Leave-one-out cross-validation for non-factorizable normal models

Paul-Christian Bürkner[1], Jonah Gabry[2], & Aki Vehtari[3]

[1] Department of Psychology, University of Münster, Germany
[2] Institute for Social and Economic Research in Policy, Columbia University, USA
[3] Department of Computer Science, Aalto University, Finland

Abstract

Cross-validation can be used to measure a model's predictive accuracy for instance for the purpose of model comparison or selection. As exact cross-validation is often practically infeasible for Bayesian models because it requires too much time, approximate cross-validation methods have been developed; most notably methods for leave-one-out cross-validation (LOO-CV). However, standard LOO-CV requires the likelihood to be factorizable, that is the observations have to be conditionally independent given the model parameters. Unfortunately, some important statistical models most notably in the context of temporal and spatial statistics are non-factorizable, but LOO-CV may still be an important measure for these models. For this reason, we derive how to compute and validate exact and approximate LOO-CV for non-factorizable models that follow a multivariate normal likelihood.

Keywords: cross-validation, Pareto-smoothed importance-sampling, non-factorizable models, SAR models

1 Introduction

After fitting a statistical model, we often want to measure its predictive accuracy, for instance for the purpose of model comparison or selection (Ando & Tsay, 2010; Geisser & Eddy, 1979; Vehtari & Lampinen, 2002; Vehtari & Ojanen, 2012). In the absence of actual new data to predict, one general approach to evaluating a model's predictive accuracy is cross-validation (Vehtari & Lampinen, 2002). When doing cross-validation, the data is split into two subsets. Based on the first subset we fit the statistical model and then evaluate its

Correspondence concerning this article should be addressed to Aki Vehtari, Department of Computer Science, Aalto University, Finland. E-mail: Aki.Vehtari@aalto.fi

predictive accuracy for the second subset. We may do this once or many times each time leaving out another subset.

One widely applied type of cross-validation is *leave-one-out cross-validation* (LOO-CV), where each time a single observation is left out and then predicted based on the model fit to the remaining data (Vehtari et al., 2017b). Predictive accuracy is evaluated by first computing the expected log predictive density of the left-out observation and then taking the sum of these values over all observations to obtain the expected log predictive density as a single measure of predictive accuracy. Unfortunately, exact LOO-CV is costly as it requires to fit the model as many times are there are observations in the data. Depending on the size of the data, complexity of the model, and estimation method, this can be practically infeasible as it simply requires too much time (Vehtari et al., 2017b). For this reason, approximate versions of LOO-CV have been developed, most notably approximations via Pareto-smoothed importance-sampling (PSIS-LOO-CV; Vehtari et al., 2017b, 2017a), which is applicable to Bayesian models.

A standard assumption of any such LOO-CV approach is that the joint likelihood of the model over all observations has to be factorizable, that is the observations have to be pairwise conditionally independent given the model parameters. The purpose of the present paper is to generalize PSIS-LOO-CV to non-factorized or non-factorizable models where observations are dependent even after conditioning on the model parameters.

1.1 Approximate LOO-CV using integrated importance-sampling

We start by introducing the mathematical basis of approximative LOO-CV. We index observations by i and denote the corresponding response value by y_i. Further, we use y to indicate the response vector of all observations and y_{-i} to indicate the response vector without the ith value. Model parameters are referred to as θ. Throughout, a Bayesian model specification and estimation via Markov chain Monte Carlo (MCMC) methods is assumed. To obtain the leave-one-out predictive density $p(y_i \,|\, y_{-i})$ we need to integrate over θ:

$$p(y_i \,|\, y_{-i}) = \int p(y_i \,|\, y_{-i}, \theta) \, p(\theta \,|\, y_{-i}) \, d\theta. \tag{1}$$

Here, $p(\theta \,|\, y_{-i})$ is the leave-one-out posterior distribution for θ, that is, the posterior distribution for θ obtained by fitting the model while holding out the ith observation.

