BGGM: Bayesian Gaussian Graphical Models in R

Donald R. Williams[1] **and Joris Mulder**[2]

1 Department of Psychology, University of California, Davis **2** Department of Methodology and Statistics, Tilburg University

Submitted: 10 January 2020
Published: (pre-print)

Summary

Gaussian graphical models (GGM) allow for learning conditional (in)dependence structures that are encoded by partial correlations. Whereas there are several `R` packages for classical methods (see Kuismin & Sillanpää, 2017, Table 1), there are only two that implement a Bayesian approach (Leday & Richardson, 2018; Mohammadi & Wit, 2015). These are exclusively focused on identifying the graphical structure. The package **BGGM** not only contains novel Bayesian methods for this purpose, but it also includes Bayesian methodology for extending inference beyond identifying non-zero relations.

BGGM is built around two approaches for *Bayesian* inference–estimation and hypothesis testing. This distinction is arbitrary (see Rouder, Haaf, & Vandekerckhove, 2018), but is used to organize this work. The former focuses on the posterior distribution and includes extensions to assess predictability (Haslbeck & Waldorp, 2018), as well as methodology to compare partial correlations. The latter includes methods for Bayesian hypothesis testing, in both exploratory and confirmatory contexts, with the novel matrix-F prior distribution (Mulder & Pericchi, 2018). This allows for testing the null hypothesis of conditional independence, as well as inequality and equality constrained hypotheses. Further, there are several approaches for comparing GGMs across any number of groups. The package also includes a suite of options for model checking. Together, **BGGM** is a comprehensive toolbox for Gaussian graphical modeling in `R`.

Estimation

There are two possibilities for *estimating* GGMs. The first is an analytic solution and the second samples from the posterior distribution (described in Williams, 2018). Sampling is recommended because the samples are required for various functions in **BGGM** (e.g., posterior predictive checks and prediction).

Structure Learning

Structure learning refers to determining which partial correlations are non-zero. This is implemented with:

```
# data (p = 5 for demonstrative purposes)
Y <- BGGM::bfi[,1:5]

# fit model
fit <- estimate(Y = Y,
                analytic = FALSE,
```

```
                 iter = 1000)

# select graph
E <- select(fit_analytic, ci_width = 0.95)

# summary
summary(E)

# output

BGGM: Bayesian Gaussian Graphical Models
---
Type: Selected Graph (Analytic Solution)
Credible Interval: 95 %
Connectivity: 80 %
---
Call:
select.estimate(x = fit_analytic, ci_width = 0.95)
---
Selected:

Partial correlations

      1     2     3    4    5
1  0.00 -0.24 -0.11 0.00 0.00
2 -0.24  0.00  0.29 0.16 0.16
3 -0.11  0.29  0.00 0.18 0.36
4  0.00  0.16  0.18 0.00 0.12
5  0.00  0.16  0.36 0.12 0.00
---

Adjacency

  1 2 3 4 5
1 0 1 1 0 0
2 1 0 1 1 1
3 1 1 0 1 1
4 0 1 1 0 1
5 0 1 1 1 0
---
```

It is customary to plot the estimated structure. The implement ion for plotting `E` is described below, as the same call is also used for the hypothesis testing methodology.

The partial correlations are plotted with:

```
# summarize the posterior distributions
fit_summary <- summary(fit, cred = 0.95)

# plot summary
fig_1a <- plot(fit_summary)
```

The object `fit_summary` includes the partial correlation that have been summarized with the posterior mean, standard deviation, and a given credible interval. The object `fig_1a`

