

A review of different parameterizations used in R and BUGS

by David LeBauer, Ben Bolker, Michael C. Dietze

Abstract The ability to generate models in the BUGS simplifies the development and analysis of Bayesian models. Software written in the BUGS language includes OpenBUGS, WinBUGS, and JAGS. A number of packages provide an access to BUGS based software, enabling users to leverage the power of R when pre-processing data and analyzing model output.

However, many common probability distributions have different default parameterizations in the R versus BUGS languages. As part of a semi-automated meta-analysis module in the PEcAn workflow (LeBauer et al., 2012), we developed a function to transform parameterizations between the R and BUGS languages. Although the probability distribution functions are documented in the respective software, we are not aware of any comprehensive treatment of the different parameterizations used by BUGS and R, or a single location in which transformations between these languages are documented. Here we provide an overview of distributions for which the default parameterizations used by R and BUGS are different, including the transformations required to convert from one language to another and a R function that encapsulates these transformations.

A comparison of parameterizations

Table 1 summarizes the different parameterizations used in R (R Development Core Team, 2011) and BUGS (Plummer, 2011).

Table 1: Summary of different parameterizations of common distributions used by R and BUGS. The random variable x is implicit in all of the BUGS “Use” parameterizations. * non-default parameterizations in R - these are not used in the `r2bugs.distributions` function; to use these parameterizations, the second argument *must* be named. For clarity and ease of reference, parameterizations follow the JAGS and R documentation; thus, the table includes some equivalent variables with different names (e.g. for Gamma, r in BUGS and a in R are precisely the same), and equivalent expressions with different forms (which motivates this article).

Distribution	Language	Parameterization	Use
Normal	R	$\frac{1}{\sqrt{2\pi\sigma}} \exp(-\frac{(x-\mu)^2}{2\sigma^2})$	<code>dnorm(x, μ, σ)</code>
	BUGS	$\sqrt{\frac{\tau}{2\pi}} \exp(-(x-\mu)^2\tau)$	<code>dnorm(μ, τ)</code>
log-Normal	R	$\frac{1}{\sqrt{2\pi\sigma x}} \exp(-\frac{(\log x - \mu)^2}{2\sigma^2})$	<code>dlnorm(x, μ, σ)</code>
	BUGS	$\frac{\sqrt{\tau}}{x} \exp(-\frac{\tau(\log(x)-\mu)^2}{2})$	<code>dlnorm(μ, τ)</code>
Binomial	R	$\binom{n}{x} p^x (1-p)^{n-x}$	<code>dbinom(x, n, p)</code>
	BUGS	same	<code>dbin(p, n)</code>
Negative Binomial	R	$\frac{\Gamma(x+n)}{\Gamma(n)\Gamma(x)} p^n (1-p)^x$	<code>dnbinom(x, n, p)</code>
	R*	$\frac{\Gamma(k+x)}{\Gamma(k)\Gamma(x)} (\frac{k}{k+\mu})^k (\frac{\mu}{k+\mu})^x$	<code>dnbinom(x, n, mu = μ)*</code>
	BUGS	$(\frac{x+n-1}{x}) p^x (1-p)^n$	<code>dnegbin(p, r)</code>
Weibull	R	$\frac{b}{\Gamma(b)} (\frac{x}{\lambda})^{b-1} \exp(-(\frac{x}{\lambda})^b)$	<code>dweibull(x, a, b)</code>
	BUGS	$v\lambda x^{v-1} \exp(-\lambda x^v)$	<code>dweib(v, λ)</code>
Gamma	R	$\frac{1}{\Gamma(a)} x^{a-1} \exp(-x)$	<code>dgamma(x, a, r)</code>
	R*	$\frac{1}{s^a \Gamma(a)} x^{a-1} \exp(-x/s)$	<code>dgamma(x, a, scale = s)*</code>
	BUGS	$\frac{\lambda^r x^{r-1} \exp(-\lambda x)}{\Gamma(r)}$	<code>dgamma(r, λ)</code>

Converting from R to BUGS parameterizations

The Normal and log-normal are parameterized in terms of τ (precision) by BUGS and in terms of σ (standard deviation) or σ^2 (variance) by R. The Negative binomial distribution size parameter is discrete in BUGS (size, prob), but the size parameter is continuous in R (size, mu). The Weibull distribution has parameters (v = shape, λ = lambda) in BUGS, in R, it has either (a =shape, b =scale) or (a =shape, b =rate). The default parameterization of the Gamma in R is (a =shape, r = rate) whereas in BUGS the parameterization is (r = rate, λ = shape). Thus, the order of the shape and rate parameters is reversed in BUGS (which uses rate, shape). R also allows the Gamma to accept (shape, scale) but this requires that the scale argument is named (e.g. `dgamma(x, a, scale = b)`).

The Beta, Poisson, Exponential, and Uniform distributions have the same parameterizations in both BUGS and R. A summary of these distributions is provided in table 1; the conversions required to transform parameters from R to BUGS (and BUGS to R) is provided in the table 2 and the functions `r2bugs` and `bugs2r` in the appendix.

It is also important to note that order matters, particularly in BUGS which doesn't have named arguments. For example, the R parameterization of the Binomial distribution, `dbinom(x,size,prob)`, has the same parameters but they are in reversed order in the BUGS distribution, `dbin(p,n)`.

