Bayesian Analysis of Growth Rates of Rats Using Hierarchical Normal Model

Niranjan Adhikari

September 2016

Introduction

The Hierarchical Model is the model where parameters of a probability model have their own probability model with some other set of parameters. In Bayesian practice, we treat the parameters of a probability model as a random variable. Thus, it is quite reasonable for this parameter to have their own model at a higher level of the hierarchy.

Suppose the random variable **X** is distributed over the probability model ***F*** with parameter $\boldsymbol{\theta}$ then the parameter $\boldsymbol{\theta}$ has another probability model ***G*** with parameter $\boldsymbol{\omega}$, so we can write the model as

$$\mathbf{X} \sim \mathbf{F}\ (\boldsymbol{\theta})$$
$$\boldsymbol{\theta} \sim \mathbf{G}\ (\boldsymbol{\omega})$$
$$\boldsymbol{\omega} \sim \mathbf{H}\ (\boldsymbol{\eta})$$

This report will examine the given data set according to the Normal Gamma Hierarchical Model using the R and JAGS software.

Aim

This report is to analyze the given data set according to Bayesian approach. The aim of the report is to see the Bayesian statistical association between time and unit growth of rats. Then, examine the mean of unit growth rate over time, overall variation of the weight over the weeks and finally, calculate the overall growth rates of rats.

Method

The '*data.csv'* file contains dataset in the 31x5 matrix format where each row define a rat and each column corresponds to a week. Let, Y_{ij} define the weight of i^{th} rat at j^{th} week, where i = 1, 2, ……, 30 and j = 1, …., 5. The dataset has been taken from section 6 Gelfand et al (1990, JASA 85: 972-985).

The Nominal Hierarchical Model is defined as follows:

$$Y_{ij} \sim Normal\ (\mu_{ij}\ , \tau_c)$$

Where,

$$\mu_{ij} = \alpha_i + \beta_i\ (x_j - \bar{x})$$

Here we have the regression model for the mean (μ_{ij}) of normal likelihood (Y_{ij}). In this model, μ_{ij} defines the linear model for the mean amount of weight gained by each rat in each week.

$$\alpha_i \sim Normal\ (\alpha_c\ , \tau_\alpha)$$

And

$$\beta_i \sim Normal\ (\beta_c\ , \tau_\beta)$$

Here, x shows the age of the rats for each measurement, $\bar{x}$ is mean and τ parameters represent the corresponding variance. So, we have linear growth model here.

Then, another level of hierarchy is defined as follows:

$$\tau_c \sim Gamma\ (0.001, 0.001)$$
$$\tau_\alpha \sim Gamma\ (0.001, 0.001)$$
$$\tau_\beta \sim Gamma\ (0.001, 0.001)$$
$$\alpha_c \sim Normal\ (0, 10^{-6})$$
$$\beta_c \sim Normal\ (0, 10^{-6})$$

In this model, β_c determines the mean unit growth, the precession of data shown by $\sigma = 1/\tau_c$ and overall mean growth rate is determined by $\alpha_0 = \alpha_c + \beta_c \bar{x}$

