Journal of Statistical Software

MMMMMM YYYY, Volume VV, Issue II. *doi: 10.18637/jss.v000.i00*

Learning permutation symmetries of a Gaussian vector with gips in R

Adam Chojecki
Warsaw University of Technology

Paweł Morgen
Warsaw University of Technology

Bartosz Kołodziejek
Warsaw University of Technology

Abstract

We implement the Bayesian model selection procedure within Gaussian vectors invariant under the permutation subgroup introduced in Graczyk, Ishi, Kołodziejek, and Massam (2022a).

Keywords: Bayesian model selection, permutation symmetry, high-dimensional statistics, dimensionality reduction, R.

1. Introduction

The study of hidden structures in the data is one of the biggest challenges in modern mathematical statistics and machine learning Hastie, Tibshirani, and Friedman (2009). In the era of Big Data, the number of variables p usually significantly exceeds the number of observations n, which makes classical methods inapplicable. The study of covariance matrix is the basic way to describe the dependency structure of a random vector and provides a convenient way to quantify the dependencies between variables. One solution to the problem of insufficient number of observations relative to the number of variables is to restrict ourselves to models with lower dimensionality. For this purpose, so-called graphical models have been introduced, where a conditional independence structure (so-called Markov structure) is imposed on the distribution of a random vector. Such structures are conveniently described by graphs and allow to significantly reduce the dimensionality of the problem. However, if the graph is not sparse enough (the size of the largest clique still significantly exceeds the sample size), then such a procedure may not allow for a reliable estimation of the covariance matrix. If

data is insufficient and some inference must be performed, one has to propose additional assumptions or restrictions. In such a situation, colored graphical models should be considered, where in addition to conditional independence, certain symmetries of the covariance matrix are imposed. A rich family of such symmetry conditions can be expressed using language of permutations. This idea was introduced in Andersson (1975); Andersson and Madsen (1998) and Højsgaard and Lauritzen (2008). In the latter paper, three types of such models (RCOP among them) were introduced to describe situations where some entries of concentration or partial correlation matrices are approximately equal. These equalities can be represented by a colored graph. The RCOP model, apart from Markov structure, permits additional invariance of the distribution with respect to some permutation subgroup. We say that a distribution of a p-dimensional random vector Z is invariant under permutation subgroup Γ on $V = \{1, \ldots, p\}$ if Z has the same distribution as $(X_{\sigma(i)})_i$ for any permutation $\sigma \in \Gamma$ Andersson (1975). This property is called the permutation symmetry of the distribution of Z and allows for a significant dimensionality reduction of the model. When the conditional dependency graph is known to be the complete graph on V, then such model was studied in Graczyk *et al.* (2022a). In that paper the authors introduced a Bayesian model selection procedure in the Gaussian setting. Equivalently, assuming a prior distribution on the parameters, they described the posterior probability of a given model. This allows one to find "the best permutation group under which the data is invariant". Not only this results in dimensionality reduction but also provides a simple and natural interpretability of the results: if $(X_i, X_j) \stackrel{d}{=} (X_j, X_i)$, then one can say that both X_i and X_j play a similar role in the model.

In the present paper, we present a package **gips**, which contains an implementation of the model selection procedure from Graczyk *et al.* (2022a). The **gips** package, described in this paper, will help you with two things:
1. Finding hidden permutation symmetries between variables (exploratory analysis).
2. Estimating covariance assuming known permutation symmetry.
To our best knowledge, there are no software packages in R (and other programming languages), that tackle the subject of permutation symmetry. We work exclusively with zero mean Gaussian vectors, however the method can be applied to centered data and, if the sample size n is reasonably large, to standardized data. If the full symmetry is assumed, that is, Z is invariant under the whole symmetric group, then the Maximum Likelihood Estimator (MLE) requires $n = 1$ sample only to exists only. The same phenomena applies when the normal sample is invariant under a cyclic subgroup generated by one cycle of length p. Although it is natural to consider permutations symmetries along with conditional independence structure, we will follow Graczyk *et al.* (2022a) and consider only the case when the Markov structure is trivial, i.e. there are no conditional independencies among the variables. This already allows for a significant dimensionality reduction together with an easy interpretation. The development of the method to take into account non trivial Markov structure is a matter of future study and we will consider extending the package if new theory is developed. However, a simple heuristic can be used to find non-trivial Markov structure using our model - see (Graczyk *et al.* 2022a, Section 1.2) and (Graczyk, Ishi, Kołodziejek, and Massam 2022b, Section 4.1).

Even though there are no other software packages in R that find permutation symmetry, we decided to compare results of our model with the canonical methods of attacking the high-dimensionality problems via RIDGE or LASSO estimation and model selection (used, for example, in **glasso**, **huge**, **hdi** and **rags2ridges** R packages). Those methods are equivalent to

estimation with constraints, or conversely to Bayesian estimation with Gaussian or Laplace priors, respectively. We compare the results of **gips** with these methods in the following sections.

The GLASSO method allows for a simultaneous model selection in the space of graphical models and estimation of the precision matrix. Due to the nature of the problem, we believe that the goal of finding permutation symmetry cannot be solved by penalized likelihood methods, which are generally much faster than Bayesian methods. We hope that our method will also find many applications.

The paper is structured as follows.

1.1. Motivation behind permutation symmetries

We argue that it is natural to expect certain symmetries in various applications, which strengthens the argument for investigating permutation symmetry of the data. There are natural symmetries in the data from gene expression. Namely, expression of a given gene is triggered by binding the transcription factors to the gene transcription factor binding sites. The transcription factors are the proteins produced by other genes, say regulatory genes. In the gene network there are often many genes triggered by the same regulatory genes and it makes sense to assume that their relative expressions depend on the abundance of proteins of the regulatory genes (i.e. gene expressions) in a similar way Graczyk *et al.* (2022a). Extracting permutation symmetries can be used to identify genes having the same function or groups of genes having similar interactions or regulated by the same mechanism. They are especially useful in revealing the structures of gene regulatory networks Kotiang and Eslami (2020). Moreover, the authors of Gao and Massam (2015) mention examples of social networks where geographical or social group clusters force additional symmetries to be taken into account. In the studying of dynamics of the human brain, one believes that the human brain has a natural symmetric structure on the left and right hemispheres Ranciati, Roverato, and Luati (2021).

