	3000	26.63708114	24.930282067	1.10313302	621.51896394	0.3467789604	99.93	18.25	0.02041	2.1365	65.73
2	delta[1]	delta	1	3000	-0.0605741	1.5780603826	-0.0577204	2.4902745712	5.2651096041	11.46	-0.06721	-12.05	-1.876	1.692
3	delta[2]	delta	2	3000	0.375608218	1.3135699507	0.12012019	1.7254660155	3.9762815768	10.19	0.3925	-6.826	-1.1635	1.8665
4	delta[3]	delta	3	3000	0.893037103	1.2864807354	-0.0875969	1.6550326926	2.2982085407	7.951	0.8974	-0.048	-0.5834	2.3955
5	delta[4]	delta	4	3000	1.664046959	1.2983249229	0.10190683	1.6857451631	2.1514752598	8.375	1.6085	-6.697	0.199	3.219
6	delta[5]	delta	5	3000	0.553245414	1.3279883076	0.20267103	1.7634999058	6.7209786227	15.09	0.5501	-7.298	-0.9575	2.04
7	delta[6]	delta	6	3000	0.566029353	1.3314326337	-0.1367976	1.7727128581	2.7655565627	7.602	0.5562	-9.556	-1.0075	2.082
8	delta[7]	delta	7	3000	-0.16210578	1.58556413	-0.4309007	2.5140136104	5.7990972119	7.602	-0.1351	-16.47	-1.9965	1.6336
9	delta[8]	delta	8	3000	-0.206556803	2.4696494323	-0.0374644	6.0991683183	3.1817268086	17.81	-0.1718	-14.31	-3.07	2.5875
10	mu[1]	mu	1	3000	0.731519339	0.5548388908	-0.8417992	0.3078461947	6.8118860482	4.449	0.7628	-4.062	0.14485	1.293
11	mu[2]	mu	2	3000	0.252234799	0.4961424499	-0.1773653	0.2451573306	7.416356079	3.001	0.23525	-5.03	-0.26305	0.81495
12	mu[3]	mu	3	3000	0.526916059	0.4953875894	0.27259134	0.2483901893	4.7451164958	4.451	0.5124	-2.779	-0.01624	1.0765
13	mu[4]	mu	4	3000	1.238953486	0.5331263386	1.23312418	0.394722891	9.304522439	7.718	1.145	-1.019	0.6605	1.9475
14	mu[5]	mu	5	3000	4.676099167	0.5715701419	-1.1001139	0.3265924271	2.7637747851	6.792	4.775	0.8475	3.9325	5.27
15	mu[6]	mu	6	3000	4.133931767	0.4786931605	0.04757807	0.229147142	2.9523977202	6.774	4.106	0.9503	3.608	4.72
16	mu[7]	mu	7	3000	5.366895333	0.5415760395	0.96349092	0.2933040066	24.26059374	13.91	5.399	1.986	4.7625	5.994
17	mu[8]	mu	8	3000	4.5829778	0.5791824063	0.79869325	0.3354522597	7.9043618734	8.918	4.5565	0.1524	4.004	5.166
18	sigma2inv	sigma2inv	1	3000	8.447894677	7.5173891195	1.74068392	5.6511139174	4.4925582882	60.09	6.195	0.08833	1.347	18.3