Model Diagram is shown demonstrated below,

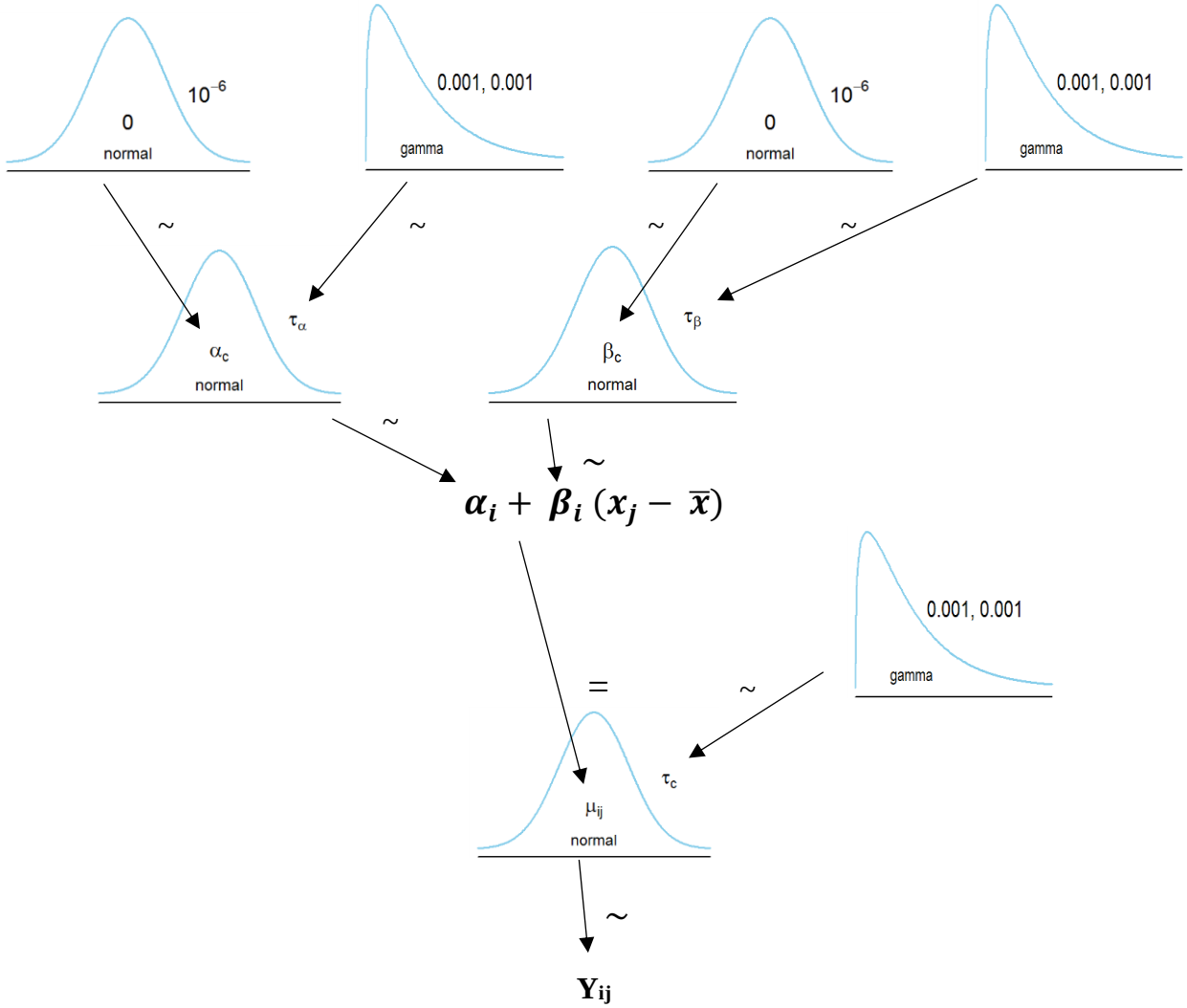

Fig 1: Normal Gamma Hierarchical Model

Our Posterior distribution becomes

$$p(\mu, \alpha, \beta / y) \propto p\,(y/\mu)\;\; p\,(\mu\,/\,\alpha, \beta)\;\; p(\alpha, \beta)$$