The discovery of hidden symmetries can make a fundamental contribution to understanding complex mechanisms. Extracting patterns from the expression profile would provide great insight into gene function and regulatory systems Toh and Horimoto (2002). Clustering genes with expression profiles can be utilized to predict the functions of gene products with functions that are unknown, and to identify sets of genes that are regulated by the same mechanism.

We note that due to the nature of the problem, the model is not scale invariant int he following sense: if a vector $(Z_i)_{i\in V}$ is invariant under a permutation subgroup, then the vector $(a_i Z_i)_{i\in V}$ is in general not invariant under any permutation subgroup. Thus, it is best to apply our procedure to situation when all variables are measured on some common scale. We point out however that there are many such examples. For example, the data from gene expression are on the same scale in the sense that they are results of experiments of the same type, measured in the same gauges. For further references see the Introduction in Graczyk *et al.* (2022a) and and Gehrmann (2011); Gao and Massam (2015); Massam, Li, and Gao (2018); Li, Gao, and Massam (2020); Li, Sun, Wang, and Gao (2021); Ranciati *et al.* (2021).

1.2. Arbitrary permutation symmetries vs cyclic permutation symmetries

As was observed in Graczyk *et al.* (2022a), model selection within arbitrary permutation

subgroup is a very hard task, not only because of theoretical, but more importantly because of technical issues. In Graczyk *et al.* (2022a) the general model was developed, but it was applied only to cyclic subgroups, which are subgroups generated by a single permutation. Such restriction allows to develop efficient methods for finding the so-called structure parameters of the model, which are indispensable for performing model selection procedure. All technical details will be presented in the forthcoming sections.

Moreover, we claim that cyclic subgroups constitute reach enough family, see Section 4.1 in Graczyk *et al.* (2022a). Since they correspond to simpler symmetries, they should be also easier interpretable. Even though our procedure searches among cyclic subgroups only, it can give valuable information when the true subgroup is not cyclic, see Section 3.3 of Graczyk *et al.* (2022b). In fact, if the posterior probabilities (which are calculated by **gips**) are high for several groups, it is natural to expect that the data will be invariant under the group containing those subgroups.

2. Models and software

2.1. Preliminaries

After this informal introduction, let us define the key concepts and present the theory behind **gips** package in a formal manner.

Fix $p \in \{1, 2, \ldots\}$. Let $Z = (Z_1, ..., Z_p)^\top$ be a multivariate random variable following a centered Gaussian model $\mathrm{N}_p(0, \Sigma)$. Let $\mathrm{Sym}(p; \mathbb{R})$ and $\mathrm{Sym}^+(p; \mathbb{R})$ denote the space of $p \times p$ symmetric matrices and the corresponding cone of positive definite matrices. Let $V = \{1, ...p\}$ be a finite index set. Finally, let $\mathfrak{S}_p$ denote the symmetric group on V, that is, the group of all permutations on $\{1, ..., p\}$ with function composition as group operation.

For a subgroup $\Gamma \subset \mathfrak{S}_p$, we define the colored space, i.e., the space of symmetric matrices invariant under Γ,

$$\mathcal{Z}_\Gamma := \{S \in \mathrm{Sym}(p; \mathbb{R}) \colon S_{ij} = S_{\sigma(i)\sigma(j)} \text{ for all } \sigma \in \Gamma\},$$

and the colored cone of positive definite matrices valued in $\mathcal{Z}_\Gamma$,

$$\mathcal{P}_\Gamma := \mathcal{Z}_\Gamma \cap \mathrm{Sym}^+(p; \mathbb{R}).$$

This definition is closely connected with the key idea of this paper, so let us dive into it further.

2.2. Permutation symmetry

Let Γ be an arbitrary subgroup of $\mathfrak{S}_p$. We say that the distribution of Z is invariant by a subgroup Γ if Z has the same distribution as $(Z_{\sigma(i)})_{i \in V}$ for all $\sigma \in \Gamma$. This invariance property can be expressed as a condition on the covariance matrix: Z is invariant by Γ if and only if for all $i, j \in V$,

$$\Sigma_{ij} = \Sigma_{\sigma(i)\sigma(j)} \quad \text{for all} \quad \sigma \in \Gamma. \tag{1}$$

If $\Gamma = \mathfrak{S}_p$, then the above conditions imply that all diagonal entries of Σ are the same and, similarly, the off-diagonal entries are the same. On the other hand, if Γ is the trivial subgroup,

i.e., $\Gamma = \{\mathrm{id}\}$, then (1) does not impose any restrictions on the entries of Σ. If Γ is non-trivial, then the sample size n required for the MLE to exist is lower than p, see subsection 2.4

2.3. Other preliminaries

Note that the mapping $\Gamma \mapsto \mathcal{Z}_\Gamma$ is not one-to-one. In particular, it is easy to see that with $p = 3$ we have $\mathcal{Z}_{\langle(1,2,3)\rangle} = \mathcal{Z}_{\mathfrak{S}_3}$. In view of (1), we see that Z is invariant by Γ if and only if $\Sigma \in \mathcal{P}_\Gamma$.

Each permutation $\sigma \in \mathfrak{S}_p$ can be represented in a cyclic form. E.g., if $\sigma\colon 1 \mapsto 2, 2 \mapsto 1, 3 \mapsto 3$, then $\sigma = (1,2)(3)$. The identity permutation is denoted by id. The number of cycles, denoted hereafter by C_σ, is the same for all such cyclic representations. Note that in C_σ, we also count the cycles of length 1.