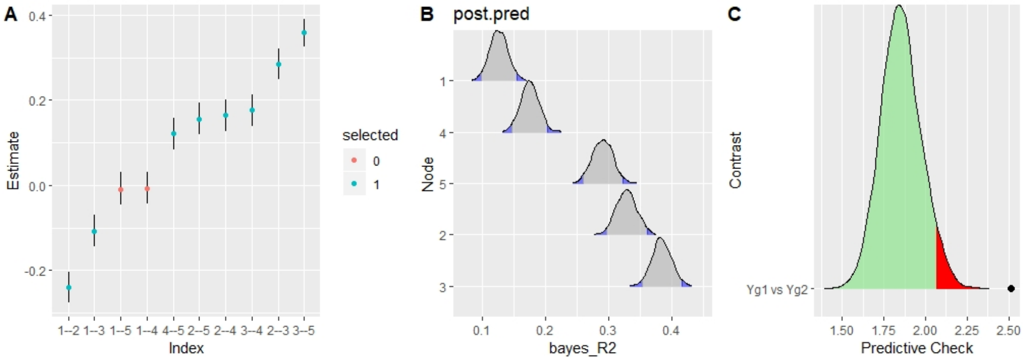

Figure 1: A) Partial correlations summarized with posterior means and 95% credible intervals. The red points denote intervals that excluded zero. B) Bayesian variance explained summarized with probability densities. C) Posterior predictive (symmetric) KL-divergence. The red area corresponds to the critical region and the black point is the observed value.

is a `ggplot` (Wickham, 2016), which allows for further customization. This is possible with all `plot()` functions in **BGGM**. An example is provided below.

Predictability

A central aspect of **BGGM** is to extend inference beyond the individual partial correlations. Assessing *predictability* provides a measure of "self-determination" (Haslbeck & Waldorp, 2018), for example, how much variance is explained by the variables included in the model. To this end, **BGGM** provides several options to assess predictability . In this example, we compute Bayesian R^2 (Gelman, Goodrich, Gabry, & Vehtari, 2019):

```
# bayes r2
r2 <- bayes_R2(fit)

fig_1b <- plot(r2,
               type = "ridgeline")

# output
BGGM: Bayesian Gaussian Graphical Models
---
Metric: Bayes R2
Type: post.pred
Credible Interval: 0.95
---
Estimates:

 Node Post.mean Post.sd Cred.lb Cred.ub
    1      0.13    0.01    0.10    0.15
    2      0.33    0.02    0.30    0.36
    3      0.38    0.02    0.35    0.41
    4      0.18    0.01    0.15    0.20
    5      0.29    0.02    0.26    0.32
```

`fig_1b` is made with the help of the **ggridges** package (Wilke, 2018).

Comparing GGMs

There is additional methodology that allows for comparing GGMs (described in Williams et al., 2019). For the estimation based methods, there are three possibilities, including `ggm_compare_estimate()`, `assess_predictability()` and `ggm_compare_ppc()`. We encourage user to explore all of those functions. The following is based on the posterior predictive distribution. In this example, we use data from a resilience survey to compare GGMs between males and females (Briganti & Linkowski, 2019).

```
# data
Ym <- subset(rsa, gender == "M",
             select = - gender)
Yf <- subset(rsa, gender == "F",
             select = - gender )

# predictive check
ppc <- ggm_compare_ppc(Ym, Yf)

fig_1c <- plot(ppc)

# summary
summary(ppc)

# output
BGGM: Bayesian Gaussian Graphical Models
---
Type: GGM Comparison (Global Predictive Check)
Posterior Samples: 5000
  Group 1: 278
  Group 2: 397
Variables (p): 33
Edges: 528
---
Call:
ggm_compare_ppc(Ym, Yf)
---
Estimates:

     contrast      KLD p_value
 Y_g1 vs Y_g2 2.512792       0
---
note:
p_value = p(T(Y_rep) > T(Y)|Y)
KLD = (symmetric) Kullback-Leibler divergence
```

Note that this method can be used to compare any number of groups.

Hypothesis Testing

The following methods were described in Williams & Mulder (2019). Note that each function has `summary()`, `print()`, and `plot()` functions. The implementation is similar to the estimation based methods, and thus not included here.

Structure Learning

The Bayes factor methods allow for gaining evidence for null effects, as well as for one-sided hypothesis testing.

```
# data
Y <- bfi[,1:5]

# fit model
fit <- explore(Y)

# select graph
E <- select(fit,
            BF_cut = 3,
            alternative = "greater")
```

The object `E` includes the selected graphs for which there was evidence for a positive effect and a null effect. This can be plotted with `plot(E)`. An example is provided below.