Result and Discussion

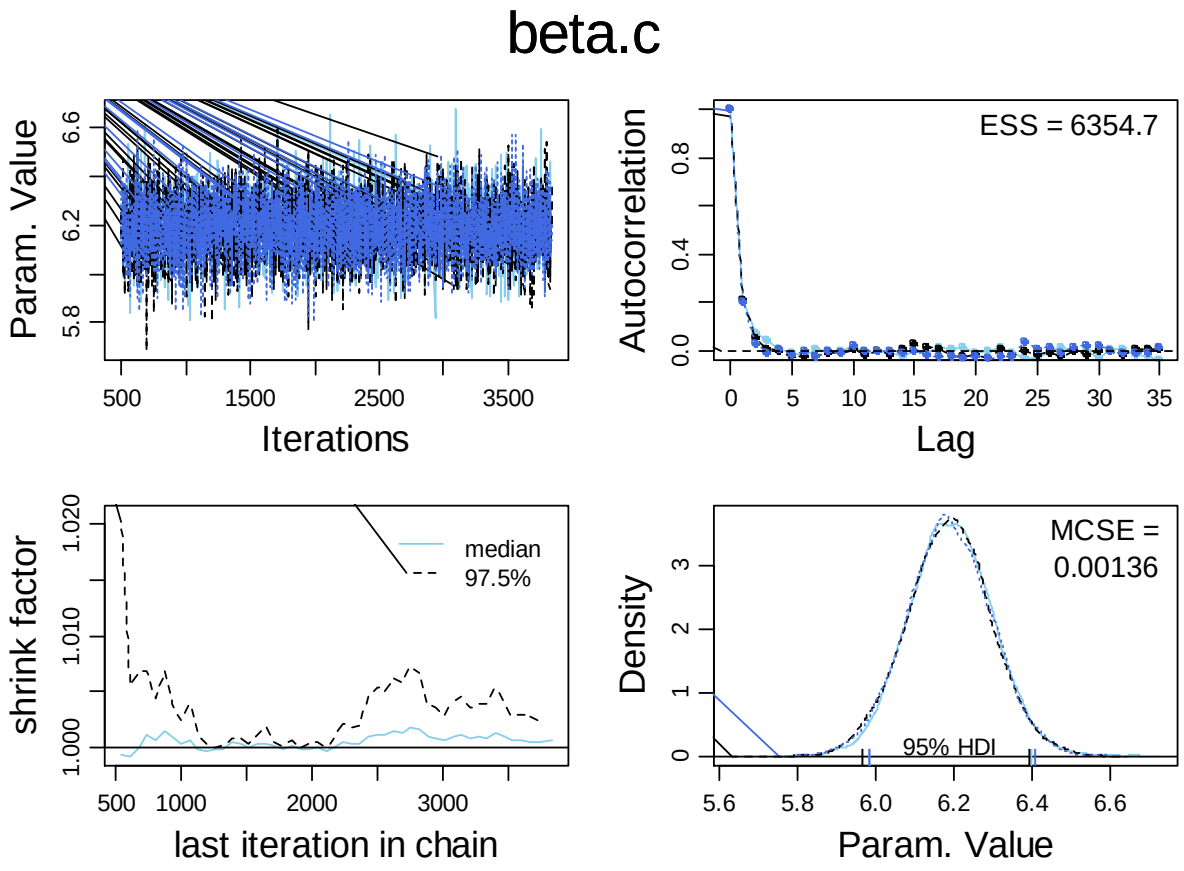

Fig 2: diagnostic plots for the mean of unit growth

From the upper left panel of the trace, the plot shows that the chain is fairly smooth and overlap to each other which is a good sign of the representativeness. However, potential scale reduction factor or shrink factor varies between 1.00 to 1.01, implies that the chain is not fully convergence. But, lower right panel the density plot of the chains overlap very well, indicates that the chains are producing the representativeness form the posterior distribution. Noticed that the density plots displace the estimated 95% HDI for each chain.