We say that a permutation subgroup $\Gamma \subset \mathfrak{S}_p$ is cyclic if $\Gamma = \{\sigma, \sigma^2, \ldots, \sigma^N\} =: \langle\sigma\rangle$ for some $\sigma \in \mathfrak{S}_p$, where N is the smallest positive integer for which $\sigma^N = \mathrm{id}$. Then N is called the order of the subgroup Γ. If p_i is the length of ith cycle in a cyclic decomposition of $\sigma \in \mathfrak{S}_p$, then N equals the least common multiple of $p_1, p_2, \ldots, p_{C_\sigma}$. We note that the mapping $\sigma \mapsto \langle\sigma\rangle$ is not one-to-one. Indeed, we have $\langle\sigma\rangle = \langle\sigma^k\rangle$ for all $k = 1, \ldots, N-1$, which are coprime with N. We identify each cyclic permutation subgroup by its generator which is smallest in the lexicografic order.

The important feature of cyclic subgroups is that they correspond to different colored spaces. More precisely, if $\mathcal{Z}_{\langle\sigma\rangle} = \mathcal{Z}_{\langle\sigma'\rangle}$ for some $\sigma, \sigma' \in \mathfrak{S}_p$, then $\langle\sigma\rangle = \langle\sigma'\rangle$.

2.4. The MLE in the Gaussian model invariant by permutation symmetry

Let $Z^{(1)}, \ldots, Z^{(n)}$ be an i.i.d. sample from $\mathrm{N}_p(0, \Sigma)$. Thanks to equality restrictions in (1), the permutation invariant models have fewer parameters to estimate. Therefore the sample size required for the MLE to exist is lower than p. Assume that $\Sigma \in \mathcal{P}_\Gamma$, where Γ is a cyclic subgroup, say $\Gamma = \langle\sigma\rangle$. Then, the MLE of Σ exists if and only if

$$n \geq n_0 := C_\sigma. \tag{2}$$

In particular, if $\sigma = \mathrm{id}$, then no restrictions are imposed on Σ, and we recover the well-known condition that the number of samples n has to be greater or equal to the number of variables $C_{\mathrm{id}} = p$. However, if σ consists of a single cycle, i.e., $C_\sigma = 1$, then the MLE always exists. This remarkable observation is crucial in the high dimensional setting. It is therefore of interest to develop an efficient tool for finding the permutation symmetry in the data.

If (2) is satisfied, then the MLE of Σ is given by

$$\hat{\Sigma} = \pi_\Gamma\left(\frac{1}{n}\sum_{i=1}^{n} Z^{(i)} \cdot (Z^{(i)})^\top\right),$$

where π_Γ is the orthogonal projection on the colored space $\mathcal{Z}_\Gamma$, which is defined by

$$\pi_\Gamma(X) = \frac{1}{\#\Gamma}\sum_{\sigma\in\Gamma} \sigma \cdot X \cdot \sigma^\top,$$

where we identify the permutation σ with its permutation matrix.

2.5. Bayesian model

Now we focus on methods aiming to discover a permutation symmetry in the data. Naturally, the stochastic nature of problem means, that we will not observe equalities in the simplest covariance estimator, $Z^\top Z$. In fact, in a sample of limited size equalities of the form as 1 occur with probability 0. What is needed is a method of inferring *reasonable* permutation from the data.

The following model was introduced in Graczyk *et al.* (2022a). Suppose that the multivariate Gaussian sample $Z^{(1)}, \ldots, Z^{(n)}$ given $\{K = k, \Gamma = c\}$ consists of i.i.d. $\mathrm{N}_p(0, k^{-1})$ random vectors with $k \in \mathcal{P}_c$. Let Γ be uniformly distributed on the set $\mathcal{C} := \{\langle\sigma\rangle : \sigma \in \mathfrak{S}_p\}$ of cyclic subgroups of $\mathfrak{S}_p$. Assume that K given $\{\Gamma = c\}$ follows the Diaconis-Ylvisaker conjugate prior (Diaconis and Ylvisaker (1979)) distribution defined by its density

$$f_{K|\Gamma=c}(k) = \frac{1}{I_c(\delta, D)} \mathrm{Det}(k)^{(\delta-2)/2} e^{-\frac{1}{2}\mathrm{Tr}[D\cdot k]} \mathbf{1}_{\mathcal{P}_c}(k),$$

where $\delta > 0$ and $D \in \mathcal{P}_\Gamma$ are the parameters and $I_c(\delta, D)$ is the normalizing constant, $c \in \mathcal{C}$. Then, it is easily seen that the posterior probability is proportional to

$$\mathsf{P}\left(\Gamma = c | Z^{(1)}, \ldots, Z^{(n)}\right) \propto \frac{I_c(\delta + n, D + U)}{I_c(\delta, D)}, \qquad c \in \mathcal{C}. \tag{3}$$

with $U = \sum_{i=1}^n Z^{(i)} \cdot (Z^{(i)})^\top$. In order to exploit (3), one has to be able to calculate, or at least approximate, the quotients of the normalizing constants.

Following the Bayesian paradigm, we choose the cyclic subgroup with the highest posterior probability, i.e.,

$$\hat{\Gamma} = \arg\max_{c\in\mathcal{C}} \mathsf{P}\left(\Gamma = c | Z^{(1)}, \ldots, Z^{(n)}\right). \tag{4}$$

Graczyk *et al.* (2022a) gave general formulas for these normalizing constants. They depend on the hyper-parameters $(\delta, D$ and the so-called structure constants, which are the parameters of the block decomposition of the colored space $\mathcal{Z}_\Gamma$, which we introduce hereafter.

Although the choice of hyper-parameters is not scale invariant, it is a common practice in similar models to take $\delta = 3$ and $D = I_p$, Massam *et al.* (2018). These are the default parameters in our method, however we recommend not to choose any particular value of D, but consider $D = d\, I_p$ for several values of $d > 0$. Results of Section 3.3 (see also (Graczyk *et al.* 2022a, Section 4.2)) indicate that big values of d favour big symmetries. Thus, by such exploratory analysis a user can tune parameter d so that the resulting model is most meaningful to him.

2.6. Block decomposition of the colored space

Let $\Gamma = \langle\sigma\rangle$ be a cyclic group of order N. We present the steps required to find the ingredients, which are necessary for the calculation of the normalizing constants $I_\Gamma(\delta, D)$.

