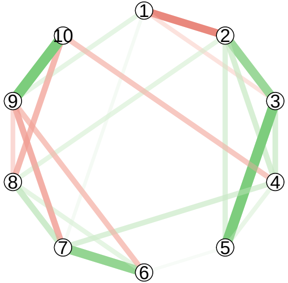
BGGM: A R Package for Bayesian Gaussian Graphical Models

Donald R. Williams

25 May 2019

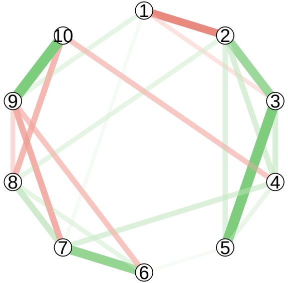
Introduction

- Gaussian graphical models capture condition dependencies between random variables



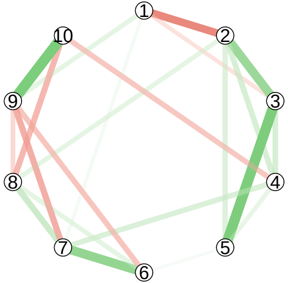
Introduction

- Gaussian graphical models capture condition dependencies between random variables
- Non-zero relations imply pairwise, conditional dependent relations, in which all other variables in the model have been controlled for



Introduction

- Gaussian graphical models capture condition dependencies between random variables
- Non-zero relations imply pairwise, conditional dependent relations, in which all other variables in the model have been controlled for
- This powerful framework for learning the structure of multivariate data has been used across the sciences



Introduction

- To date, the most common applications have been gene co-expression and functional connectivity "networks"
  - Common to have more variables p than observations n (i.e., high dimensional data)
  - Requires some form of regularization to make estimation possible (e.g., lasso)

Introduction

- To date, the most common applications have been gene co-expression and functional connectivity "networks"
  - Common to have more variables p than observations n (i.e., high dimensional data)
  - Requires some form of regularization to make estimation possible (e.g., lasso)
- Numerous R packages–e.g., **glasso**, **huge**, and **flare**
  - Limited to *only* point estimates

Introduction

- To date, the most common applications have been gene co-expression and functional connectivity "networks"
  - Common to have more variables p than observations n (i.e., high dimensional data)
  - Requires some form of regularization to make estimation possible (e.g., lasso)
- Numerous R packages–e.g., **glasso**, **huge**, and **flare**
  - Limited to *only* point estimates
- High-dimensional inference is an active area of research
  - Confidence intervals and p-values provided in **SILGGM**

Introduction

- More recently, GGMs have become increasingly popular in the social-behavioral sciences

Introduction

- More recently, GGMs have become increasingly popular in the social-behavioral sciences
  - Regularization is not only unnecessary, but actually has worse performance than non-regularized methods

Introduction

- More recently, GGMs have become increasingly popular in the social-behavioral sciences
  - Regularization is not only unnecessary, but actually has worse performance than non-regularized methods
- Related publications:

Introduction

- More recently, GGMs have become increasingly popular in the social-behavioral sciences
  - Regularization is not only unnecessary, but actually has worse performance than non-regularized methods
- Related publications:

MULTIVARIATE BEHAVIORAL RESEARCH
https://doi.org/10.1080/00273171.2019.1575716

Routledge
Taylor & Francis Group

Check for updates

On Nonregularized Estimation of Psychological Networks

Donald R. Williams, Mijke Rhemtulla, Anna C. Wysocki, and Philippe Rast
Department of Psychology, University of California, Davis, CA, USA

Introduction

- More recently, GGMs have become increasingly popular in the social-behavioral sciences
  - Regularization is not only unnecessary, but actually has worse performance than non-regularized methods
- Related publications:

British Journal of Mathematical and Statistical Psychology (2019)

1

The British Psychological Society

www.wileyonlinelibrary.com

Back to the basics: Rethinking partial correlation network methodology

Donald R. Williams* and Philippe Rast
University of California, Davis, California, USA

Introduction

- We need methods specifically for $(p < n)$

Introduction

- We need methods specifically for $(p < n)$
  - Extend inference beyond point estimates

Introduction

- We need methods specifically for ($p < n$)
  - Extend inference beyond point estimates
  - Testing for invariances–i.e., the null hypothesis of conditional independence

Introduction

- We need methods specifically for ($p < n$)
  - Extend inference beyond point estimates
  - Testing for invariances–i.e., the null hypothesis of conditional independence
- Only one R package for Bayesian GGMs (**BDgraph**)
  - Limited to identifying non-zeros

Introduction

- We need methods specifically for ($p < n$)
  - Extend inference beyond point estimates
  - Testing for invariances–i.e., the null hypothesis of conditional independence
- Only one R package for Bayesian GGMs (**BDgraph**)
  - Limited to identifying non-zeros
- **BGGM** includes many novel (Bayesian) methods, each of which goes beyond detecting non-zero effects

Introduction

- We need methods specifically for ($p < n$)
  - Extend inference beyond point estimates
  - Testing for invariances–i.e., the null hypothesis of conditional independence
- Only one R package for Bayesian GGMs (**BDgraph**)
  - Limited to identifying non-zeros
- **BGGM** includes many novel (Bayesian) methods, each of which goes beyond detecting non-zero effects
  - Edge set identification

Introduction

- We need methods specifically for $(p < n)$
  - Extend inference beyond point estimates
  - Testing for invariances–i.e., the null hypothesis of conditional independence
- Only one R package for Bayesian GGMs (**BDgraph**)
  - Limited to identifying non-zeros
- **BGGM** includes many novel (Bayesian) methods, each of which goes beyond detecting non-zero effects
  - Edge set identification
  - Exploratory and **Confirmatory** hypothesis testing

Introduction

- We need methods specifically for ($p < n$)
  - Extend inference beyond point estimates
  - Testing for invariances–i.e., the null hypothesis of conditional independence
- Only one R package for Bayesian GGMs (**BDgraph**)
  - Limited to identifying non-zeros
- **BGGM** includes many novel (Bayesian) methods, each of which goes beyond detecting non-zero effects
  - Edge set identification
  - Exploratory and **Confirmatory** hypothesis testing
  - Bayesian measures of prediction error

Introduction

- We need methods specifically for ($p < n$)
  - Extend inference beyond point estimates
  - Testing for invariances–i.e., the null hypothesis of conditional independence
- Only one R package for Bayesian GGMs (**BDgraph**)
  - Limited to identifying non-zeros
- **BGGM** includes many novel (Bayesian) methods, each of which goes beyond detecting non-zero effects
  - Edge set identification
  - Exploratory and **Confirmatory** hypothesis testing
  - Bayesian measures of prediction error
  - Methods to compare any number of GGMs

Introduction

- We need methods specifically for ($p < n$)
  - Extend inference beyond point estimates
  - Testing for invariances–i.e., the null hypothesis of conditional independence
- Only one R package for Bayesian GGMs (**BDgraph**)
  - Limited to identifying non-zeros
- **BGGM** includes many novel (Bayesian) methods, each of which goes beyond detecting non-zero effects
  - Edge set identification
  - Exploratory and **Confirmatory** hypothesis testing
  - Bayesian measures of prediction error
  - Methods to compare any number of GGMs
  - etc. described in 3 first-author papers (in review): Journal of Mathematical Psychology, Psychological Methods,...