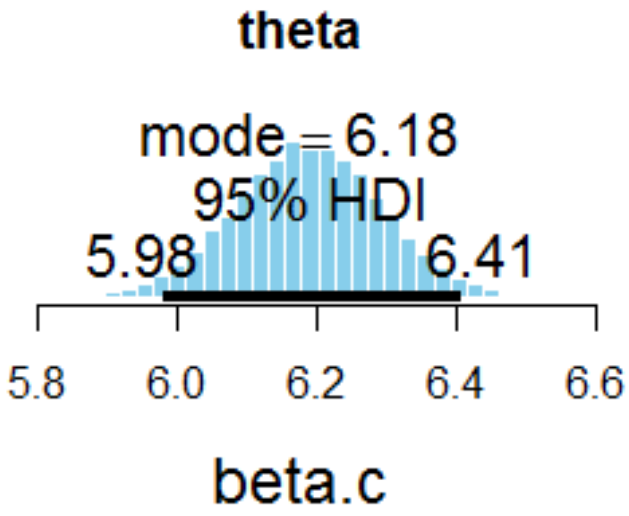

Fig 3: diagnostic plots for the mean of unit growth

We know that for this model, β_c represents the mean of unit growth of rats over the time of measurement. Thus, from our posterior pot, it is observed that with the 95% confidence the mean of unit growth rate lies between 5.96 to 6.41.

