DOI: 10.31276/VMOSTJOSSH.64(1).85-96

Bayesvl: an R package for user-friendly Bayesian regression modelling

Quan-Hoang Vuong[1, 2], Minh-Hoang Nguyen[1], Manh-Toan Ho[1*]

[1]*Centre for Interdisciplinary Social Research, Phenikaa University, Hanoi, Vietnam*
[2]*Vietnam Institute for Advanced Study in Mathematics, Ministry of Education and Training*

Received 5 October 2021; accepted 23 February 2022

Abstract:

Compared with traditional statistics, only a few social scientists employ Bayesian analyses. The existing software programs for implementing Bayesian analyses such as OpenBUGS, WinBUGS, JAGS, and rstanarm can be daunting given that their complex computer codes involve a steep learning curve. In contrast, this paper introduces a new open software for implementing Bayesian network modelling and analysis: the bayesvl R package. The package aims at providing an intuitive gateway for beginners of Bayesian statistics to construct and analyse mathematical models in social sciences. To achieve this aim, the bayesvl package integrates three core functions seamlessly: (i) designing Bayesian network models using directed acyclic graphs (DAGs) of bnlearn, (ii) generating attractive visualization of ggplot2, and (iii) simulating data and computing posterior distribution using the Markov chain Monte Carlo (MCMC) algorithms of rstan and rethinking. A case example illustrates how the bayesvl package helps leverage users' intuition in creating and evaluating mathematical models of their social scientific problems while minimizing the daunting aspect of writing complex computer codes.

Keywords: **Bayesian network, bayesvl, ggplot2, mathematical model, MCMC.**

Classification number: **7**

Introduction

While there are many straightforward philosophical reasons for the greater widespread use of Bayesian statistics, it appears that, in practice, Bayesian statistical analysis is still limited to only a minority of social and behavioural scientists. The Bayesian approach is methodologically similar to the way one's intuition works [1]. There are several factors that provide Bayesian statistics with strengths over traditional statistics, especially in the realm of social sciences and humanities. Firstly, the Bayesian inference process incorporates both mathematics and background knowledge in the model [2, 3]. Second, actual observations are the primary concern when checking the strength of evidence, unlike the assumption of an infinite number of observations in conventional statistics. Moreover, even when background knowledge is incorporated, the final results can still be computed appropriately with one model able to be quantitatively compared with other models [3, 4]. Here,

[*] *Corresponding author: Email: toan.homanh@phenikaa-uni.edu.vn*

the Bayesian approach enables researchers to utilise mathematical tools with intuition and common sense [5]. In essence, when using Bayesian statistics, a researcher will use prior distributions to reflect their initial beliefs (informed background knowledge, sometimes even intuition), then update their initial beliefs with new data to form posterior distributions [6-8]. However, social scientists are still more familiar with the conventional statistics [9]. For instance, studies in top sociology journals using Bayesian statistics are virtually non-existent [10].

One of the factors that contribute to this issue is the availability of friendly software for conventional statistics such as SPSS, Stata, or SAS. As for the lack of intuitive software, the current software packages for Bayesian analysis such as require OpenBUGS [11], JAGS [12], MCMCglmm [13], Stan [14], brms [15], rethinking [16], and rstanarm [17] require users to be highly familiar with a command-line interface. The users must code the model from scratch, which can entail a steep learning curve for many statistical novices. Thus, we created an R package called bayesvl to help social scientists focus on their research problems without being caught up in computational details. With relative ease-of-use and minimal coding, the bayesvl R package can help create Bayesian network models and automatically generate stan codes of the graphical structure of Bayesian networks to perform sampling and parameter learning. It can also perform the Hamiltonian MCMC simulations and provide multiple visualization tools such as a Bayesian network graph, a bar chart for conditional probabilities, a trace plot, a Gelman shrink factor plot, an autocorrelation effect plot, a highest posterior distribution intervals plot, a density plot, a pairwise parameter comparison plot, etc. Moreover, it can produce a number of statistical and computational checks such as calculating conditional probabilities, the effective sample size, or r-hat statistics. Finally, it supports a detailed model comparison process with the Pareto-smoothed importance sampling leave-one-out cross-validation approach (PSIS-LOO-CV) [18], the widely applicable information criterion (WAIC) [19], and the Bayesian stacking weights [4, 20]. In this paper, we will briefly introduce the core functions of bayesvl and demonstrate its functions with a real-life example.

The bayesvl R package

The bayesvl project was launched in 2017 [21, 22] and the package was eventually published in CRAN [23] and Github [24] in 2019. There were several components that contributed to our conception of the software: the visualization capability of R and the causality and uncertainty inherent to Bayesian Network modelling [25]. Simulated data using the MCMC algorithm makes this package more scientific for social science research in the age of Big Data and is visually appealing and intuitive to the readers [26]. To this end, the bayesvl R package was developed referring to rethinking [16] and rstanarm [17] for MCMC simulation method; bnlearn [27, 28] for Bayesian network modelling; and ggplot2 for beautiful and flexible data visualization.

The model fitting procedure of the bayesvl package is relatively simple. First, it is recommended that users install version 3.5.1 or more recent versions of R software. Then, the ggplot2, rstan, and the bayesvl packages need to be installed. Then, a diagram needs to be built to visualize the relationship between variables.

Moreover, visualization supports four cognitive mechanisms: reinterpretation, abstraction, combination, and mapping [29, 30]. The bayesvl package provides simple code to help researchers build the diagram. Other packages for Bayesian statistics such as brms, MCMCglmm, rstanarm, and rethinking tend to require the user to code mathematical formulae, which can be intimidating. Thus, the simplicity of bayesvl code helps researchers ease into to the world of Bayesian statistics. Next, the Stan code for model fitting can be created automatically with the bayesvl package's link to rstan. Finally, the bayesvl package allows the MCMC results to be viewed in both numbers and charts.