100データセットが生成

1つ1つをモンテカルロ  
標本と考え、平均を計算

## モンテカルロシミュレーション結果の加工

事後分布と信用区間のモンテカルロ平均をグラフ化



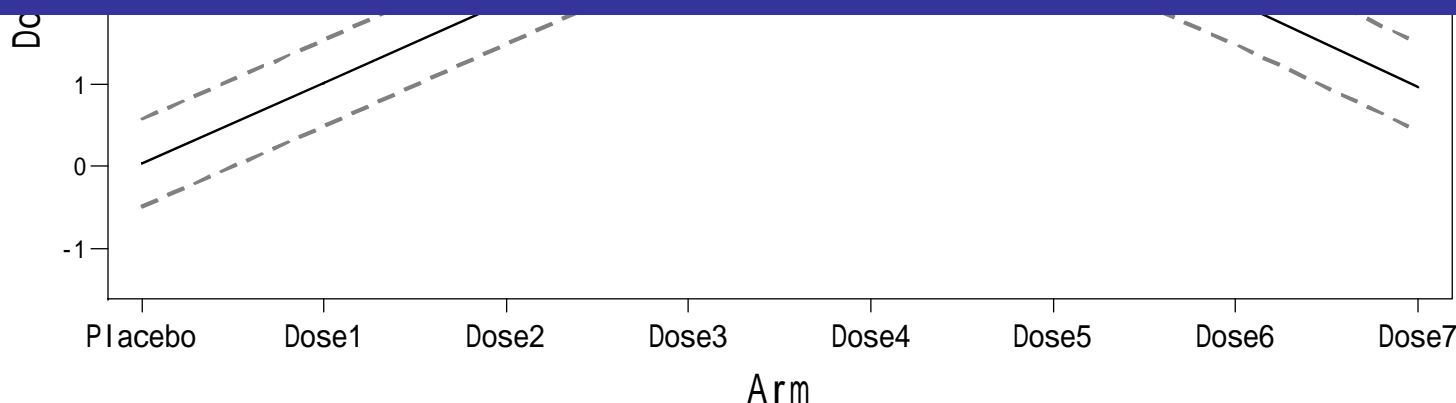


## モンテカルロシミュレーション結果の加工

事後分布と信用区間のモンテカルロ平均をグラフ化



SASからWinBUGSを制御し、実行することでモンテカルロシミュレーションも容易に



## 発表構成

- はじめに(背景)
- SAS-WinBUGSによるベイズ分析
  - ・データ解析
  - ・モンテカルロシミュレーション
- おわりに

## おわりに

本発表で紹介したプログラムを用いることで、WinBUGSを用いたベイズ分析においても、通常のSASによる解析と同様の手順で実行が可能



**ベイズモデルの解析でも通常の解析と同様に**

データハンドリングの必要なデータの解析  
複数のデータに対する解析  
層別解析などの同じ処理の繰り返しのSASマクロ化  
モンテカルロシミュレーション

## 参考文献

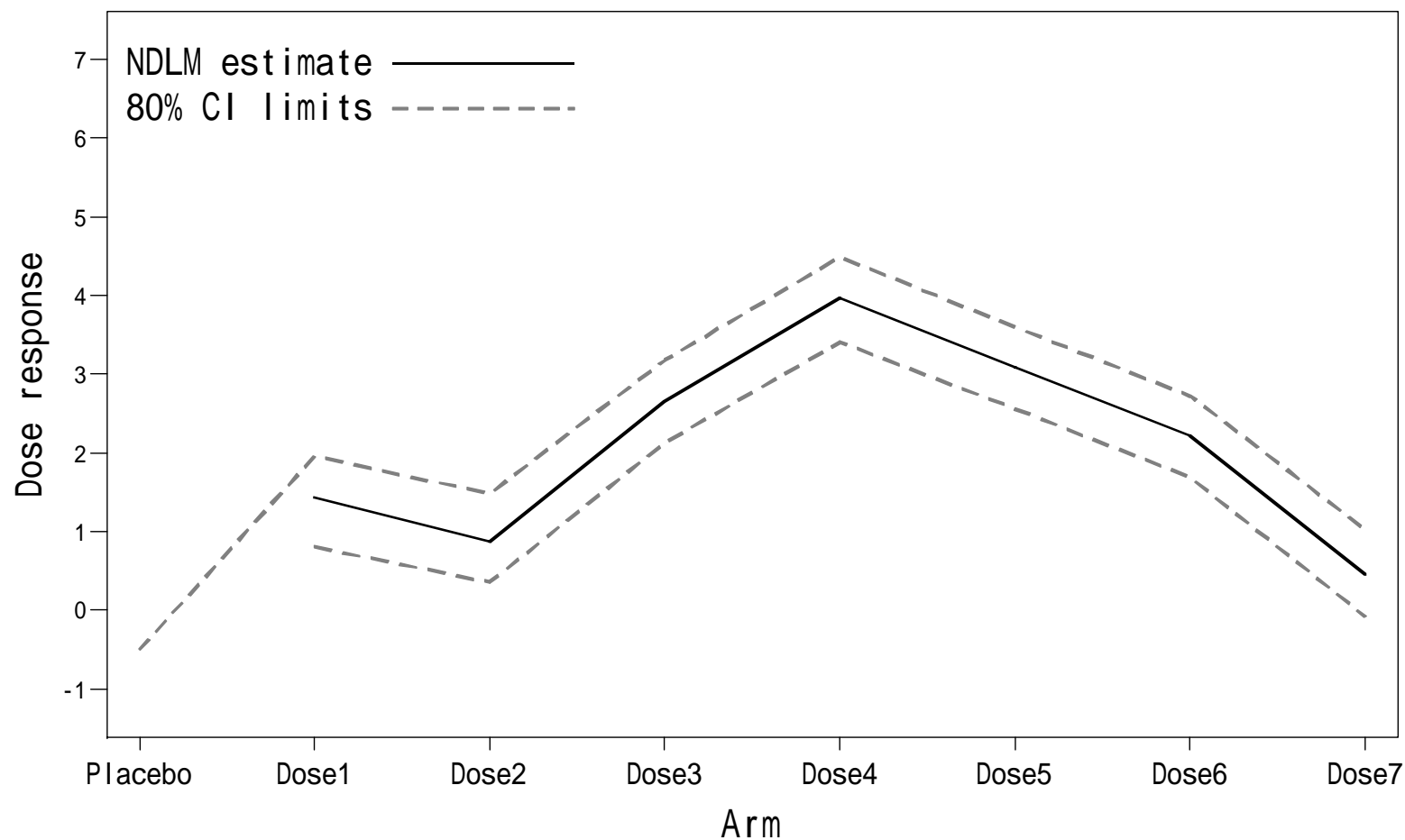
1. 中妻照雄. 入門ベイズ統計学. 朝倉書店; 2007
2. 統計数理研究所 2008年公開講座配布資料. マルコフ連鎖モンテカルロ法の基礎と実践.
3. Michael K. Smith, Ieuan Jones, Mark F. Morris, Andrew P. Grieve and Keith Tan, Implementation of a Bayesian adaptive design in a proof of concept study, Pharmaceut. Statist. 2006; 5: 39–50
4. Zhiyong Zhang , John J. McArdle , Lijuan Wang. Fumiaki Hamagami. A SAS Interface for Bayesian Analysis With WinBUGS. Structural Equation Modeling: A Multidisciplinary Journal. 2008;15:4,705 — 728