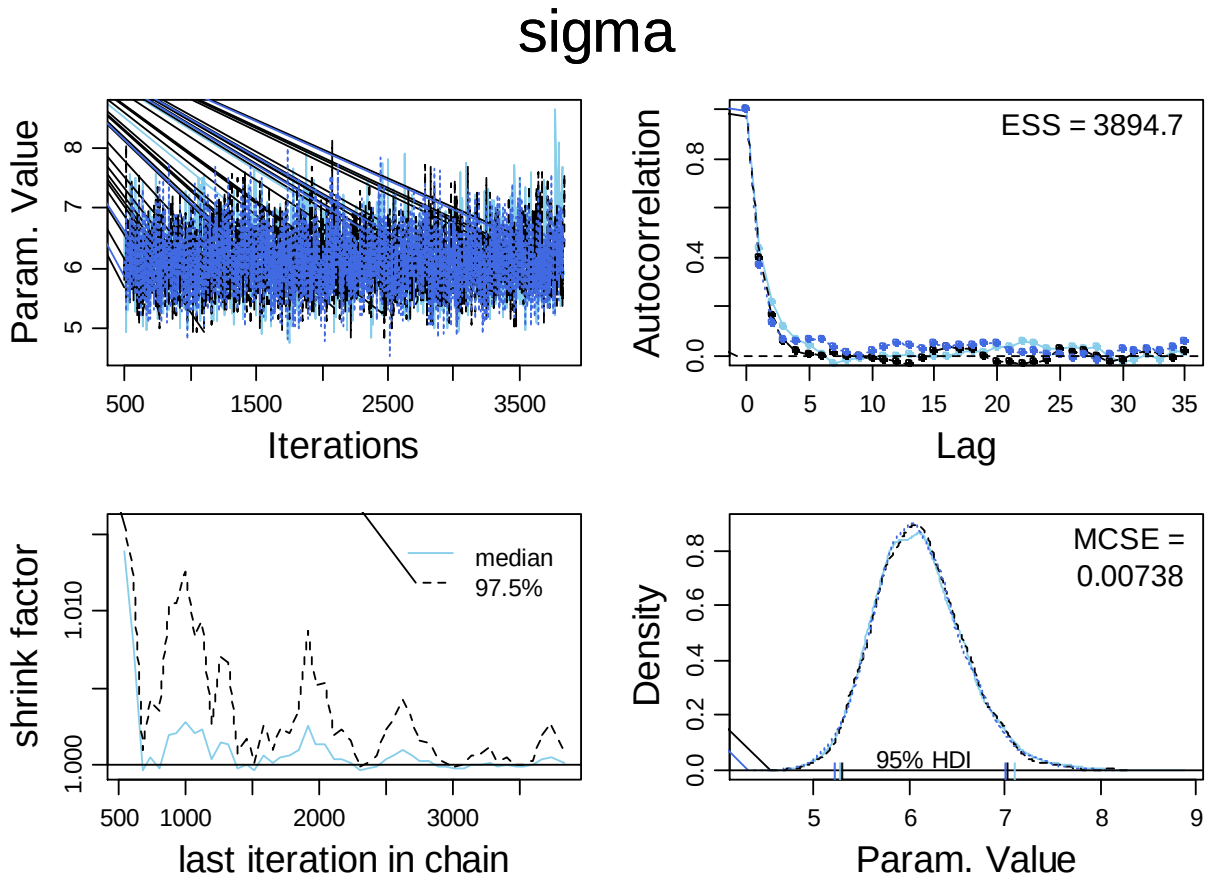

Fig 4: diagnostic plots for overall precision

From the upper left panel of the trace, the plot shows that the chain is fairly smooth and overlap to each other which is a good sign of the representativeness. However, potential scale reduction factor or shrink factor varies implies that the chain is not fully convergence. But, lower right panel the density plot of the chains overlap very well within the burn-in period, indicates that the chains are producing the representativeness form the posterior distribution. Noticed that the density plots displace the estimated 95% HDI for each chain.

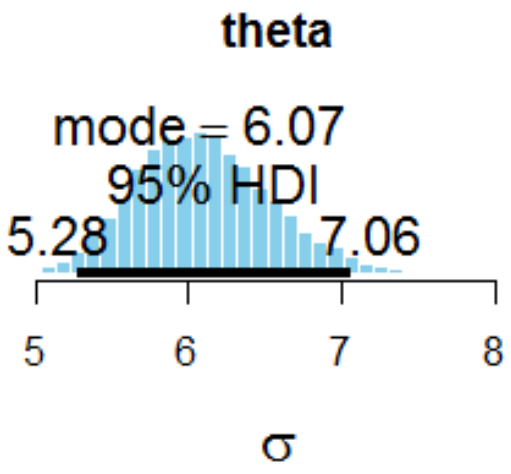

Fig 5: Posterior histogram for overall precision

Since sigma determines the precession of the data, thus from the posterior HDI plot it is inferred that with 95% confidence the variation of the measurement of the data set is lies between the 5.26 to7.06.

