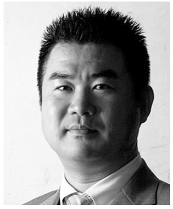
空間の統計学(11)：ベイズ空間計量経済学③

慶應義塾大学総合政策学部准教授
古谷　知之（Furutani Tomoyuki）
■兵庫県生まれ。2001年東京大学大学院工学系研究科博士課程修了。博士（工学）。東京大学大学院助手、慶應義塾大学環境情報学部専任講師を経て、07年４月より現職。専門分野：空間統計学、都市交通計画、観光政策。



1. はじめに

空間計量経済モデルの中には、空間上で離散的に発生するイベントを説明したり、表現したりするモデルがあります。例えば、交通事故や人の死亡などのカウント・データや、目的地選択行動や選挙投票行動などの行動データを扱うモデルです。データの性質上、多くのサンプル数を集めることが困難であるために、政策的な分析や意志決定を行う際には、少数サンプルでもモデル推定が必要とされる場合があります。また、集団間或いは地域間の差異が顕著である場合には、個人・グループや地域の異質性を考慮したモデリングが要求されます。今回と次回は、こうしたカウント・データや離散選択行動データのベイジアン・モデリングについて紹介します。今回は、ロジットモデルに代表される離散選択モデル、とりわけ、３つ以上の選択肢に対する選択行動データを前提に、多項ロジットモデル、ネスティッドロジットモデル、Mixedロジットモデルを扱います。なお、カテゴリカルデータを使ったモデリングについては、文献[1]などにも詳しいです。

演習には、離散選択問題の代表例とも言える交通手段選択行動データを用いることにします。Rのmlogitパッケージで提供される交通手段選択データ「Mode」に、選択肢利用可能性などの項目を任意に追加しました。Modeデータを使って、「車（運転）」、「車（同乗）」、「鉄道」、「バス」の４種類の交通手段選択行動を、移動費用と所要時間とで説明する離散選択モデルを推定することができます。

Rでは、上述のmlogitパッケージを用いて、（条件付き）多項ロジットモデル、ネスティッドロジットモデル、Mixedロジットモデルを推定できます。mclogitパッケージを用いても、多項ロジットモデルやMixedロジットモデルを推定できます。また、**optim( )**関数を用いて、モデル独自の効用関数を直接定義し、モデル推定を行うことができます。多項ロジットモデルのベイズ推定を行うには、パッケージMCMCpackを用いてもできますし、JAGSやWinBUGSのコードを書いてR2jagsやR2WinBUGSなどの

パッケージを用いてもできます。

本稿では、誌面の都合上、最尤法を使った推定方法についてはmlogitを使った方法を紹介します。対数尤度関数を記述しoptim( )関数を使って解く方法は、筆者のHP（http://web.sfc.keio.ac.jp/~maunz/wiki/）などを参照してください。ベイズ法を使った推定方法については、JAGSコードを記述し、R2jagsパッケージを使って計算する方法を紹介します。

2. 離散選択モデルの最尤推定

⑴　多項ロジットモデル

多項ロジットモデルは、3種類以上の異なる選択肢から1つの選択肢を選ぶような選択行動を表現するモデルとして使われます。演習用データを例に挙げると、「車（運転）」、「車（同乗）」、「鉄道」、「バス」の4選択肢から1つを選ぶような場合です。各選択肢に対して、選択効用関数を定式化します。顕示選好データを用いる場合、実際に選択した場合の選択肢については、顕在化した行動として観測されますが、選択しなかった選択肢については観測されないため、選択しなかった選択肢についての効用関数も定式化すると共に、効用関数を規定する変数のデータも示すことが必要となります。