5. 伊庭幸人, 種村正美, 大森裕浩, 和合肇, 佐藤整尚, 高橋明彦. 計算統計Ⅱ マルコフ連鎖モンテカルロ法とその周辺. 岩波書店; 2005

## 参考文献

6. Berry DA, MullerP, Grieve AP, Smith MK, Parke T, Blazek R, Mitchard N, Krams M. Adaptive Bayesian designs for dose-ranging trials. In Case studies in Bayesian statistics V, Carlin B, Carriquiry A, Gatsonis C, Gelman A, Kass RE, Verdine I, West M (eds). Springer: Berlin, 2002; pp. 99–181.
7. Orgogozo and Gary A. Ford Michael Krams, Kennedy R. Lees, Werner Hacke, Andrew P. Grieve, Jean-Marc, Acute Stroke Therapy by Inhibition of Neutrophils (ASTIN) An Adaptive Dose-Response Study of UK-279,276 in Acute Ischemic Stroke, Stroke 2003;34;2543-2548;
8. Spiegelhalter D, Thomas A, Best N. WinBUGS version 1.4.1 User Manual. Medical Research Council Biostatistics Unit, Cambridge, <http://www.mrc-bsu.cam.ac.uk/bugs/WinBUGS/> 2004.
9. 簗谷千凰彦, 大津泰介, 清水玄彦, 中妻照雄, 高木康順, 隅田和人. 計量経済学のフロンティア. 慶応義塾大学出版会; 2006