Let p_i be the length of ith cycle in a cyclic decomposition of $\sigma \in \mathfrak{S}_p$ and let $\{i_1, \ldots, i_{C_\sigma}\}$ be a complete system of representatives of the cycles of σ. Let $(e_i)_{i=1}^p$ be the standard basis of $\mathbb{R}^p$.

1. For $c = 1, \ldots, C_\sigma$ calculate $v_1^{(c)}, \ldots, v_{p_c}^{(c)} \in \mathbb{R}^p$ as

$$
\begin{aligned}
v_1^{(c)} &:= \sqrt{\frac{1}{p_c}} \sum_{k=0}^{p_c-1} e_{\sigma^k(i_c)}, \\
v_{2\beta}^{(c)} &:= \sqrt{\frac{2}{p_c}} \sum_{k=0}^{p_c-1} \cos\Big(\frac{2\pi\beta k}{p_c}\Big) e_{\sigma^k(i_c)} \qquad (1 \le \beta < p_c/2), \\
v_{2\beta+1}^{(c)} &:= \sqrt{\frac{2}{p_c}} \sum_{k=0}^{p_c-1} \sin\Big(\frac{2\pi\beta k}{p_c}\Big) e_{\sigma^k(i_c)} \qquad (1 \le \beta < p_c/2), \\
v_{p_c}^{(c)} &:= \sqrt{\frac{1}{p_c}} \sum_{k=0}^{p_c-1} \cos(\pi k) e_{\sigma^k(i_c)} \qquad (\text{if } p_c \text{ is even}).
\end{aligned}
$$

2. Construct an orthogonal matrix U_Γ by arranging column vectors $\{v_k^{(c)}\}$, $1 \le c \le C_\sigma$, $1 \le k \le p_c$, in the following way: we put $v_k^{(c)}$ earlier than $v_{k'}^{(c')}$ if
(i) $\frac{[k/2]}{p_c} < \frac{[k'/2]}{p_{c'}}$, or
(ii) $\frac{[k/2]}{p_c} = \frac{[k'/2]}{p_{c'}}$ and $c < c'$, or
(iii) $\frac{[k/2]}{p_c} = \frac{[k'/2]}{p_{c'}}$ and $c = c'$ and k is even and k' is odd.

3. For $\alpha = 0, 1, \ldots, \lfloor \frac{N}{2} \rfloor$ calculate

$$
\begin{aligned}
r_\alpha^* &= \#\{c \in \{1, \ldots, C_\sigma\} \colon \alpha\, p_c \text{ is a multiple of } N\}, \\
d_\alpha^* &= \begin{cases} 1 & (\alpha = 0 \text{ or } N/2), \\ 2 & (\text{otherwise}). \end{cases}
\end{aligned}
$$

Then set $L = \#\{\alpha \colon r_\alpha^* > 0\}$, $r = (r_\alpha^* \colon r_\alpha^* > 0)$ and $d = (d_\alpha^* \colon r_\alpha^* > 0)$.

In the definition of r_α^* we treat 0 as a multiple of N and so $r_0^* = C_\sigma$.

Having the orthogonal matrix U_Γ, it is easy to find the block decomposition of the colored space. For each $S \in \mathcal{Z}_\Gamma$, we have

$$
U_\Gamma^\top \cdot S \cdot U_\Gamma = \begin{pmatrix} x_1 & & \\ & \ddots & \\ & & x_L \end{pmatrix}, \tag{5}
$$

where $x_i \in \mathrm{Sym}(r_i d_i; \mathbb{R})$. The parameters (the structure constants) of this block decomposition are crucial for finding the normalizing constant

$$
I_\Gamma(\delta, D) = \int_{\mathcal{P}_\Gamma} \mathrm{Det}(k)^{(\delta-2)/2} e^{-\frac{1}{2}\mathrm{Tr}[D \cdot k]} dk.
$$

2.7. Normalizing constants

We note that the formulas for normalizing constants for arbitrary subgroup $\Gamma \subset \mathfrak{S}_p$ are presented in Graczyk *et al.* (2022a). Here we specialize these formulas to cyclic subgroups, which allows a significant simplification.

In the previous subsection, we defined structure constants $(r_i, d_i)_{i=1}^L$. We have

$$I_\Gamma(\delta, D) = e^{-A_\Gamma(\delta-2)/2-B_\Gamma} \prod_{i=1}^{L} \Gamma_i\left(1 + d_i(\delta + r_i - 3)/2\right) \gamma_\Gamma\left(D/2, \delta\right),$$

where $A_\Gamma = \sum_{i=1}^L r_i\, d_i \log d_i$, $B_\Gamma = \frac{1}{2}\sum_{i=1}^L r_i(1 + (r_i - 1)d_i/2) \log d_i$ and

$$\Gamma_i(\lambda) = (2\pi)^{r_i(r_i-1)d_i/4} \prod_{k=1}^{r_i} \Gamma(\lambda - (k-1)d_i/2).$$

For $S \in \mathcal{Z}_\Gamma$ let x_i denote the $(r_i d_i) \times (r_i d_i)$ matrix defined in (5), $i = 1, \ldots, L$. Then,

$$\gamma_\Gamma(S, \delta) = \prod_{i=1}^{L} \mathrm{Det}(x_i)^{-(\delta+r_i-3)/2-1/d_i}.$$

2.8. Searching for a permutation with MAP probability

The quotient from 3 is a way to numerically evaluate a permutation (more precisely, a cyclic group generated by a permutation). Finding a permutation with a high evaluation score is still challenging, because of the huge size of space of possible permutation symmetries.

Recall that $\mathcal{C}$ is the number of cyclic subgroups of $\mathfrak{S}_p$. When p is small (e.g., less than 9), we can calculate posterior probabilities (3) for all $c \in \mathcal{C}$ and find $\hat{\Gamma}$ from (4) based on exact calculations.

The cardinality of $\mathcal{C}$ grows super-exponentially with p; In particular, for $p = 150$, the cardinality of $\mathcal{C}$ is roughly 10^{250} (see OEIS[1] sequence A051625). This makes it computationally infeasible to calculate the quotients (3) for all $c \in \mathcal{C}$.