Introduction

This remainder of this talk will describe these methods

(Overview of an (in prep) Journal of Statistical Software paper)

The Gaussian Graphical Model

Psychological "Networks"

Captures undirected, conditional relations, that are typically visualized to infer the underlying structure

The Gaussian Graphical Model

Psychological "Networks"

Captures undirected, conditional relations, that are typically visualized to infer the underlying structure

- The undirected graph is $G = (V, E)$
  - Vertex set $V = \{1, 2, ..., p\}$ (i.e., nodes or variables)
  - Edge set $E \subset V \times V$

The Gaussian Graphical Model

Psychological "Networks"

Captures undirected, conditional relations, that are typically visualized to infer the underlying structure

- The undirected graph is $G = (V, E)$
  - Vertex set $V = \{1, 2, ..., p\}$ (i.e., nodes or variables)
  - Edge set $E \subset V \times V$
- Let $\mathbf{Y} = (\mathbf{y}_1, ..., \mathbf{y}_p)'$ be a $n \times p$ matrix
  - Each $\mathbf{y}$ is a n dimensional vector that is indexed by the graphs vertices.

The Gaussian Graphical Model

Psychological "Networks"

Captures undirected, conditional relations, that are typically visualized to infer the underlying structure

- The undirected graph is $G = (V, E)$
  - Vertex set $V = \{1, 2, ..., p\}$ (i.e., nodes or variables)
  - Edge set $E \subset V \times V$
- Let $\mathbf{Y} = (\mathbf{y}_1, ..., \mathbf{y}_p)'$ be a $n \times p$ matrix
  - Each $\mathbf{y}$ is a n dimensional vector that is indexed by the graphs vertices.
- Assume that $\mathbf{Y} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$
  - Mean vector $\boldsymbol{\mu} = (0_1, ..., 0_p)'$
  - $p \times p$ positive definite covariance matrix $\boldsymbol{\Sigma}$

The Gaussian Graphical Model

Psychological "Networks"

Captures undirected, conditional relations, that are typically visualized to infer the underlying structure

- The undirected graph is $G = (V, E)$
  - Vertex set $V = \{1, 2, ..., p\}$ (i.e., nodes or variables)
  - Edge set $E \subset V \times V$
- Let $\mathbf{Y} = (\mathbf{y}_1, ..., \mathbf{y}_p)'$ be a $n \times p$ matrix
  - Each $\mathbf{y}$ is a n dimensional vector that is indexed by the graphs vertices.
- Assume that $\mathbf{Y} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$
  - Mean vector $\boldsymbol{\mu} = (0_1, ..., 0_p)'$
  - $p \times p$ positive definite covariance matrix $\boldsymbol{\Sigma}$
- Denote the precision matrix $\boldsymbol{\Theta} = \boldsymbol{\Sigma}^{-1}$

The Gaussian Graphical Model

Conditional Relations

The graph, or the underlying conditional (*in*)dependence structure, is obtained from the off-diagonal elements–i.e., $\theta_{ij} \in \mathbf{\Theta}_{ij}, 1 \leq i < j \leq p$.

The Gaussian Graphical Model

Conditional Relations

The graph, or the underlying conditional (*in*)dependence structure, is obtained from the off-diagonal elements–i.e., $\theta_{ij} \in \mathbf{\Theta}_{ij}, 1 \leq i < j \leq p$.

- Conditional **In**dependence
  - $\mathbf{Y}_u \perp\!\!\!\perp \mathbf{Y}_k \mid \mathbf{Y}_{V_{\setminus\{u,k\}}} \notin E$

The Gaussian Graphical Model

Conditional Relations

The graph, or the underlying conditional (*in*)dependence structure, is obtained from the off-diagonal elements–i.e., $\theta_{ij} \in \mathbf{\Theta}_{ij}, 1 \leq i < j \leq p$.

- Conditional **In**dependence
  - $\mathbf{Y}_u \perp\!\!\!\perp \mathbf{Y}_k \mid \mathbf{Y}_{V_{\setminus \{u,k\}}} \notin E$
  - Evidence for the null hypothesis !

The Gaussian Graphical Model

Conditional Relations

The graph, or the underlying conditional (*in*)dependence structure, is obtained from the off-diagonal elements–i.e., $\theta_{ij} \in \mathbf{\Theta}_{ij}, 1 \leq i < j \leq p$.

- Conditional **In**dependence
  - $\mathbf{Y}_u \perp\!\!\!\perp \mathbf{Y}_k \mid \mathbf{Y}_{V_{\setminus \{u,k\}}} \notin E$
  - Evidence for the null hypothesis !
- Conditional **D**ependence
  - $\mathbf{Y}_u \not\perp\!\!\!\perp \mathbf{Y}_k \mid \mathbf{Y}_{V_{\setminus \{u,k\}}} \in E$

The Gaussian Graphical Model

Conditional Relations

The graph, or the underlying conditional (*in*)dependence structure, is obtained from the off-diagonal elements–i.e., $\theta_{ij} \in \mathbf{\Theta}_{ij}, 1 \leq i < j \leq p$.

- Conditional **In**dependence
  - $\mathbf{Y}_u \perp\!\!\!\perp \mathbf{Y}_k \mid \mathbf{Y}_{V_{\setminus \{u,k\}}} \notin E$
  - Evidence for the null hypothesis !
- Conditional **D**ependence
  - $\mathbf{Y}_u \not\perp\!\!\!\perp \mathbf{Y}_k \mid \mathbf{Y}_{V_{\setminus \{u,k\}}} \in E$
- Partial Correlations:

The Gaussian Graphical Model

Conditional Relations

The graph, or the underlying conditional (*in*)dependence structure, is obtained from the off-diagonal elements–i.e., $\theta_{ij} \in \mathbf{\Theta}_{ij}, 1 \leq i < j \leq p$.

- Conditional **In**dependence
  - $\mathbf{Y}_u \perp\!\!\!\perp \mathbf{Y}_k \mid \mathbf{Y}_{V_{\setminus \{u,k\}}} \notin E$
  - Evidence for the null hypothesis !
- Conditional **D**ependence
  - $\mathbf{Y}_u \not\perp\!\!\!\perp \mathbf{Y}_k \mid \mathbf{Y}_{V_{\setminus \{u,k\}}} \in E$
- Partial Correlations:

$$\rho_{ij} = \frac{-\theta_{ij}}{\sqrt{\theta_{ii}\theta_{jj}}} \tag{1}$$

The Gaussian Graphical Model

Conditional Relations

The graph, or the underlying conditional (*in*)dependence structure, is obtained from the off-diagonal elements–i.e., $\theta_{ij} \in \boldsymbol{\Theta}_{ij}, 1 \leq i < j \leq p$.