個人$n(=1, \cdots, N)$の交通手段選択肢$i(=1, 2, 3, 4)$に対する選択効用U_{in}を、次式のように表します。ここで、変数x_{in1}は所要時間、x_{in2}は費用、β_0、β_1、β_2は未知パラメータ、ε_{in}は誤差項を意味します。

$$U_{in} = \beta_0 + \beta_1 x_{in1} + \beta_2 x_{in2} + \varepsilon_{in}$$

このうち、$V_{in} = \beta_0 + \beta_1 x_{in1} + \beta_2 x_{in2}$を確定項と呼びます。多項選択モデルの場合、基準となる選択肢の定数項＝0となります。この離散選択モデルは、ε_{in}が正規分布の時にプロビットモデル、ε_{in}がガンベル分布の時にロジットモデルとなります。このとき、個人nにとっての交通手段選択iの選択確率は、次式のようになります。

$$P_{in} = \frac{\exp(V_{in})}{\sum_{i=1}^{4} \exp(V_{in})}$$

mlogitパッケージを使うと、以下のようにして多項ロジットモデルを最尤推定できます。データを読み込んだ後、**mlogit.data**( )関数を使って、mlogit分析用のデータフォーマットに変換し、**mlogit**( )関数を使ってロジットモデルを推定します。

```
library(mlogit)
data <- read.table("modechoice.csv",
sep=",", header=T)
TM.data <- mlogit.data(data,
varying=c(9:16), shape="wide",
choice="choice", alt.levels=
c("car", "carpool", "rail", "bus"),
alt.var="mode")
nml.out <- mlogit(choice~cost+time,
data=TM.data)
summary(nml.out)
```

この結果は、以下のようになります。

多項ロジットモデルの推定結果（最尤推定）

```
> nml.out <- mlogit(choice~cost+time,data=TM.data)
> summary(nml.out)

Call:
mlogit(formula = choice ~ cost + time, data = TM.data, method = "nr",
    print.level = 0)

Frequencies of alternatives:
    car carpool    rail     bus
0.17881 0.48124 0.07064 0.26932

nr method
5 iterations, 0h:0m:0s
g'(-H)^-1g = 6.07E-07
optimum reached

Coefficients :
            Estimate Std. Error  t-value  Pr(>|t|)
altcarpool  3.2924661  0.3172767  10.3773 < 2.2e-16 ***
altrail    -0.9051585  0.2459427  -3.6804 0.0002329 ***
altbus      0.6277690  0.1633612   3.8428 0.0001216 ***
cost       -0.7723478  0.0919795  -8.3970 < 2.2e-16 ***
time       -0.0853574  0.0077484 -11.0161 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Log-Likelihood: -354.45
McFadden R^2:  0.34811
Likelihood ratio test : chisq = 378.56 (p.value=< 2.22e-16)
```

⑵　ネスティッドロジットモデル

離散選択モデルを扱う場合、分析の単位となる個人にとって、2つの選択肢に関する選択肢の比率が、他のいかなる選択肢の効用の確定項から影響を受けない、という性質があり、これをIIA（Independence from Irrelevant Alternatives）特性と呼びます。しかし、選択肢が増えると、選択肢の特徴に類似性が高い選択肢の選択確率を過大に評価し、それ以外の選択肢については選択確率を過小に評価する傾向が生じます。こうした問題に対処するため、選択構造を段階的にするネスティッドロジットモデルや、選択肢間の類似性を明示できるmixedロジットモデルが使われています。

演習用データで用いている交通手段選択行動をネスティッドロジットモデルで表すとすると、車で行くか公共交通で行くかを決めた後、車なら「車（運転）」か「車（同乗）」、公共交通なら「鉄道」か「バス」かを、段階的に選択するような選択構造が考えられます。このとき、車と公共交通を選択する上位の選択効用と、各交通手段を選択する下位の選択効用を、以下のように表せます。ここで、上位の選択肢を$j(=1,2)$とします。

上位：　$V_{jn} = \lambda_{1n}\Lambda_{1n} + \lambda_{2n}\Lambda_{2n}$

$$\Lambda_{1n} = \ln\sum_{i=1}^{2} \exp(V_{in})$$

$$\Lambda_{2n} = \ln\sum_{i=3}^{4} \exp(V_{in})$$

下位：　$V_{in} = \beta_0 + \beta_1 x_{in1} + \beta_2 x_{in2}$

Λ_{1n}やΛ_{2n}をログサム項または合成効用と呼びます。スケールパラメータと呼ばれるλ_{1n}・λ_{2n}は値が0から1の範囲の時、想定した段階構成が適切であると考え、0以下または1以上の場合、段階構成を再検討する必要があると考えます。

mlogit( )関数の中で、nestsオプションを設定することにより、ネスティッドロジットモデルを推定できます。

```
nl.out <- mlogit(choice˜cost+time,
TM.data, shape="long", method="bfgs",
nests=list(public=c("rail","bus"),
private=c("car","carpool")),
alt.levels=c("car","carpool",
"rail","bus"))
summary(nl.out)
```