We propose to use a method using Monte Carlo Markov Chain. Below we define an irreducible Markov chain $(\sigma_t)_t$ that travels even bigger space, $\mathfrak{S}_p$, and run the Metropolis-Hastings algorithm for finding pre-estimates. Then, we take into account the fact that the actual search space is $\mathcal{C}$ and obtain the estimates of the posterior probabilities. The Metropolis-Hastings algorithm gives statistical guarantees that the estimates will converge to the actual values as the number of iterations goes to infinity.

The Metropolis-Hastings algorithm

A transposition is a permutation whose cyclic representation consists of one cycle with two elements and other length-1 cycles. Let $\mathcal{T}$ denote the set of all transpositions.

Starting from an arbitrary permutation $\sigma_0 \in \mathfrak{S}_p$, repeat the following two steps for $t = 1, \ldots, T$:

1. Sample x_t uniformly from the set $\mathcal{T}$ and set $\sigma' = \sigma_{t-1} \circ x_t$;
2. Accept the move $\sigma_t = \sigma'$ with probability

$$\min\left\{1, \frac{I_{\langle\sigma'\rangle}(\delta + n, D + U)\; I_{\langle\sigma_{t-1}\rangle}(\delta, D)}{I_{\langle\sigma'\rangle}(\delta, D)\; I_{\langle\sigma_{t-1}\rangle}(\delta + n, D + U)}\right\}.$$

   If the move is rejected, set $\sigma_t = \sigma_{t-1}$.

[1] The On-Line Encyclopedia of Integer Sequences, https://oeis.org/.

In this way, we obtain a sequence of permutations $(\sigma_t)_{t=1}^T$, where T is the number of steps made by the above algorithm.

Let Φ be the Euler's totient function, i.e., $\Phi(n) = \#\{k \in \{1, \ldots, n\} \colon k \text{ and } n \text{ are coprime}\}$. We have as $T \to \infty$, $c \in \mathcal{C}$,

$$\hat{\pi}_c := \frac{\sum_{t=1}^T \mathbf{1}(\langle \sigma_t \rangle = c)}{\Phi(c) \sum_{t=1}^T 1/\Phi(\langle \sigma_t \rangle)} \xrightarrow{a.s.} \mathsf{P}\left(\Gamma = c | Z^{(1)}, \ldots, Z^{(n)}\right).$$

In practice, one has the approximation $\hat{\pi}_c \approx \mathsf{P}\left(\Gamma = c | Z^{(1)}, \ldots, Z^{(n)}\right)$ for large T. The estimator of the group with highest posterior probability is given by

$$\hat{\Gamma} = \arg\max_{c \in (\sigma_t)_{t=1}^T} \mathsf{P}\left(\Gamma = c | Z^{(1)}, \ldots, Z^{(n)}\right).$$

The maximum above is taken over all permutations visited by the Metropolis-Hastings algorithm.

2.9. Centering and standardizing data

We again stress that due to the nature of the problem, considered model is not invariant under changing the scale of variables: if Z is invariant by a subgroup, then a random vector $\mathrm{diag}(\alpha) \cdot Z$ for $\alpha \in \mathbb{R}^p$ is in general not invariant under any permutation subgroup.

Therefore it is recommended to apply our procedure to data which has comparable scales and keep all variables in the same units.

The default model applied to the zero mean Gaussian sample. If $Z^1, \ldots, Z^{(p)}$ is an i.i.d. sample from $\mathrm{N}_p(\mu, \Sigma)$, then we can center the data and take this fact into account by setting the parameter `was_mean_estimated = TRUE`.

If the sample size n is reasonably large, then it is a common practice to assume that the standardized normal sample (which has multivariate t-distribution) is Gaussian. After standardization, the empirical covariance has the unit diagonal. Such approach may favor cyclic subgroups whose generators consists of a single cycle as they correspond to matrices with constant diagonal. We believe that such analyses in certain situations can yield interesting conclusions, we do not recommend standardizing the data for small n.

3. Illustrations

The primary use case of **gips** package is finding a permutation with high a posteriori probability in the Bayesian model introduced in previous sections The package implements the Metropolis-Hastings algorithm described in previous section in the `find_MAP` method called on `gips` objects.

3.1. Toy example

Let us illustrate the concept of permutation symmetry using package **gips** in a simple use case. We'll be using small `aspirin` dataset from **HSAUR** package. We will specify a permutation here manually. It can be done algorithmically using `find_MAP` function, but this feature is not the main focus of this section.

```
library(gips)
data("aspirin", package="HSAUR")

Z <- aspirin
Z
```

```
#>   dp   tp   da   ta
#>   67  624   49  615
#>   64   77   44  757
#>   126  850  102  832
#>   38  309   32  317
#>   52  406   85  810
#>  219 2257  346 2267
#>  1720 8600 1570 8587
```

After loading data and some preprocessing, we are ready to proceed with creating a `"gips"` object. Such objects contain information about the (covariance) matrix, the search process and the found permutation, under which the data seems invariant. Here also `plot` method for them is demonstrated.

Since no permutation is specified, the identity permutation is applied, which corresponds to no permutation symmetry.

```
g <- gips(S, number_of_observations)
add_labels(plot(g, type = "heatmap"))
```