- Conditional **In**dependence
  - $\mathbf{Y}_u \perp\!\!\!\perp \mathbf{Y}_k \mid \mathbf{Y}_{V_{\setminus \{u,k\}}} \notin E$
  - Evidence for the null hypothesis !
- Conditional **D**ependence
  - $\mathbf{Y}_u \not\!\perp\!\!\!\perp \mathbf{Y}_k \mid \mathbf{Y}_{V_{\setminus \{u,k\}}} \in E$
- Partial Correlations:

$$\rho_{ij} = \frac{-\theta_{ij}}{\sqrt{\theta_{ii}\theta_{jj}}} \tag{1}$$

- E via hypothesis testing–e.g.,

$$\begin{aligned} \mathcal{H}_0 &: \rho_{ij} = 0 \\ \mathcal{H}_1 &: \rho_{ij} \neq 0 \end{aligned} \tag{2}$$

Method Organization

BGGM is built around two approaches for Bayesian inference: estimation and hypothesis testing

Method Organization

BGGM is built around two approaches for Bayesian inference: estimation and hypothesis testing

- Estimation
  - Similar to frequentist methods, but allow for probabilistic inference
  - e.g., Credible interval exclusion of zero, or directional posterior probabilities $p(\rho_{ij} > 0|\mathbf{Y})$

Method Organization

BGGM is built around two approaches for Bayesian inference: estimation and hypothesis testing

- Estimation
  - Similar to frequentist methods, but allow for probabilistic inference
  - e.g., Credible interval exclusion of zero, or directional posterior probabilities $p(\rho_{ij} > 0|\mathbf{Y})$
  - Prior distributions (often) play little role in inference, especially as $n \rightarrow \infty$

Method Organization

BGGM is built around two approaches for Bayesian inference: estimation and hypothesis testing

- Estimation
  - Similar to frequentist methods, but allow for probabilistic inference
  - e.g., Credible interval exclusion of zero, or directional posterior probabilities $p(\rho_{ij} > 0|\mathbf{Y})$
  - Prior distributions (often) play little role in inference, especially as $n \to \infty$
- Hypothesis Testing
  - Bayesian model comparison with the Bayes factor
  - (relative) Evidence between competing models–e.g., $\mathcal{M}_0$ vs. $\mathcal{M}_1$

Method Organization

BGGM is built around two approaches for Bayesian inference: estimation and hypothesis testing

- Estimation
  - Similar to frequentist methods, but allow for probabilistic inference
  - e.g., Credible interval exclusion of zero, or directional posterior probabilities $p(\rho_{ij} > 0|\mathbf{Y})$
  - Prior distributions (often) play little role in inference, especially as $n \rightarrow \infty$
- Hypothesis Testing
  - Bayesian model comparison with the Bayes factor
  - (relative) Evidence between competing models–e.g., $\mathcal{M}_0$ vs. $\mathcal{M}_1$
  - Prior distributions play a critical role (for all n), as they capture scientific expectations–i.e., actual predictions

Method Organization

BGGM is built around two approaches for Bayesian inference: estimation and hypothesis testing

- Estimation
  - Similar to frequentist methods, but allow for probabilistic inference
  - e.g., Credible interval exclusion of zero, or directional posterior probabilities $p(\rho_{ij} > 0|\mathbf{Y})$
  - Prior distributions (often) play little role in inference, especially as $n \rightarrow \infty$

I will focus on the estimation based methods

Estimation

Estimation based methods

These following methods are described in Williams (2018).

Williams, D. R. (2018). Bayesian Inference for Gaussian Graphical Models: Structure Learning, Explanation, and Prediction. PsyArXiv

Estimation

Estimation based methods

These following methods are described in Williams (2018).

Williams, D. R. (2018). Bayesian Inference for Gaussian Graphical Models: Structure Learning, Explanation, and Prediction. PsyArXiv

The joint posterior density for the precision matrix follows

$$p(\mathbf{\Theta}|\mathbf{Y}) \propto p(\mathbf{Y}|\mathbf{\Theta})p(\mathbf{\Theta}), \tag{3}$$

where $\mathbf{Y}$ is a $n \times p$ matrix drawn from a multivariate normal distribution–i.e.,

$$\mathbf{Y} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Theta}^{-1}). \tag{4}$$

Estimation

When using the conjugate Wishart prior, $\mathcal{W}(k, \mathbf{I}_p)$, with k degrees of freedom and identity matrix $\mathbf{I}_p$, the posterior distribution also has a Wishart distribution–i.e.,

$$\mathbf{\Theta}|\mathbf{Y} \sim \mathcal{W}\big(k + n, (\boldsymbol{S} + \epsilon\mathbf{I}_p)^{-1}\big), \tag{5}$$

where $\boldsymbol{S}$ is the sums of squares matrix $\mathbf{Y}'\mathbf{Y}$ and ϵ is a constant.

Estimation

When using the conjugate Wishart prior, $\mathcal{W}(k, \mathbf{I}_p)$, with k degrees of freedom and identity matrix $\mathbf{I}_p$, the posterior distribution also has a Wishart distribution–i.e.,

$$\mathbf{\Theta}|\mathbf{Y} \sim \mathcal{W}\big(k + n, (\boldsymbol{S} + \epsilon\mathbf{I}_p)^{-1}\big), \tag{5}$$

where $\boldsymbol{S}$ is the sums of squares matrix $\mathbf{Y}'\mathbf{Y}$ and ϵ is a constant.

The posterior mode then has a closed form

$$\text{argmax}_{\mathbf{\Theta}} p(\mathbf{\Theta}|\mathbf{Y}) = (k + n - p - 1)(\boldsymbol{S} + \epsilon\mathbf{I}_p)^{-1}. \tag{6}$$

Estimation

The posterior variance also has a closed form–i.e.,

$$\text{Var}(\boldsymbol{\Theta}|\mathbf{Y}) = (k+n)\big((\boldsymbol{S}+\epsilon\mathbf{I}_p)^{-1^2} + \boldsymbol{d}\boldsymbol{d}'\big), \tag{7}$$

where $\boldsymbol{d} = \text{diag}(\boldsymbol{S}+\epsilon\mathbf{I}_p)$.

Estimation

The posterior variance also has a closed form–i.e.,

$$\mathrm{Var}(\boldsymbol{\Theta}|\mathbf{Y}) = (k+n)\big((\boldsymbol{S}+\epsilon\mathbf{I}_p)^{-1^2} + \boldsymbol{d}\boldsymbol{d}'\big), \tag{7}$$

where $\boldsymbol{d} = \mathrm{diag}(\boldsymbol{S}+\epsilon\mathbf{I}_p)$.

By setting $k = p + 1$, and ϵ to a small value, say, 1.0×10^{-10}, results (approximately) in the maximum likelihood estimate $n(\boldsymbol{S}^{-1})$.