Case example

In the following section, we will examine the model fitting procedures and graphical ability of the bayesvl package through a real-life example: a nested multi-level Bayesian network with varying intercept, which analyses how the satisfaction of medical patients are affected by their socioeconomic, residence, and insurance status, and the outcome of their medical treatment. The dataset for this real-life example was deposited in the Open Science Framework (OSF)'s depository and contained 1042 observations on health insurance, health care, and socioeconomic status [31, 32].

Model construction

The statistical problem at hand is to investigate the level of a patient's financial

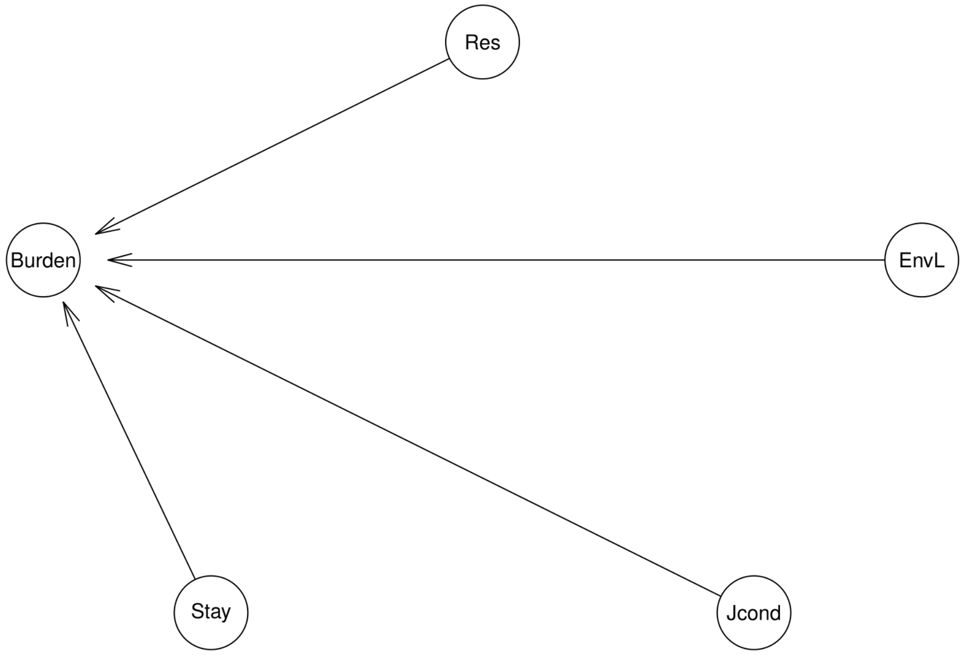

Fig. 1. A nested multi-level Bayesian network with varying intercepts analyses how a patient's financial burden is associated with their residence, length of hospital stay, employment condition, and the amount of 'thank you money'.

burden given their socio-residence status, length of hospital stay, employment condition, and the amount of 'thank you money' included in the treatment fee. First, one can create a visual representation of the relationships among the variables, i.e., a relationship tree. Fig. 1 presents a visualization of this example.

Explanatory variables:

- *Res:* whether a patient lives in the same region as the hospital; yes=1; no=2.
- *Stay:* the amount of time a patient stays at the hospital; under 10 days (short)=0; or more than 10 days (long)=1.
- *Jcond:* the employment condition of the patient; stable=3; unstable=2; or unemployed=1.
- *Envl:* the amount of 'thank you money' that the patient must pay with the treatment fee; high (>15%)=3; medium (7-15%)=2; low (<7%)=1.

Response variable:

- *Burden:* the self-reported financial situation of the patient after the treatment; minimally affected (A)=1; adversely affected (B)=2; destitute (C)=3; adversely destitute (D)=4.

As one can see, the model is a nested multi-level Bayesian network, which enables simultaneous analyses of individual quantities to be performed [2]. This is also a direct example of how Bayesian modelling leverages the background knowledge of social science researchers [9]. It is true that data sets that pool data over multiple units are commonplace in social sciences; for example, people can attend different schools, companies trade in different markets, and votes live in different communities. These causal structures are captured in Bayesian hierarchical models.

In a simple regression model, we have the following mathematical formula:

$$y_i = \alpha + \beta x_i + \epsilon_i \qquad (1)$$

in which $\epsilon_i \sim N(0, \sigma)$.

With the varying intercept multi-level model, the mathematical formula has the following form:

$$y_i=\alpha_{j[i]}+\beta x_i+\varepsilon_i \qquad (2)$$

Applying Eq. (2) to the model in Fig. 1, we have:

$$y_{burden[i]} = \alpha_{Res}[x_{Res[i]}] + \beta_{stay}x_{stay[i]} + \beta_{Jcond}x_{Jcond\,[i]} + \beta_{Envl}x_{Envl\,[i]} + \epsilon_{burden[i]} \qquad (3)$$

In the model, β_{stay}, β_{Jcond} and β_{Envl} are three regression coefficients indicating the slopes of regression lines or the rate of change in y_{burden} as x_{stay}, x_{Jcond}, and x_{Envl} change, respectively, while α_{Res} stands for a varying intercept.

