



WebBUGS: Conducting Bayesian Statistical Analysis Online

Zhiyong Zhang

University of Notre Dame

Abstract

A web interface, named **WebBUGS**, is developed to conduct Bayesian analysis online over the Internet through **OpenBUGS** and R. **WebBUGS** can be used with the minimum requirement of a web browser both remotely and locally. **WebBUGS** has many collaborative features such as email notification and sharing. **WebBUGS** also eases the use of **OpenBUGS** by providing built-in model templates, data management module, and other useful modules. In this paper, the use of **WebBUGS** is illustrated and discussed.

Keywords: **WebBUGS**, **OpenBUGS**, R, online Bayesian analysis.

1. Introduction

The popularity of Bayesian methods has grown rapidly in many disciplines such as biology, medical research, physics, and social, behavioral, and educational sciences (Bacco, D'Amore, and Scalfari 2004; Gelman, Carlin, Stern, Vehtari, and Rubin 2013; Gill 2007; Grandy Jr. and Schick 2007). Recently, Bayesian analysis has also become more and more appealing to applied researchers whose primary aim is to answer substantive research questions through applications of Bayesian modeling (e.g., Muthén and Asparouhov 2012; Rupp, Dey, and Zumbo 2004). There are a number of factors that make the Bayesian framework appealing as pointed out by Muthén and Asparouhov (2012). First, Bayesian analysis performs better for small-sample problems and large-sample theory is often not required. Second, it can be less computationally demanding, especially for some complex models involving multilevel structures, missing data, and non-normal data. Many models that involve high-dimensional integration are computationally intensive or even impossible to estimate. In this situation, Bayesian methods can be used as a computational tool for conducting high-dimensional integration to obtain model parameter estimates based on Markov chain Monte Carlo (MCMC) methods. Third, Bayesian analysis can also incorporate useful prior information in statistical inference. In social and

educational research, ample public data and information are available and should inform new research design and data analysis. Bayesian analysis provides a natural way to incorporate such information (e.g., Zhang, Hamagami, Wang, Grimm, and Nesselroade 2007). Fourth, Bayesian analysis allows the development and use of new types of models and data analysis that otherwise may not be possible. Many such examples can be seen in the literature (e.g., Congdon 2003; Wang and McArdle 2008).

In recent scholarship, the primary question about Bayesian analysis has shifted from “What is Bayesian analysis?” and “Why Bayesian analysis?” to “How to conduct Bayesian analysis?” However, given its advantages, Bayesian data analysis is still not as widely used as it should be. One important barrier to its wider use is the lack of accessible software (e.g., Martin 2008; Moore 1997; Winkler 2001). The most popular (and perhaps most powerful) Bayesian software is BUGS, which includes **WinBUGS**, **OpenBUGS** and **JAGS** (Lunn, Spiegelhalter, Thomas, and Best 2009; Plummer 2003). The availability of BUGS has largely advanced the adoption of Bayesian analysis in the applied community.

In this paper, I introduce a newly developed web interface for BUGS, named **WebBUGS**, to conduct Bayesian analysis within an Internet browser using **OpenBUGS** and R (R Core Team 2014) as its back-end on a remote server. Compared to the traditional interface of **WinBUGS** and **OpenBUGS**, **WebBUGS** has several distinctive features. First, **WebBUGS** has few requirements for use. In fact, only a web browser, which is often pre-installed on a system, is needed to use **WebBUGS** for Bayesian analysis. Therefore, a user can potentially conduct Bayesian data analysis through Internet Explorer on a desktop PC, Safari on a MacBook, or even Android Browser on a smartphone.

Second, **WebBUGS** seamlessly integrates **OpenBUGS** and R. **WebBUGS** calls for **OpenBUGS** for MCMC methods and uses R to process the generated Markov chains for convergence diagnostics and output analysis. It also allows a user to customize the analysis of generated Markov chains through R online.

Third, **WebBUGS** can make the analysis of large scale data sets possible. Bayesian analysis on average is much more computationally intensive than frequentist data analysis. For analyzing large-scale data sets, personal desktops and laptops may not have sufficient capabilities to conduct complex Bayesian analyses. However, **WebBUGS** can be set up on a central powerful computer and be shared by more than one researcher. For example, the current **WebBUGS** server is powered by 64 CPU cores and 128GB of memory.

Fourth, **WebBUGS** provides a data management module that can be used to easily convert a data set to **OpenBUGS** compatible format. This feature largely removes a big hurdle of using BUGS especially for users who are not familiar with R.

Fifth, **WebBUGS** also includes modules for conducting certain types of analysis, such as meta-analysis, through a graphical user interface. These modules allow novel users to carry out routine Bayesian analysis. **WebBUGS** further provides built-in BUGS model code that can help users learn to use BUGS.

Sixth, **WebBUGS** can largely boost the productivity of Bayesian data analysis by providing email notification and sharing features. For example, at the end of the analysis, an email will be sent to the user for notification if required. Furthermore, a user can easily share a data analysis with collaborators by sharing the link with a password of the analysis. The analysis by a user can be saved on the server for future reference in one central location. This can be useful for classroom teaching and conference presentations.

In the following, Bayesian analysis is briefly introduced in Section 2. Then, the use of **WebBUGS** is demonstrated in Section 3. After that, the features of **WebBUGS** are discussed in Section 4 and some modules of **WebBUGS** are presented in Section 5. For those who are interested in setting up their own **WebBUGS** server, instructions are provided in Section 6. Finally, future developments of **WebBUGS** are discussed in Section 7.

2. Brief introduction to Bayesian analysis

Introductory textbooks on Bayesian analysis include Gelman *et al.* (2013), Gill (2007) and Kruschke (2011); more advanced discussions are given by Lee (2007) and Song and Lee (2012). Here, I only provide a brief introduction of essential concepts needed to begin using the **WebBUGS** software.

Bayesian methods are based on Bayes' theorem which has the following form

$$p(\boldsymbol{\theta}|\mathbf{Y}) = \frac{p(\boldsymbol{\theta})p(\mathbf{Y}|\boldsymbol{\theta})}{p(\mathbf{Y})},$$

where $\boldsymbol{\theta}$ represents a vector of all the model parameters in a model and \mathbf{Y} represents all available data. $p(\boldsymbol{\theta})$ is the prior distribution that summarizes the prior information or knowledge of $\boldsymbol{\theta}$ before data collection. $p(\mathbf{Y}|\boldsymbol{\theta})$ is the distribution of the data given a model and is also the likelihood function $L(\boldsymbol{\theta}|\mathbf{Y})$ in the frequentist framework. $p(\boldsymbol{\theta}|\mathbf{Y})$ is the posterior distribution that incorporates information from both prior information and currently collected data. $p(\mathbf{Y}) = \int_{\boldsymbol{\theta}} p(\boldsymbol{\theta})p(\mathbf{Y}|\boldsymbol{\theta})d\boldsymbol{\theta}$ is the marginal distribution of \mathbf{Y} , which is not related to model parameters. Therefore, the posterior distribution is proportional to the product of the prior distribution and the likelihood function such that

$$p(\boldsymbol{\theta}|\mathbf{Y}) \propto p(\boldsymbol{\theta})p(\mathbf{Y}|\boldsymbol{\theta}) = p(\boldsymbol{\theta})L(\boldsymbol{\theta}|\mathbf{Y}).$$

All information regarding the model parameters $\boldsymbol{\theta}$ can be obtained from the posterior distribution $p(\boldsymbol{\theta}|\mathbf{Y})$. For example, the density function can be plotted directly for one- or two-dimensional $\boldsymbol{\theta}$. For high-dimensional problems, the posterior means and the standard deviations can be obtained either analytically or through MCMC methods. MCMC methods generate random numbers from the posterior distribution and construct the parameter estimates using the average and standard deviation of the generated numbers.

There are two practical issues in applying MCMC methods. The first is to diagnose the convergence of Markov chains. The second is to decide the length of Markov chains. Brooks and Roberts (1998) and Cowles and Carlin (1996) discussed many different methods for testing convergence. Here, I focus on the Geweke test (Geweke 1992) and visually inspecting the trace plots of Markov chains.

Geweke (1992) proposed a convergence diagnostic method for Markov chains based on a test for equality of the means of the first and last part of a Markov chain. If the samples are drawn from the stationary distribution of the chain, the two means are equal and the Geweke statistic asymptotically follows a standard normal distribution. Let $\{\theta^t : t = 1, \dots, n_A\}$ and $\{\theta^t : t = n - n_B + 1, \dots, n\}$ with $n - n_B + 1 > n_A$ denote two subsets of a Markov chain.

$$\bar{\theta}_A = \frac{1}{n_A} \sum_{t=1}^{n_A} \theta^t$$

and

$$\bar{\theta}_B = \frac{1}{n_B} \sum_{t=n-n_B+1}^n \theta^t$$

are the sample means and $\hat{S}_A(0)$ and $\hat{S}_B(0)$ are the spectral densities at zero of the two subsets. If the Markov chain is stationary, then

$$Z = \frac{(\bar{\theta}_A - \bar{\theta}_B)}{\sqrt{\hat{S}_A(0)/n_A + \hat{S}_B(0)/n_B}} \sim N(0, 1)$$

as $n \rightarrow \infty$. Geweke (1992) suggested the use of $n_A = n/10$ and $n_B = n/2$.