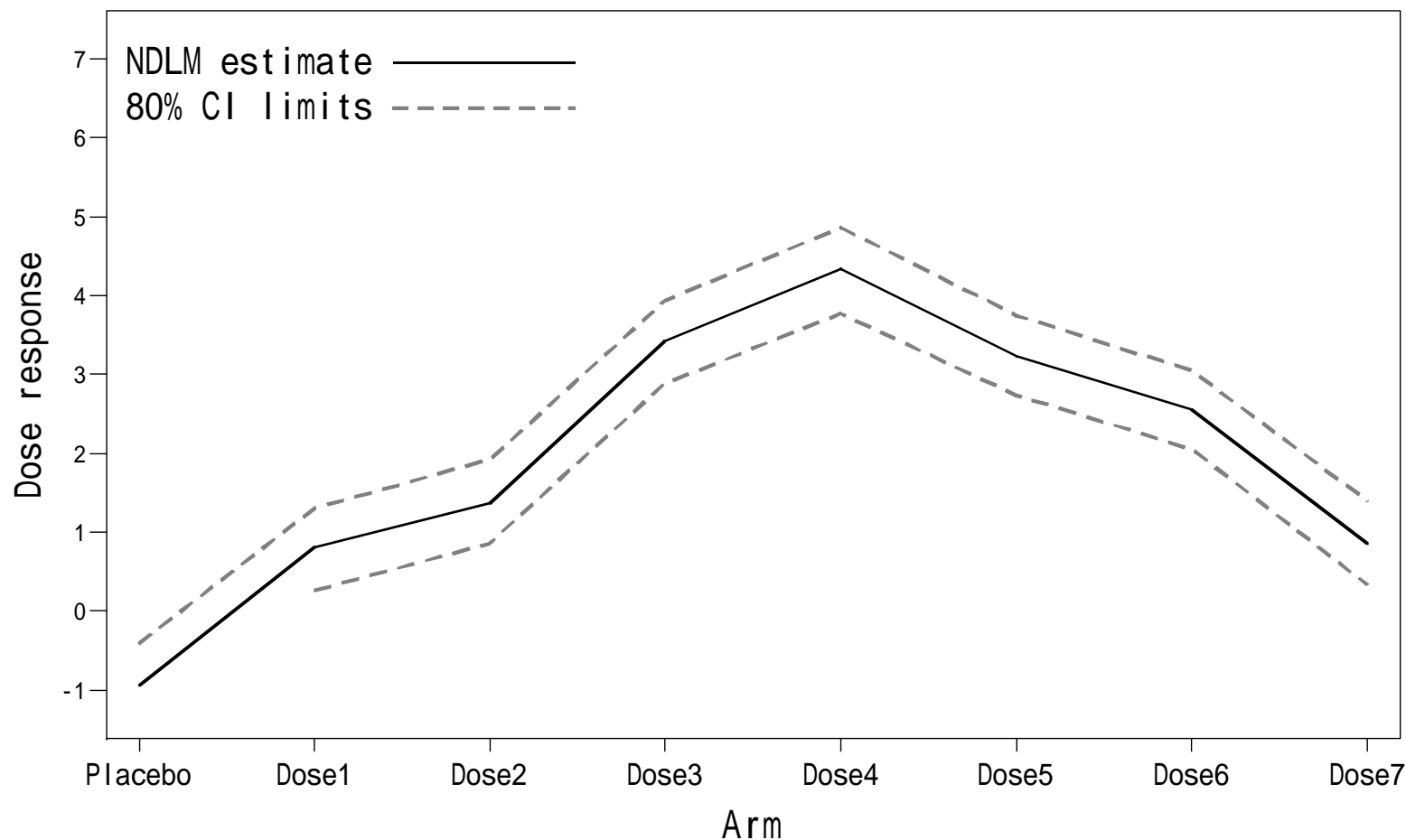
# Back up モンテカルロシミュレーションの回数

## 2回



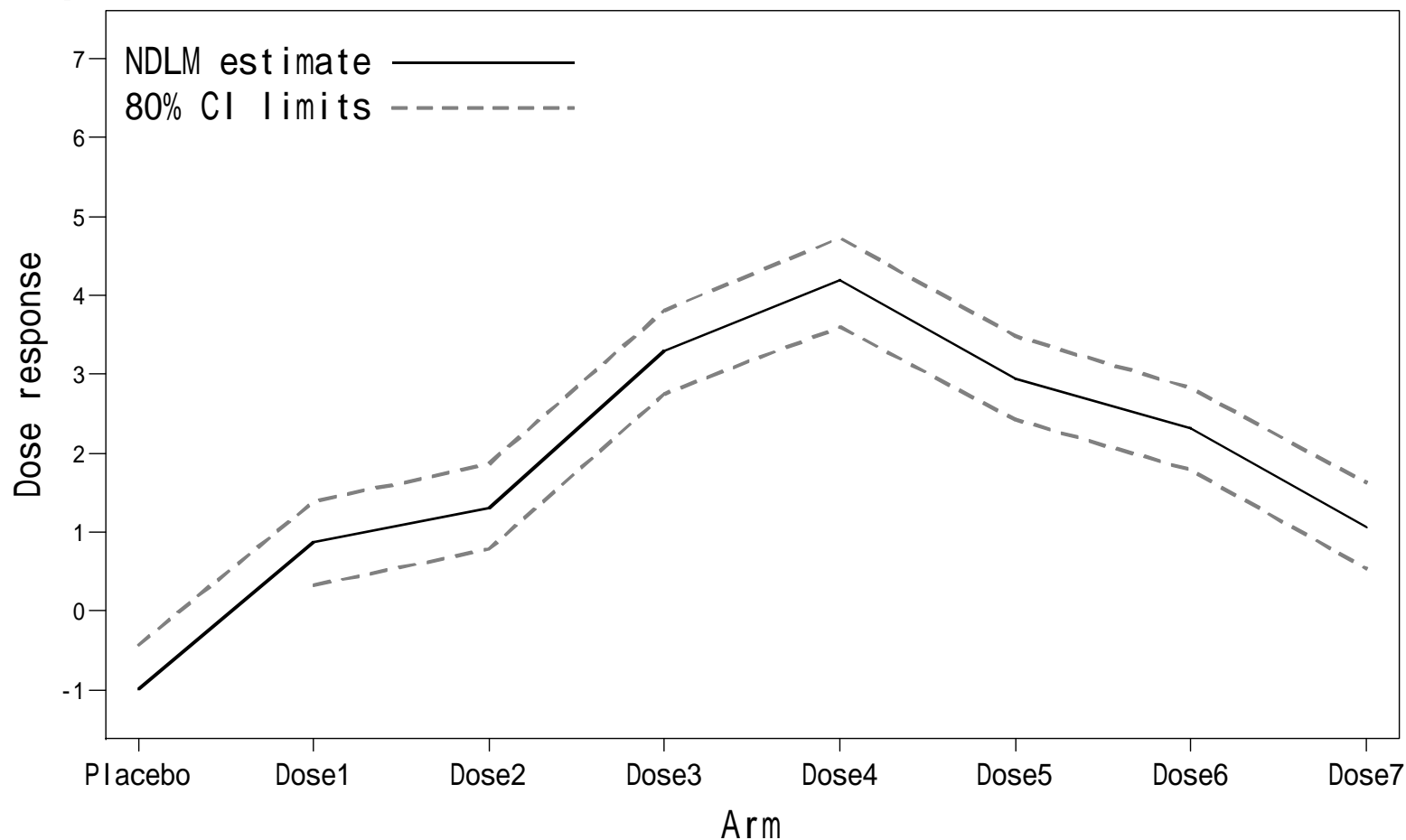
# Back up モンテカルロシミュレーションの回数

## 5回



# Back up モンテカルロシミュレーションの回数

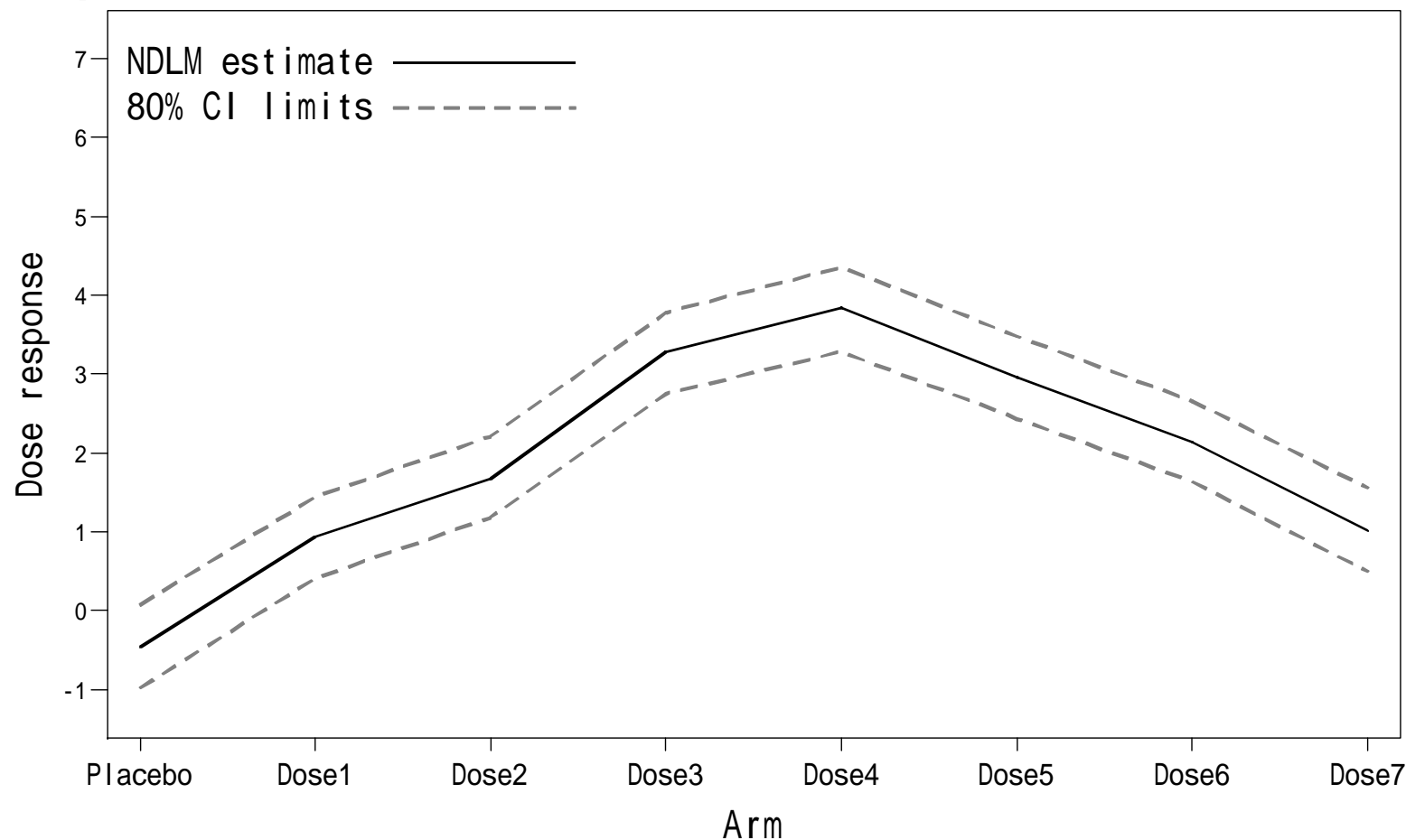
## 10回





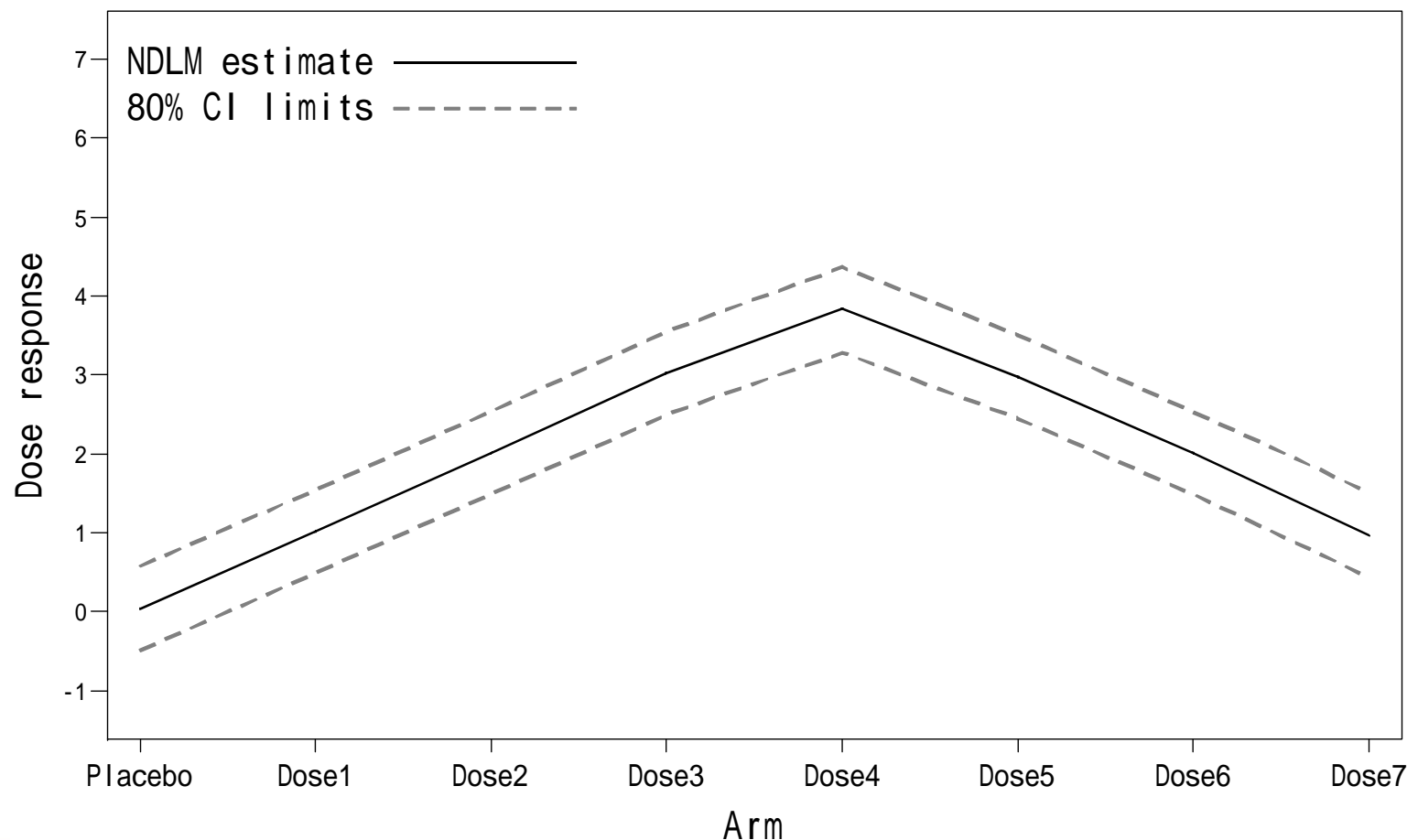
# Back up モンテカルロシミュレーションの回数

## 20回



# Back up モンテカルロシミュレーションの回数

## 100回



# Back up WinBUGSでの解析

Log