Once the user has a clear picture of the causal and correlational structure of the variables, the next step is to code the Bayesian network model. The following box presents how the nodes and arcs are added to the model.

```
# Add nodes to model
model <- bayesvl()
model <- bvl_addNode(model, “Burden”, “norm”)
model <- bvl_addNode(model, “Stay”, “binom”)
model <- bvl_addNode(model, “Jcond”, “norm”)
model <- bvl_addNode(model, “EnvL”, “norm”)
model <- bvl_addNode(model, “Res”, “cat”)
# Add arcs to model
model <- bvl_addArc(model, “Stay”, “Burden”, “slope”)
model <- bvl_addArc(model, “Jcond”, “Burden”, “slope”)
model <- bvl_addArc(model, “EnvL”, “Burden”, “slope”)
model <- bvl_addArc(model, “Res”, “Burden”, “varint”)
```

A relational diagram is constructed with two commands: bvl_addNode and bvl_addArc. When using bvl_addNode, the statistical distribution of any variable is chosen with “norm” for normal distribution, “binom” for binominal distribution, or “cat” categorical distribution. Meanwhile, the code bvl_addArc is used for setting the mathematical relationship between two nodes, for example, with a fixed-effect model (“slope”), varying slope model (“varint”), varying intercept model (“varslope”), or a mixed-effect model (“varpars”). For multilevel modelling, random-intercept (“varslope”) and mixed-effect models (“varpars”) are used.

Automatic generation of stan codes

The bayesvl package allows the users to easily generate the stan codes for the model using the following commands:

```
#  Generate the stan code for model
model_string <- bvl_model2Stan(model)
cat(model_string)
```

As a result, we have the following stan codes automatically generated from building the graphical structure of the Bayesian network.

```
functions{
    int numLevels(int[] m) {
       int sorted[num_elements(m)];
       int count = 1;
       sorted = sort_asc(m);
       for (i in 2:num_elements(sorted)) {
         if (sorted[i] != sorted[i-1])
            count = count + 1;
       }
       return(count);
    }
}
data{
    // Define variables in data
    int<lower=1> Nobs;   // Number of observations (an
integer)
    real Burden[Nobs];   // outcome variable
    int<lower=0,upper=1> Stay[Nobs];
    real Jcond[Nobs];
    real EnvL[Nobs];
    int NRes;
    int<lower=1,upper=NRes> Res[Nobs];
}
transformed data{
    // Define transformed data
}
parameters{
    // Define parameters to estimate
    real<lower=0> sigma_Burden;
    real b_Stay_Burden;
    real b_Jcond_Burden;
    real b_EnvL_Burden;
    real a0_Res;
    real<lower=0> sigma_Res;
    vector[NRes] u_Res;
}
transformed parameters{
    // Transform parameters
    real mu_Burden[Nobs];
    vector[NRes] a_Res;
    // Varying intercepts definition
    for(k in 1:NRes) {
       a_Res[k] = a0_Res + u_Res[k];
    }

    for (i in 1:Nobs) {
```

MCMC simulation and convergence diagnostics

Next, using the following commands, we can fit the model and execute the MCMC estimation for the model.

```
# Fit the model
model <- bvl_modelFit(model, dat, warmup = 2000,
iter = 5000, chains = 4, cores = 4)
summary(model)
```

The summary of the model is given in Table 1.

Table 1. Simulation result.

	Mean	SD	n_eff	Rhat
b_Stay_Burden	0.22	0.05	8933	1
b_Jcond_Burden	-0.50	0.04	8173	1
b_EnvL_Burden	-0.06	0.03	8613	1
a_Res[Yes]	2.76	0.12	7895	1
a_Res[No]	3.58	0.11	7846	1
a0_Res	2.79	3.13	1563	1.01
sigma_Res	3.66	3.82	2151	1

Two values were used to diagnose the convergence of the model: n_eff (effective sample size) and Rhat value. In standard practice, n_eff is good when it reaches above 1000 samples. Rhat is roughly 1, which means all chains have the same distribution. Meanwhile, the model should be checked again when Rhat is greater than 1.1. In Table 1, n_eff is greater than 1400 and Rhat equals 1, which suggests that the model is successfully converged.

A fundamental aspect of the MCMC approach is to visually diagnose the convergence key indicators such as Markov chains, the Shrink factor, and the autocorrelation phenomenon. The following examples help us understand this visual diagnostics process.