By nature, Markov chains always have autocorrelation. For two Markov chains, the one with higher autocorrelation provides less information about the posterior distribution than the one with smaller autocorrelation. In other words, a longer Markov chain is needed to accurately describe a posterior if the autocorrelation is higher. To characterize the information in a Markov chain, I use the statistic called effective sample size (ESS). The ESS is the equivalent sample size assuming no autocorrelation. For two Markov chains with the same length, the one with larger ESS provides more information. A practical rule of thumb is to get an ESS of at least 400 to determine if the length of a Markov chain is sufficient.

Suppose θ is the unknown variance in a normal model with unknown mean and variance $N(\mu, \theta)$. To estimate θ through MCMC methods such as a Gibbs sampling algorithm, one needs to diagnose the convergence of θ . For illustration, Figures 1 and 2 present trace plots for 4 possible Markov chains of θ . Figure 1 on the left displays a “well-behaved” Markov chain of θ . The Markov chain fluctuates around its center with similar variation. The Geweke statistic is 0.17, also indicating the convergence of the Markov chain. Furthermore, the effective sample size is 4668, close to the total number of iterations 5000. This indicates that the autocorrelation for this Markov chain is small.

The Markov chain of θ in Figure 1 on the right seems to converge well based on the trace plot and the Geweke statistic (-1.82). However, the ESS is only 122 indicating high autocorrelation. Therefore, a longer Markov chain is needed if one wants to make inference of the posterior from which the Markov chain is generated.

There appear two parts in the Markov chain of θ in Figure 2 on the left. There is a decline trend in the first part up to around the 1000th iteration. The second part seems very stable as in the Markov chain in Figure 1 on the left. In this situation, the first part of the Markov chain is often discarded as burn-in. The Geweke statistic for the whole Markov chain is 6.39 indicating non-convergence of the overall chain. However, after the burn-in period is discarded, the Geweke statistic becomes 0.58. The ESS is 3512 indicating low autocorrelation for the kept part of the Markov chain.

Figure 2 on the right portrays an ill-behaved Markov chain of θ . First, there appears a growth trend. Second, it seems the autocorrelation is high. Furthermore, the Geweke statistic is -16.38 , also indicating non-convergence. A Markov chain like this cannot be used for further inference.

For a converged Markov chain with sufficient effective sample size, one can construct posterior mean, posterior standard deviation, as well as equal-tail and highest posterior density (HPD) credible intervals for each model parameter for inference. Suppose the generated Markov

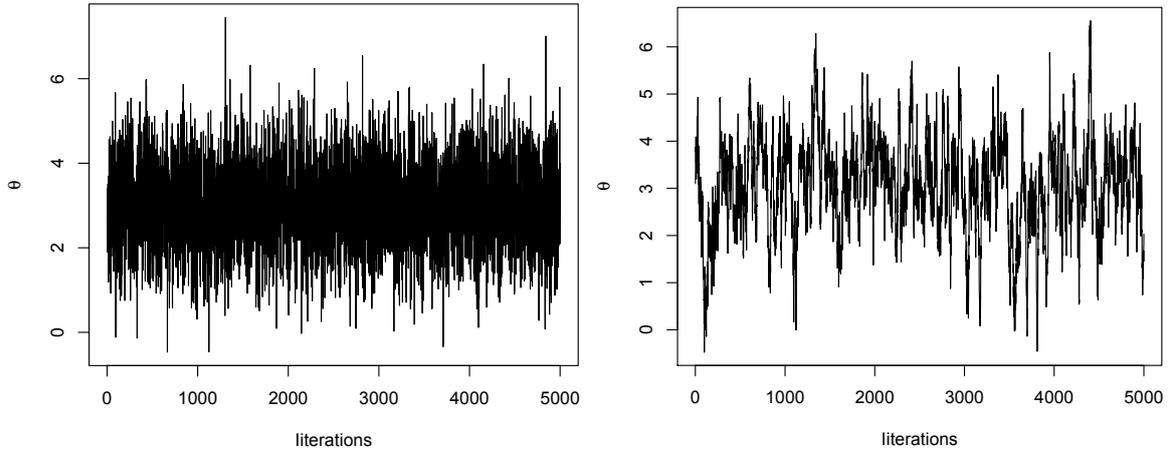


Figure 1: Left: Convergence with low autocorrelation. Geweke = 0.17 & ESS = 4668. Right: Convergence with high autocorrelation. Geweke = -1.82 & ESS = 122.

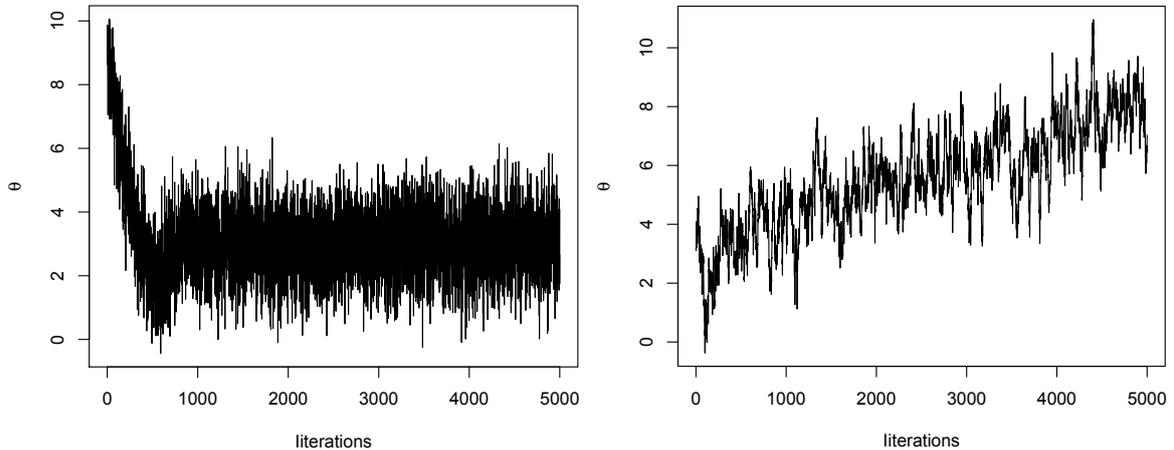


Figure 2: Left: Convergence with low autocorrelation after 1,000 iterations. Geweke = 6.39 for the whole Markov chain. Geweke = 0.58 & ESS = 3512 for the kept iterations. Right: Non-convergence. Geweke = -16.36 .

chain after the burn-in period for θ is $\theta_i, i = 1, \dots, n$ where n is the number of iterations. Then, a point estimate for θ can be constructed by the sample mean of the Markov chain

$$\bar{\theta} = \frac{1}{n} \sum_{i=1}^n \theta_i.$$

The standard deviation of θ is given by

$$s.d.(\theta) = \frac{1}{n-1} \sum_{i=1}^n (\theta_i - \bar{\theta})^2.$$

Credible intervals for θ can also be constructed based on the Markov chain. The most widely used credible intervals include the equal-tail credible interval and the HPD credible interval. A $100(1 - \alpha)\%$ equal-tail credible interval is $[\theta^{\alpha/2}, \theta^{1-\alpha/2}]$ where the lower and upper bounds

are the $100\alpha/2$ th and $100(1 - \alpha/2)$ th percentiles of the Markov chain, respectively. The HPD credible interval is the credible interval that covers $100(1 - \alpha)\%$ region of the density formed by the Markov chain but at the same time has the smallest interval width. For symmetrical posteriors, the equal-tail credible interval is the same as the HPD credible interval. For non-symmetrical posteriors, the HPD credible interval has smaller width than the equal-tail credible interval.

2.1. Bayesian analysis using BUGS

A comprehensive coverage of the use of BUGS can be found in textbooks such as Lunn, Jackson, Best, Thomas, and Spiegelhalter (2012) and Ntzoufras (2009). Typical BUGS code for Bayesian analysis consists of three parts: model specification, data, and initial values. The model specification part is used to specify the likelihood $L(\boldsymbol{\theta}|\mathbf{Y})$ and prior distribution $p(\boldsymbol{\theta})$. Both data and initial values can be provided using BUGS list format. A list can include a scalar, a vector, and/or an array. Although the rectangular format of data can be used, it is less flexible.

Once the model, data, and initial values are ready, one can compile the model, load data and initial values, set up the parameters to monitor, and generate Markov chains for parameters. One then examines convergence of the Markov chains, which can typically be done using the R package `coda` (Plummer, Best, Cowles, and Vines 2006). Finally, one can obtain the parameter estimates, standard deviations, and credible intervals.

BUGS is probably by far the most widely used software for Bayesian analysis. However, the learning curve of it is very steep especially for researchers in the education, behavioral, and social sciences. Based on my own experience of teaching graduate level Bayesian classes and statistical consulting, students and applied researchers often lacked the programming skills to use BUGS. For example, BUGS requires the specification of a model using R-like language and organization of data in specific formats, which often makes it difficult to use for novel users. This motivated the initial development of **WebBUGS**. The current version of **WebBUGS** follows the same structure as BUGS to perform Bayesian analysis but simplifies many of the steps for users. In addition, it has many other distinctive features as discussed in Sections 1 and 4.

3. Use of WebBUGS

The use of **WebBUGS** is demonstrated through an example of Bayesian meta-analysis of correlation with power prior. Meta-analysis is a statistical method to combine findings from multiple studies to get a more comprehensive understanding of the population under study (e.g., Hunter and Schmidt 2004). Because studies may differ in quality or sample size, Zhang, Jiang, and Liu (2013a) proposed a Bayesian meta-analysis method that controls the contribution of each study to the final results through a power prior (e.g., Ibrahim and Chen 2000).