Confirmatory Hypothesis Testing

GGMs are typically data driven and thus inherently exploratory. Another key contribution of **BGGM** is extending hypothesis testing beyond exploratory and to confirmatory in GGMs (Williams & Mulder, 2019). The former is essentially feeding the data to the functions in **BGGM** and seeing what comes back. In other words, there are no specific, hypothesized models under consideration. On the other hand, confirmatory hypothesis testing allows for comparing theoretical models or (actual) predictions. A researcher may expect, for example, that a set of partial correlations is larger than another set. This is tested with:

```
# define hypothesis
hypothesis <- c("(A1--A2, A1--A3) >
                 (A1--A4, A1--A5)")

# test inequality contraint
test_order <-  confirm(Y = Y,
                       hypothesis  = hypothesis,
                       prior_sd = 0.5, iter = 5000,
                       cores = 2)
# output
BGGM: Bayesian Gaussian Graphical Models
Type: Confirmatory Hypothesis Testing
---
Call:
confirm(Y = Y, hypothesis = hypothesis, prior_sd = 0.5, iter = 5000,
    cores = 2)
---
Hypotheses:

 H1 (A1--A2,A1--A3)>(A1--A4,A1--A5)
 Hc                          'not H1'
---
Posterior prob:
```

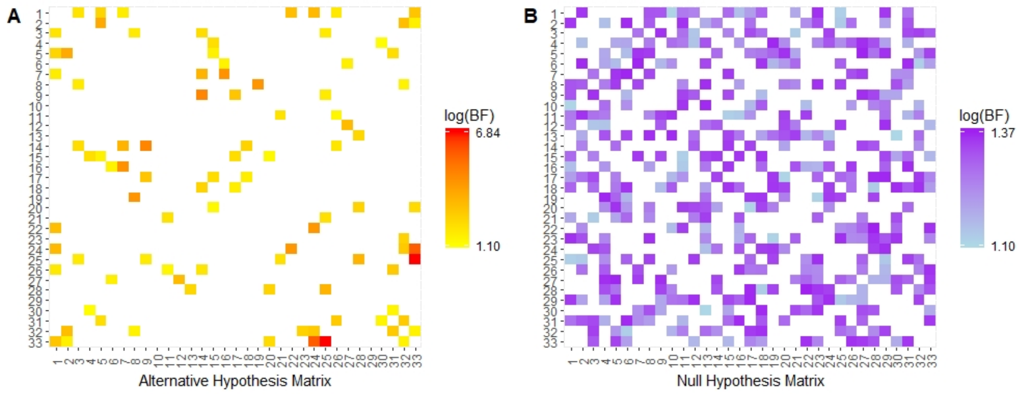

Figure 2: A) Heatmap depiticing partial correlation differences in favor of a difference. B) Heatmap depicting partial correlation differences in favor of the null hypothesis (no difference)

```
 p(H1|Y) = 0
 p(Hc|Y) = 1
---
Bayes factor matrix:
   H1  Hc
H1  1 Inf
Hc  0   1
---
note: equal hypothesis prior probabilities
```

Note that `A1--A2` denotes the partial correlation between variables A1 and A2. Any number of hypothesis can be tested. They just need to be separated by a a semi-colon, e.g., `hypothesis = c(A1--A2 > 0; A1--A2 = 0)`, which also demonstrates that it is possible to simultaneously test both inequality (`>` or `<`) and equality (`=`) restrictions.

Comparing GGMs

The Bayes factor approach can also be used to compare GGMs (see Williams et al., 2019).

```
# compare
bf_comp <- ggm_compare_bf(Ym, Yf,
                          prior_sd = 0.20)

# select
sel <- select(bf_comp,
              BF_cut = 3)

# figure 2
fig_2 <- plot(sel)
```

The results are provided in Figure 2. Note that confirmatory hypothesis testing is also possible. This uses the same arguments as those provided above.

Plotting GGMs

Plotting both the estimate and hypothesis testing graphs uses the same arguments. The following is an example of plotting a graph estimated with `explore`.