Figure 2 presents the trace plots for all

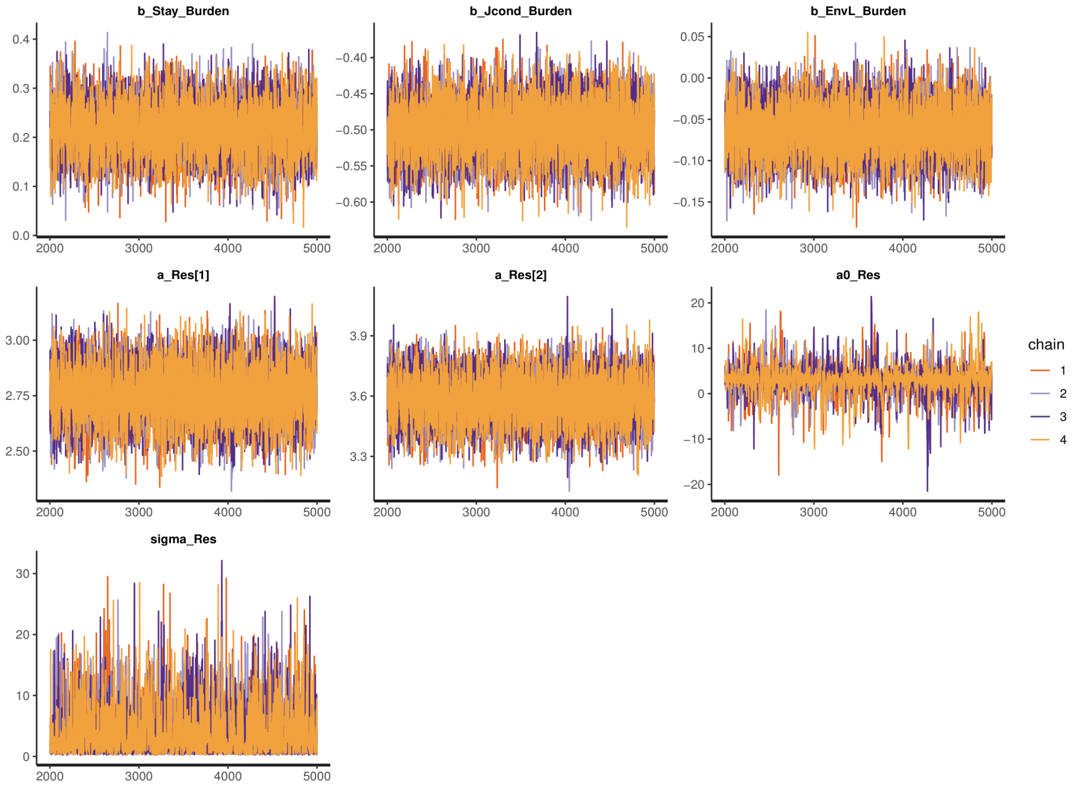

Fig. 2. The trace plot for each variable of the Bayesian regression model.

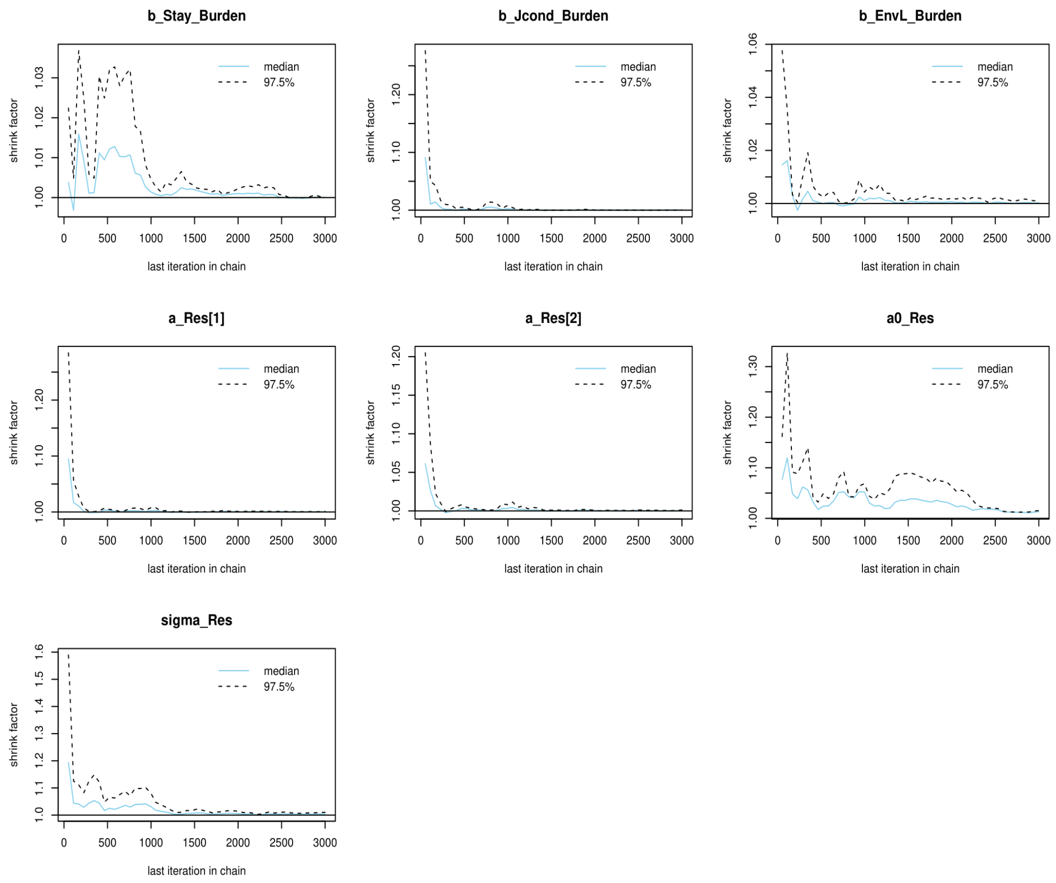

Fig. 3. The Gelman plot for each variable of the Bayesian regression model.

model variables, which show no divergent chains. The figure can be generated with the command bvl_plotTrace.

Figure 3 shows the convergence of all variables in the model according to the Gelman shrink factor (or potential scale reduction factor). To generate the visualization, the command bvl_plotGelmans is used. Similarly, Figs. 4 and 5 also illustrate different values for technical validation.

Figure 4 presents the autocorrelation factor (ACF) for each variable in the model (using the command bvl_plotAcfs). Technically, the MCMC algorithm produces samples that correlate with each other but are not independent. Therefore, the ACF plot allows researchers to check whether autocorrelation eventually stops.

Figure 5 provides technical validation for the posterior distribution of each variable in the model. The illustration can be generated using bvl_plotParams.

