Translating Probability Distributions: From R to BUGS and Back Again

by David S. LeBauer, Michael C. Dietze, Ben M. Bolker

Abstract The ability to implement statistical models in the BUGS language facilitates Bayesian inference by automating MCMC algorithms. Software written in the BUGS language include OpenBUGS, WinBUGS, and JAGS. A suite of R packages integrate this software seamlessly into the R environemnt for use preprocessing data and analyzing model output. However, the R and BUGS languages use inconsistent representations of common probability distributions, and this creates the potential for error and confusion when using both languages. Here we review different parameterizations used by the R and BUGS languages, describe how to translate between the languages, and provide two R functions `r2bugs.distributions` and `bugs2r.distributions` that transform parameterizations from R to BUGS, and back.

To support the use of informative prior distributions in meta-analysis (LeBauer et al., 2012), we developed functions to translate parameterizations of common probability distributions from the R to the BUGS implementations. 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, as well as R functions to translate between R and BUGS.

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

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).

Translating R parameterizations to BUGS (and back again)

Translating R paramterizations to BUGS requires three straightforward but error-prone steps. The first step is to compare how each language implements the supported probability distribution, in many cases using distinct formulae. Table 1 summarizes the different parameterizations used in R and the JAGS implementation of BUGS (Plummer, 2010). The second step is to identify relationships between parameters, including transformations and the order of arguments to a function (Table 2). Transformations used to convert from R to BUGS are provided in Table 2. Finally, the spelling of some function names is changed (Table 3).

The **Normal** and **log-normal** distributions use mean (μ) as the first argument in both languages, but the second argument is standard deviation (σ) in R and precision ($\tau = 1/\sigma^2$) in BUGS. The **Negative binomial** distribution uses a continuous size parameter (n) in R and a discrete size parameter (r) in BUGS; both languages use a probability parameter (p) by default, but the order of parameters is reversed. The R functions `*nbinom` can also use μ (where $p = n/(n+\mu)$). The **Weibull** distribution has shape (a) and scale (b) parameters in R, but has shape (ν) and lambda (λ, where $\lambda = 1/b$) in BUGS. The default parameterization of the **Gamma** in R uses shape (a) and rate (r) whereas BUGS reverses and renames these parameters: rate (r) and shape (λ). R also allows the Gamma to accept shape and scale param-

eters, if the scale argument is explicitly named (e.g. `dgamma(x,a,scale = b)`). The **Beta**, **Poisson**, **Exponential**, and **Uniform** distributions have the same parameterizations in both BUGS and R.

The order of parameters matters, since argument names are not used in BUGS and are optional in R. This is especially important because the order of parameters for the **Binomial** and **Negative Binomial** distributions are switched when translating between BUGS and R (Tables 1,2). Alternative parameterizations of the **Gamma** and **Negative Binomial** distributions are provided in R, but their use requires that the parameters be given as named arguments.

All of these considerations are integrated into the R functions `r2bugs.distributions` and `bugs2r.distributions` that translate the parameterization of common probability distributions between the R and BUGS languages (see Appendix).

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)

Table 2: Transformations required to convert from R to BUGS parameterizations

In addition to different parameterizations, four distributions have different naming conventions (Table 3).

Distribution	R	BUGS
Binomial	`dbinom`	`dbin`
Negative Binomial	`dnbinom`	`dnegbin`
χ^2	`dchisq`	`dchisqr`
Weibull	`dweibull`	`dweib`

Table 3: Distributions with different naming conventions

Acknowlegements

This collaboration 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

David LeBauer, Dan Wang, Katie Richter, Carl Davidson, and Michael C. Dietze. Facilitating feedbacks between field measurements and ecosystem models. *Ecological Monographs*, 2012. ISSN 0012-9615. doi: 10.1890/12-0137.1.

Martyn Plummer. JAGS Version 3.1.0 user manual, 2010. URL http://sourceforge.net/projects/mcmc-jags/.

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

Michael C. Dietze
Department of Earth And Environment
Boston University, USA

Ben M. Bolker
Department of Mathematics and Statistics
McMaster University, Canada

Appendix: R functions r2bugs and bugs2r to translate between R and BUGS parameterizations of common probability distributions

```
r2bugs.distributions <-
  function(priors, direction = 'r2bugs')

  priors$distn  <- as.character(priors$distn)
  priors$parama <- as.numeric(priors$parama)
  priors$paramb <- as.numeric(priors$paramb)

  ## index dataframe according to distribution

  ## norm, lnorm use the same transformation
  norm   <- priors$distn %in% c('norm', 'lnorm')

  ## the following match both R and BUGS names
  weib   <- grepl("weib", priors$distn)
  gamma  <- priors$distn == 'gamma'
  chsq   <- grepl("chisq", priors$distn)
  bin    <- priors$distn %in% c('binom', 'bin')
  nbin   <- priors$distn %in% c('nbinom', 'negbin')

  exponent <- ifelse(direction == "r2bugs", -2, -0.5)
  ## Convert sd to precision for norm & lnorm
  priors$paramb[norm] <-  priors$paramb[norm] ^ exponent
  if(direction == 'r2bugs')
    ## Convert R parameter b to
    ## BUGS parameter lambda by l = (1/b)^a
    priors$paramb[weib] <-
      (1 / priors$paramb[weib]) ^ priors$parama[weib]
   else if (direction == 'bugs2r')
    ## Convert BUGS parameter lambda
    ## to BUGS parameter b by b = l^(-1/a)
    priors$paramb[weib] <-
      priors$paramb[weib] ^ (- 1 / priors$parama[weib])


  ## Reverse parameter order
  ## for binomial and negative binomial
  priors[bin | nbin, c('parama', 'paramb')] <-
    priors[bin | nbin, c('paramb', 'parama')]

  ## Translate distribution names
  if(direction == "r2bugs")
    priors$distn[weib] <- "weib"
    priors$distn[chsq] <- "chisqr"
    priors$distn[bin]  <- "bin"
    priors$distn[nbin] <- "negbin"
   else if(direction == "bugs2r")
    priors$distn[weib] <- "weibull"
    priors$distn[chsq] <- "chisq"
    priors$distn[bin]  <- "binom"
    priors$distn[nbin] <- "nbinom"

  return(priors)


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