推定結果は、以下のようになります。

ネスティッドロジットモデルの推定結果（最尤推定）

```
> summary(nl.out)

Call:
mlogit(formula = choice ~ cost + time, data = TM.data, nests = list
(public = c("rail",
    "bus"), private = c("car", "carpool")), shape = "long", method =
"bfgs",
    alt.levels = c("car", "carpool", "rail", "bus"))

Frequencies of alternatives:
    car carpool    rail     bus
0.17881 0.48124 0.07064 0.26932

bfgs method
10 iterations, 0h:0m:0s
g'(-H)^-1g = 5.59E-08
optimum reached

Coefficients :
                Estimate Std. Error t-value  Pr(>|t|)
altcarpool      3.413966   0.439369  7.7702 7.772e-15 ***
altrail        -1.887281   0.661711 -2.8521  0.004343 **
altbus          0.499683   0.213501  2.3404  0.019262 *
cost           -0.806950   0.120310 -6.7072 1.983e-11 ***
time           -0.095322   0.011151 -8.5485 < 2.2e-16 ***
lambda.public   1.821938   0.461174  3.9507 7.794e-05 ***
lambda.private  0.995307   0.168501  5.9068 3.487e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Log-Likelihood: -350.76
McFadden R^2:  0.35491
Likelihood ratio test : chisq = 385.95 (p.value=< 2.22e-16)
```

⑶　Mixedロジットモデル

選択肢間の類似性を考慮するために、選択肢特性ベクトルz_{in}と確率変数ベクトルμ（平均0の確率分布に従う）とを用いて、効用関数を次式のように表します。

$$U_{in} = V_{in} + [\mu z_{in} + \varepsilon_{in}]$$

Rでは、以下のようにして推定します。

```
mixed.out <- mlogit(
choice˜cost+time¦gend,data=TM.data,
rpar=c(cost='n',time='n'),
correlation=TRUE, R=10, tol=10)
summary(mixed.out)
```

推定結果は、以下のようになります。

Mixedロジットモデルの推定結果（最尤推定）

```
> summary(mixed.out)

Call:
mlogit(formula = choice ~ cost + time, data = TM.data, rpar = c(cost =
"n",
    time = "n"), R = 10, correlation = TRUE, tol = 10)

Frequencies of alternatives:
    car carpool    rail     bus
0.17881 0.48124 0.07064 0.26932

bfgs method
3 iterations, 0h:0m:0s
g'(-H)^-1g =  4.38
optimum reached

Coefficients :
            Estimate Std. Error t-value  Pr(>|t|)
altcarpool  3.395534   0.370257  9.1708 < 2.2e-16 ***
altrail    -0.837322   0.258609 -3.2378  0.001205 **
altbus      0.657207   0.167174  3.9313  8.45e-05 ***
cost       -0.777664   0.260481 -2.9855  0.002831 **
time       -0.048407   0.037917 -1.2766  0.201728
cost.cost   0.038845   1.020443  0.0381  0.969635
cost.time   0.011413   0.329491  0.0346  0.972367
time.time   0.042860   0.093188  0.4599  0.645564
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Log-Likelihood: -354.75
McFadden R^2:  0.34756
Likelihood ratio test : chisq = 377.97 (p.value=< 2.22e-16)

random coefficients
     Min..Min.     1st Qu.      Median        Mean     3rd Qu.
cost      -Inf -0.80386435 -0.77766406 -0.77766406 -0.75146378
time      -Inf -0.07832317 -0.04840695 -0.04840695 -0.01849074
     Max..Max.
cost       Inf
time       Inf
```