Zhang *et al.* (2013a) discussed several types of meta-analysis models. As an example, I focus on the random-effects meta-analysis of correlation. Suppose there are m studies that report the sample correlation between two variables. Each study reports a sample correlation r_i with the corresponding sample size n_i . Let $z_i = \frac{1}{2} \ln \left(\frac{1+r_i}{1-r_i} \right)$ denote the Fisher z -transformation of

Sample size (n)	Correlation (r)	Reliability (a)
215	0.21	1
132	0.25	1
309	0.12	1
117	0.33	1
307	0.40	0.80
1212	0.34	1
175	0.11	1
380	0.15	1
86	0.11	1
74	0.36	1
361	0.14	1

Table 1: Example data.

r_i and $\zeta_i = \frac{1}{2} \ln \left(\frac{1+\rho_i}{1-\rho_i} \right)$ be the Fisher z -transformation of the population correlation. Then, $z_i \sim N(\zeta_i, \phi_i)$ with $\phi_i = (n_i - 3)^{-1}$.

A random-effects model can be written as a two-level model,

$$\begin{cases} z_i = \zeta_i + e_i \\ \zeta_i = \beta + v_i \end{cases} \quad (1)$$

where $\text{VAR}(e_i) = \phi_i$ and $\text{VAR}(v_i) = \tau$. In the model, each z_i has its own mean ζ_i and the grand mean of ζ_i is β . Based on the Fisher z -transformation, $z_i \sim N(\zeta_i, \phi_i)$. It is often assumed that v_i has a normal distribution and therefore $\zeta_i \sim N(\beta, \tau)$. Let $\alpha_i, 0 \leq \alpha_i \leq 1$, denote the power to control the contribution of the i th study. Zhang *et al.* (2013a) showed that using the power prior is equivalent to modifying the distribution of z_i to be $N(\zeta_i, \phi_i/\alpha_i)$. For the random-effects model, one has the fixed-effects parameter β and the random-effects parameter τ . The parameter τ represents the between-study variability. The parameter β can be transformed back to the correlation that represents the overall correlation across all studies.

For illustration, a subset of the data used in Zhang *et al.* (2013a) is used here, in which correlations between high-performance work systems (HPWS) and financial performance from 11 studies as well as sample size and reliability of each study are available. The data are given in Table 1. To use **WebBUGS** to fit the random-effects meta-analysis model, first open a web browser, for example, Firefox, Internet Explorer, or Safari. In the address bar, input the web address of **WebBUGS**. For our server, the web address is <http://WebBUGS.psychstat.org/>. Users will then be prompted to log in using their username and password as shown in Figure 3. There are two reasons for requiring a user to register to use **WebBUGS**. First, this will prevent a user's analyses to be viewed by other users. Second, a user can access his/her analyses in the future. For demonstration, a user `jssdemo` was created with the password `jssdemo`.

After login, a user will be directed to the index page as shown in Figure 4 where the user can start a new Bayesian analysis. On this page, a user can provide a **BUGS** model, data and initial values. The name of the analysis can also be provided to identify the analysis in the future. If the checkbox for **Email notification** is checked, the user will receive an email notification of the completion of the analysis.

Welcome

Login

Username

Password

[Register](#) | [Forgot Password?](#) | [FAQ](#)

WebBUGS Admin » [Login](#) | [Logout](#) | [Profile](#) | [Forgot password](#)

Figure 3: The login page. A user has to log in before conducting data analysis. For testing, `jssdemo` can be used both as username and password.

Welcome JSS » [New](#) | [All Analyses](#) | [Modules](#) | [FAQ](#)

Model

Name of analysis Email notification

1. Model: Type or select a model [Help](#)

2. Data: Input or select data [Help](#)

3. Initial values: Multiple sets of initial values are allowed. [Help](#)

WebBUGS Admin » [Login](#) | [Logout](#) | [Profile](#) | [Forgot password](#)

Figure 4: The index page. A new analysis starts here.

3.1. Model

A model in the BUGS format needs to be specified in the `Model` field of the web page. A user can type or copy the BUGS code for a model. To ease its use, **WebBUGS** provides a list of built-in models for different types of analysis. For example, for meta-analysis, if one chooses the `Meta-analysis` model from the drop-down list, the BUGS code will be filled in the `Model` field as shown in Figure 5. The template code can be modified for specific data analysis, e.g., reflecting the data and the model. For the model used here, `m`, `r`, `n`, and `a` are data that should be supplied. The parameters `beta` and `pre.tau` need to be initialized using initial values.

Welcome JSS » [New](#) | [Current](#) | [Re-run](#) | [Copy](#) | [Email](#) | [Share](#) | [All Analyses](#) | [Modules](#) | [FAQ](#)

Model

Name of analysis Email notification

1. Model: Type or select a model [Help](#)

```

model{
  for (i in 1:m){
    z[i] <- .5*log((1+r[i])/(1-r[i]))
    pre.phi[i] <- (n[i]-3)*a[i]
    z[i] ~ dnorm(zeta[i], pre.phi[i])
    zeta[i] ~ dnorm(beta, pre.tau)
  }
  beta ~ dnorm(0, 1.0E-6)
  rho <- (exp(2*beta)-1)/((exp(2*beta)+1))
  pre.tau ~ dgamma(.001,.001)
  tau <- 1/pre.tau
}

```

2. Data: Input or select data [Convert data](#) [Help](#)

```

list(m = 11, r = c(0.210,0.252,0.123,0.330,0.400,0.340,0.110,0.147,0.110,0.360,0.138), n =
c(215,132,309,117,307,1212,175,380,86,74,361), a =
c(1.0,1.0,1.0,1.0,0.8,1.0,1.0,1.0,1.0,1.0,1.0))

```

3. Initial values: Multiple sets of initial values are allowed. [Convert data](#) [Help](#)

```

list(beta = 0, pre.tau = 1 )
list(beta = 1, pre.tau = 1 )
list(beta = -1, pre.tau = 1 )

```

[WebBUGS](#) [Admin](#) » [Login](#) | [Logout](#) | [Profile](#) | [Forgot password](#)

Figure 5: The input specifications for the meta-analysis example. The **Model**, **Data**, and **Initial values** fields are required and the **Name of analysis** field is optional.

3.2. Data

Data should be provided in the format of a list as discussed in the **OpenBUGS** manual. The rectangular format of data is not allowed because **WebBUGS** provides flexible ways to convert data to a list, which can largely reduce potential formatting errors of data input.

A set of data can be supplied in three ways. First, one can simply copy and paste BUGS format data into the **Data** field. Second, if a user already has a set of data saved on the web server, it can be selected using the drop-down list. Note the drop-down list automatically displays all available data sets. Instruction on how to upload and manipulate data can be found in Section 5.1. Third, data can be converted directly in the **Data** field. The third method is used in the example.

In the meta-analysis example, one needs to know the total number of studies m , also known as, the sample size, the data for the correlation variable r , the corresponding sample size n and the power a . To input such data, one can use the code shown below. The data are formatted in a way that the data name is first given, followed by the equal sign $=$ and then the data values. The data values are separated by one or more white spaces. Data for two different data names are separated by the semicolon $;$. Note that *no* semicolon should be used at the end of data input. Data can be provided on several lines for a given variable.

```

m = 11; r = 0.210 0.252 0.123 0.330 0.400 0.340 0.110 0.147 0.110 0.360
0.138; n = 215 132 309 117 307 1212 175 380 86 74 361;
a = 1.0 1.0 1.0 1.0 0.8 1.0 1.0 1.0 1.0 1.0 1.0

```

Welcome **JSS** » [New](#) | [Current](#) | [Re-run](#) | [Copy](#) | [Email](#) | [Share](#) | [All Analyses](#) | [Modules](#) | [FAQ](#)

Controlling MCMC

1. Names of the parameters to monitor: separated by space

2. Number of total updates

3. Length of burn in: i.e. number of iterations to discard at the beginning

4. Thinning rate: must be a positive integer.

5. Random seed for OpenBUGS. Must be an integer between 1-14.

[WebBUGS](#) [Admin](#) » [Login](#) | [Logout](#) | [Profile](#) | [Forgot password](#)

Figure 6: Specify parameters to control MCMC.

After inputting the data, one can click the **Convert data** button to convert the data to BUGS format as shown in the **Data** field of Figure 5 and also as shown below. It is always recommended to check whether data are converted correctly.

```
list(m = 11, r = c(0.210, 0.252, 0.123, 0.330, 0.400, 0.340, 0.110, 0.147,
0.110,0.360, 0.138), n = c(215, 132, 309, 117, 307, 1212, 175, 380, 86,
74, 361), a = c(1.0, 1.0, 1.0, 1.0, 0.8, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0))
```

3.3. Initial values

Initial values also have to be provided in the BUGS list format. The initial values can be converted in the same way as data conversion. In addition, multiple sets of initial values using multiple lists are allowed, when multiple-chain analysis is conducted. For the meta-analysis example, three sets of initial values are supplied. The initial values can be typed as below with the vertical bar | to separate each set of values. By clicking the button **Convert data**, the initial values in the field of **Initial values** of Figure 5 are generated.

```
beta = 0; pre.tau = 1 | beta = 1; pre.tau = 1 | beta = -1; pre.tau = 1
```

3.4. Control MCMC

With the model, data, and initial values provided, by clicking on the **Next** button in Figure 5, a user will be directed to a page as shown in Figure 6. Note that **WebBUGS** checks the functions and distributions used in a model and if an error is found, it will prompt users to make correction. On this page, the user can specify the parameters to be monitored. For the meta-analysis example, two parameters are monitored: **beta** and **tau**. Note that multiple parameters are separated by white space. Only parameters used in the model can be specified