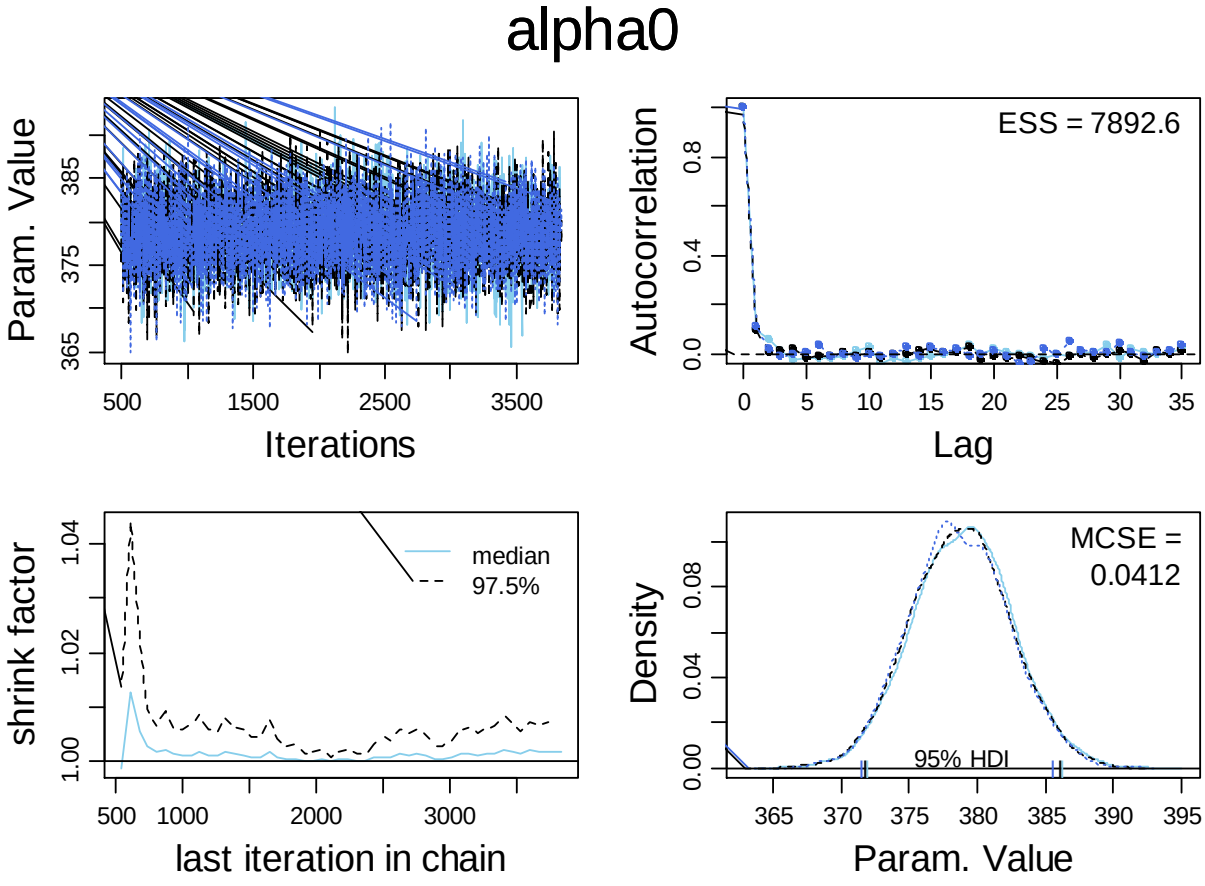

Fig 6: diagnostic plots for overall mean growth rate

From the upper left panel of the trace, the plot shows that the chain is fairly smooth and overlap to each other which is the good sign of the representativeness. Further, potential scale reduction factor or shrink factor lies between 1.00 to 1.01, implies that the chain is mostly convergence. Finally, lower right panel the density plot of the chains overlaps very well indicates that the chains are producing the representativeness form the posterior distribution. Noticed that the density plots displace the estimated 95% HDI for each chain.

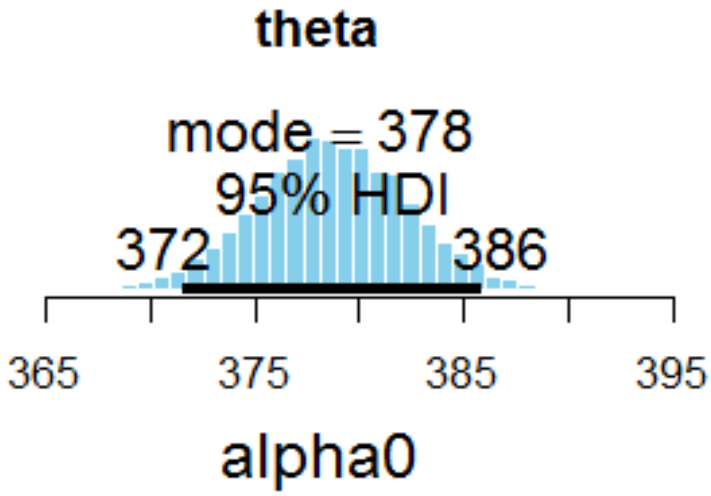

Fig 7: Posterior histogram for overall mean growth rate

According to our model structure, $\alpha_0 = \alpha_c + \beta_c \bar{x}$ determines the overall mean growth rate of rats. The above posterior histogram insists that with 95% confidence the overall growth rate for 5 weeks of weight measurement is lies between 372 to 386.

Conclusion

Hence, Bayesian data analysis has been performed to see the growth rates of rats over the time of measurement, with above plots it is clearly seen that the chains are well overlapped in MCMC. The chains are well representativeness of the posterior distribution. Thus, posterior HDI histograms have been used to estimate the value of mean unit growth, the variation of measurement and overall unit growth rate of rats.

References

John K. Kruschke, Doing Bayesian Data Analysis A Tutorial with R, JAGS, and Stan, Ed. 2, 2014, Elsevier.
Gelman A, Carlin J B, Stern H B, Dunson D B, Vehtari A, Rubin D B, Bayesian data analysis, 3rd Edition, 2014, CRC Press.