3. 離散選択モデルのベイズ推定

⑴ 多項ロジットモデル

JAGSやWinBUGSを用いて離散選択モデルを推定する場合、尤度関数を直接記述する必要があります。多項ロジットモデルのJAGSコード（mnl.txt）は、次のようになります。JAGSのコードは、文献[2]を参考にしました。選択肢利用可能性を、availという変数（０、１の２値）で表しています。

mnl.txt

```
model {
L <- sum(LL[,])
for (j in 1:np) {
beta[j] ˜ dnorm(0,0.001) }
for (i in 1:hh) {
y[i,1:nc] ˜ dmulti(p[i,1:nc],1)
Lkd[i] <- exp(sum(LL[i,1:nc]))
for (k in 1:nc) {
LL[i,k] <- y[i,k]*log(p[i,k])*avail[i,k]
phi1[i,k] <- exp(beta[1]*time[i,k]+
beta[2]*cost[i,k]+ beta[3]*cons2[i,k]+
beta[4]*cons3[i,k]+ beta[5]*cons4[i,k])
phi2[i,k] <- phi1[i,k]*avail[i,k]
p[i,k]  <- phi1[i,k] / sum(phi2[i,])
}}}
```

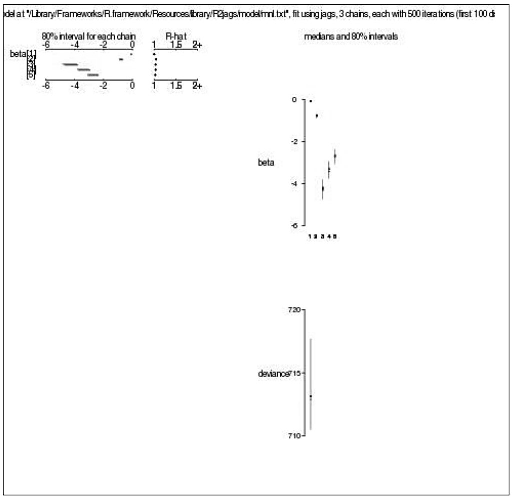
R2jagsを使って、モデルを推定します。

```
library(R2jags)
data <- read.table("modechoice.csv",
sep=",", header=T) summary(data)
hh<-nrow(data) # サンプル数
nc<-4 # 選択肢数
np <- 5 # パラメータ数
y <- as.matrix(data[,3:6])
time <- as.matrix(data[,13:16])
cost <- as.matrix(data[,9:12])
cons2 <- matrix(0,nrow=hh,ncol=nc)
cons2[,2] <- 1
cons3 <- matrix(0,nrow=hh,ncol=nc)
cons3[,3] <- 1
cons4 <- matrix(0,nrow=hh,ncol=nc)
cons4[,4] <- 1
avail <- as.matrix(data[,18:21])
# 初期値
in1 <- list(beta=c(0,0,0,0,0))
in2 <- list(beta=c(1,1,1,1,1))
in3 <- list(beta=c(-1,-1,-1,-1,-1))
beta<-c(0,0,0,0,0)
Data <- list("hh","nc","np","y", "time",
"cost","cons2","cons3","cons4","avail")
inits <- list(in1, in2, in3)
parameters <- c("beta")
model.file <- system.file(
package="R2jags", "model", "mnl.txt")
mnl.jags <- jags(data=Data, inits=inits,
```

```
n.iter=500, parameters, n.burnin=100,
n.chains=3, model.file=model.file)
print(mnl.jags, digits=3)
```

推定結果は、以下のようになります（図１）。

図１　多項ロジットモデルの推定結果（ベイズ推定）



mnl.txtの推定結果

```
> print(mnl.jags, digits=3)
Inference for Bugs model at "/Library/Frameworks/R.framework/Resources/
library/R2jags/model/mnl.txt", fit using jags,
 3 chains, each with 500 iterations (first 100 discarded)
 n.sims = 1200 iterations saved
            mean    sd    2.5%     25%     50%     75%   97.5%
beta[1]   -0.086 0.008  -0.101  -0.090  -0.086  -0.081  -0.071
beta[2]   -0.785 0.089  -0.966  -0.843  -0.777  -0.725  -0.627
beta[3]   -4.254 0.375  -4.990  -4.504  -4.247  -3.993  -3.583
beta[4]   -3.337 0.316  -3.998  -3.539  -3.320  -3.126  -2.747
beta[5]   -2.706 0.281  -3.295  -2.885  -2.689  -2.517  -2.184
deviance 713.710 2.971 709.745 711.527 713.066 715.316 720.963
          Rhat n.eff
beta[1]  1.002  1100
beta[2]  1.043    51
beta[3]  1.033    68
beta[4]  1.034    63
beta[5]  1.026    85
deviance 1.010   500

For each parameter, n.eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor (at convergence, Rhat=1).

DIC info (using the rule, pD = var(deviance)/2)
pD = 4.4 and DIC = 718.1
DIC is an estimate of expected predictive error (lower deviance is
better).
```

