

Using JAGS within R: visiting classic BUGS Examples

Mauro Gasparini

May 28, 2025

A lot of different software exists to approximate posterior distributions in complex and hierarchical Bayesian models using MCMC (Markov Chain Monte Carlo) simulation. In this note we focus on JAGS, which can be considered one of the mature implementations of several BUGS versions developed in the last 30 years. JAGS will be invoked within R, which is a universally known software environment for statistical analysis. The R2jags library will serve the purpose. There exists an analogous interface Python to JAGS called PyJAGS.

The reason to focus on JAGS and BUGS is that there are many “classic BUGS” examples which give an overview of several different (bio)statistical problems and applications. In addition, there are also several books focusing on different applications of BUGS, not necessarily bio.

Download and install JAGS from

<https://sourceforge.net/projects/mcmc-jags/files/latest/download>

Download the JAGS version of the Classic Bugs files from

<https://sourceforge.net/projects/mcmc-jags/files/Examples/4.x/classic-bugs.tar.gz/download>

and install them in your favorite directory (YFD).

Then get into Rstudio and install, one for all, the relevant package to run JAGS from within R:

```
#install.packages("R2jags")    #this has to be done only once  
library(R2jags)
```

```
## Loading required package: rjags  
## Loading required package: coda  
## Linked to JAGS 4.3.2  
## Loaded modules: basemod,bugs  
##  
## Attaching package: 'R2jags'  
## The following object is masked from 'package:coda':  
##  
##      traceplot
```

For illustration, we run the first example, called RATS, in the first Volume of classic BUGS examples. To do so, first set current directory to YFD/classic-bugs/vol1/rats then run the example

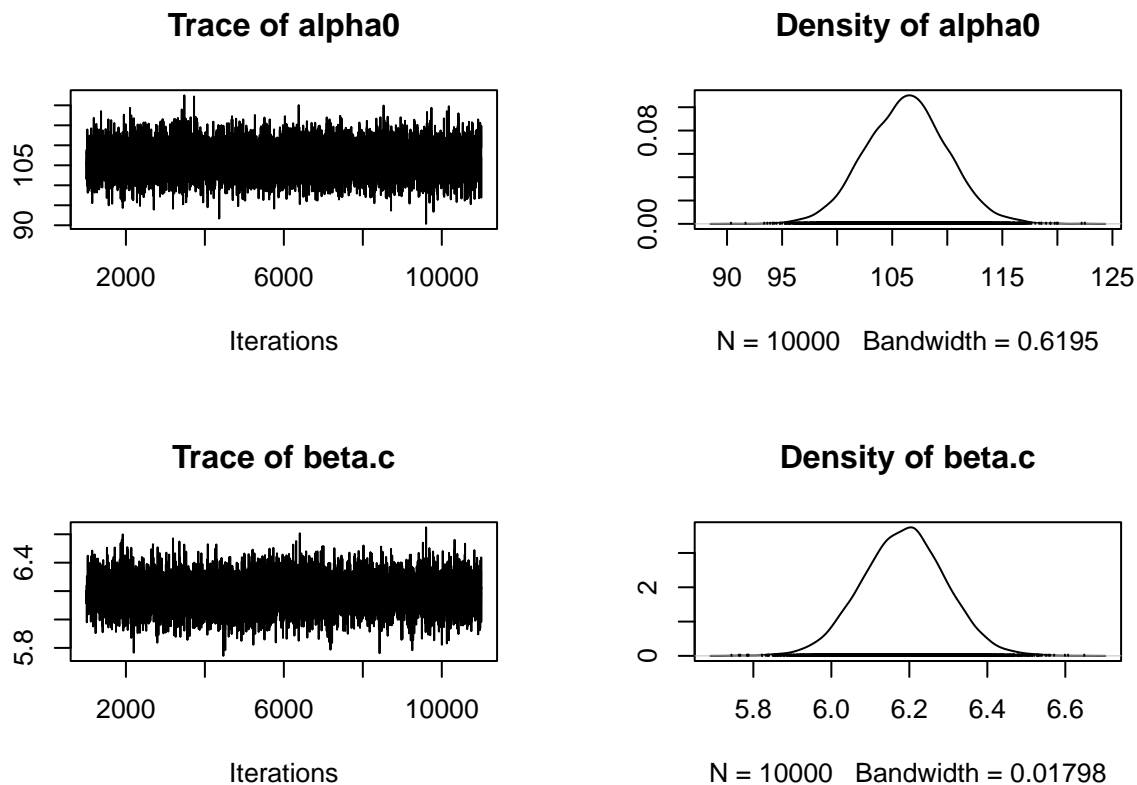
```
#setwd("~/bruno/teach/MML/jags/classic-bugs/vol1/rats")  
setwd("~/teach/MML/jags/classic-bugs/vol1/rats")  
source("test1.R")
```

```
## Compiling model graph  
##   Resolving undeclared variables  
##   Allocating nodes
```

```
## Graph information:
##   Observed stochastic nodes: 150
##   Unobserved stochastic nodes: 65
##   Total graph size: 537
##
## Initializing model
##
##       alpha0       beta.c
## 0.0245092751 0.0003007395
## OK
```

What happened is that the MCMC simulation has run and has output object `x` as a result. Object `x` which can be explored in this way

```
plot(x)
```



```
summary(x)
```

```
##
## Iterations = 1001:11000
## Thinning interval = 1
## Number of chains = 1
## Sample size per chain = 10000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##           Mean      SD Naive SE Time-series SE
## alpha0 106.307 3.6939 0.036939      0.041242
## beta.c   6.186 0.1079 0.001079      0.001361
##
```

```
## 2. Quantiles for each variable:
##
##          2.5%    25%    50%    75%   97.5%
## alpha0 99.132 103.793 106.337 108.734 113.536
## beta.c  5.975   6.115   6.188   6.258   6.397
```

You will find a nice description of the dataset and the Bayesian model in the .pdf file WinBUGS_Vol1.pdf.

Many more examples are in the three volumes, and even more complete descriptions of the examples are contained in the following book, which also contains some theory and extensive practical suggestions.

The BUGS Book: A Practical Introduction to Bayesian Analysis (2013). by David Lunn, Chris Jackson, Nicky Best, Andrew Thomas, David Spiegelhalter. Chapman & Hall.