Estimation

The posterior variance also has a closed form–i.e.,

$$\text{Var}(\boldsymbol{\Theta}|\mathbf{Y}) = (k+n)\big((\boldsymbol{S}+\epsilon\mathbf{I}_p)^{-1^2} + \boldsymbol{d}\boldsymbol{d}'\big), \tag{7}$$

where $\boldsymbol{d} = \text{diag}(\boldsymbol{S}+\epsilon\mathbf{I}_p)$.

By setting $k = p + 1$, and ϵ to a small value, say, 1.0×10^{-10}, results (approximately) in the maximum likelihood estimate $n(\boldsymbol{S}^{-1})$.

Consequently, assuming that $p(\theta_{ij}|\mathbf{Y})$ is normally distributed, allows for constructing credible intervals and computing posterior probabilities.

Estimation

The posterior variance also has a closed form–i.e.,

$$\text{Var}(\boldsymbol{\Theta}|\mathbf{Y}) = (k+n)\big((\boldsymbol{S}+\epsilon\mathbf{I}_p)^{-1^2} + \boldsymbol{d}\boldsymbol{d}'\big), \tag{7}$$

where $\boldsymbol{d} = \text{diag}(\boldsymbol{S}+\epsilon\mathbf{I}_p)$.

By setting $k = p + 1$, and ϵ to a small value, say, 1.0×10^{-10}, results (approximately) in the maximum likelihood estimate $n(\boldsymbol{S}^{-1})$.

Consequently, assuming that $p(\theta_{ij}|\mathbf{Y})$ is normally distributed, allows for constructing credible intervals and computing posterior probabilities. The former can be computed as

$$\int_l^u p(\theta_{ij}|\mathbf{Y})d\theta_{ij} = 1 - \alpha, \tag{8}$$

where l and u denote the lower and upper bounds of the interval.

Estimation

A posterior probability can also be computed, for example with

$$\int_0^{\infty} p(\theta_{ij}|\mathbf{Y})d\theta_{ij}, \tag{9}$$

which corresponds to the posterior probability of a positive effect. Both can be computed analytically with the point estimate in (Equation 6) and variance in (Equation 7).

Estimation

A posterior probability can also be computed, for example with

$$\int_0^\infty p(\theta_{ij}|\mathbf{Y})d\theta_{ij}, \tag{9}$$

which corresponds to the posterior probability of a positive effect. Both can be computed analytically with the point estimate in (Equation 6) and variance in (Equation 7).

In **BGGM**, this simple solution can be used for determining E and analytically deriving "network" predictability

Estimation

A posterior probability can also be computed, for example with

$$\int_0^\infty p(\theta_{ij}|\mathbf{Y})d\theta_{ij}, \tag{9}$$

which corresponds to the posterior probability of a positive effect. Both can be computed analytically with the point estimate in (Equation 6) and variance in (Equation 7).

In **BGGM**, this simple solution can be used for determining E and analytically deriving "network" predictability

There is one drawback of this analytic form. Namely, because there are no posterior samples, this limits its applicability to some of the functions in **BGGM**

Estimation

BGGM include an additional approach for conveniently drawing samples from the posterior distribution.

Estimation

BGGM include an additional approach for conveniently drawing samples from the posterior distribution.

Assuming the Jeffrey's prior, $p(\mathbf{\Theta}^{-1}) \propto |\mathbf{\Theta}^{-1}|^{-(p+1)/2}$, allows for sampling directly from the a Wishart distribution–i.e.,

$$\mathbf{\Theta} \sim \mathcal{W}(n-1, \boldsymbol{S}^{-1}). \tag{10}$$

Estimation

BGGM include an additional approach for conveniently drawing samples from the posterior distribution.

Assuming the Jeffrey's prior, $p(\mathbf{\Theta}^{-1}) \propto |\mathbf{\Theta}^{-1}|^{-(p+1)/2}$, allows for sampling directly from the a Wishart distribution–i.e.,

$$\mathbf{\Theta} \sim \mathcal{W}(n-1, \boldsymbol{S}^{-1}). \tag{10}$$

These samples, $s = 1, ..., S$, are used to compute the posterior distribution for the $p \times p$ partial correlation matrix, with $\rho_{ij} \in \boldsymbol{\mathcal{P}}$–i.e.,

$$\boldsymbol{\mathcal{P}}^{(s)} = -\big([\text{diag}(\boldsymbol{\sigma})^{(s)}]^{-1}\mathbf{\Theta}^{(s)}[\text{diag}(\boldsymbol{\sigma})^{(s)}]^{-1}\big), \tag{11}$$

where $\boldsymbol{\sigma}$ are the square roots of diag($\mathbf{\Theta}$) and multiplying by -1 reverses the direction ($\pm$).

Estimation

Because ρ is a standardized effect, and thus each ρ_{ij} is on the same scale, **BGGM** allows for defining a *region of practical equivalence*.

Estimation

Because ρ is a standardized effect, and thus each ρ_{ij} is on the same scale, **BGGM** allows for defining a *region of practical equivalence*.

Rather than determining E in reference to 0, a neighborhood around zero is defined (a null area).

Estimation

Because ρ is a standardized effect, and thus each ρ_{ij} is on the same scale, **BGGM** allows for defining a *region of practical equivalence*.

Rather than determining E in reference to 0, a neighborhood around zero is defined (a null area).

Consequently, given some region, $|\rho_{ij}| \leq \vartheta_0$, there is support for null values when the posterior probability is above a pre-determined threshold,

$$\int_{-\vartheta_0}^{\vartheta_0} p(\rho_{ij}|\mathbf{Y}) d\rho_{ij} > 1 - \alpha, \tag{12}$$

where $p(\rho_{ij}|\mathbf{Y})$ is the posterior distribution.

Estimation

Conversely, determining practically meaningful edges, and thus belonging to E, are determined with

$$\int_{-\infty}^{-\vartheta_0} p(\rho_{ij}|\mathbf{Y}) d\rho_{ij} + \int_{\vartheta_0}^{\infty} p(\rho_{ij}|\mathbf{Y}) d\rho_{ij} > 1 - \alpha. \tag{13}$$

Estimation

Conversely, determining practically meaningful edges, and thus belonging to E, are determined with

$$\int_{-\infty}^{-\vartheta_0} p(\rho_{ij}|\mathbf{Y})d\rho_{ij} + \int_{\vartheta_0}^{\infty} p(\rho_{ij}|\mathbf{Y})d\rho_{ij} > 1 - \alpha. \tag{13}$$

This is implemented in **BGGM** for both determining conditional dependent and practically independent relations, as well as comparing two edges (e.g., $\rho_{1,2} - \rho_{1,3}$).

Estimation

Conversely, determining practically meaningful edges, and thus belonging to E, are determined with

$$\int_{-\infty}^{-\vartheta_0} p(\rho_{ij}|\mathbf{Y})d\rho_{ij} + \int_{\vartheta_0}^{\infty} p(\rho_{ij}|\mathbf{Y})d\rho_{ij} > 1 - \alpha. \tag{13}$$

This is implemented in **BGGM** for both determining conditional dependent and practically independent relations, as well as comparing two edges (e.g., $\rho_{1,2} - \rho_{1,3}$).