The Pareto-smoothed importance sampling (PSIS) diagnostic plot in Fig. 6 illustrates the model's goodness-of-fit given the underlying data. The PSIS diagnostic plot is used together with the leave-one-out cross-validation (LOO) technique to validate the model's performance. In the bayesvl package, the LOO-PSIS diagnostic can be used with the following command:

```
loo<-bvl_stanLoo(model)
plot(loo)
```

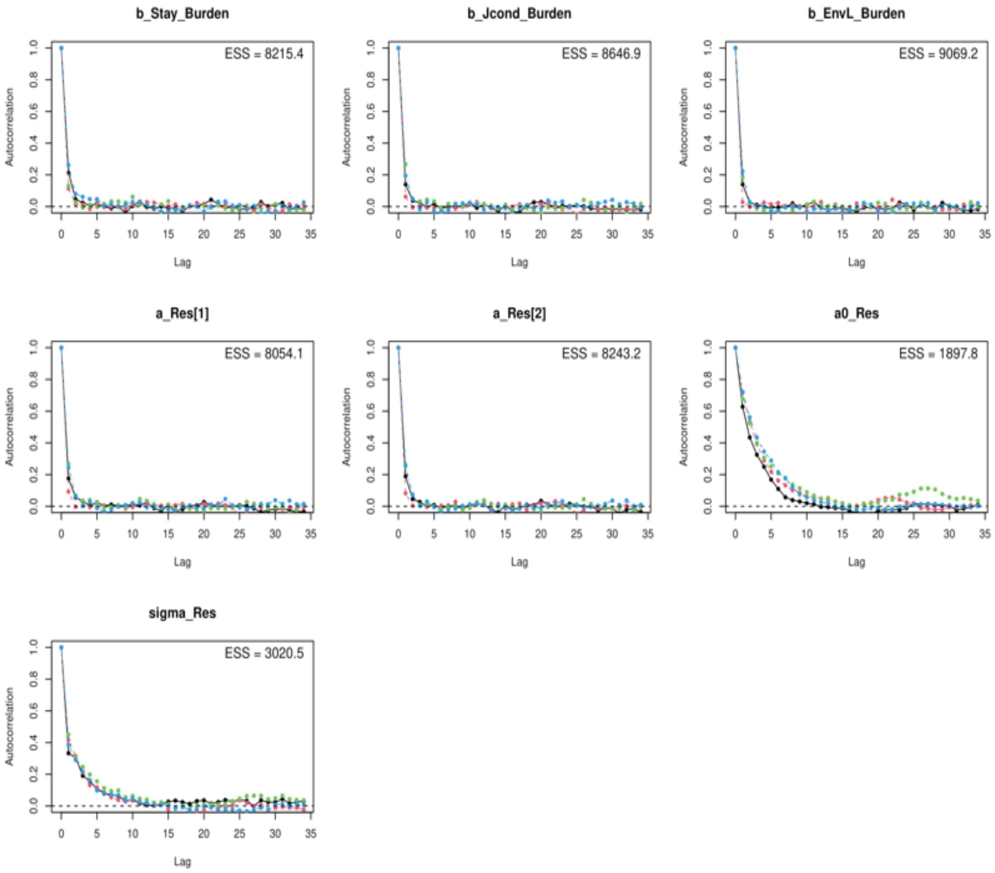

Fig. 4. The autocorrelation factor plot for each variable of the Bayesian regression model.

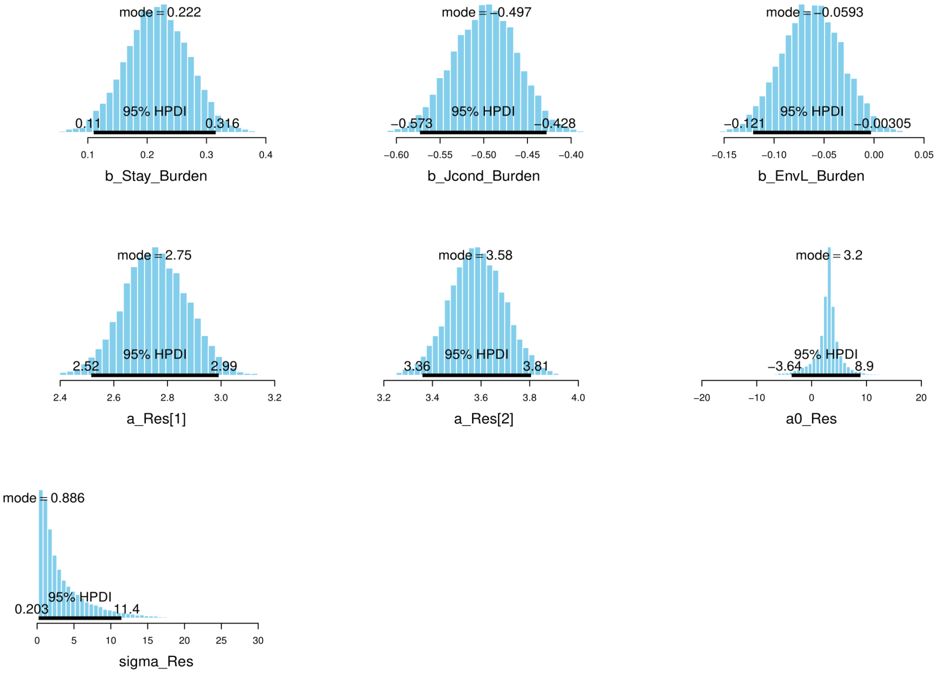

Fig. 5. The parameter plot for each variable of the Bayesian regression model.