Node statistics

node	mean	sd	MC error	2.5%	median	97.5%	start	sample
W	26.64	24.93	0.4682	0.4994	18.26	90.04	1001	3000
delta[1]	-0.06057	1.578	0.0282	-3.074	-0.06641	2.954	1001	3000
delta[2]	0.3756	1.313	0.02478	-2.314	0.3939	2.962	1001	3000
delta[3]	0.893	1.286	0.02302	-1.694	0.8976	3.404	1001	3000
delta[4]	1.664	1.298	0.02327	-0.9324	1.609	4.343	1001	3000
delta[5]	0.5632	1.328	0.02423	-2.119	0.5835	3.321	1001	3000
delta[6]	0.566	1.331	0.02497	-2.132	0.5562	3.356	1001	3000
delta[7]	-0.1621	1.585	0.0296	-3.268	-0.1348	2.919	1001	3000
delta[8]	-0.2066	2.469	0.04666	-5.246	-0.1715	4.64	1001	3000
mu[1]	0.7315	0.5548	0.01041	-0.6016	0.7629	1.778	1001	3000
mu[2]	0.2522	0.4961	0.008694	-0.7136	0.2353	1.27	1001	3000
mu[3]	0.5269	0.4983	0.0091	-0.4565	0.5125	1.58	1001	3000
mu[4]	1.209	0.533	0.01158	0.2683	1.145	2.458	1001	3000
mu[5]	4.676	0.5715	0.01126	3.264	4.775	5.576	1001	3000
mu[6]	4.134	0.4786	0.009745	3.21	4.106	5.129	1001	3000
mu[7]	5.367	0.5415	0.008905	4.175	5.399	6.289	1001	3000
mu[8]	4.583	0.5791	0.01171	3.525	4.557	5.875	1001	3000
sigma2inv	8.448	7.516	0.1444	0.5654	6.203	27.88	1001	3000