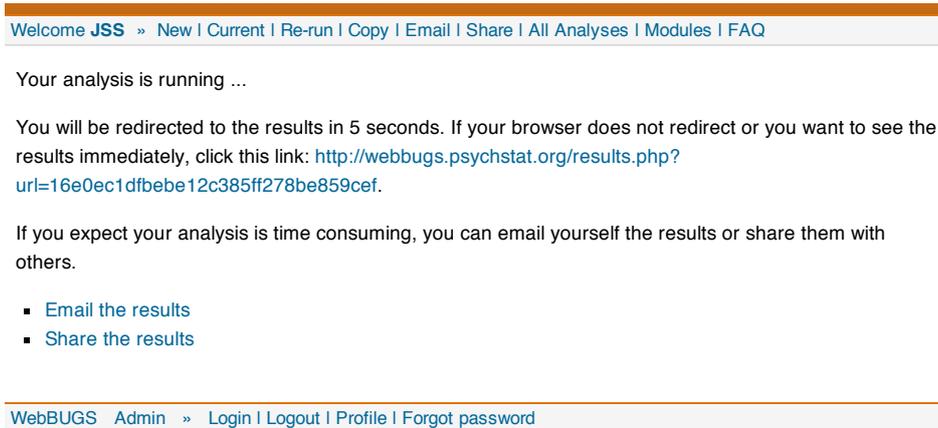


Figure 7: A user will see this page if the update cannot be completed immediately. This page will refresh every 10 seconds to show analysis results until the analysis is completed.

here. By default, 10,000 updates are carried out with the first 4,000 updates discarded as burn-in. The defaults are used for the meta-analysis example. The default thinning rate and random number seed are 1, which can be modified by a user.

3.5. Run analysis

By clicking on the **Next** button in Figure 6, a user will typically see the output page shown in Figure 7. This page will stay for 5 seconds to wait for the results. After 5 seconds, the page will be refreshed automatically to display the results. If an analysis takes less than 5 seconds, a user will see all the results immediately. Otherwise, the page will refresh itself till all results are shown. If an analysis is expected to take a significant amount of time, a user can send the link of the results to his/her email account to view the results later.

3.6. Output

The output of the meta-analysis example is shown in Figures 8–11. Figure 8 shows the first part of the typical **WebBUGS** output including model parameter estimates and DIC. The **Parameter estimates** table (Table 1 in the output) summarizes information for each model parameter monitored. Specifically, for each parameter, the following statistics are reported. First, the posterior mean, posterior median, and posterior standard deviation are given. Second, ESS and the ratio of Monte Carlo error and posterior standard deviation, M/SD , are calculated for each parameter. If the ESS is smaller than 400 or M/SD is larger than 0.05, it is highlighted in red indicating that more updates are needed to obtain an accurate parameter estimate. Third, the 95% percentile credible interval and HPD credible interval are produced for each parameter. Finally, for single chain analysis, the Geweke statistic is provided for convergence diagnostics (Geweke 1992) and for multiple chain analysis, the Gelman-Rubin test statistic is provided (Gelman and Rubin 1992). If the Markov chain for a parameter does not pass the test, the statistic will be highlighted in red. The DIC (Table 2 in the output) along with its components for the model is summarized in the DIC table.

The second part of the output includes the trace, density, and autocorrelation (TDA) plots for each parameter and the deviance of the model. For example, Figures 9 and 11 display

Welcome [Johnny Zhang](#) » [New](#) | [Current](#) | [Re-run](#) | [Copy](#) | [Email](#) | [Share](#) | [All Analyses](#) | [Modules](#) | [FAQ](#)

Results

The program started to run at 22:20:30 on Jan 09, 2014 .

Links to the full model and R coda analysis: [Full model](#) [R CODA](#) [Download coda](#)

More updates: **Number of updates** **Burn-in**

Table 1. Parameter estimates

	Mean	SD	ESS	M/SD	Percent CI		Median	HPD CI		G-R
beta	0.235*	0.041	7727	0	0.153	0.316	0.235	0.152	0.314	1
tau	0.013	0.01	6174	0	0.003	0.039	0.01	0.001	0.031	1.003

Note. * represents a significant parameter;
M/SD is the ratio of Monte Carlo error and standard deviation;
G-R is the Gelman-Rubin diagnostic statistic.

Table 2. DIC

	Dbar	Dhat	pD	DIC
z	-27.64	-35.63	7.997	-19.64
total	-27.64	-35.63	7.997	-19.64

Figure 8: Parameter estimates and DIC of the meta-analysis example.

Trace, density, and autocorrelation plots

Figure 1. Plot for beta. [PDF plot](#)

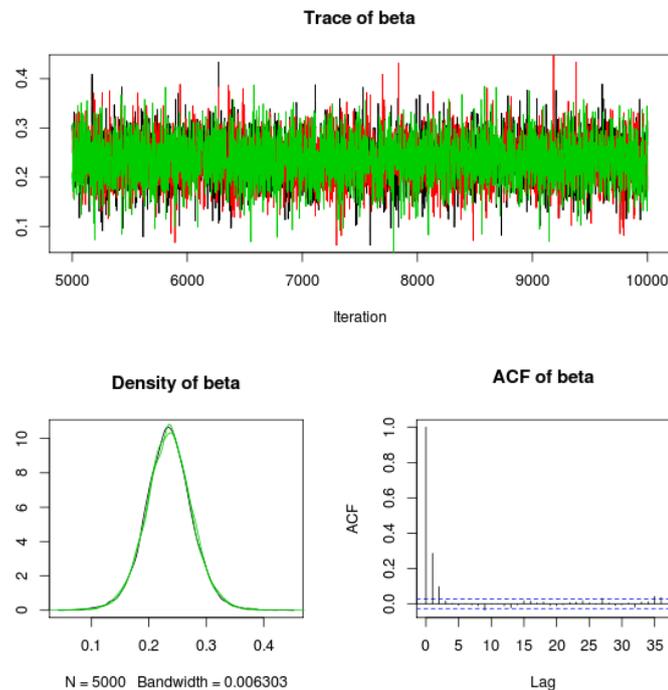


Figure 9: TDA for the parameter **beta**.

the TDAs for the parameters **beta** and **tau**, respectively, in the meta-analysis example. The TDA for the deviance of the model is also produced in the output as shown in Figure 10. Note that the PDF format of the plots can be downloaded using the link [PDF plot](#) for better

Figure 2. Plot for deviance. PDF plot

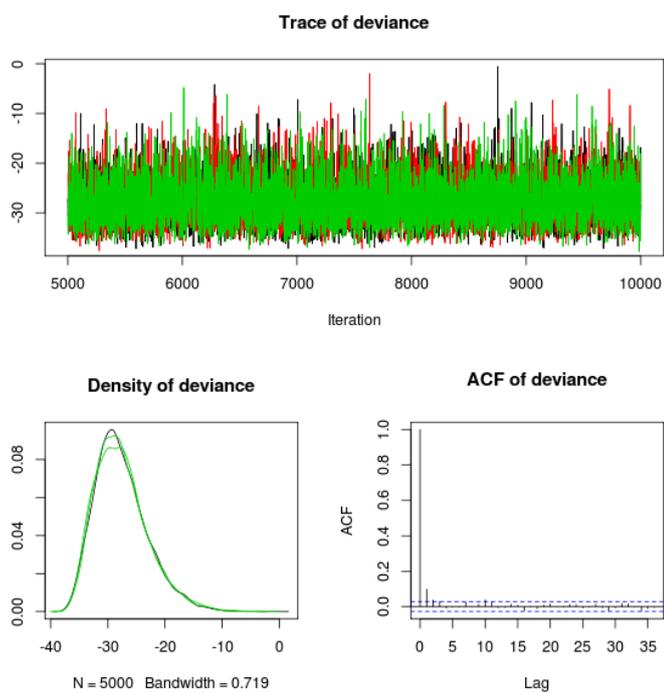


Figure 10: TDA for the deviance of the meta-analysis model.

Figure 3. Plot for tau. PDF plot

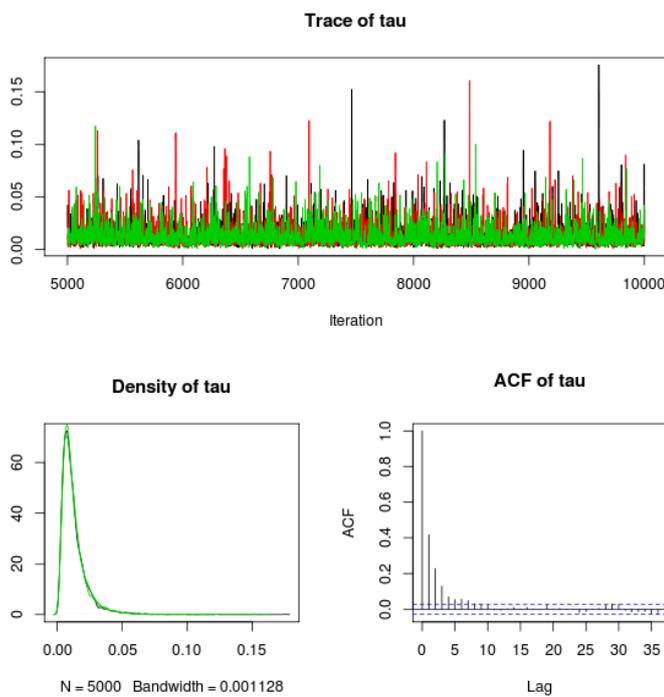


Figure 11: TDA for the parameter τ .

quality.