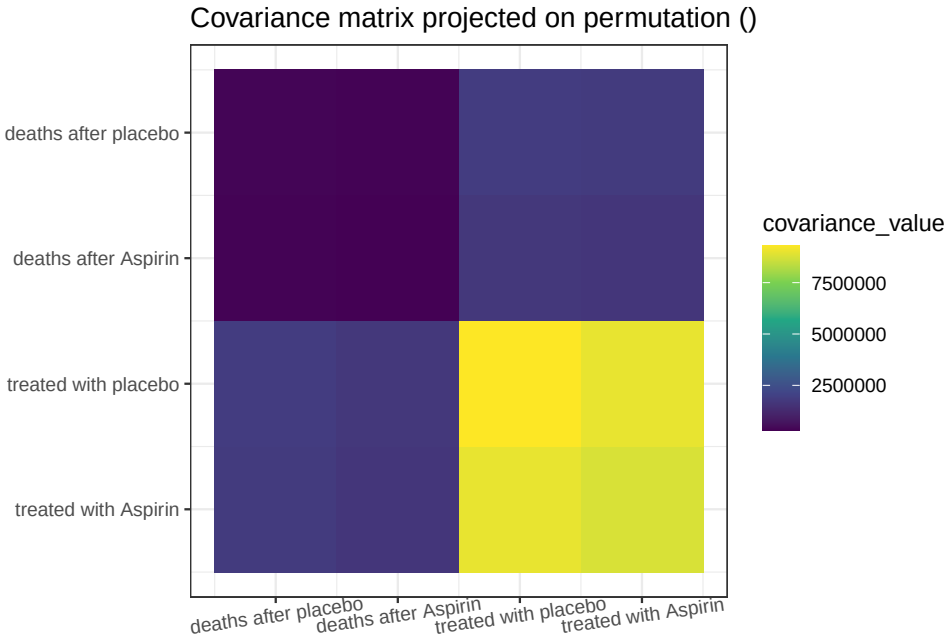

Figure 1: The heatmap of the the empirical covariance matrix.

We can see some strong similarities between the covariances of columns 3 and 4. Those have similar variances (`S[3,3]` $\approx$ `S[4,4]`), and their covariances with the rest of the columns are alike (`S[1,3]` $\approx$ `S[1,4]` and `S[2,3]` $\approx$ `S[2,4]`).

Using the language of permutations one can expect, that **the true covariance matrix is invariant under** $(3, 4)$ **permutation**. This means, that:

- $\mathsf{VAR}[Z_3] = \mathsf{VAR}[Z_4]$
- $\mathsf{COV}[Z_1, Z_3] = \mathsf{COV}[Z_1, Z_4]$
- $\mathsf{COV}[Z_2, Z_3] = \mathsf{COV}[Z_2, Z_4]$

Under this new assumption, we can give a new estimate for covariance matrix. Theoretically, we are projecting the matrix on the space of positive definite matrices invariant under permutation $(3, 4)$. In practice, we replace pairs of entries mentioned above with their averages.

```
g_with_perm <- gips(S, number_of_observations, perm="(3,4)")
S_projected <- project_matrix(S, g_with_perm)
round(S_projected)
```

```
#>         [,1]    [,2]    [,3]    [,4]
#> [1,]  381405  345527 1839144 1839144
#> [2,]  345527  316411 1687459 1687459
#> [3,] 1839144 1687459 9030113 8991343
#> [4,] 1839144 1687459 8991343 9030113
```

```
add_labels(plot(g_with_perm, type = "heatmap"))
```

This `S_projected` matrix can now be interpreted as a more stable covariance matrix estimator. We can also interpret the data suggesting there is, for example, the same covariance of "number of deaths after Aspirin" with "number of people treated with X" no matter if the "X" represents the placebo or Aspirin.

Since we have only 4 variables, the space of permutation symmetries to consider is small. We can calculate the a posteriori probabilities directly for every permutation (a *brute force* approach) as follows.

```
g_MAP <- find_MAP(g, optimizer = "brute_force",
                  return_probabilities = TRUE,
                  save_all_perms = TRUE)
```

```
#> ================================================================================
#> ================================================================================
```

```
g_MAP
```

```
#> The permutation (3,4)
#>  - was found after 24 log_posteriori calculations
#>  - is 106222567640989 times more likely than the starting, () permutation.
```

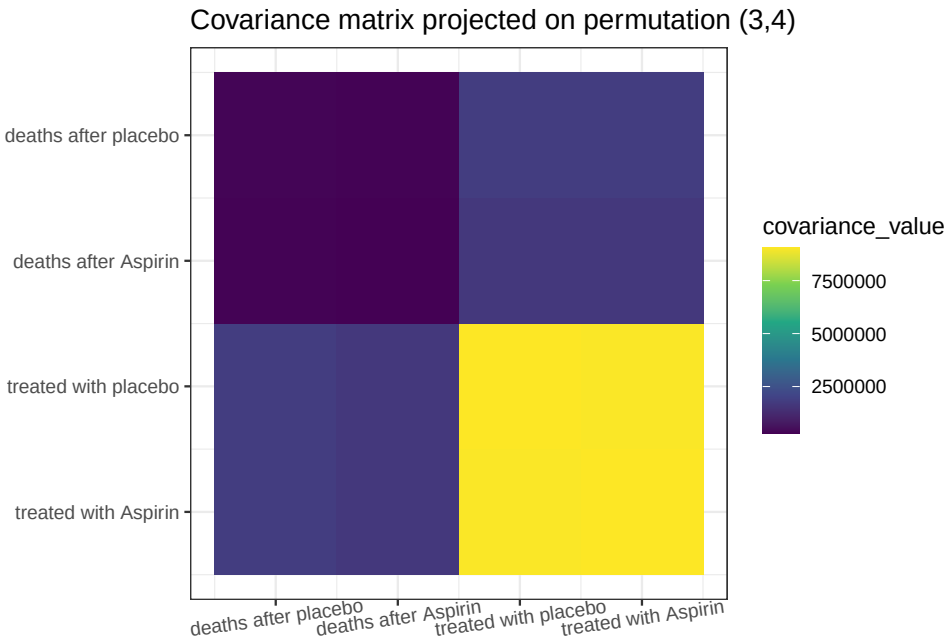

Figure 2: The heatmap of the maximul likelihood estimator for the covariance matrix Σ assuming invariance under the subgroup $\Gamma = \langle (3,4) \rangle$.

```
prob_AP <- get_probabilities_from_gips(g_MAP)
prob_AP
```

```
#>           ()        (3,4)        (2,3)      (2,3,4)        (2,4)        (1,2)
#> 6.839794e-15 7.265405e-01 3.124315e-20 1.888644e-10 2.532751e-20 2.043752e-07
#>   (1,2)(3,4)      (1,2,3)    (1,2,3,4)    (1,2,4,3)      (1,2,4)        (1,3)
#> 2.105929e-01 5.321630e-13 3.498803e-08 5.406414e-08 1.605486e-13 2.552393e-12
#>      (1,3,4)   (1,3)(2,4)    (1,3,2,4)        (1,4)   (1,4)(2,3)
#> 1.579596e-03 2.237285e-18 6.128671e-02 3.573103e-12 2.218938e-18
```

Take a look at the permutations with largest AP probabilities:

```
sort(prob_AP, decreasing = TRUE)[1:5]
```

```
#>        (3,4)   (1,2)(3,4)    (1,3,2,4)      (1,3,4)        (1,2)
#> 7.265405e-01 2.105929e-01 6.128671e-02 1.579596e-03 2.043752e-07
```

We see that the permutation subgroup $\langle (1,2)(3,4) \rangle$ is roughly 3.4 times less likely than "the best" permutation subgroup $\langle (3,4) \rangle$.