**Sample Monitor Tool**

node  chains  to  percentiles

beg  end  thin

clear set trace history density

stats coda quantiles bgr diag auto cor

percentiles

2.5  
5  
10  
25  
median  
75  
90  
95  
97.5

# Back up SAS-WinBUGSでの解析

VIEWTABLE: Work.Stat														
	parameter	prm_grp	prm_seq	非欠損値数	平均	標準偏差	歪度	分散	尖度	最大値	中央値	最小値	10.0000 パーセント点	90.0000 パーセント
1	W	w	1	3000	26.63708114	24.930282067	1.10313302	621.51896394	0.3467789804	99.93	18.25	0.02041	2.1365	65.73
2	delta[1]	delta	1	3000	-0.0605741	1.5780603826	-0.0577204	2.4902745712	5.2651096041	11.46	-0.06721	-12.05	-1.876	1.692
3	delta[2]	delta	2	3000	0.375608218	1.3135699507	0.12012019	1.7254660155	3.9762815768	10.19	0.3925	-6.826	-1.1635	1.8665
4	delta[3]	delta	3	3000	0.893037103	1.2864807354	-0.0875959	1.6550326826	2.2982085407	7.951	0.8974	-6.048	-0.5834	2.3955
5	delta[4]	delta	4	3000	1.664046959	1.2983624929	0.10190963	1.6857451631	2.1514752598	8.375	1.6085	-6.697	0.199	3.219
6	delta[5]	delta	5	3000	0.563245414	1.3279683376	0.20267103	1.7634999058	6.7209786227	15.09	0.5831	-7.298	-0.9575	2.04
7	delta[6]	delta	6	3000	0.566029353	1.3314326337	-0.1367976	1.7727128581	2.7665595527	7.602	0.5562	-9.556	-1.0075	2.082
8	delta[7]	delta	7	3000	-0.16210578	1.58556413	-0.4309007	2.5140136104	5.7990972119	7.602	-0.1351	-16.47	-1.9965	1.6335
9	delta[8]	delta	8	3000	-0.20656803	2.4696494323	-0.0374644	6.0991683183	3.1817268086	17.81	-0.1718	-14.31	-3.07	2.5875
10	mu[1]	mu	1	3000	0.731519339	0.5548388908	-0.8417992	0.3078461947	6.8118860482	4.449	0.7628	-4.062	0.14485	1.293
11	mu[2]	mu	2	3000	0.252234799	0.4961424499	-0.1773653	0.2461573306	7.416356079	3.001	0.23525	-5.03	-0.26305	0.81495
12	mu[3]	mu	3	3000	0.526916099	0.4983875894	0.27259134	0.2483901893	4.7451164958	4.451	0.5124	-2.779	-0.01624	1.0765
13	mu[4]	mu	4	3000	1.208953486	0.5331263368	1.23312418	0.284223691	9.3064522439	7.718	1.145	-1.019	0.6605	1.8475
14	mu[5]	mu	5	3000	4.676099167	0.5715701419	-1.1001139	0.3266924271	2.7637747851	6.792	4.775	0.8475	3.9325	5.27
15	mu[6]	mu	6	3000	4.133931767	0.4786931605	0.04757807	0.229147142	2.9523977202	6.774	4.106	0.9503	3.608	4.72
16	mu[7]	mu	7	3000	5.366895333	0.5415760395	0.96349092	0.2933046066	24.26059374	13.91	5.399	1.986	4.7625	5.894
17	mu[8]	mu	8	3000	4.5829778	0.5791824063	0.79869325	0.3354522597	7.9043618734	8.918	4.5565	0.1524	4.004	5.166
18	sigma2inv	sigma2inv	1	3000	8.447894677	7.5173891195	1.74068392	56.511139174	4.4925582882	60.09	6.195	0.08833	1.347	18.3

## Back up モンテカルロシミュレーション 実行時間

### 使用PCスペック

OS:Windows XP 物理メモリ:2GB CPU: Intel Core 2 Duo 2.13G

**MC100回で・・・15分くらい**

**モデルの複雑さ、Burn inの数、サンプリング数、  
サンプリング間隔などにも依存**