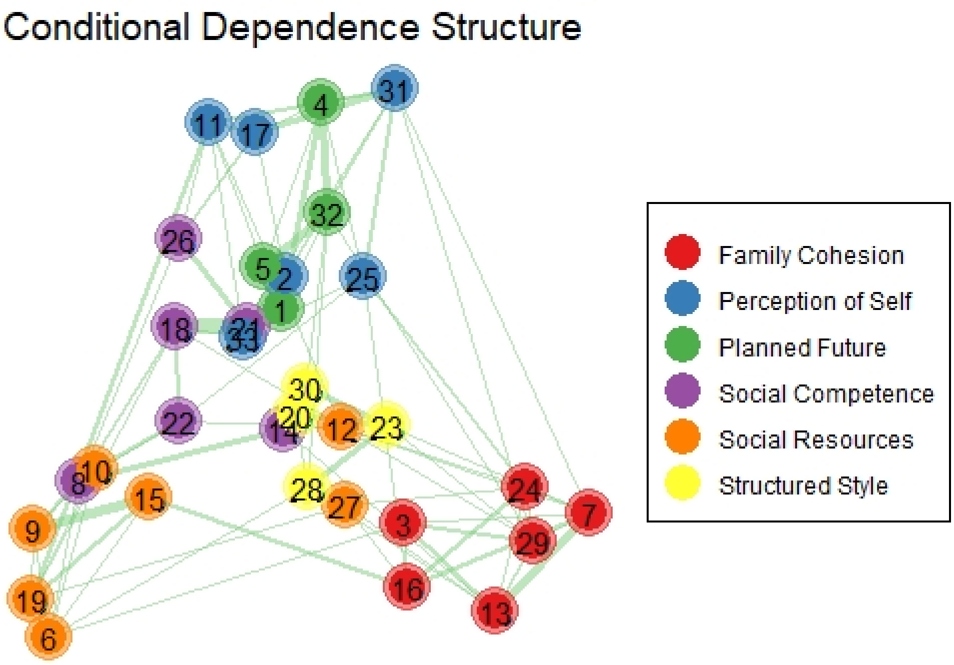

Figure 3: Estimated conditional dependence structure

```
# resilience scale
Y <- subset(rsa, select = - gender)

# fit model
fit <- explore(Y)

# select graph
E <- select(fit,
            BF_cut = 3,
            alternative = "greater")

# plot graph
fig_3 <- plot(E, layout = "mds",
              node_labels_color = "black",
              node_groups = BGGM::rsa_labels,
              txt_size = 4, node_outer_size = 8,
              node_inner_size = 6,
              edge_multiplier = 5,
              alpha = 0.3)$plt +
  # remove legend name and set palette
  scale_color_brewer(name = NULL,
                     palette = "Set1")  +
  # add title
  ggtitle("Conditional Dependence Structure") +
  # make title larger and add box around legend
  theme(plot.title = element_text(size = 14),
        legend.background = element_rect(color = "black"))
```