These posterior samples are also used to compute Bayesian R^2 and Bayesian leave-one-out cross-validation.

Estimation: Structure Learning

Estimation: Structure Learning

The analytic form is implemented with:

```r
library(BGGM)
# fit model
Y <- BGGM::bfi[,1:5]
# fit model (analytic = T)
fit_analytic <- estimate(Y, analytic = T)
```

Estimation: Structure Learning

The analytic form is implemented with:

```
library(BGGM)
# fit model
Y <- BGGM::bfi[,1:5]
# fit model (analytic = T)
fit_analytic <- estimate(Y, analytic = T)
```

```
# print
print(fit_analytic)
```

```
## BGGM: Bayesian Gaussian Graphical Models
## ---
## Type: Estimation (Analytic Solution)
## Posterior Samples:
## Observations (n): 2709
## Variables (p): 5
## Edges: 10
## ---
## Call:
## estimate.default(x = Y, analytic = T)
## ---
## Date: Thu May 23 11:36:07 2019
```

Estimation: Structure Learning

The edge set E is selected with:

```r
# select E
E <- select(fit_analytic, ci_width = 0.95)
summary(E)
```

```
## 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
```

Estimation: Structure Learning

The analytic solution estimates $\mathbf{\Theta}$, and thus, further information (e.g., credible intervals) is not provided.

Estimation: Structure Learning

The analytic solution estimates $\boldsymbol{\Theta}$, and thus, further information (e.g., credible intervals) is not provided.

This is because the non-standardized off-diagonal elements are in the opposite direction of the partial correlations, which in our experience, can lead to confusion.

Estimation: Structure Learning

The analytic solution estimates $\boldsymbol{\Theta}$, and thus, further information (e.g., credible intervals) is not provided.

This is because the non-standardized off-diagonal elements are in the opposite direction of the partial correlations, which in our experience, can lead to confusion.

The sampling based method can be used to summarize the partial correlations–i.e.,

Estimation: Structure Learning

The analytic solution estimates $\mathbf{\Theta}$, and thus, further information (e.g., credible intervals) is not provided.

This is because the non-standardized off-diagonal elements are in the opposite direction of the partial correlations, which in our experience, can lead to confusion.

The sampling based method can be used to summarize the partial correlations–i.e.,

```
# fit model
fit_sampling <- estimate(Y, analytic = F)
# select E
E <- select(fit_sampling, ci_width = 0.95,
            rope = NULL, prob = NULL)
```

Estimation: Structure Learning

This object is then summarized with:

```
summary(E, summarize = T, digits = 2)
```

```
## BGGM: Bayesian Gaussian Graphical Models
## ---
## Type: Selected Graph (Sampling)
## Credible Interval: 95 %
## Connectivity: 80 %
## ---
## Call:
## select.estimate(x = fit_sampling, ci_width = 0.95, rope = NULL,
##     prob = NULL)
## ---
## Estimates:
##
##  egde post_mean post_sd   2.5%  97.5%
##  1--2   -0.2400   0.018 -0.275 -0.205
##  1--3   -0.1075   0.019 -0.145 -0.069
##  2--3    0.2869   0.017  0.253  0.320
##  1--4   -0.0071   0.019 -0.045  0.030
##  2--4    0.1648   0.019  0.128  0.201
##  3--4    0.1774   0.018  0.141  0.214
##  1--5   -0.0095   0.019 -0.047  0.028
##  2--5    0.1558   0.019  0.119  0.192
##  3--5    0.3582   0.016  0.326  0.389
##  4--5    0.1216   0.019  0.085  0.158
## ---
```

Estimation: Structure Learning

A key feature of GGMs is the tradition of visualizing the underlying structure.

Estimation: Structure Learning

A key feature of GGMs is the tradition of visualizing the underlying structure.

To this end, BGGM offers several plotting options.

Estimation: Structure Learning

A key feature of GGMs is the tradition of visualizing the underlying structure.

To this end, BGGM offers several plotting options.

The partial correlations can be plotted with:

Estimation: Structure Learning

A key feature of GGMs is the tradition of visualizing the underlying structure.

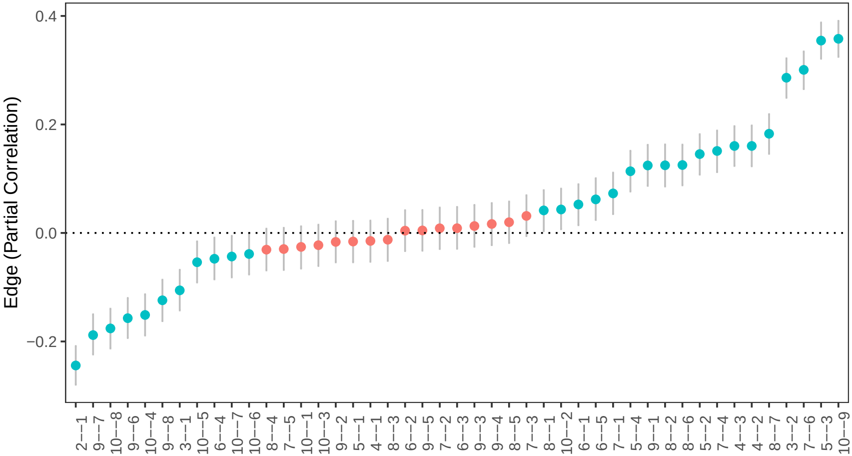
To this end, BGGM offers several plotting options.

The partial correlations can be plotted with:

```
# p = 10
Y <- BGGM::bfi[,1:10]
# sample posterior
fit_sampling <- estimate(Y, analytic = F)
# plot
plot_1 <- plot(fit_sampling, ci_width = 0.95,
               width = 0.1, size = 2) +
               coord_cartesian() +
               theme(axis.text.x = element_text(angle = 90))
```

Note the plots can be further customized with **ggplot2**.

Estimation: Structure Learning



Estimation: Structure Learning

BGGM also includes two options for visualizing E.