3.2. Searching for permutation symmetry

In general setting a permutation with high a posteriori probability can be found using `find_MAP` method. By default, it uses the Metropolis-Hastings algorithm.

Let us present the capabilities of **gips** package using breast cancer data from `GEOQuery` package from BioConductor. Code for downloading and minimal preprocessing is available in Supplementary Material.

```
Z <- breast_cancer
dim(Z)

#> [1]  58 150

Z[1:5, 1:5]

#>           1053_at 200039_s_at 200053_at 200079_s_at 200628_s_at
#> GSM79115 6.524984    9.459735  8.192045    9.391399    7.767257
#> GSM79116 6.733606    8.376470  7.957888    8.893947    7.669449
#> GSM79118 6.430577    8.689633  7.759223    8.938420    7.221506
#> GSM79122 6.741080    9.174062  8.348095    9.702200    7.245702
#> GSM79123 6.537532    9.289905  7.809236    9.477653    7.394035
```

Indeed, we observe, that we have fewer observations that variables. Let us search for permutation symmetries. Create `gips` object and run `find_MAP` method on it:

```
S <- cov(Z)

set.seed(1234)
g <- gips(S, 58, was_mean_estimated = TRUE)

set.seed(1234)
%# Takes in our case 2-3 hours
g_MAP <- find_MAP(g, max_iter = 150000, optimizer = "MH")
g_MAP

#> The permutation (1,8,10,69)(2,122,121,16,132,140,150,49,34,92)(3,43,127,20,101,131,
21,119)(4,89)(5,81,72,103,37,31,130,137,87,59)(6,138,93,60,61,64,107,147,118,26,63,136,
145,70,88)(7,124,114,142,143,91,50,74,22,125,36)(9,104,39,76)(11,79,19,126)(12,75,129,
40,117,100,146,108,97,94,95,148,62,58,66,32,134,33)(13,113,110,46,105,57)(14,28,53,71)
(15,90,80,29)(17,78,98,135)(18,82,77,84,106,99,123,139)(23,30,52,96,149,144,38,116)(24,
109,111,73,42,35,67,102)(25,44,120,112)(27,56,51,45)(41,47)(65,86,128,133)(85,141,115)
#>  - was found after 150000 log_posteriori calculations
#>  - is Inf times more likely than the starting, () permutation.
```

3.3. Hyperparameter's influence

Drawing conclusions about hyperparameter's influence on method's outcome is hard and must be done with caution. They influence directly the shape of a posteriori distribution and therefore change both the theoretical MAP and the difficulty of the optimization problem, thus MAP solution obtainable in a given computational budget.

In order to partially investigate the characteristics of a posteriori distribution, we have considered δ to be 3 (default) and 10 and D of the form $d \cdot I_p$, where I_p is the identity matrix. Comparisons are conducted with $n = p = 20$ across strengths of structure of true covariance matrices. By *strength of structure* we mean strictness of the equality conditions imposed on

a matrix. We take n_0, the number of samples needed for existence of MLE, as its measure; the greater n_0, the weaker the structure. We have used matrices invariant by permutation groups

large structure $\langle(1, 2, \ldots, 20)\rangle$,

moderate structure $\langle(1, 2, 3)(4, 5, 6)(7, 8, 9)(10, 11)(12, 13)(14, 15)\rangle$,

no structure $\langle\mathrm{id}\rangle$.

Experiment in each setting was repeated 7 times. `gips` method was used with 150 000 iterations.

We took a closer look at two aspects: the overall shape of the (estimated) a posteriori distribution on n_0's (so summed probabilities of permutation groups with equal n_0) and at variability of the distribution, expressed as its entropy.

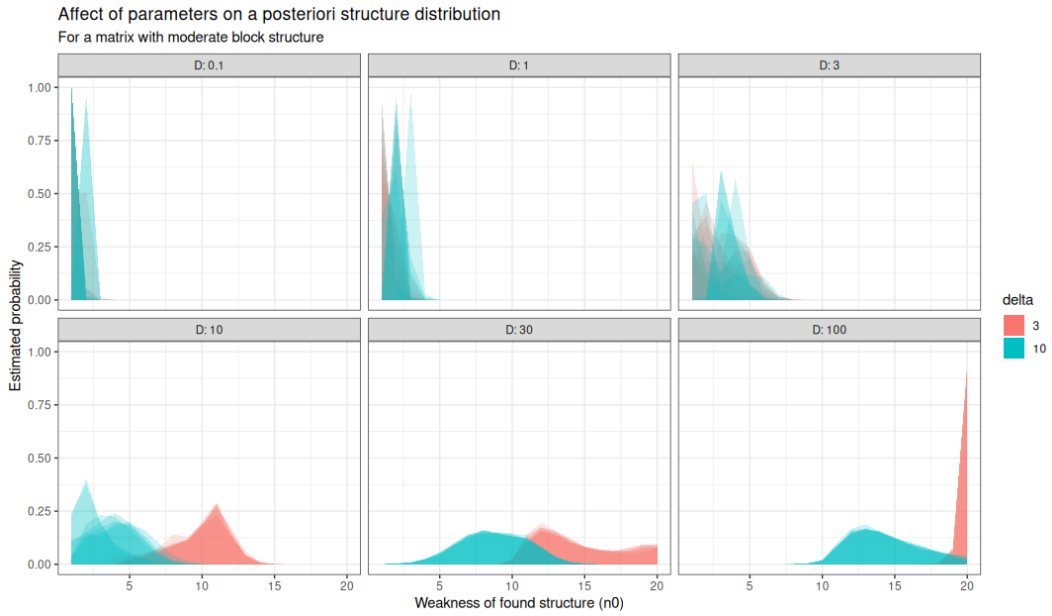

Figure 3: A posteriori distribution on structure strength estimated by MH algorihm. On the plot, the distributions found across 7 runs for each configuration are overlayed. Runs were done per 150 000 iterations each.