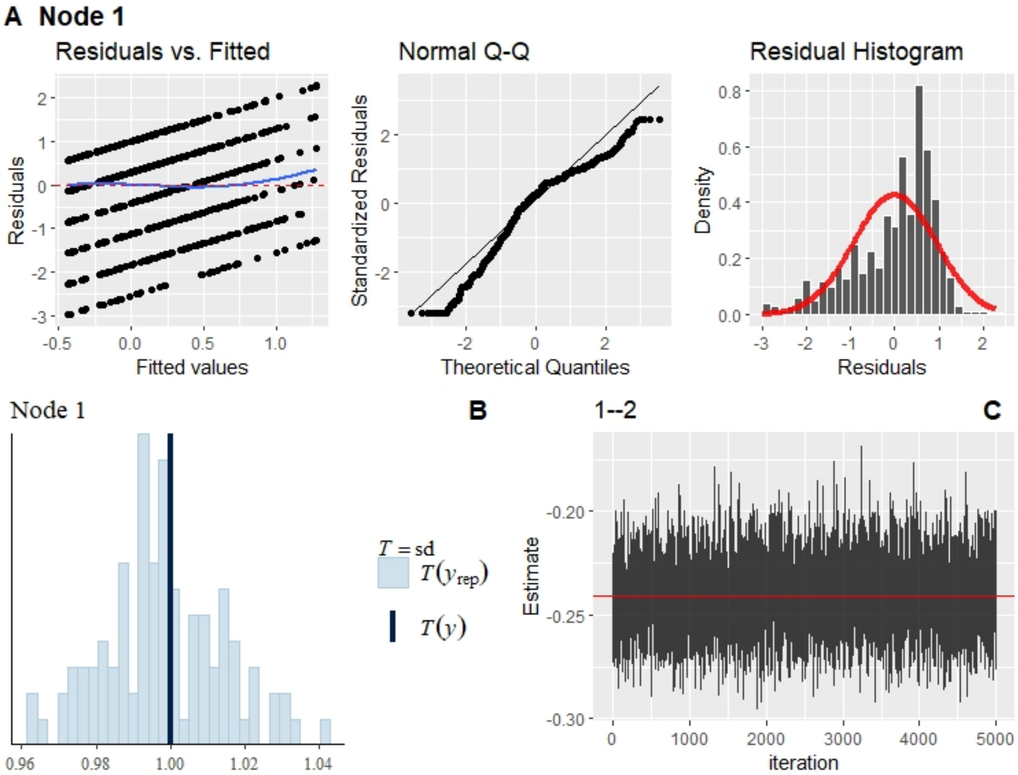

Figure 4: A) Regression diagnostics. B) Posterior predictive check of the standard deviation. C) MCMC trace plot for a partial correlation.

Model Checking

The methods in **BGGM** use custom samplers to estimate the partial correlations. There are several functions to monitor convergence. Additionally, there are a variety of methods to check the adequacy of the fitted model.

Regression Diagnostics

GGMs have a direct correspondence to multiple regression (Kwan, 2014; Stephens, 1998). Hence regression diagnostics can be used to evaluate the fitted model.

```
# data
Y <- bfi[,1:5]

# fit model
fit <- estimate(Y)

# plot
fig_4a <- diagnostics(fit, iter = 100)
```

The object `fig_4a` contains one plot for each variable in model (in this case five). One of those plots is included in Figure 4 (panel A).

Posterior Predictive Checks

In **BGGM**, posterior predictive checks are carried out with the `R` package **bayesplot** (Gabry, Simpson, Vehtari, Betancourt, & Gelman, 2019). Some internal code was borrowed from **brms** to achieve consistent between packages .

```
fig_4b <- pp_check(fit,
                   type = "stat",
                   stat = "sd")
```

The object `fig_4b` also contains one plot for each variable in model. One of those plots is included in Figure 4 (panel B). Note that this implementation is based on the internals of the `pp_check()` function from **brms** (Bürkner, 2018), but adapted for GGMs. This ensures that there is consistency between packages using **bayesplot**.

MCMC Convergence

BGGM uses Gibbs samplers to estimate GGMs. The models are not terribly difficult to estimate, so convergence should typically not be an issue. To verify convergence, however, **BGGM** provide both trace and acf plots. An example of the former is implemented with:

```
fig_4c <- convergence(fit,
                      type = "trace",
                      param = "1--2")
```

This plot is in Figure 4 (panel C). Note that it is possible to check the convergence of several parameters (e.g., `param = c(1--2, 1--3)`).

Discussion

There are several additional functions in **BGGM** not discussed in this work. For example, there are a variety of options for predictability (e.g., mean squared error), plotting capabilities for `confirm()`, among others. We encourage users to explore the package documentation. We are committed to further developing **BGGM**. In the next version, all of the methods will accommodate ordinal, binary, and mixed data. Currently, there is one **shiny** application freely available online that implements the `confirm()` method (link) (Chang, Cheng, Allaire, Xie, & McPherson, 2019). In the near future, all of the methods in **BGGM** will be implemented in a **shinydashboard** (Chang & Borges Ribeiro, 2018).