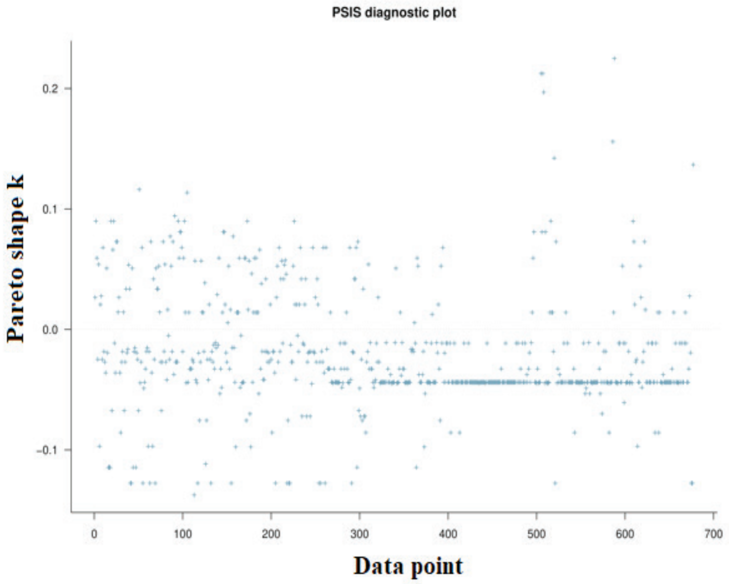

Fig. 6. The LOO plot for each variable of the Bayesian regression model.

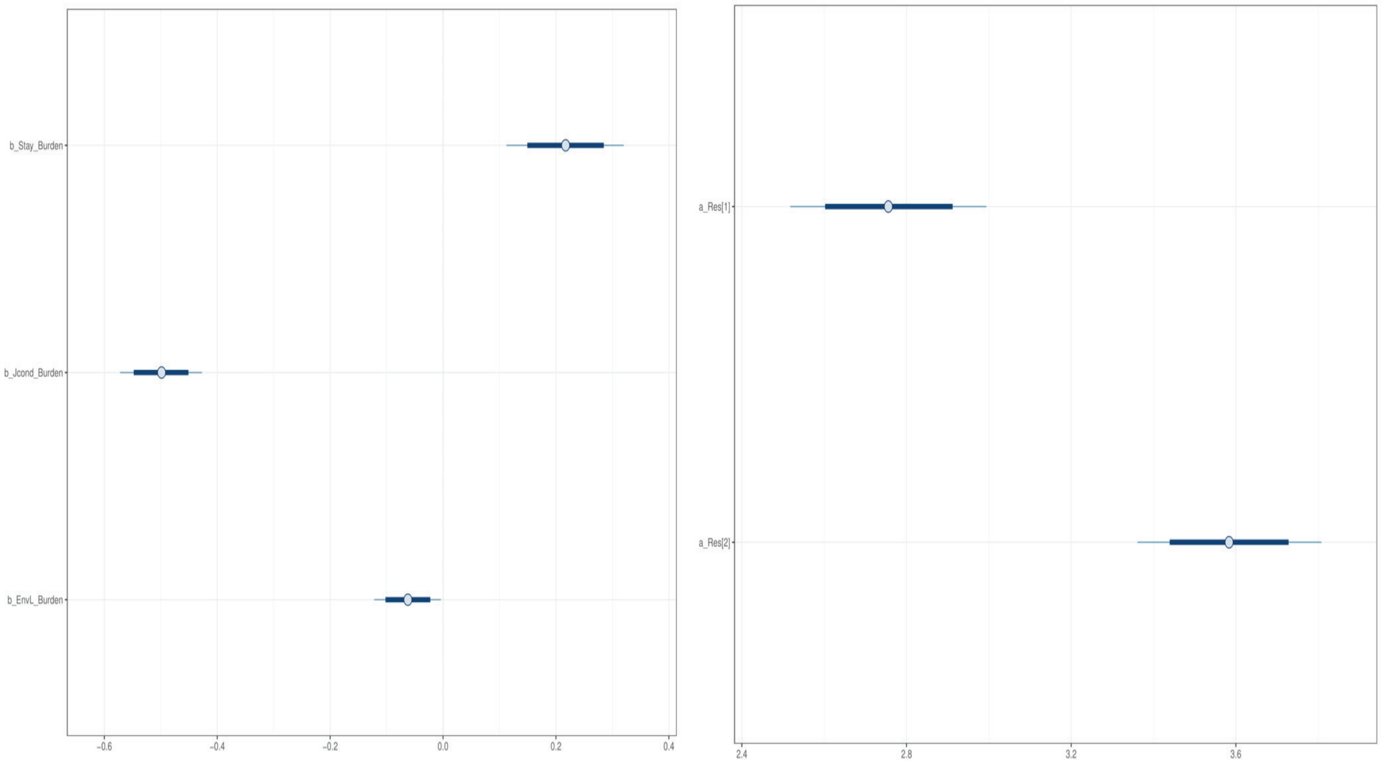

Fig. 7. The interval plot of the coefficients' posterior distribution.

Regression results

The range of posterior distribution (Fig. 7) can be visualized using the command bvl_plotIntervals.

Or we can view the coefficients in a different style using the command bvl_plotDensity (Fig. 8).