From the trace plot of each parameter in the TDA plots, one can conclude the Markov chains converged well for the meta-analysis example. Furthermore, the ACF plots show relative small autocorrelation, indicating that the Markov chains mixed well. These findings are consistent with the Gelman-Rubin test of convergence and the effective sample size. The density plots show that the distribution of the parameter `beta` is approximately symmetric while the distribution of the parameter `tau` is skewed to the right.

Given the convergence of the Markov chains, one can interpret the results of the meta-analysis. For example, the overall effect for the correlation between HPWS and financial performance is about 0.235 with the HPD interval [0.152, 0.314], reflecting a small to medium, statistically significant effect.

4. Other features of WebBUGS

In addition to the basic functionality of **WebBUGS** as illustrated in the previous section, **WebBUGS** has many other useful features.

4.1. Email notification

By checking the checkbox of **Email notification** on the top right corner of the index page as shown in Figure 4, a user can receive email notification of the completion of an analysis. A link is provided in the email and the user can click on it to view the analysis. This feature is especially useful when an analysis is expected to take a substantial amount of time.

4.2. Manage analyses

A user can view all of his/her analyses by clicking the **All Analyses** link. All the analyses for the user will be listed in a table with the name of the analysis and the time when the analysis was conducted. For example, all the analyses conducted by the user `jssdemo` are listed in Figure 12. A user can only view his/her own analysis. By clicking on the name of the analysis, a user will be taken to the output of the analysis. To delete an analysis, one can check the checkbox of the analysis and hit the **Delete** button at the bottom. Everything related to the analysis will be removed and cannot be recovered once a user has chosen to delete the analysis.

4.3. Share an analysis

An analysis can be easily shared with others. To share the current analysis, click the **Share** link. A page as in Figure 13 will be shown. On this page, a user can specify the name of the person to share the analysis with and his/her email address. By clicking on the **share** button, an email with the link to the shared analysis will be sent to the email address provided. The recipient of the email can view the shared analysis by following the link in the email. To share an old analysis, one can first access the results of it from a list of analyses and then share it using the above procedure. Note that the recipient can only view the specific analysis shared with him/her. If the user wants to share another analysis with the same or another recipient, the above procedure needs to be repeated.

Welcome **JSS** » [New](#) | [Current](#) | [Re-run](#) | [Copy](#) | [Email](#) | [Share](#) | [All Analyses](#) | [Modules](#) | [FAQ](#)

All Analyses

You can view or delete your previous analysis here.

	Name of analysis	Time
<input type="checkbox"/>	Meta-analysis of correlation	January 1, 2014, 10:01 pm
<input type="checkbox"/>	Rats: a normal hierarchical model for JSS	December 25, 2011, 3:59 pm
<input type="checkbox"/>	Dogs: loglinear model for binary data	December 22, 2011, 11:27 am
<input type="checkbox"/>	Pumps: an example with two chains	December 22, 2011, 10:21 am
<input type="checkbox"/>	Rats: a normal hierarchical model	December 19, 2011, 7:49 pm
<input type="checkbox"/>	Re-run the Rats example	December 19, 2011, 7:44 pm

Total: 1.

WebBUGS Admin » [Login](#) | [Logout](#) | [Profile](#) | [Forgot password](#)

Figure 12: List of all analyses by the user `jssdemo`.

Welcome **Johnny Zhang** » [New](#) | [Current](#) | [Re-run](#) | [Copy](#) | [Email](#) | [Share](#) | [All Analyses](#) | [Modules](#) | [FAQ](#)

Share analysis

Names of the recipient:

Email address:

WebBUGS Admin » [Login](#) | [Logout](#) | [Profile](#) | [Forgot password](#)

Figure 13: An analysis can be shared.

4.4. Multiple-chain analysis

A multiple-chain analysis can be initialized in **WebBUGS** by providing multiple sets of initial values. For example, the meta-analysis was conducted using three chains. Therefore, for the initial values field, three sets of starting values were used. **WebBUGS** used the three sets of initial values to initialize three chains.

4.5. Highlighted output

The results in the `Parameter estimates` table are highlighted in different ways. First, if a single chain does not converge according to the Geweke test, e.g., a Geweke statistic larger than 1.96 by default, the Geweke statistic will be highlighted in red. For the multiple chain analysis, if the Gelman-Rubin statistic is larger than 1.05, the statistic will be highlighted. If either the Geweke statistic or the Gelman-Rubin statistic for a parameter is highlighted, a longer Markov chain or a different burn-in period may be needed. Second, if the M/SD for a given parameter is larger than 0.05, it will be highlighted. If a Markov chain is converged, a large M/SD value typically means that a longer Markov chain is needed. Third, if the effective sample size is smaller than 400, it will be highlighted to alert a user to increase the length of the Markov chains.

WEBBUGS
Conducting Bayesian Analysis Online

Welcome [Johnny Zhang](#) » [New](#) | [All Analyses](#) | [Modules](#) | [FAQ](#)

Type R commands for direct analysis of sample output

CODA files will be read automatically by the following R codes.

```
library("coda")
mcmc.out<-read.openbugs("coda")
```

Please type R code you want to implement for coda analysis.

```
summary(mcmc.out)
gelman.plot(mcmc.out)
```

WebBUGS Admin » [Login](#) | [Logout](#) | [Profile](#) | [Forgot password](#)

Figure 14: CODA analysis through R.

4.6. Change the number of updates and burn-in

Many times one may need to increase the number of updates and change the burn-in period. This can be done on the results page. On the results page shown in Figure 8, there is a section on the top called **More updates**. In the field of **Number of updates**, a user can specify how many more updates to conduct. A user can also change the burn-in period by specifying a number in the **Burn-in** field. If a user only wants to change the burn-in, the **Number of updates** field can be filled with 0. By clicking on the **Update** button, the results will be updated.

4.7. R CODA analysis

The output from **WebBUGS** can be further analyzed in R, for example, using the **coda** package (Plummer *et al.* 2006). High quality plots in PDF format can be generated and additional convergence diagnostics can be conducted. To initialize the CODA analysis, on the output page, click the **R CODA** link. Then a page as in Figure 14 will be shown. A user can input R code to conduct desired analyses. For example, in Figure 14, I aim to obtain the summary statistics for the current CODA and get a Gelman-Rubin convergence diagnostic plot for model parameters using the following R code:

```
R> summary(mcmc.out)
R> gelman.plot(mcmc.out)
```

By clicking the **submit** button, the results will be shown for the current CODA analysis. The Web browser may refresh itself to display all results. If your Web browser does not refresh automatically, please refresh it manually. For example, the output for the analysis in Figure 14 is given below. Note the first part reminds users of the generated plot and provides a link to the plot (the plot is given in Figure 15). The second part is the typical output of an R CODA analysis. Each session of CODA analysis needs to be conducted separately but the number of commands are not limited.

CODA analysis

Figures have been generated in the analysis, [click here](#) to view the plot.

```
> library("coda")
Loading required package: lattice
> mcmc.out <- read.openbugs("coda")
> summary(mcmc.out)
```

```
Iterations = 1:10000
Thinning interval = 1
Number of chains = 3
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
beta	0.2343	0.040604	2.344e-04	3.311e-04
deviance	-27.6414	4.618114	2.666e-02	3.284e-02
tau	0.0127	0.009967	5.754e-05	9.411e-05

2. Quantiles for each variable:

	2.5%	25%	50%	75%	97.5%
beta	0.153200	0.208700	0.2345	0.25980	0.31510
deviance	-34.780000	-30.980000	-28.2700	-25.01000	-16.87000
tau	0.002703	0.006534	0.0101	0.01577	0.03816