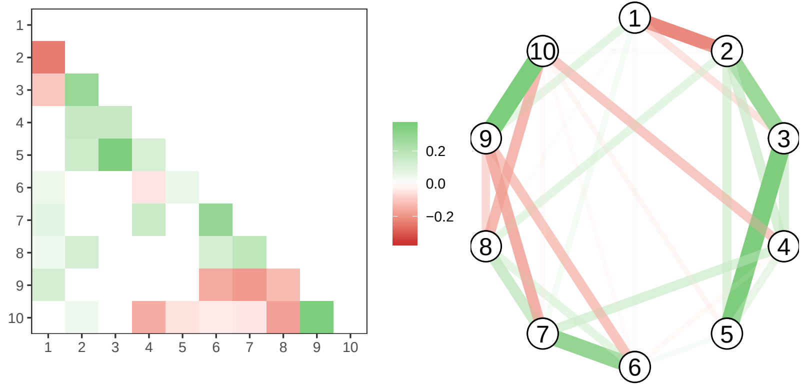
A heatmap is plotted with:

```
# E
E <- select(fit_sampling, ci_width = 0.95)
# plot
plot_2 <-plot(E, type = "heatmap",
               lower_tri = TRUE)
```

On the other hand, a “network” plot follows:

```
# plot
plot_3 <-plot(E, type = "network",
              lower_tri = TRUE,
              node_outer = 10,
              node_inner = 9,
              node_text_size = 6)
```

Estimation: Structure Learning



Estimation: Structure Learning

BGGM extends inference beyond identifying non-zero partial correlations. The region of practical equivalence can be used for this purpose, as it allows for determining which relations are practically zero.

```
# p = 10
Y <- BGGM::bfi[,1:10]
# sample posterior
fit_sample <- estimate(Y, samples = 5000, analytic = F)
# E
E <- select(fit_sample, rope = 0.1, prob = 0.95)
```

Estimation: Structure Learning

```
head(E, nrow = 10, summarize = T, digits = 2)
```

```
## BGGM: Bayesian Gaussian Graphical Models
## ---
## Type: Selected Graph (Sampling)
## Probability: 0.95
## Region of Practical Equivalence:[-0.1, 0.1]
## Connectivity: 31.1 %
## ---
## Call:
## select.estimate(x = fit_sample, rope = 0.1, prob = 0.95)
## ---
## pr_out: post prob outside of rope
## pr_in: post prob inside of rope
## ---
## Estimates:
##
##  egde post_mean post_sd pr_out  pr_in
##  1--2    -0.244   0.018   1.00 0.0000
##  1--3    -0.105   0.020   0.61 0.3916
##  2--3     0.286   0.018   1.00 0.0000
##  1--4    -0.015   0.019   0.00 1.0000
##  2--4     0.161   0.019   1.00 0.0008
##  3--4     0.160   0.019   1.00 0.0004
##  1--5    -0.016   0.019   0.00 1.0000
##  2--5     0.145   0.019   0.99 0.0080
##  3--5     0.354   0.017   1.00 0.0000
##  4--5     0.114   0.019   0.77 0.2286
## ---
```

Estimation: Structure Learning

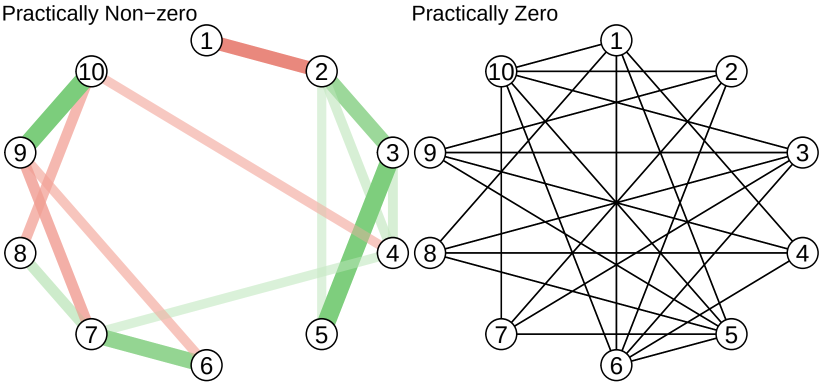
In this case, the plot function returns two objects: (1) the selected, non-zero, edges; (2) those for which there is support for the null values.

```
plts <- plot(E, type = "network",
             layout ='circle',
             node_outer = 10,
             node_inner = 9,
             node_text_size = 6)

# practically non-zero
plot_4 <- plts$plot_nonzero +
             ggtitle("Practically Non-zero") +
             theme(plot.title = element_text(size = 15))

# practically zero
plot_5 <- plts$plot_zero +
              ggtitle("Practically Zero") +
              theme(plot.title = element_text(size = 15))
```

Estimation: Structure Learning



Estimation: Edge Differences

Differences between partial correlations are often tested in GGMs. In **BGGM**, it is possible to use posterior probabilities to determine which edges are practically equivalent.

Estimation: Edge Differences

Differences between partial correlations are often tested in GGMs. In **BGGM**, it is possible to use posterior probabilities to determine which edges are practically equivalent.

This is implemented with:

```
# p = 10
Y <- BGGM::bfi[,1:10]
# sample posterior
fit_sample <- estimate(Y, analytic = F)

# edge difference
edge_difference <- edge_compare(fit_sample,
                                contrast =  list("1--5 - 1--3",
                                                 "1--2 - 1--6",
                                                 "1--4 - 1--7",
                                                 "1--5 - 1--10",
                                                 "1--2 - 1--9"),
                                ci_width = 0.95,
                                rope = 0.1)
```

Estimation: Edge Differences

```r
summary(edge_difference)
```

```
## BGGM: Bayesian Gaussian Graphical Models
## ---
## Type: Edge comparison(s)
## Region of Practical Equivalence:[-0.1, 0.1]
## ---
## Call:
## edge_compare.estimate(x = fit_sample, contrast = list("1--5 - 1--3",
##     "1--2 - 1--6", "1--4 - 1--7", "1--5 - 1--10", "1--2 - 1--9"),
##     ci_width = 0.95, rope = 0.1)
## ---
## Posterior Estimates:
##
##      contrast post_mean post_sd pr_out  pr_in
##   1--5 - 1--3    0.0911  0.0318  0.392  0.608
##   1--2 - 1--6   -0.2972  0.0263      1      0
##   1--4 - 1--7   -0.0871  0.0295 0.3316 0.6684
##  1--5 - 1--10    0.0106  0.0271  6e-04 0.9994
##   1--2 - 1--9   -0.3684  0.0265      1      0
```

Estimation: Prediction

Estimation: Prediction

- Direct Correspondence between the elements of $\boldsymbol{\Theta}$ and regression (coefficients and error variance)

Estimation: Prediction

- Direct Correspondence between the elements of $\mathbf{\Theta}$ and regression (coefficients and error variance)
- Recall:

Estimation: Prediction

- Direct Correspondence between the elements of $\mathbf{\Theta}$ and regression (coefficients and error variance)
- Recall:
  - $\mathbf{Y}'\mathbf{Y}$ ($\mathbf{\Sigma}_{\text{MLE}} = n^{-1}\mathbf{Y}'\mathbf{Y}$)

Estimation: Prediction

- Direct Correspondence between the elements of $\boldsymbol{\Theta}$ and regression (coefficients and error variance)
- Recall:
  - $\mathbf{Y}'\mathbf{Y}$ ($\boldsymbol{\Sigma}_{\text{MLE}} = n^{-1}\mathbf{Y}'\mathbf{Y}$)
  - $(\mathbf{Y}'\mathbf{Y})'$ (non-normalized precision matrix)