To avoid the cost of sampling from N leave-one-out posteriors, it is possible to take the posterior draws $\theta^{(s)}$ $(s = 1, \ldots, S)$, from the *full* posterior $p(\theta \,|\, y)$, and then approximate the above integral using integrated importance sampling (see Section 3.6.1 in Vehtari, Mononen, Tolvanen, Sivula, & Winther, 2016):

$$p(y_i \,|\, y_{-i}) \approx \frac{\sum_{s=1}^S p(y_i \,|\, y_{-i}, \, \theta^{(s)}) \, w_i^{(s)}}{\sum_{s=1}^S w_i^{(s)}}. \tag{2}$$

In the above equation, $w_i^{(s)}$ are importance weights to be computed as follows. First we compute the raw importance ratios

$$r_i^{(s)} \propto \frac{1}{p(y_i \,|\, y_{-i},\, \theta^{(s)})}, \tag{3}$$

and then stabilize them using Pareto-smoothed importance-sampling to obtain the weights $w_i^{(s)}$ (Vehtari et al., 2017b, 2017a). The resulting approximation is referred to as PSIS-LOO-CV (Vehtari et al., 2017b).

2 Leave-one-out cross validation for non-factorizable models

When computing approximate LOO-CV after fitting a Bayesian model, the first step is to calculate the *pointwise* log-likelihood for every response value y_i, $i = 1, \ldots, N$. This is straightforward for *factorizable* models in which response values are conditionally independent given the model parameters θ and the likelihood can be written in the familiar form

$$p(y \,|\, \theta) = \prod_{i=1}^{N} p(y_i \,|\, \theta). \tag{4}$$

The function p will be either a probability density function (PDF) or a probability mass function (PMF) depending on whether we have a continuous or discrete outcome. When $p(y)$ can be factorized in this way, the conditional pointwise log-likelihood can be obtained easily by computing $\log p(y_i \,|\, \theta)$ for each i. We then save each of these individual contributions to the log-likelihood rather than simply summing them to obtain the total log-likelihood.

The situation is more complicated for *non-factorizable* models in which response values are not conditionally independent. When there is residual dependency even after accounting for the model parameters θ, the conditional pointwise log-likelihood has the general form $\log p(y_i \,|\, y_{-i}, \theta)$, where, again, y_{-i} denotes all response values except observation i.

2.1 LOO-CV for multivariate normal models

Although computing the pointwise log-likelihood for non-factorizable models is often impossible, there is a large class of multivariate normal models for which an analytical solution is available. These equations were initially derived by Sundararajan and Keerthi (2001) with a focus on the special case of a zero-mean Gaussian process model with prior covariance K and residual standard deviation σ,

$$y \sim \mathrm{N}(0,\, K + \sigma^2 I), \tag{5}$$

where I is the identity matrix of appropriate dimension and $C = K + \sigma^2 I$ is the covariance matrix of the model. Sundararajan and Keerthi's equations are not usually used for LOO-CV of Gaussian process models, as in most the cases, Gaussian processes are combined with a factorizable likelihood so that simpler equations for univariate distributions can be applied. What makes Sundararajan and Keerthi's equations helpful for the purpose of the present paper, is our observation that their derivations make no use of the special form of C for Gaussian process models and thus immediately generalize to the case of an arbitrary invertible covariance matrix C. For such models, the LOO predictive mean and standard deviation can be computed as follows:

$$\begin{aligned} \mu_{\tilde{y},-i} &= y_i - \bar{c}_{ii}^{-1} g_i \\ \sigma_{\tilde{y},-i} &= \sqrt{\bar{c}_{ii}^{-1}}, \end{aligned} \tag{6}$$

where $g_i = \left[C^{-1} y\right]_i$ and $\bar{c}_{ii} = \left[C^{-1}\right]_{ii}$. The log predictive density of the ith observation is then computed as

$$\log p(y_i \,|\, y_{-i}, \theta) = -\frac{1}{2}\log(2\pi) - \frac{1}{2}\log\sigma^2_{-i} - \frac{1}{2}\frac{(y_i - \mu_{-i})^2}{\sigma^2_{-i}}. \tag{7}$$