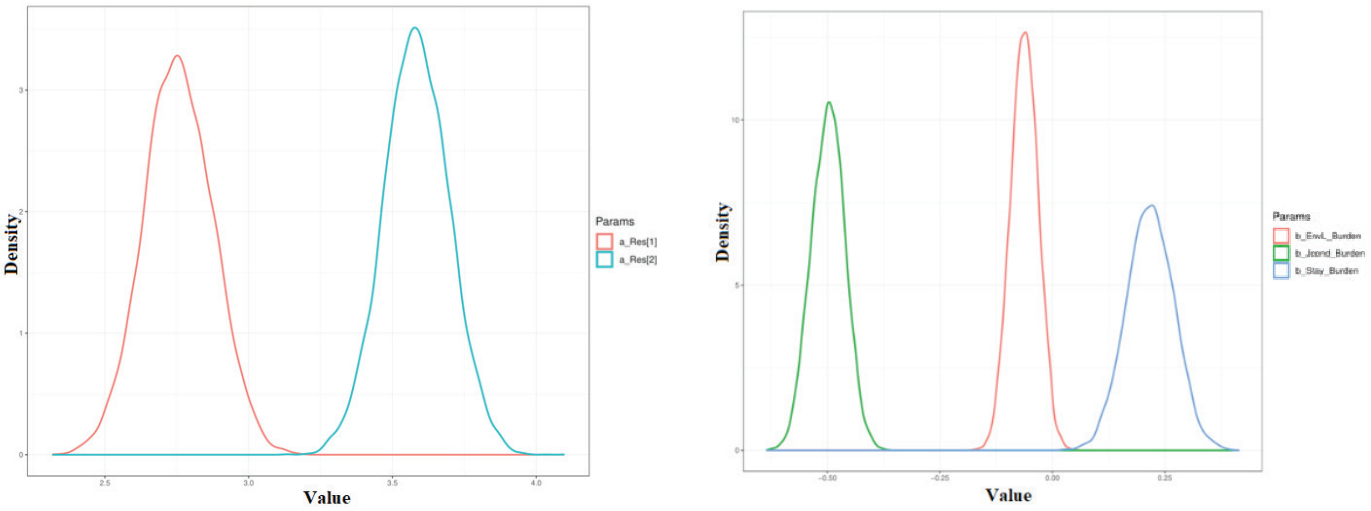

Fig. 8. Plotting the coefficients' posterior distribution.

Discussion

As demonstrated in the example above, the bayesvl R package can be useful for social sciences and humanities researchers to gain familiarity with Bayesian statistics. First, the bayesvl R package allows users to visualize their model in the form of a Bayesian network, and makes it is relatively easy to code the graphical structures of any Bayesian network with only two commands: bvl_addNode and bvl_addArc. Bayesian network modelling is suitable for handling problems due to the veracity of data [25] as well as ensuring the examination of the models. As a visual presentation, researchers can visually inspect the formulated correlational structures [33, 34]. With the link to ggplot2, bayesvl allows quick and easy visualization of the data as well as of the posterior distribution. Good data visualization can help researchers quickly identify errors in the data [35] and point them toward possible causal/correlational structures in the data. Moreover, bayesvl can help researchers with the Stan code, especially new learners who are not familiar with the coding language of Stan.

B. Aczel, et al. (2020) [3] observed that selecting from a wide variety of inference tools in Bayesian analysis can be confusing and D. Spiegelhalter (2019) [2] also noted the daunting computational nature of modern Bayesian analysis, its lack of established and widely accepted criteria for significance, and its lack of user-friendly software. However, there have been many attempts to analyse the practice and philosophy of Bayesian data analysis to point toward a more established framework of Bayesian inference. For example, J. Gabry, et al. (2019) [36] argued that Bayesian data analysis, in practice, is an iterative process of model building, inference, model checking and evaluation, and finally model expansion. In each stage, visualization is indispensable. However, A. Gelman and C.R. Shalizi (2013) [37] argued that the most successful forms of Bayesian statistics do not support that particular philosophy but align more with a form of 'hypothetico-deductivism'. Similarly, S.E. Lazic, et al. (2020) [38] argued how a Bayesian approach can deal with the problem of pseudo-replication. In building this software package, we hope to contribute to the growing spread of Bayesian statistics applications in the social sciences and humanities. We believe that an appreciation for testing and trying new methodologies will make social sciences and humanities studies more scientific and reproducible [39, 40]. Compared with the conventional frequentist regression, Bayesian regression modelling is better at dataset analysis of small sample sizes [41]. Moreover, the Bayesian approach offers a wide range of tools to construct, compare, and choose models based on their predictability performance [8]. In this new era of AI and Big Data, reproducibility and transparency are two values one must uphold, which will greatly reduce the cost of science [42].

COMPETING INTERESTS

The authors declare that there is no conflict of interest regarding the publication of this article.

REFERENCES

[1] J.K. Kruschke, T.M. Liddell (2018), "Bayesian data analysis for newcomers", *Psychonomic Bulletin & Review*, **25(1)**, pp.155-177.

[2] D. Spiegelhalter (2020), *The Art of Statistics: Learning from Data*, Penguin Random House, 448pp.

[3] B. Aczel, et al. (2020), "Discussion points for Bayesian inference", *Nature Human Behaviour*, **4**, pp.561-563.

[4] Y. Yao, et al. (2018), "Using stacking to average Bayesian predictive distributions (with discussion)", *Bayesian Analysis*, **13(3)**, pp.917-1007.

[5] J.A. Scales, R. Snieder, "To Bayes or not to Bayes?", *Geopolitics*, **62(4)**, pp.1045-1046.

[6] J. Gill (2002), *Bayesian Methods: A Social and Behavioral Sciences Approach*, Chapman and Hall/CRC, 459pp.

[7] J.K. Kruschke (2015), *Doing Bayesian Data Analysis: A Tutorial with R, JAGS, and Stan*, 2nd ed., Elsevier, 759pp.

[8] R. McElreath (2016), *Statistical Rethinking: A Bayesian Course with Examples in R and Stan*, CRC Press, 612pp.