```
> gelman.plot(mcmc.out)
```

4.8. Download the model and CODA files

After running an analysis, a user can download the full **OpenBUGS** model with model specification, data, and initial values. The full model can be either saved for future reference or being used within **OpenBUGS**. In addition, a user can download the CODA files generated from a model for other purposes, e.g., to be processed in R. The links to the full model and CODA files are provided on the output page as shown in Figure 8.

4.9. Re-run or copy an analysis

Occasionally, there may be errors in the model, data, or initial values. Error information is displayed in the output of the analysis. A quick way to correct the errors and re-run the analysis is to click on the **Re-run** link. Then a user will see a page similar to the one in Figure 5. The user can then correct possible errors on this page and redo the analysis. The user can also simply change the initial values for better convergence using this feature. If a user wants to fit a new model to the same data or the same model to a new set of data,

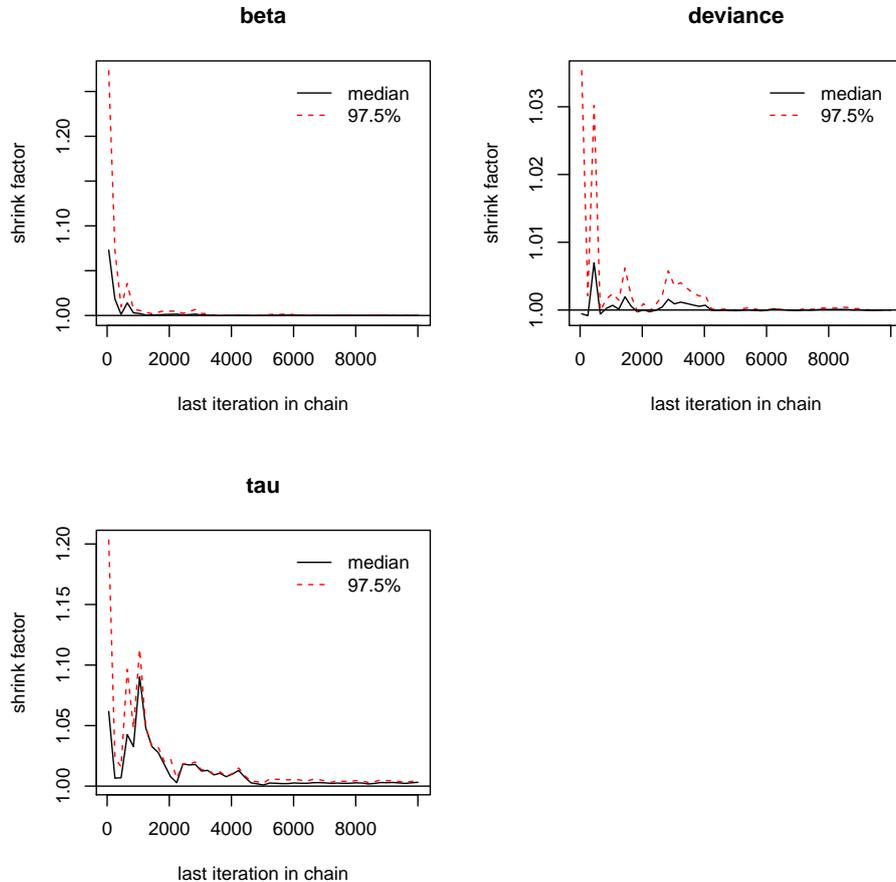


Figure 15: Gelman-Rubin convergence diagnostic plot for meta-analysis generated using R.

he/she can copy the model to a new analysis using the **Copy** link.

4.10. Wiki and online help

A Wiki is incorporated within **WebBUGS** to provide online help to users. For example, in Figure 5, a **Help** link appears at three places. If a user clicks on the link, he/she will be directed to a Wiki help page. The Wiki can also be updated easily online to include as much information as possible by any user.

5. WebBUGS modules

WebBUGS allows development of modules for Bayesian analysis. A module consists of web pages for inputting information within a web browser and R code on the server to carry out certain analyses. In this paper, I focus on the discussion of two modules: a data management module and a meta-analysis module. The full list of completed modules, 4 in total as of 2014, can be accessed through the **Modules** link.

WEBBUGS
Conducting Bayesian Analysis Online

Welcome **JSS** » [New](#) | [All Analyses](#) | [Modules](#) | [FAQ](#)

Convert input data

List of data sets

No file chosen

<input type="checkbox"/>	File name	Edit	Edit in a table	Convert to BUGS data	Time
<input type="checkbox"/>	metadata.txt	Edit	Spreadsheet edit	Convert	2014.01.02 13:41:41.
<input type="checkbox"/>	metadata.txt.bug	Edit			2014.01.02 13:43:31.

WebBUGS Admin » [Login](#) | [Logout](#) | [Profile](#) | [Forgot password](#)

Figure 16: The interface of the data management module.

WEBBUGS
Conducting Bayesian Analysis Online

Welcome **Johnny Zhang** » [New](#) | [All Analyses](#) | [Modules](#) | [FAQ](#)

Edit data

```
n c o r a x
215 0.21 1 1
132 0.252 1 1
309 0.123 1 1
117 0.33 1 1
307 0.4 0.8 0
1212 0.34 1 1
175 0.11 1 1
380 0.147 1 1
86 0.11 1 1
74 0.36 1 1
361 0.138 1 1
```

WebBUGS Admin » [Login](#) | [Logout](#) | [Profile](#) | [Forgot password](#)

Figure 17: Edit data in a textarea.

5.1. Data management module

The direct link to the data management module is <http://WebBUGS.psychstat.org/modules/data/>. Using the data management module, users can manage their data online, including uploading, deleting, editing, and converting data. The interface of the data management module is shown in Figure 16.

First, the available data sets are listed in a table. Both file names and the time last modified are shown. Second, a new data set can be uploaded. To upload a file, one can first **Choose** or **Browse** the file on the local computer and then click the **Upload** button. If a file is uploaded successfully, it will be shown in the data list. Error information will be returned if upload fails.

WEBBUGS
 Conducting Bayesian Analysis Online

Welcome **Johnny Zhang** » [New](#) | [All Analyses](#) | [Modules](#) | [FAQ](#)

Edit data

	n	cor	a	x
1	215	0.21	1	1
2	132	0.252	1	1
3	309	0.123	1	1
4	117	0.33	1	1
5	307	0.4	0.8	0
6	1212	0.34	1	1
7	175	0.11	1	1
8	380	0.147	1	1
9	86	0.11	1	1
10	74	0.36	1	1
11	361	0.138	1	1

[WebBUGS](#) [Admin](#) » [Login](#) | [Logout](#) | [Profile](#) | [Forgot password](#)

Figure 18: Edit data in a spreadsheet.

Only text files with extensions `.txt` (space separated data file) or `.csv` (comma separated value file) are allowed to be uploaded. Third, a data set can be edited either in a textarea or as a spreadsheet. By clicking on the **Edit** link in the data list, the content of the data file will be shown in a textarea for editing as shown in Figure 17. The edit can be saved by clicking the button **Save** or **Save and go back to data list**. If a data file has less than 200 rows of data, it can be edited in the spreadsheet mode as shown in Figure 18, which allows the edit of individual elements.

The data management module can also convert data, either from a data file or user input, into the **OpenBUGS** format. To convert data from a data file, click the link **Convert** in the data list. Then the screen as in Figure 19 is shown. On the screen, a user can input required data information for **OpenBUGS**. Each row of input represents a piece of information. Any R data manipulation code can be used here. After input, one can click on the button **Convert data**. The **OpenBUGS** format data are shown in a textarea that can be further edited or modified. The converted data are also saved to a file with the extension name `.bug` that can be used for Bayesian analysis in **OpenBUGS**. An example is given in Figure 19.

To convert user input data, one can click the **Convert input data** link. Then a screen as in Figure 20 will appear. On the screen, one can simply input the data. Each piece of information is separated by a semi-colon `;`. For each piece of information, its name is given first and then followed by the equal sign `=`. A scalar, vector, or matrix can be used. For a vector, a list

WEBBUGS
 Conducting Bayesian Analysis Online

[Welcome JSS](#) » [New](#) | [All Analyses](#) | [Modules](#) | [FAQ](#)

Head of the data file

	n	r	a
1	215	0.21	1
2	132	0.252	1
3	309	0.123	1
4	117	0.33	1

Type code below (See [help](#) for instruction)

```
m = nrow(data)
n = n
r = r
a = a
```

BUGS data

```
list(
m
=
11, n
=c(
215, 132, 309, 117, 307, 1212, 175, 380, 86, 74, 361
), r
=c(
2.100000E-01, 2.520000E-01, 1.230000E-01, 3.300000E-01, 4.000000E-01, 3.400000E-01, 1.100000E-01,
1.470000E-01, 1.100000E-01, 3.600000E-01, 1.380000E-01
), a
=c(
1.000000E+00, 1.000000E+00, 1.000000E+00, 1.000000E+00, 8.000000E-01, 1.000000E+00, 1.000000E+00,
1.000000E+00, 1.000000E+00, 1.000000E+00, 1.000000E+00
)
)
```

[WebBUGS](#) [Admin](#) » [Login](#) | [Logout](#) | [Profile](#) | [Forgot password](#)

Figure 19: Convert data in a data file to **OpenBUGS** format.

of values need to be provided. One can change to a new line if needed. For a matrix, the dimension of the matrix is given first and then the data information. The dimension and data values are separated by a comma ,. Figure 20 gives examples on the input of data. Note that a data matrix can be copied and pasted directly. By clicking the button **Convert data**, the **OpenBUGS** format data are shown as in Figure 20. As in the meta-analysis, multiple sets of data can be converted by separating them using a vertical bar |.

5.2. Meta-analysis module

The meta-analysis module carries out Bayesian meta-analysis with power prior proposed by Zhang *et al.* (2013a). The URL for this module is <http://WebBUGS.psychstat.org/modules/metacorr/>. The module has an interface shown in Figure 21.

To use the module, one needs to either upload a new data file or select an existing file. To upload a new file, in the DATA box, click on the button **Choose file** or **Browse ...** or other text depending on your web browser. Then, a user will be directed to select a file on

The screenshot shows the WebBUGS interface. At the top, there is an orange header with the text "WEBBUGS Conducting Bayesian Analysis Online". Below the header, there is a navigation bar with links: "Welcome JSS » New | All Analyses | Modules | FAQ". The main content area has a text input field with the instruction "Type code below (See help for instruction)". The input field contains the following text:

```
N = 10;
x = 5 2,
1 2
3 4
5 6
7 8
9 10;

y = 1 0 0 1 1
```

Below the input field is a button labeled "Convert data". Underneath the button is a text area labeled "BUGS data" containing the converted OpenBUGS code:

```
list(
N = 10,
x = structure(.Data= c(1,2,3,4,5,6,7,8,9,10), .Dim=c(5,2)),
y = c(1,0,0,1,1)
)
```

At the bottom of the interface, there is a navigation bar with links: "WebBUGS Admin » Login | Logout | Profile | Forgot password".

Figure 20: Convert user input data to **OpenBUGS** format.

his/her computer. All existing data files are automatically shown in the drop down menu of the second method. One can simply select a file to use without uploading a data file. The data should be separated by one or more white spaces. The first line of the data file will be the variable names, which will be used in the meta-analysis model.

With data, a user can provide other information required by meta-analysis. A user can choose to conduct either fixed-effects or random-effects meta-analysis. Detailed information on the model is provided using the variable names provided in the data file. Correlation is required for all analysis and can be specified using the variable names in the data set. For example, in the data set, *r* is the variable name for correlation. Therefore, *r* should be input in the field of **Correlation** in the interface. Sample size is required for all analysis and can be specified using the variable names in the data set. In the example, *n* is used in the **Sample size** field because in the data set, *n* is also the variable name for sample size. A user can also specify the variable for power assigned to each study. If left blank, the power 1 is used for all studies. A user can further specify a reliability variable to adjust for correlation. The adjusted correlation is equal to the original correlation divided by the reliability. For meta-regression analysis, covariates can be provided. More than one covariate should be separated by space.

A user can control the MCMC method and output of the meta-analysis. The total number of Monte Carlo iterations and the burn-in period can be specified. A user can also require the output of the estimates for the random effects, DIC, and diagnostic plots for all model parameters including the random effects. A funnel plot may also be requested for publication bias analysis. If one checks the option **Email notification**, an email will be sent to the user once the analysis is completed. With the required information, one can click the button **Run meta-analysis** to carry out the analysis.

The output of meta-analysis includes: (1) the model in mathematical format, (2) descriptive statistics of the data, (3) DIC if required, (4) parameter estimates, (5) trace, autocorrelation, and density plot for each parameter, (6) funnel plot if required, and (7) test of publication

WEBBUGS
 Conducting Bayesian Analysis Online

Welcome [JSS](#) » [New](#) | [All Analyses](#) | [Modules](#) | [FAQ](#)

Bayesian meta-analysis of correlation through power prior

[MANUAL](#)

DATA: Upload or select a file

Method 1: Upload data (only .txt file allowed)	
<input type="button" value="Choose File"/>	No file chosen
Method 2: Select a data file <input type="button" value="No existing data"/>	

MODEL: Provide the names of the variables in the data set

Name of model:	<input type="text" value="Meta analysis"/>
Type of model:	<input checked="" type="radio"/> Random <input type="radio"/> Fixed
Correlation (r):	<input type="text"/>
Sample size (n):	<input type="text"/>
Power (α):	<input type="text"/>
Reliability:	<input type="text"/>
Covariates (X):	<input type="text"/>

CONTROL MCMC and OUTPUT

Number of iterations:	<input type="text" value="10000"/>
Burn-in:	<input type="text" value="4000"/>
Output:	<input type="checkbox"/> Random effects <input type="checkbox"/> DIC <input type="checkbox"/> Diagnostic plot for all <input type="checkbox"/> Funnel plot <input type="checkbox"/> Email notification

[WebBUGS](#) [Admin](#) » [Login](#) | [Logout](#) | [Profile](#) | [Forgot password](#)

Figure 21: The interface of the meta-analysis module.

bias. An example output of meta-analysis can be found at <http://WebBUGS.psychstat.org/wiki/index.php?page=Meta-analysis+of+correlation>.

6. Install WebBUGS

If preferred, a user can set up their own **WebBUGS** server. The installation of **WebBUGS** requires the setup of a web server with **PHP** and **MySQL** support, as well as the installation of **R** and **OpenBUGS**. Although **WebBUGS** can be installed under both Windows and Linux operating systems (OS), it is recommended to use it under Linux OS for better performance and security. Therefore, I focus on the instruction of installing **WebBUGS** under Linux and briefly discuss how to set it up for Windows.

6.1. Install WebBUGS under Linux OS

There are many varieties of Linux distributions. One popular distribution is Ubuntu. I use

Ubuntu to illustrate the installation of **WebBUGS** but the procedure will work for or can easily be adapted to other Linux distributions.¹

Install Ubuntu

Ubuntu is freely available on its web page at <http://www.ubuntu.com/>. The installation of Ubuntu follows a standard procedure with a graphical user interface as shown at <http://www.ubuntu.com/download/>.

Set up a web server

A simple way to set up a web server for **WebBUGS** is to install **XAMPP** for Linux. **XAMPP** contains the Apache HTTP Server with **MySQL** and **PHP** support and is very easy to install. **XAMPP** can be downloaded from <http://www.ApacheFriends.org/en/xampp-linux.html>. For example, the version 1.8.2 can be downloaded through the terminal under Ubuntu using

```
$ wget http://downloads.sourceforge.net/project/xampp/XAMPP%20Linux/1.8.2\
> /xampp-linux-x64-1.8.2-3-installer.run
```

To install it, issue the command:

```
$ chmod 755 xampp-linux-1.8.2-0-installer.run
$ ./xampp-linux-1.8.2-0-installer.run
```

To start the server, issue the command:

```
$ sudo /opt/lampp/lampp start
```

Detailed instructions on how to set up **XAMPP** can be seen at <http://www.ApacheFriends.org/en/xampp-linux.html>.

Install R

To install R, issue the command:

```
$ sudo apt-get install r-base r-base-dev
```

The R package **coda** is needed for output analysis for **WebBUGS**. To install the package, start R using

```
$ sudo R
```

and within R,

```
R> install.packages("coda", lib = "/usr/local/lib/R/site-library")
```

By default, R is installed as `/usr/bin/R`.

Install OpenBUGS

OpenBUGS for Linux (using the version 3.2.1 as an example) can be downloaded from <http://OpenBUGS.net/w/Downloads/>. To install **OpenBUGS**, issue the commands below:

¹Tested for both Ubuntu Desktop and Server version 11.10.

```

$ wget http://WebBUGS.psychstat.org/public/OpenBUGS-3.2.1.tar.gz
$ tar -zxvf OpenBUGS-3.2.1.tar.gz
$ cd OpenBUGS-3.2.1
$ ./configure
$ make
$ sudo make install

```

By default, the **OpenBUGS** is installed as `/usr/local/lib/OpenBUGS`.

Install WebBUGS

To install **WebBUGS**, first download it using

```
$ wget http://WebBUGS.psychstat.org/public/WebBUGS.linux.tar.gz
```

and expand it to the folder `/opt/lampp/htdocs`. Change the permission of the folders `analysis` and `data` to `777` by using

```
$ sudo chmod -r 777 analysis data
```

A more secure way is to change the owner of the folder to the web user such as `www-data`. Note that both the `analysis` and `data` folders are protected and cannot be accessed directly.

Then, set up a **MySQL** database. The database can be created in a web browser with the address <http://localhost/phpmyadmin/> on the server. If accessing remotely on a user's computer, change `localhost` to the IP address or the domain name of the server. A database called `webbugs` can be created. Then, the following **SQL** scripts can be run to add four tables `builtin`, `modelinfo`, `share`, and `users`.

```

CREATE TABLE IF NOT EXISTS `builtin` (
  `id` int(3) NOT NULL,
  `name` varchar(254) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

```

```

CREATE TABLE IF NOT EXISTS `modelinfo` (
  `id` bigint(100) NOT NULL AUTO_INCREMENT,
  `url` varchar(120) NOT NULL,
  `base` varchar(120) NOT NULL,
  `user` varchar(120) NOT NULL,
  `email` varchar(120) NOT NULL,
  `date` varchar(120) NOT NULL,
  `note` text NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

```

```

CREATE TABLE IF NOT EXISTS `share` (
  `url` varchar(50) NOT NULL,
  `name` varchar(200) NOT NULL,

```

```

`email` varchar(200) NOT NULL,
`password` varchar(100) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

CREATE TABLE IF NOT EXISTS `users` (
  `id` bigint(20) NOT NULL AUTO_INCREMENT,
  `fullname` tinytext COLLATE latin1_general_ci NOT NULL,
  `username` varchar(200) COLLATE latin1_general_ci NOT NULL DEFAULT '',
  `email` varchar(220) COLLATE latin1_general_ci NOT NULL DEFAULT '',
  `level` tinyint(4) NOT NULL DEFAULT '1',
  `passwd` varchar(220) COLLATE latin1_general_ci NOT NULL DEFAULT '',
  `address` text COLLATE latin1_general_ci NOT NULL,
  `date` date NOT NULL DEFAULT '0000-00-00',
  `approved` int(1) NOT NULL DEFAULT '0',
  PRIMARY KEY (`id`),
  UNIQUE KEY `user_email` (`email`),
  FULLTEXT KEY `idx_search` (`fullname`,`address`,`email`,`username`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

```

```

INSERT INTO `users` (`fullname`, `username`, `email`, `level`,
`passwd`, `address`, `date`, `approved`) VALUES
('JSS', 'jssdemo', 'jssdemo@localhost', 5,
md5('jssdemo'), 'address', '2011-12-05', 1);

```

Note that the username and password jssdemo can and should be changed accordingly. After creating the database, edit the configuration file `/opt/lampp/htdocs/includes/config.php`. The current content of it is shown below.