Expressing this same equation in terms of g_i and $\bar{c}_{ii}$, the log predictive density becomes:

$$\log p(y_i \,|\, y_{-i}, \theta) = -\frac{1}{2}\log(2\pi) + \frac{1}{2}\log\bar{c}_{ii} - \frac{1}{2}\frac{g_i^2}{\bar{c}_{ii}} \tag{8}$$

(note that Vehtari et al. (2016) has a typo in the corresponding Equation 34). From these equations we can now derive a recipe for obtaining the conditional pointwise log-likelihood for *all* models that can be expressed conditionally in terms of a multivariate normal with invertible covariance matrix C.

2.2 Exact LOO-CV with re-fitting

In order to validate the approximate LOO-CV procedure, and also in order to allow exact computations to be made for a small number of leave-one-out folds for which the Pareto-k diagnostic (Vehtari et al., 2017a) indicates an unstable approximation, we need to consider how we might to do *exact* LOO-CV for a non-factorizable model. In the case of a Gaussian process that has the marginalization property, we could just drop the one row and column of C corresponding to the held out out observation. This does not hold in general for multivariate normal models, however, and to keep the original prior we may need to maintain the full covariance matrix C even when one of the observations is left out.

The solution is to model y_i as a missing observation and estimate it along with all of the other model parameters. For a conditional multivariate normal model, $\log p(y_i \,|\, y_{-i})$

can be computed as follows. First, we model y_i as missing and denote the corresponding parameter y_i^{mis}. Then, we define

$$y_{\mathrm{mis}(i)} = (y_1, \ldots, y_{i-1}, y_i^{\mathrm{mis}}, y_{i+1}, \ldots, y_N). \tag{9}$$

to be the same as the full set of observations y, except replacing y_i with the parameter y_i^{mis}.

Second, we compute the LOO predictive mean and standard deviations as above, but replace y with $y_{\mathrm{mis}(i)}$ in the computation of $\mu_{\tilde{y},-i}$:

$$\mu_{\tilde{y},-i} = y_{\mathrm{mis}(i)} - \bar{c}_{ii}^{-1} g_i, \tag{10}$$

where in this case we have

$$g_i = \left[C^{-1} y_{\mathrm{mis}(i)} \right]_i . \tag{11}$$

The conditional log predictive density is then computed with the above $\mu_{\tilde{y},-i}$ and the left out observation y_i:

$$\log p(y_i \,|\, y_{-i}, \theta) = -\frac{1}{2}\log(2\pi) - \frac{1}{2}\log \sigma^2_{\tilde{y},-i} - \frac{1}{2}\frac{(y_i - \mu_{\tilde{y},-i})^2}{\sigma^2_{\tilde{y},-i}}. \tag{12}$$

Finally, the leave-one-out predictive distribution can then be estimated as

$$p(y_i \,|\, y_{-i}) \approx \frac{1}{S}\sum_{s=1}^{S} p(y_i \,|\, y_{-i}, \theta_{-i}^{(s)}), \tag{13}$$

where $\theta_{-i}^{(s)}$ are draws from the posterior distribution $p(\theta \,|\, y_{\mathrm{mis}(i)})$.

3 Case Study

A common non-factorizable multivariate normal model is the simultaneously autoregressive (SAR) model, which is frequently used for spatially correlated data. The lagged SAR model is defined as

$$y = \rho W y + \eta + \epsilon \tag{14}$$

or equivalently

$$(I - \rho W) y = \eta + \epsilon, \tag{15}$$

where ρ is the spatial correlation parameter and W is a user-defined weight matrix. The matrix W has entries $w_{ii} = 0$ along the diagonal and the off-diagonal entries w_{ij} are larger when areas i and j are closer to each other. In a linear model, the predictor term η is given by $\eta = X\beta$ with design matrix X and regression coefficients β. However, since the above equation holds for arbitrary η, these results are not restricted to linear models. If we have $\epsilon \sim \mathrm{N}(0,\, \sigma^2 I)$, it follows that