Acknowledgements

DRW was supported by a National Science Foundation Graduate Research Fellowship under Grant No. 1650042 and JM was supported by a ERC Starting Grant (758791).

References

Briganti, G., & Linkowski, P. (2019). Item and domain network structures of the Resilience Scale for Adults in 675 university students. *Epidemiology and Psychiatric Sciences*, 1–9. doi:10.1017/S2045796019000222

Bürkner, P.-C. (2018). Advanced Bayesian multilevel modeling with the R package brms. *The R Journal*, *10*(1), 395–411. doi:10.32614/RJ-2018-017

Chang, W., & Borges Ribeiro, B. (2018). *Shinydashboard: Create dashboards with 'shiny'*. Retrieved from https://CRAN.R-project.org/package=shinydashboard

Chang, W., Cheng, J., Allaire, J., Xie, Y., & McPherson, J. (2019). *Shiny: Web application framework for r*. Retrieved from https://CRAN.R-project.org/package=shiny

Gabry, J., Simpson, D., Vehtari, A., Betancourt, M., & Gelman, A. (2019). Visualization in bayesian workflow. *J. R. Stat. Soc. A*, *182*(2), 389–402. doi:10.1111/rssa.12378

Gelman, A., Goodrich, B., Gabry, J., & Vehtari, A. (2019). R-squared for Bayesian Regression Models, *73*(3), 307–309. doi:10.1080/00031305.2018.1549100

Haslbeck, J. M. B., & Waldorp, L. J. (2018). How well do network models predict observations? On the importance of predictability in network models. *Behavior Research Methods*, *50*(2), 853–861. doi:10.3758/s13428-017-0910-x

Kuismin, M., & Sillanpää, M. (2017). Estimation of covariance and precision matrix, network structure, and a view toward systems biology. *Wiley Interdisciplinary Reviews: Computational Statistics*, *9*(6), 1–13. doi:10.1002/wics.1415

Kwan, C. C. Y. (2014). A Regression-Based Interpretation of the Inverse of the Sample Covariance Matrix. *Spreadsheets in Education*, *7*(1).

Leday, G. G. R., & Richardson, S. (2018). Fast Bayesian inference in large Gaussian graphical models, 1–26. Retrieved from http://arxiv.org/abs/1803.08155

Mohammadi, A., & Wit, E. C. (2015). Bayesian structure learning in sparse Gaussian graphical models. *Bayesian Analysis*, *10*(1), 109–138. doi:10.1214/14-BA889

Mulder, J., & Pericchi, L. (2018). The Matrix-F Prior for Estimating and Testing Covariance Matrices. *Bayesian Analysis*, (4), 1–22. doi:10.1214/17-BA1092

Rouder, J. N., Haaf, J. M., & Vandekerckhove, J. (2018). Bayesian inference for psychology, part IV: parameter estimation and Bayes factors. *Psychonomic Bulletin and Review*, *25*(1), 102–113. doi:10.3758/s13423-017-1420-7

Stephens, G. (1998). On the Inverse of the Covariance Matrix in Portfolio Analysis. *The Journal of Finance*, *53*(5), 1821–1827.

Wickham, H. (2016). *Ggplot2: Elegant graphics for data analysis*. Springer-Verlag New York. Retrieved from https://ggplot2.tidyverse.org

Wilke, C. O. (2018). *Ggridges: Ridgeline plots in 'ggplot2'*. Retrieved from https://CRAN.R-project.org/package=ggridges

Williams, D. R. (2018). Bayesian Inference for Gaussian Graphical Models: Structure Learning, Explanation, and Prediction. doi:10.31234/OSF.IO/X8DPR

Williams, D. R., & Mulder, J. (2019). Bayesian Hypothesis Testing for Gaussian Graphical Models: Conditional Independence and Order Constraints. *PsyArXiv*. doi:10.31234/osf.io/ypxd8

Williams, D. R., Rast, P., Pericchi, L. R., & Mulder, J. (2019). Comparing Gaussian Graphical Models with the Posterior Predictive Distribution and Bayesian Model Selection. doi:https://doi.org/10.31234/osf.io/yt386