```

// Website url
define("WWWURL", "XXXXXX");
define("BUGSPATH", "XXXXXX");
define("SERVERURL", "XXXXXX");
define("SERVERPATH", "XXXXXX");

//User Database
define ("DB_HOST", "localhost");
define ("DB_USER", "XXXXXX");
define ("DB_PASS", "XXXXXX");
define ("DB_NAME", "webstats");

//Email through smtp
define ("EMAIL_ADDRESS", "XXXXXX");
define ("EMAIL_PASS", "XXXXXX");

```

The paths in the file should be specified correctly. WWWURL is the web address to access the server. BUGSPATH is the command to start **OpenBUGS**. SERVERURL is the folder where the analysis is saved and SERVERPATH is the path where **WebBUGS** is installed. The **MySQL**

username `DB_USER` and password `DB_PASS` should also be specified. The default installation of **XAMPP** uses `root` as the username with an empty password. **WebBUGS** by default uses gmail to send emails. To use it, specify the `EMAIL_ADDRESS` and `EMAIL_PASS` for your gmail account.

Now, one can start to use **WebBUGS**. In a web browser, type in `http://localhost/`. If installed correctly, one should see the page as shown in Figure 3. The default user name and password `jssdemo` can be used to log in. One should change the password for `jssdemo` after login if not doing so during the phase of creating the database.

6.2. Install WebBUGS under Windows

To install **WebBUGS** under Windows, download the Windows version of **XAMPP** (<http://www.Apachefriends.org/en/xampp-windows.html>), **R** (<http://CRAN.R-project.org/bin/windows/>), **OpenBUGS** (<http://OpenBUGS.net/w/Downloads/>), and **WebBUGS** (<http://WebBUGS.psychstat.org/jss/WebBUGS.windows.zip>) using the given links.² It is recommended to install **XAMPP**, **R**, and **OpenBUGS** in one folder and avoid spaces in all the directories. A sample configure file (`C:/xampp/htdocs/includes/config.php`) for **WebBUGS** on Windows looks like

```
// Website url
define("WWWURL", "http://localhost/");
define("BUGSPATH", "C:/xampp/OpenBUGS/OpenBUGS321/OpenBUGS");
define("SERVERURL", "C:/xampp/htdocs/analysis");
define("SERVERPATH", "C:/xampp/htdocs");
//User Database
define ("DB_HOST", "localhost"); // MySQL server address
define ("DB_USER", "root"); // MySQL user name
define ("DB_PASS", ""); // MySQL user password
define ("DB_NAME", "webbugs"); // MySQL database name
//Email through smtp
define ("EMAIL_ADDRESS", "XXXXX"); // Email address to send messages
define ("EMAIL_PASS", "XXXX"); // Password of the email
```

7. Future work

Future development of **WebBUGS** will focus on several areas. First, **WebBUGS** will incorporate **JAGS** and **stan** (Stan Development Team 2014) so that a user can select which software to use for MCMC. Second, the interface of **WebBUGS** will be improved as requested by the users. Third, in the future, a dedicated web server and computing server will be used to run **WebBUGS**, which currently uses a single machine as web and computing server.

Fourth, most of **WebBUGS**'s future development will focus on the development of modules for **WebBUGS**. Presently, the data management module only supports plain text data files. A development goal is to allow the use of data files from other statistical packages such as **SAS** (SAS Institute Inc. 2011) and **SPSS** (IBM Corporation 2013). **WebBUGS** allows basic

²The installation was tested under Windows 7.

analysis of MCMC output. In the future, a CODA module will be developed for more flexible analysis of MCMC output.

Other modules will also be developed including the regression module, the item response theory (IRT) module, and the growth curve analysis module. The regression module will allow a user to conduct multiple regression, logistic regression, Poisson regression, and tobit regression by simply specifying the dependent and independent variables. The IRT module will initially include the one-parameter model, the two-parameter model, and the three-parameter model. The growth curve analysis module will allow the Bayesian estimation of linear growth curve models, quadratic growth curve models, exponential growth curve models, tobit growth curve models, and robust growth curve models (e.g., Wang, Zhang, McArdle, and Salthouse 2008; Zhang, Lai, Lu, and Tong 2013b).

References

- Bacco MD, D'Amore G, Scalfari F (eds.) (2004). *Applied Bayesian Statistical Studies in Biology and Medicine*. Springer-Verlag, Norwell.
- Brooks SP, Roberts GO (1998). "Assessing Convergence of Markov Chain Monte Carlo Algorithms." *Statistics and Computing*, **8**(4), 319–335.
- Congdon P (2003). *Applied Bayesian Modelling*. John Wiley & Sons, New York.
- Cowles MK, Carlin BP (1996). "Markov Chain Monte Carlo Convergence Diagnostics: A Comparative Review." *Journal of the American Statistical Association*, **91**(434), 883–904.
- Gelman A, Carlin JB, Stern HS, Vehtari A, Rubin DB (2013). *Bayesian Data Analysis*. 3rd edition. Chapman & Hall/CRC, Boca Raton.
- Gelman A, Rubin D (1992). "Inference from Iterative Simulation Using Multiple Sequences." *Statistical Science*, **7**(4), 457–511.
- Geweke J (1992). "Evaluating the Accuracy of Sampling-Based Approaches to Calculating Posterior Moments." In JM Bernardo, JO Berger, AP Dawid, AFM Smith (eds.), *Bayesian Statistics 4*, pp. 169–193. Clarendon Press, Oxford.
- Gill J (2007). *Bayesian Methods: A Social and Behavioral Sciences Approach*. 2nd edition. Chapman & Hall/CRC, Boca Raton.
- Grandy Jr WT, Schick LH (eds.) (2007). *Maximum Entropy and Bayesian Methods (Fundamental Theories of Physics)*. Springer-Verlag, Boston.
- Hunter JE, Schmidt FL (2004). *Methods of Meta-Analysis: Correcting Error and Bias in Research Findings*. Sage, Newbury Park.
- IBM Corporation (2013). *IBM SPSS Statistics 22*. IBM Corporation, Armonk. URL <http://www.ibm.com/software/analytics/spss/>.
- Ibrahim JG, Chen MH (2000). "Power Prior Distributions for Regression Models." *Statistical Science*, **15**(1), 46–60.

- Kruschke JK (2011). *Doing Bayesian Data Analysis: A Tutorial with R and BUGS*. Academic Press, Burlington.
- Lee SY (2007). *Structural Equation Modeling: A Bayesian Approach*. John Wiley & Sons, New York.
- Lunn D, Jackson C, Best N, Thomas A, Spiegelhalter D (2012). *The BUGS Book: A Practical Introduction to Bayesian Analysis*. Chapman & Hall/CRC, Boca Raton.
- Lunn D, Spiegelhalter D, Thomas A, Best N (2009). “The **BUGS** Project: Evolution, Critique and Future Directions.” *Statistics in Medicine*, **28**(25), 3049–3082.
- Martin AD (2008). “Bayesian Analysis.” In JM Box-Steffensmeier, HE Brady, D Collier (eds.), *Oxford Handbook of Political Methodology*. Oxford University Press, Oxford.
- Moore DS (1997). “Bayes for Beginners? Some Pedagogical Questions.” In S Panchapakesan, N Balakrishnan, Gupta, S Swarup (eds.), *Advances in Statistical Decision Theory and Applications*, pp. 3–17. Birkhäuser, Boston.
- Muthén B, Asparouhov T (2012). “Bayesian SEM: A More Flexible Representation of Substantive Theory.” *Psychological Methods*, **17**(3), 313–335.
- Ntzoufras I (2009). *Bayesian Modeling Using WinBUGS*. John Wiley & Sons, Hoboken.
- Plummer M (2003). “**JAGS**: A Program for Analysis of Bayesian Graphical Models Using Gibbs Sampling.” In K Hornik, F Leisch, A Zeileis (eds.), *Proceedings of the 3rd International Workshop on Distributed Statistical Computing (DSC 2003)*. Vienna, Austria.
- Plummer M, Best N, Cowles K, Vines K (2006). “**coda**: Convergence Diagnosis and Output Analysis for MCMC.” *R News*, **6**(1), 7–11. URL <http://CRAN.R-project.org/doc/Rnews/>.
- R Core Team (2014). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL <http://www.R-project.org/>.
- Rupp AA, Dey DK, Zumbo BD (2004). “To Bayes or Not to Bayes, from Whether to When: Applications of Bayesian Methodology to Modeling.” *Structural Equation Modeling*, **11**(3), 424–451.
- SAS Institute Inc (2011). *The SAS System, Version 9.3*. SAS Institute Inc., Cary. URL <http://www.sas.com/>.
- Song XY, Lee SY (2012). *Basic and Advanced Bayesian Structural Equation Modeling: With Applications in the Medical and Behavioral Sciences*. John Wiley & Sons, New York.
- Stan Development Team (2014). “**Stan**: A C++ Library for Probability and Sampling, Version 2.4.” URL <http://mc-stan.org/>.
- Wang L, McArdle JJ (2008). “A Simulation Study Comparison of Bayesian Estimation with Conventional Methods for Estimating Unknown Change Points.” *Structural Equation Modeling*, **15**(1), 52–74.

- Wang L, Zhang Z, McArdle JJ, Salthouse TA (2008). “Investigating Ceiling Effects in Longitudinal Data Analysis.” *Multivariate Behavioral Research*, **43**(3), 476–496.
- Winkler RL (2001). “Why Bayesian Analysis Hasn’t Caught on in Healthcare Decision Making.” *International Journal of Technology Assessment in Health Care*, **17**(1), 56–66.
- Zhang Z, Hamagami F, Wang L, Grimm KJ, Nesselroade JR (2007). “Bayesian Analysis of Longitudinal Data Using Growth Curve Models.” *International Journal of Behavioral Development*, **31**(4), 374–383.
- Zhang Z, Jiang K, Liu H (2013a). “Bayesian Meta-Analysis of Correlation Coefficients through Power Prior.” arXiv:1401.2054 [stat.ME], URL <http://arxiv.org/abs/1401.2054>.
- Zhang Z, Lai K, Lu Z, Tong X (2013b). “Bayesian Inference and Application of Robust Growth Curve Models Using Student’s t Distribution.” *Structural Equation Modeling*, **20**(1), 47–78.

Affiliation:

Zhiyong Zhang
Department of Psychology
University of Notre Dame
118 Haggard Hall
Notre Dame, IN 46556, United States of America
E-mail: zzhang4@nd.edu
URL: <http://nd.psychstat.org/>