Estimation: Prediction

- Direct Correspondence between the elements of $\mathbf{\Theta}$ and regression (coefficients and error variance)
- Recall:
  - $\mathbf{Y}'\mathbf{Y}$ ($\mathbf{\Sigma}_{\text{MLE}} = n^{-1}\mathbf{Y}'\mathbf{Y}$)
  - $(\mathbf{Y}'\mathbf{Y})'$ (non-normalized precision matrix)
- $\beta = (\mathbf{X}'\mathbf{X})'\mathbf{X}'\mathbf{y}$

Estimation: Prediction

For $j = 1, ..., p$, let $\mathbf{y} = V_j$ and $\mathbf{X} = V_{\backslash\{j\}}$. Then fit the pth regression model–i.e.,

Estimation: Prediction

For $j = 1, ..., p$, let $\mathbf{y} = V_j$ and $\mathbf{X} = V_{\setminus\{j\}}$. Then fit the pth regression model–i.e.,

$$\mathbf{y} = \boldsymbol{X}\beta^{(j)} + \varepsilon, \tag{14}$$

where ε is an n-dimensional vector, with the mean as a vector of zeroes, and the covariance matrix as $\sigma^2\mathbf{I}_n$.

Estimation: Prediction

For $j = 1, ..., p$, let $\mathbf{y} = V_j$ and $\mathbf{X} = V_{\setminus\{j\}}$. Then fit the pth regression model–i.e.,

$$\mathbf{y} = \boldsymbol{X}\beta^{(j)} + \varepsilon, \tag{14}$$

where ε is an n-dimensional vector, with the mean as a vector of zeroes, and the covariance matrix as $\sigma^2\mathbf{I}_n$.

$\beta^{(j)}$ denotes the $(p-1)$ dimensional vector of coefficients for the jth regression model.

Estimation: Prediction

The regression coefficients and error variances then correspond to the off-diagonal and diagonal elements of $\mathbf{\Theta}$–i.e.,

Estimation: Prediction

The regression coefficients and error variances then correspond to the off-diagonal and diagonal elements of $\mathbf{\Theta}$–i.e.,

$$\theta_{ij} = \frac{-\beta_{ij}}{\sigma_j^2} \text{ and } \theta_{jj} = \frac{1}{\sigma_j^2}, \tag{15}$$

where θ_{ij} denotes the covariance corresponding to ith row and jth column of $\mathbf{\Theta}$.

Estimation: Prediction

The regression coefficients and error variances then correspond to the off-diagonal and diagonal elements of $\mathbf{\Theta}$–i.e.,

$$\theta_{ij} = \frac{-\beta_{ij}}{\sigma_j^2} \text{ and } \theta_{jj} = \frac{1}{\sigma_j^2}, \tag{15}$$

where θ_{ij} denotes the covariance corresponding to ith row and jth column of $\mathbf{\Theta}$.

Consequently, for all posterior samples, $s = 1, ..., S$,

$$\beta_{ij}^{(s)} = \frac{-\theta_{ij}^{(s)}}{\theta_{jj}^{(s)}} \text{ and } \sigma_j^{2^{(s)}} = \frac{1}{\theta_{jj}^{(s)}} \tag{16}$$

results in the posterior distribution for each regression coefficient and residual variance.

Estimation: Prediction

It follows that **BGGM** can also be used for the purpose of Bayesian multiple regression–i.e.,

Estimation: Prediction

It follows that **BGGM** can also be used for the purpose of Bayesian multiple regression–i.e.,

```
fit <- estimate(Y, samples = 5000)
coefficients(fit, node = 1, ci_width = 0.95)
```

```
## BGGM: Bayesian Gaussian Graphical Models
## ---
## Type: Inverse to Regression
## ---
## Call:
## BGGM:::beta_summary(x = fit, node = node, ci_width = ci_width,
##     samples = samples)
## ---
## Estimates:
##
##  node post_mean post_sd   2.5%  97.5%
##     2    -0.277   0.021 -0.318 -0.238
##     3    -0.124   0.022 -0.166 -0.081
##     4    -0.015   0.021 -0.055  0.026
##     5    -0.018   0.021 -0.059  0.024
##     6     0.056   0.021  0.017  0.096
##     7     0.081   0.021  0.038  0.122
##     8     0.044   0.020  0.004  0.085
##     9     0.141   0.021  0.100  0.184
##    10    -0.028   0.021 -0.069  0.012
## ---
```

Estimation: Bayesian R^2

In-sample prediction error:

Estimation: Bayesian R^2

In-sample prediction error:

```r
# training data
Y_train <- BGGM::bfi[1:100,1:10]
# fit to training data
fit_train <- estimate(Y_train, samples = 5000)
# compute Bayes R2
train_R2 <- predict(fit_train,
                    ci_width = 0.90,
                    samples = 1000,
                    measure = "R2")
# summary for first 2 rows
head(train_R2, nrow = 2)
```

```
## BGGM: Bayesian Gaussian Graphical Models
## ---
## Type: In-sample predictive accuracy
## Measure: Variance Explained (R2)
## ---
## Call:
## predict.estimate(fit = fit_train, ci_width = 0.9, samples = 1000,
##     measure = "R2")
## ---
## Estimates:
##
##  node post_mean    post_sd       2.5%     97.5%
##     1 0.1670357 0.06560828 0.04998099 0.2973988
##     2 0.2917582 0.06505191 0.16503668 0.4128562
## ---
```

Estimation: Bayesian R^2

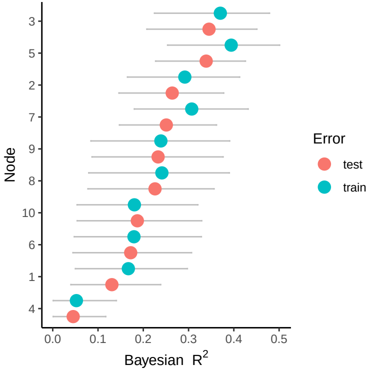
Out-of-sample prediction error:

```r
# test data
Y_test <-  BGGM::bfi[101:2000,1:10]
# predict test data
test_R2 <- predict(fit_train, ci_width = 0.90,
                   test_data = Y_test,
                   samples = 1000, measure = "R2")
```

The work flow is completed by visualizing Bayesian R^2 for each node-i.e.,

```r
# prior training and test error in the same plot
plt_6 <- plot(x1 = train_R2, x2 =  test_R2, order = "test")
```

Estimation: Bayesian R^2



Estimation: Bayesian Leave-one-Out Cross-Validation

BGGM also includes Bayesian leave-one-out cross-validation.

Estimation: Bayesian Leave-one-Out Cross-Validation

BGGM also includes Bayesian leave-one-out cross-validation.