⑵　ネスティッドロジットモデル

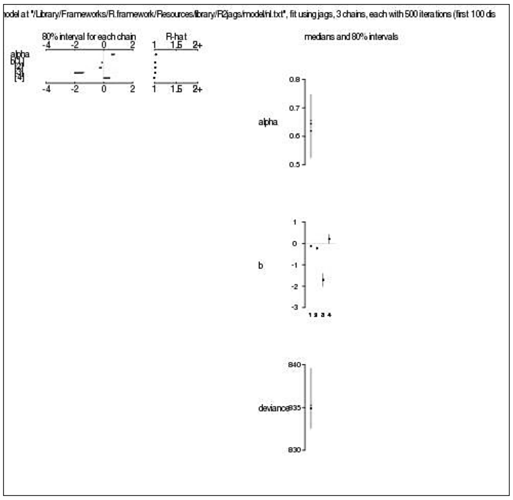
ネスティッドロジットモデルのJAGSコード（nl.txt）は、次のようになります。ここでは、下位の階層については、選択肢のいずれかが選択可能な場合を想定して、尤度関数を記述しています。

nl.txt

```
model {
for (i in 1:hh) {
for (k in 1:nc) {
LL[i,k] <- y[i,k]*log(p[i,k])*avail[i,k]
}}
L <- sum(LL[,])
for (i in 1:hh) {
for (k in 1:nc) {
phi1[i,k] <-
exp(time[i,k]*b[1]+cost[i,k]*b[2]+
cons2[i,k]*b[3]+cons4[i,k]*b[4])
phi2[i,k] <- phi1[i,k]*avail[i,k]
}}
for (i in 1:hh) {
y[i,1:nc] ~ dmulti(p[i,1:nc],1)
l1[i] <- log(sum(phi1[i,1:2]))
l2[i] <- log(sum(phi1[i,3:4]))
p[i,1] <-
(exp(alpha*l1[i])/(exp(alpha*l1[i])+
exp(alpha*l2[i])))*
(phi1[i,1]/sum(phi2[i,1:2]))
p[i,2] <-
(exp(alpha*l1[i])/(exp(alpha*l1[i])+
exp(alpha*l2[i])))*
(phi1[i,2]/sum(phi2[i,1:2]))
p[i,3] <-
(exp(alpha*l2[i])/(exp(alpha*l1[i])+
exp(alpha*l2[i])))*
(phi1[i,3]/sum(phi2[i,3:4]))
p[i,4] <-
(exp(alpha*l2[i])/(exp(alpha*l1[i])+
exp(alpha*l2[i])))*
(phi1[i,4]/sum(phi2[i,3:4])) }
for (j in 1:np) {
b[j] ~ dnorm(0,0.001)
beta[j] <- b[j]*alpha }
alpha ~ dexp(1) }
```

誌面の都合上、Rコードは省略しますが、推定結果は次のようになります（図２）。

図２　ネスティッドロジットモデルの推定結果（ベイズ推定）



nl.txtの推定結果

```
> print(nl.jags, digits=3)
Inference for Bugs model at "/Library/Frameworks/R.framework/Resources/
library/R2jags/model/nl.txt", fit using jags,
 3 chains, each with 500 iterations (first 100 discarded)
 n.sims = 1200 iterations saved
            mean    sd    2.5%     25%     50%     75%   97.5%
alpha      0.638 0.088   0.482   0.579   0.636   0.691   0.830
b[1]      -0.119 0.012  -0.144  -0.126  -0.119  -0.111  -0.097
b[2]      -0.227 0.060  -0.359  -0.263  -0.225  -0.184  -0.122
b[3]      -1.706 0.244  -2.249  -1.855  -1.692  -1.533  -1.287
b[4]       0.206 0.173  -0.133   0.094   0.211   0.322   0.540
deviance 835.619 2.904 831.779 833.474 835.024 837.091 842.765
          Rhat n.eff
alpha    1.034    83
b[1]     1.013   220
b[2]     1.025    84
b[3]     1.016   130
b[4]     1.001  1200
deviance 1.001  1200

For each parameter, n.eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor (at convergence, Rhat=1).

DIC info (using the rule, pD = var(deviance)/2)
pD = 4.2 and DIC = 839.8
DIC is an estimate of expected predictive error (lower deviance is
better).
```