[9] S. Jackman (2009), *Bayesian Analysis for the Social Sciences*, NY: John Wiley & Sons, 573pp.

[10] S.M. Lynch, B. Bartlett (2019), "Bayesian statistics in sociology: past, present, and future", *Annual Review of Sociology*, **45**, pp.47-68.

[11] D. Spiegelhalter, et al. (2006), *OPENBUGS User Manual Version 2.20*, MRC Biostatistics Unit, Cambridge.

[12] mcmc-jags. sourceforge.net/.

[13] J.D. Hadfield (2010), "MCMC methods for multi-response generalized linear mixed models: the MCMCglmm R package", *Journal of Statistical Software*, **33(2)**, pp.1-22.

[14] B. Carpenter, et al. (2017), "Stan: a probabilistic programming language", *Journal of Statistical Software*, **76(1)**, pp.1-32.

[15] P.C. Bürkner (2017), "Brms: an R package for bayesian multilevel models using stan", *Journal of Statistical Software*, **80(1)**, pp.1-28.

[16] R. McElreath (2018), *Statistical Rethinking: a Bayesian Course with Examples in R and Stan*, 1st ed., Chapman and Hall/CRC, 483pp.

[17] A. Gelman, et al. (2019), "R-squared for Bayesian regression models", *The American Statistician*, **73(3)**, pp.307-309.

[18] A. Vehtari, A. Gelman, J. Gabry (2017), "Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC", *Statistics and Computing*, **27(5)**, pp.1413-1432.

[19] S. Watanabe (2013), "A widely applicable Bayesian information criterion", *Journal of Machine Learning Research*, **14**, pp.867-897.

[20] mc-stan.org/loo/articles/loo2-weights.html.

[21] M.T. Ho, Q.H. Vuong (2019), "The values and challenges of 'openness' in addressing the reproducibility crisis and regaining public trust in social sciences and humanities", *European Science Editing*, **45(1)**, pp.14-17.

[22] Q.H. Vuong, M.T. Ho, V.P. La (2019), "'Stargazing' and p-hacking behaviours in social sciences: some insights from a developing country", *European Science Editing*, **45(2)**, pp.54-55.

[23] cran.r-project.org/web/packages/bayesvl/index.html.

[24] Q.H. Vuong, V.P. La (2018), *BayesVL Package for Bayesian Statistical Analyses in R.* Github, version 0.8.5.

[25] H. Njah, S. Jamoussi, W. Mahdi (2019), "Deep Bayesian network architecture for big data mining", *Concurrency and Computation: Practice and Experience*, **31(2)**, DOI: 10.1002/cpe.4418.

[26] Q.H. Vuong, N.K. Napier (2017), "Academic research: the difficulty of being simple and beautiful", *European Science Editing*, **43(2)**, pp.32-33.

[27] M. Scutari, J.B. Denis (2015), *Bayesian Networks: With Examples in R.* 2015, Boca Raton: CRC Press, 225p.

[28] M. Scutari (2010), "Learning Bayesian networks with the bnlearn R package", *Journal of Statistical Software*, **35(3)**, pp.1-22.

[29] L. Martin, D.L. Schwartz (2014), "A pragmatic perspective on visual representation and creative thinking", *Visual Studies*, **29(1)**, pp.80-93.

[30] J.H. Mathewson (1999), "Visual-spatial thinking: an aspect of science overlooked by educators", *Science Education*, **83(1)**, pp.33-54.

[31] M.T. Ho, et al. (2019), "Health care, medical insurance, and economic destitution: a dataset of 1042 stories", *Data*, **4(2)**, DOI: 10.3390/data4020057.

[32] M.T. Ho (2019), *Health Care, Medical Insurance, and Economic Destitution: A Dataset of 1042 Stories*, Open Science Framework, osf.io/2k8nd/.

[33] J. Wang, et al. (2014), *A Scalable Data Science Workflow Approach for Big Data Bayesian Network Learning*, 2014 IEEE/ACM International Symposium on Big Data Computing, DOI:10.1109/BDC.2014.10.

[34] C. Champion, C. Elkan (2017), "Visualizing the consequences of evidence in bayesian networks", *Computer Science*, 9pp.

[35] Q.H. Vuong, et al. (2018), "An open database of productivity in Vietnam's social sciences and humanities for public use", *Scientific Data*, **5**, DOI: 10.1038/sdata.2018.188.

[36] J. Gabry, et al. (2019), "Visualization in Bayesian workflow", *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, **182(2)**, pp.389-402.

[37] A. Gelman, C.R. Shalizi (2013), "Philosophy and the practice of Bayesian statistics", *British Journal of Mathematical and Statistical Psychology*, **66(1)**, pp.8-38.

[38] S.E. Lazic, et al. (2020), "A Bayesian predictive approach for dealing with pseudoreplication", *Scientific Reports*, **10**, DOI: 10.1038/s41598-020-59384-7.

[39] V. Amrhein, S. Greenland, B. McShane (2019), "Scientists rise up against statistical significance", *Nature*, **567**, pp.305-307.

[40] G. D'Oca, I. Hrynaszkiewicz (2015), "Palgrave communications' commitment to promoting transparency and reproducibility in research", *Palgrave Communications*, **1(1)**, DOI:10.1057/palcomms.2015.13.

[41] J. Piironen, A. Vehtari (2017), "Comparison of Bayesian predictive methods for model selection", *Statistics and Computing*, **27(3)**, pp.711-735.

[42] Q.H. Vuong (2018), "The (ir)rational consideration of the cost of science in transition economies", *Nature Human Behaviour*, **2**, DOI: 10.1038/s41562-017-0281-4.