$$(I - \rho W)y \sim \mathrm{N}(\eta, \sigma^2 I). \tag{16}$$

For the purpose of computing LOO-CV, it makes sense to rewrite the SAR model in slightly different form. Conditional on ρ, η, and σ, if we write

$$y - (I - \rho W)^{-1}\eta \sim \mathrm{N}(0, \sigma^2 (I - \rho W)^{-1}(I - \rho W)^{-T}), \tag{17}$$

or more compactly, with $\widetilde{W} = (I - \rho W)$,

$$y - \widetilde{W}^{-1}\eta \sim \mathrm{N}(0, \sigma^2 (\widetilde{W}^T \widetilde{W})^{-1}), \tag{18}$$

then this has the same form as the zero mean Gaussian process from above. Accordingly, we can compute the leave-one-out predictive densities with the equations from Sundararajan and Keerthi (2001), replacing y with $(y - \widetilde{W}^{-1}\eta)$ and taking the covariance matrix C to be $\sigma^2 (\widetilde{W}^T \widetilde{W})^{-1}$.

3.1 Neighborhood Crime in Columbus, Ohio

In order to demonstrate how to carry out the computations implied by these equations, we will first fit a lagged SAR model to data on crime in 49 different neighborhoods of Columbus, Ohio during the year 1980. The data was originally described in Anselin (1988) and ships with the spdep package (Bivand & Piras, 2015).

In addition to the loo package (Vehtari, Gelman, & Gabry, 2018), for this analysis we use the brms interface (Bürkner, 2017, 2018) to Stan (Carpenter et al., 2017) to generate a Stan program and fit the model, and also the bayesplot (Gabry & Mahr, 2018) and ggplot2 (Wickham, 2016) packages for plotting. The three variables in the data set relevant to this example are: `CRIME`: the number of residential burglaries and vehicle thefts per thousand households in the neighborhood, `HOVAL`: housing value in units of $1000 USD, and `INC`: household income in units of $1000 USD. We will also use the object `COL.nb`, which is a list containing information about which neighborhoods border each other. From this list we will be able to construct the weight matrix to used to help account for the spatial dependency among the observations. The complete R code for this case study can be found at (http://mc-stan.org/loo/articles/loo2-non-factorizable.html).

A model predicting `CRIME` from `INC` and `HOVAL`, while accounting for the spatial dependency via an SAR structure, can be specified in brms as follows:

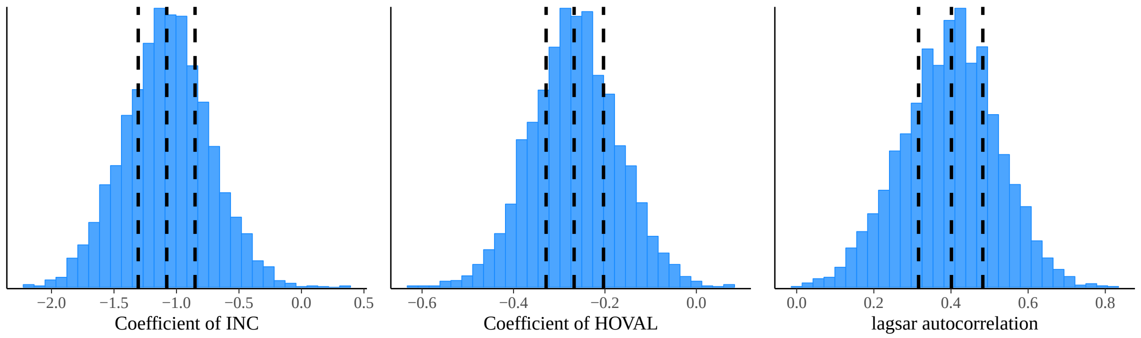

Figure 1. Posterior distribution of selected parameters of the lagged SAR model along with posterior median and 50% central interval.

```
brm(CRIME ~ INC + HOVAL, data = COL.OLD, autocor = cor_lagsar(COL.nb))
```

In Figure 1, we see that both higher income and higher housing value predict lower crime rates in the neighborhood. Moreover, there seems to be substantial spatial correlation between adjacent neighborhoods, as indicated by the posterior distribution of the `lagsar` parameter.