Appendix

NormalHierarchicalModel.R

```
# ---- Import data.csv file into R ----
data <- read.csv("data.csv")

Y = as.matrix(data) # The y values are the component named y.

x = c(8.0, 15.0, 22.0, 29.0, 36.0)
xbar = mean(x)
N= nrow(Y)
t = ncol(Y)
dataList = list(x = x, xbar = xbar, N = N, t = t, Y= Y)

modelString = "
model
{
  for( i in 1 : N ) {
    for( j in 1 : t ) {
      Y[i , j] ~ dnorm(mu[i , j],tau.c)
      mu[i , j] <- alpha[i] + beta[i] * (x[j] - xbar)
    }
    alpha[i] ~ dnorm(alpha.c,alpha.tau)
    beta[i] ~ dnorm(beta.c,beta.tau)
  }
  tau.c ~ dgamma(0.001,0.001)
  sigma <- 1 / sqrt(tau.c)
  alpha.c ~ dnorm(0.0,1.0E-6)
  alpha.tau ~ dgamma(0.001,0.001)
  beta.c ~ dnorm(0.0,1.0E-6)
  beta.tau ~ dgamma(0.001,0.001)
  alpha0 <- alpha.c + xbar * beta.c
}
"

writeLines( modelString , con="TEMPmodel.txt" ) # write to file

thetaInit = list(alpha = c(250, 250, 250, 250, 250, 250, 250, 250, 250, 250, 250, 250, 250, 250,
250,
            250, 250, 250, 250, 250, 250, 250, 250, 250, 250, 250, 250, 250, 250, 250),
    beta  = c(6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6,
            6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6, 6),
    alpha.c = 150, beta.c = 10,
```

```
    tau.c = 1, alpha.tau = 1, beta.tau = 1)

library(rjags)

jagsModel = jags.model( file="TEMPmodel.txt" ,
                    # the name of the file in which the model
                    # specification is stored
                    data=dataList ,
                    # the list of data
                    inits=thetaInit ,
                    # the list of initial values
                    # to let JAGS to create its own
                    # initial values for the chains,
                    # simply omit this argument entirely
                    n.chains=3 ,
                    # the number of chains to be generated
                    n.adapt=500
                    # the number of steps to take for adapting
                    # (or tuning) the samplers
)

update( jagsModel ,   # tell the name of the object
       # that include the model to JAGS
       n.iter=500    # specify the length of the burn-in period
)

codaSamples = coda.samples( jagsModel ,
                        # previously created JAGS model object
                        variable.names=c("alpha0","beta.c","sigma") ,
                        # specify which parameters will have
                        # their values recorded during the
                        # MCMC walk
                        n.iter=3334
                        # specify the number of iterations for
                        # each chain
)

source("DBDA2E-utilities.R")
# ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ---- ----

# The following line give the diagnostic plots for overall mean growth rate ---
diagMCMC( codaObject=codaSamples , parName="alpha0" )
```

```
# the following line give the diagnostic plots for mean of unit growth ---
diagMCMC( codaObject=codaSamples , parName="beta.c" )

# The following line give diagnostic plots for overall precision ---
diagMCMC( codaObject=codaSamples , parName="sigma" )

# The following plotPost() function give the  posterior histogram and posterior estimates for overall mean growth rate ---
plotPost( codaSamples[,"alpha0"] , # the element of the posterior
        # samples to be plotted
        main="theta" ,          # main title
        xlab=bquote(alpha0)      # x-axis label
)

# The  following plotPost() function give the posterior histogram and posterior estimates for mean of unit growth ---
plotPost( codaSamples[,"beta.c"] , # the element of the posterior
        # samples to be plotted
        main="theta" ,          # main title
        xlab=bquote(beta.c)      # x-axis label
)

# The following plotPost() function give the posterior histogram and posterior estimates for overall precision ---
plotPost( codaSamples[,"sigma"] , # the element of the posterior
        # samples to be plotted
        main="theta" ,          # main title
        xlab=bquote(sigma)      # x-axis label
)
```