⑶　Mixedロジットモデル

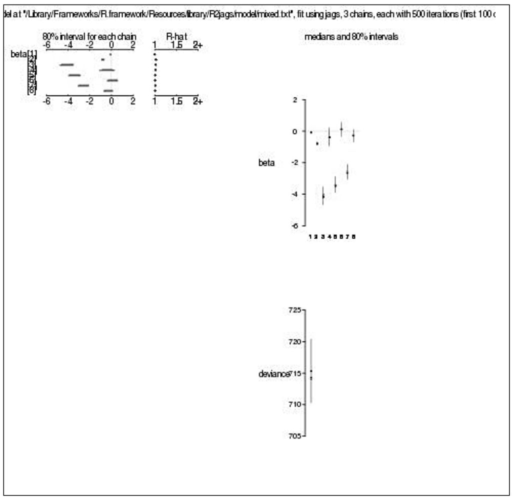
mixedロジットモデルのJAGSコードは、次のようになります。

mixed.txt

```
model {
L <- sum(LL[,])
for (j in 1:np) {
beta[j] ˜ dnorm(0,0.001) }
for (i in 1:hh) {
y[i,1:nc] ˜ dmulti( p[i,1:nc] , 1 )
InvL[i] <- 1/exp(sum(LL[i,]))
for (k in 1:nc) {
LL[i,k] <- y[i,k]*log(p[i,k])*avail[i,k]
phi1[i,k] <- exp(beta[1]*time[i,k]+
beta[2]*cost[i,k]+ beta[3]*cons2[i,k]+
beta[4]*age2[i,k]+beta[5]*cons3[i,k]+
beta[6]*age3[i,k]+ beta[7]*cons4[i,k]+
beta[8]*age4[i,k])
phi2[i,k] <- phi1[i,k]*avail[i,k]
p[i,k]  <- phi1[i,k] / sum(phi2[i,])
}}}
```

モデルの推定結果は、以下のようになります(図３)。

図３　Mixedロジットモデルの推定結果（ベイズ推定）



mixed.txtの推定結果

```
> print(mixed.jags, digits=3)
Inference for Bugs model at "/Library/Frameworks/R.framework/Resources/
library/R2jags/model/mixed.txt", fit using jags,
 3 chains, each with 500 iterations (first 100 discarded)
 n.sims = 1200 iterations saved
            mean    sd    2.5%     25%     50%     75%   97.5%
beta[1]   -0.088 0.008  -0.104  -0.093  -0.087  -0.082  -0.073
beta[2]   -0.791 0.090  -0.957  -0.858  -0.788  -0.728  -0.615
beta[3]   -4.105 0.441  -4.908  -4.444  -4.099  -3.778  -3.279
beta[4]   -0.388 0.459  -1.328  -0.684  -0.380  -0.089   0.491
beta[5]   -3.411 0.384  -4.095  -3.690  -3.427  -3.122  -2.670
beta[6]    0.112 0.358  -0.596  -0.127   0.115   0.357   0.791
beta[7]   -2.591 0.353  -3.268  -2.842  -2.598  -2.339  -1.939
beta[8]   -0.284 0.315  -0.934  -0.480  -0.271  -0.061   0.283
deviance 715.017 4.185 708.903 712.114 714.396 717.194 725.147
          Rhat n.eff
beta[1]  1.001  1200
beta[2]  1.028    74
beta[3]  1.023    94
beta[4]  1.019   820
beta[5]  1.010   200
beta[6]  1.008  1200
beta[7]  1.012   170
beta[8]  1.008  1000
deviance 1.031    87

For each parameter, n.eff is a crude measure of effective sample size,
and Rhat is the potential scale reduction factor (at convergence, Rhat=1).

DIC info (using the rule, pD = var(deviance)/2)
pD = 8.6 and DIC = 723.6
DIC is an estimate of expected predictive error (lower deviance is
better).
```

＊参考文献

[1] 金 明哲(2007)：Rによるデータサイエンス：森北出版.
[2] P. Congdon(2003)：Applied Baysian Modelling：Wiley.