3.2 Approximate and exact LOO-CV

After fitting the model, the next step is to compute the pointwise log-likelihood values needed for approximate LOO-CV. To do this we use the recipe laid out in Section 2.

The quality of the PSIS-LOO approximation can be investigated graphically by plotting the Pareto-k estimate for each observation. Ideally, they should not exceed 0.5, but in practice the algorithm turns out to be robust up to values of 0.7 (Vehtari et al., 2017b, 2017a). In Figure 2, we see that the fourth observation is problematic and so may reduce the accuracy of the LOO-CV approximation.

The PSIS-LOO-CV to approximation of the expected log predictive density for new data reveals $\mathrm{elpd}_{\mathrm{approx}}$ = -187.25. This result still needs to be validated against exact LOO-CV, which is somewhat more involved, as we need to re-fit the model N times each time leaving out a single observations. For the lagged SAR model, we cannot just ignore the held-out observation entirely as this will change the prior of the other observations. In other words, the lagged SAR model does not have the marginalization property that holds, for instance, for Gaussian process models. Instead, we have to model the held-out observation as a missing value, which is to be estimated along with the other model parameters (see the case study section in http://mc-stan.org/loo/articles/loo2-non-factorizable.html for details on the R code).

A first step in the validation of the pointwise predictive density is to compare the distribution of the implied response values for the left-out observation using the pointwise mean and standard deviation from (6) to the distribution of the y_i^{mis} posterior-predictive values estimated as part of the model. If the pointwise predictive density is correct, the two

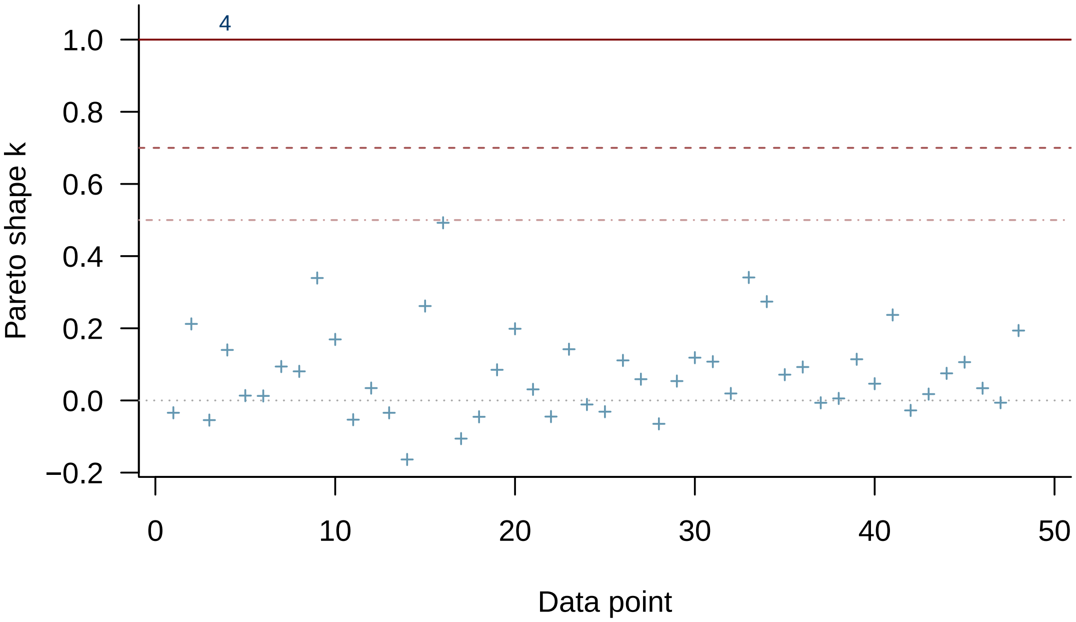

Figure 2. PSIS diagnostic plot showing the Pareto-k-estimate of each observation.

distributions should match very closely (up to sampling error). In Figure 3, we overlay these two distributions for the first four observations and see that they match very closely (as is the case for all 49 observations of in this example).