It is implemented with:

```
# p = 10
Y <- BGGM::bfi[1:1000,1:10]
# sample posterior
fit_sample <- estimate(Y, samples = 5000)
# Bayesian LOO
bayes_loo <- loocv(fit_sample)
```

Estimation: Bayesian Leave-one-Out Cross-Validation

```r
# nodewise loo summary
summary(bayes_loo)
```

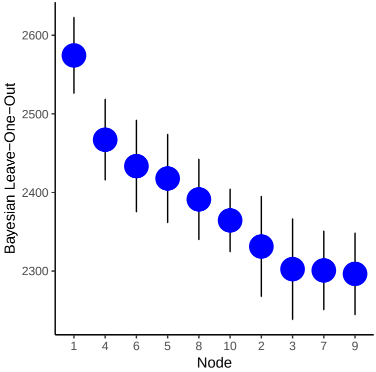
```
## BGGM: Bayesian Gaussian Graphical Models
## ---
## Type: Leave-One-Out Prediction Error (Bayesian)
## ---
## Call:
## loocv.default(x = fit_sample)
## ---
## Estimates:
##
##   node      loo   loo_se
##     1 2574.364 48.55682
##     2 2331.292 63.81872
##     3 2302.470 64.33004
##     4 2467.128 51.75399
##     5 2417.796 56.24886
##     6 2433.493 58.62935
##     7 2300.869 50.29901
##     8 2391.216 51.31349
##     9 2296.485 52.30362
##    10 2364.485 40.15936
## ---
```

Estimation: Bayesian Leave-one-Out Cross-Validation

The results are plotted with:

```
# plot CV error
plt_7 <- plot(bayes_loo, size = 8) +
          theme_classic() +
          ylab("Bayesian Leave-One-Out")
```

Estimation: Bayesian Leave-one-Out Cross-Validation



Estimation: Analytic Leave-one-Out Cross-Validation

BGGM includes a convenient, analytic form, for computing leave-one-out (*loo*) prediction error:

Estimation: Analytic Leave-one-Out Cross-Validation

BGGM includes a convenient, analytic form, for computing leave-one-out (*loo*) prediction error:

- Let $\mathbf{X}^* \subseteq \mathbf{X}$ denote the selected predictors for the *p*th regression model
  - $\mathbf{H} = \mathbf{X}^*(\mathbf{X}^{*'}\mathbf{X}^*)\mathbf{X}^*$

Estimation: Analytic Leave-one-Out Cross-Validation

BGGM includes a convenient, analytic form, for computing leave-one-out (*loo*) prediction error:

- Let $\mathbf{X}^* \subseteq \mathbf{X}$ denote the selected predictors for the pth regression model
  - $\mathbf{H} = \mathbf{X}^*(\mathbf{X}^{*'}\mathbf{X}^*)\mathbf{X}^*$
  - $\mathbf{H}_{\boldsymbol{d}} = \text{diag}(\mathbf{H})$

Estimation: Analytic Leave-one-Out Cross-Validation

BGGM includes a convenient, analytic form, for computing leave-one-out (*loo*) prediction error:

- Let $\mathbf{X}^* \subseteq \mathbf{X}$ denote the selected predictors for the *p*th regression model
  - $\mathbf{H} = \mathbf{X}^*(\mathbf{X}^{*'}\mathbf{X}^*)\mathbf{X}^*$
  - $\mathbf{H}_{\boldsymbol{d}} = \text{diag}(\mathbf{H})$
  - $\hat{\mathbf{y}} = \mathbf{X}^*\hat{\beta}^*$

Estimation: Analytic Leave-one-Out Cross-Validation

BGGM includes a convenient, analytic form, for computing leave-one-out (*loo*) prediction error:

- Let $\mathbf{X}^* \subseteq \mathbf{X}$ denote the selected predictors for the pth regression model
  - $\mathbf{H} = \mathbf{X}^*(\mathbf{X}^{*'}\mathbf{X}^*)\mathbf{X}^*$
  - $\mathbf{H}_{\boldsymbol{d}} = \text{diag}(\mathbf{H})$
  - $\hat{\mathbf{y}} = \mathbf{X}^*\hat{\beta}^*$
  - $\varepsilon = \mathbf{y} - \hat{\mathbf{y}}$

Estimation: Analytic Leave-one-Out Cross-Validation

BGGM includes a convenient, analytic form, for computing leave-one-out (*loo*) prediction error:

- Let $\mathbf{X}^* \subseteq \mathbf{X}$ denote the selected predictors for the pth regression model
  - $\mathbf{H} = \mathbf{X}^*(\mathbf{X}^{*'}\mathbf{X}^*)\mathbf{X}^*$
  - $\mathbf{H}_{\boldsymbol{d}} = \mathrm{diag}(\mathbf{H})$
  - $\hat{\mathbf{y}} = \mathbf{X}^*\hat{\beta}^*$
  - $\varepsilon = \mathbf{y} - \hat{\mathbf{y}}$

$$loo = \sum_{i=1}^{n} \left(\frac{\varepsilon_i}{1 - H_{\boldsymbol{d}}^i}\right)^2$$

Estimation: Analytic Leave-one-Out Cross-Validation

This method is implemented with

```
# p = 10
Y <- BGGM::bfi[1:1000,1:10]
# analytic solution
fit_analytic <- estimate(Y, analytic = T)
# analytic LOO (PRESS; based on point estimates)
press_loo <- loocv(fit_analytic)
```

Estimation: Analytic Leave-one-Out Cross-Validation

```
summary(press_loo)
```

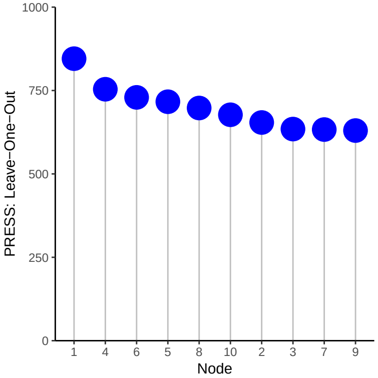
```
## BGGM: Bayesian Gaussian Graphical Models
## ---
## Type: Leave-One-Out Prediction Error (Analytic)
## ---
## Call:
## loocv.default(x = fit_analytic)
## ---
## Estimates:
##
##   node      loo      rss
##      1 845.6207 838.6759
##      2 654.2324 644.6005
##      3 634.5554 629.3224
##      4 753.9226 744.1923
##      5 716.4956 710.4838
##      6 729.5712 723.7107
##      7 633.1005 626.5797
##      8 697.7085 686.3148
##      9 630.1037 621.9319
##     10 677.4374 672.4343
## ---
```

Estimation: Analytic Leave-one-Out Cross-Validation

The results are plotted with:

```
# plot CV error
plt_8 <- plot(press_loo, size = 8) +
         theme_classic() +
         ylab("PRESS: Leave-One-Out") +
         scale_y_continuous(expand = c(0, 0),
         limit = c(0, 1000))
```

Estimation: Analytic Leave-one-Out Cross-Validation



Conclusion

Next DIPS presentation I will discuss the Bayesian hypothesis testing methods.

Then the following I will discuss the methods to compare any number of GGMs.

Conclusion

Next DIPS presentation I will discuss the Bayesian hypothesis testing methods.

Then the following I will discuss the methods to compare any number of GGMs.

Thank You !