Distribution	R to BUGS conversion
Normal, log-Normal	$\tau = 1/\sigma^2$
Binomial	reverse parameter order
Negative Binomial	reverse parameter order
Weibull	$\lambda = (1/b)^a$
Gamma	$r = a; \lambda = r$ (reverse order)

Differences in naming

In addition to different parameterizations, four distributions have different naming conventions.

Distribution	R	BUGS
Binomial	<code>dbinom</code>	<code>dbin</code>
Negative Binomial	<code>dnbinom</code>	<code>dnegbin</code>
χ^2	<code>dchisq</code>	<code>dchisqr</code>
Weibull	<code>dweibull</code>	<code>dweib</code>

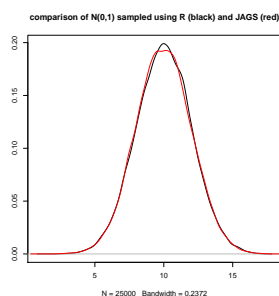
Implementation

As a simple example, we will compare the use of R and BUGS to sample from a normal distribution, $N(\mu = 10, \sigma = 2)$

```
> library(r2bugs)
> n.iter <- 100000
> set.seed(0)
> r.distn <- data.frame(distn = "norm", parama = 10, paramb = 2)
> Y.R <- do.call(paste("r", r.distn$distn, sep = ""),
+               list(n.iter/4, r.distn$parama, r.distn$paramb))
> bugs.dist <- r2bugs.distributions(r.distn)
> Y.BUGS <- r2bugs::bugs.rdist(bugs.dist, n.iter = n.iter)

> plot(density(Y.R))
> lines(density(Y.BUGS), col = 'red')
```

Here, we can visually compare the similarity of the density of 25000 samples from the same distribution implemented in R (black) and then JAGS (red).



Finally, we demonstrate how the `r2bugs.distributions` function can be used to convert a dataframe of distributions, with examples from each of the seven supported distributions (Normal, log-Normal, Weibull, Gamma, χ^2 , Binomial, and Negative-Binomial).

```
> r.distns <- data.frame(distn = c('norm', 'lnorm', 'weibull',
+                                'gamma', 'chisq', 'binom', 'nbinom'),
+                        ###      n   ln weib gam chsq bin nbin
+                        parama = c(10, 2, 10, 100, 10, 20, 10),
+                        paramb = c(1, 0.1, 10, 10, NA, 0.5, 1))
> bugs.distns <- r2bugs.distributions(r.distns)
```

Acknowledgements

This work is the result of collaboration that began on the statistical question and answer website Cross Validated (<http://stats.stackexchange.com/q/5543/1381>). Funding was provided to DSL by the Energy Biosciences Institute.

Bibliography

R Development Core Team (2011) R: A language and environment for statistical Computing. R Foundation for Statistical Computing, Vienna, Austria ISBN 3-900051-07-0 URL <http://www.R-project.org/>.

Plummer, Martyn (2011) JAGS Version 3.1.0 user manual. URL <http://sourceforge.net/projects/mcmc-jags/>.

David S. LeBauer, Dan Wang, Carl D. Davidson, Katie T. Richter and Michael C. Dietze (2012). Facilitating feedbacks between field measurements and ecosystem models. Ecological Monographs.

David S. LeBauer
 Department of Plant Biology
 Energy Biosciences Institute
 University of Illinois
 USA
dlebauer@illinois.edu

Ben M. Bolker
 Department of Mathematics and Statistics
 McMaster University
 Canada
bolker@mcmaster.ca

Appendix: R functions `r2bugs` and `bugs2r` to convert from R to BUGS parameterizations

```
r2bugs <- function(distribution,
                  parama,
                  paramb,
                  direction = "r2bugs") {

  norm  <- distn %in% 'norm'
  lnorm <- distn %in% 'lnorm'
  weib  <- distn %in% 'weibull'
  bin   <- pdistn %in% 'binom'

  ## Convert sd to precision for norm & lnorm
  paramb[norm | lnorm] <-
    1/paramb[norm | lnorm]^2
  if(direction == "r2bugs"){
```

```
    ## Convert R parameter b to JAGS
    ## parameter lambda by  $l = (1/b)^a$ 
    paramb[weib] <-
      1 / paramb[weib]^parama[weib]
  } else if (direction == "bugs2r") {
    paramb[weib] <-
      exp(-paramb[weib] / parama[weib])
  }
  ## Reverse parameter order for binomial
  bin.a <- parama[bin]
  bin.b <- paramb[bin]
  parama[bin] <- bin.b[bin]
  paramb[bin] <- bin.a[bin]

  ## Translate distribution names
  rnames <- c("weibull", "binom", "nbinom")
  bugnames <- c("weib", "bin", "negbin")
  if(direction == "r2bugs") {
    from <- rnames
    to <- bugnames
  } else if (direction == "bugs2r") {
    from <- bugnames
    to <- rnames
  }

  distn <- gsub(to, from, distn)
  return(data.frame(distn = distn,
                    parama = parama,
                    paramb=paramb))
}

bugs2r <- function(..., direction = "bugs2r"){
  return(r2bugs(..., direction))
}
```