In the final step, we compute the pointwise predictive density based on the exact LOO-CV and compare it to the approximate PSIS-LOO-CV result computed earlier. The results of the approximate ($\text{elpd}_{\text{approx}}$ = -187.25) and exact LOO-CV ($\text{elpd}_{\text{exact}}$ = -188.64) are similar but not as close as we would expect if there were no problematic observations. We can investigate this issue more closely by plotting the approximate against the exact pointwise ELPD values.

In Figure 4, the fourth data point – the observation flagged as problematic by the PSIS-LOO approximation – is colored in red and is the clear outlier. Otherwise, the correspondence between the exact and approximate values is strong. In fact, summing over the pointwise ELPD values and leaving out the fourth observation yields practically equivalent results for approximate and exact LOO-CV ($\text{elpd}_{\text{approx},-4}$ = -172.94 vs. $\text{elpd}_{\text{exact},-4}$ = -173.03). From this we can conclude that the difference we found when including *all* observations does not indicate an error in the implementation of the approximate LOO-CV but rather a violation of its assumptions.

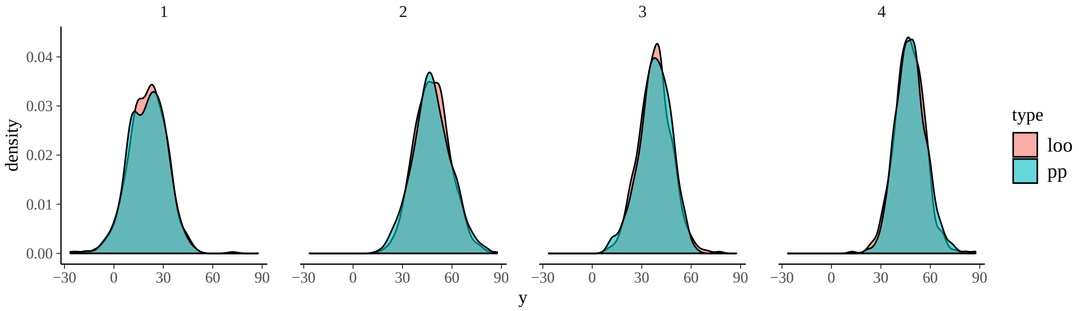

Figure 3. Implied response values of the first four observations computed (a) after model fitting (type = ’loo’) and (b) as part of the model in the form of posterior-predictive draws for the missing observation (type = ’pp’).

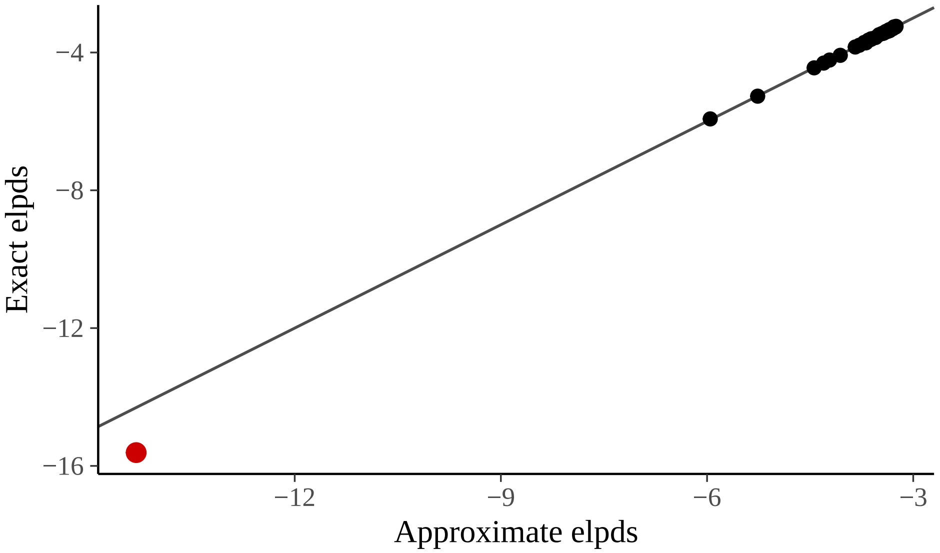

Figure 4. Comparison of approximate and exact pointwise elpd values for the SAR model. Problematic observations are marked as red dots.

4 Conclusion

In summary, we have shown how to set up and validate approximate and exact LOO-CV for non-factorizable multivariate normal models. We demonstrated the usefulness of our approach with a case study involving the non-factorizable spatial SAR model. Although we motivated the present paper by means of non-factorizable models (i.e. models that cannot be factorized at all), we note that our approach also works for any Bayesian model that can be expressed in terms of a multivariate normal likelihood. That is, we can also apply it to models that are factorizable but for which the factorized representation is difficult to compute or not available to the researcher for some other reasons.

5 Acknowledgements

We thank Daniel Simpson for useful discussion and the Academy of Finland (grants 298742, 313122) for partial support of this work.

References

Ando, T., & Tsay, R. (2010). Predictive likelihood for Bayesian model selection and averaging. *International Journal of Forecasting*, *26*(4), 744–763.

Anselin, L. (1988). *Spatial econometrics: Methods and models*. Dordrecht: Kluwer Academic.

Bivand, R., & Piras, G. (2015). Comparing implementations of estimation methods for spatial econometrics. *Journal of Statistical Software*, *63*(18), 1–36. Retrieved from http://www.jstatsoft.org/v63/i18/

Bürkner, P.-C. (2017). brms: An R package for Bayesian multilevel models using Stan. *Journal of Statistical Software*, *80*(1), 1–28. doi:10.18637/jss.v080.i01

Bürkner, P.-C. (2018). Advanced bayesian multilevel modeling with the R package brms. *The R Journal*, 395–411. Retrieved from https://journal.r-project.org/archive/2018/RJ-2018-017

Carpenter, B., Gelman, A., Hoffman, M., Lee, D., Goodrich, B., Betancourt, M., … Ridell, A. (2017). Stan: A probabilistic programming language. *Journal of Statistical Software*.

Gabry, J., & Mahr, T. (2018). *bayesplot: Plotting for Bayesian models*. Retrieved from https://CRAN.R-project.org/package=bayesplot

Geisser, S., & Eddy, W. F. (1979). A predictive approach to model selection. *Journal of the American Statistical Association*, *74*(365), 153–160.

Sundararajan, S., & Keerthi, S. S. (2001). Predictive approaches for choosing hyperparameters in Gaussian processes. *Neural Computation*, *13*(5), 1103–1118.

Vehtari, A., Gelman, A., & Gabry, J. (2017a). Pareto smoothed importance sampling. *arXiv Preprint*. Retrieved from https://arxiv.org/abs/1507.02646

Vehtari, A., Gelman, A., & Gabry, J. (2017b). Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. *Statistics and Computing*, *27*(5), 1413–1432. Retrieved from http://link.springer.com/article/10.1007/s11222-016-9696-4

Vehtari, A., Gelman, A., & Gabry, J. (2018). *loo: Efficient leave-one-out cross-validation and WAIC for Bayesian models*. Retrieved from https://github.com/stan-dev/loo

Vehtari, A., & Lampinen, J. (2002). Bayesian model assessment and comparison using cross-validation predictive densities. *Neural Computation*, *14*(10), 2439–2468.

Vehtari, A., Mononen, T., Tolvanen, V., Sivula, T., & Winther, O. (2016). Bayesian leave-one-out cross-validation approximations for Gaussian latent variable models. *Journal of Machine Learning Research*, *17*(103), 1–38. Retrieved from http://jmlr.org/papers/v17/14-540.html

Vehtari, A., & Ojanen, J. (2012). A survey of bayesian predictive methods for model assessment, selection and comparison. *Statistics Surveys*, *6*, 142–228.

Wickham, H. (2016). *ggplot2: Elegant graphics for data analysis*. Springer-Verlag New York. Retrieved from http://ggplot2.org