Clearly, matrices with higher values on diagonals and with higher δ were shifting the probability mass from stronger to weaker structures. At their lowest values, large structures were preferred; at their highest, almost whole probability mass resided in $\langle\mathrm{id}\rangle$ permutation.

A more spread out probability mass also meant less tendency to get stuck at local minimums, which meant larger acceptance rate and more consistent MH runs.

Now, to characterize more the a posteriori distribution on groups themselves. The patterns observed for matrix with moderate structure are confirmed for other matrices: extreme δ and D values mean lower entropy and thus high concentration of probability mass in few permutation groups; values in between have higher entropy and thus the mass is more spread out.

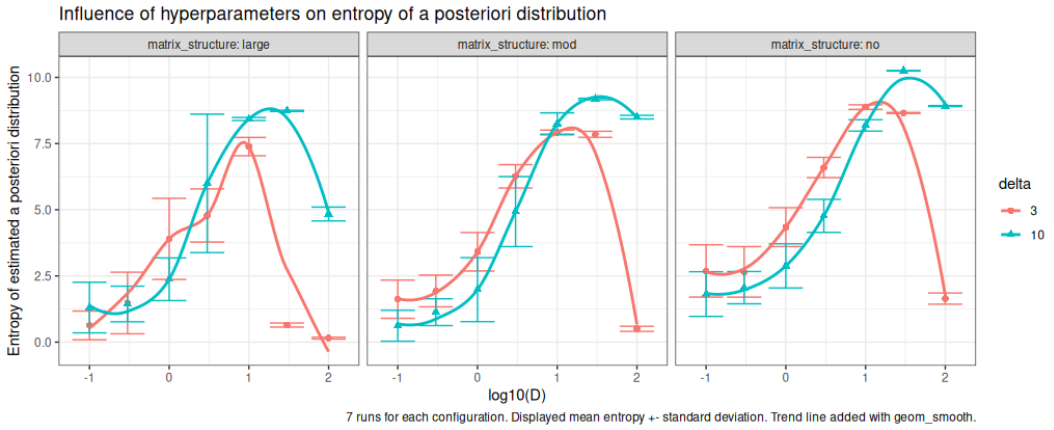

Figure 4: Entropies of a posteriori distributions on permutations on structure strength estimated by MH algorihm. On the plot preesnted are mean entropies with error bars (± 1 standard deviation) found across 7 runs for each configuration. Trend lines added with `ggplot2::geom_smooth` function. Runs were done per 150 000 iterations each.

3.4. Comparison with other algorithms

As mentioned in the introduction, there are other methods for estimating the covariance matrix in the high-dimensional setting. In this section we will compare method from **gips** package with methods implemented in **huge** and **rags2ridges** packages. They both are based on matrix penalization and have hyperparameter λ that controls the penalty's strength. They both have also hyperparameter search techniques implemented, which we shall use.

Methodology

We conducted the comparison across different sample sizes and across strengths of structure of true covariance matrices, as understood in previous section. We have used matrices invariant by permutation groups

large structure $\langle (1, 2, \ldots, 50) \rangle$,

moderate structure $\langle (1,2,3)(4,5,6)\ldots(16,17,18)(19,20)(21,22)\ldots(35,36) \rangle$,

no structure $\langle \mathrm{id} \rangle$.

For each permutation group two matrices were considered: one with substantial presence of 0 values in precision matrix, and the other without such presence. This modification was done to provide settings with LASSO method in mind; we expected it to perform better in the former than in the latter.

The comparison method is as follows:

1. Fix a sample size n and true covariance matrix Σ.
2. Generate a sample Z from $\mathrm{N}_p(0, \Sigma)$ of size n.
3. Estimate the covariance matrix using:

		sample size			
	matrix_info	5	10	20	40
1	large_nozeros	0.1	0.3	1	1
2	moderate_nozeros	0.4	1	1	1
3	none_nozeros	0.6	1	1	1
4	large_zeros	0.2	1	1	1
5	moderate_zeros	0.2	1	1	1
6	none_zeros	0	0	0.6	1

Table 1: Percentages of how often matrix produced by gips method was positive definite, depending on problem setting and sample size. For further analysis only configurations corresponding to cells with percentage > 0.5 and sample size > 5 were considered.

- from **gips** package: `find_MAP` method with default hyperparameters and `n_iter=150 000`,
- from **rags2ridges** package: `ridgeP` function with λ parameter found by `optPenalty.kCVauto` function with standard `lambdaMin=0.001, lambdaMax=100` range,
- from **huge**: function `huge` with `method="glasso"` and `lambda`...

4. Evaluate the estimation using the spectral distance to the true covariance matrix
5. Repeat 2)-4) and aggregate results

Experiment in each setting (fixed sample size and true covariance matrix) was repeated 10 times. `gips` method was used with default parameters and 300 000 iterations. Other methods were used with default hyperparameter selection techniques.

Results

First thing to note is that the gips method at times produced estimates, that were not positive definite (the found group had too weak structure). This is behaviour, which one should expect to occur, especially in extremely high-dimensional settings. More details are presented in Table 1. We notice, that for some settings the sample size was too low for our method; notable case is $n = 5$. For further analysis we discarded cases with $n = 5$ and other cases, where gips produced a valid estimate in less than 50% cases.

The results give certain hints to method's performance. First, we note the difference in discrepancy of calculated metrics: the (negative) loglikelihoods are pretty close to each other for a given setting and algorithm (with an exception for matrix `large_nozeros` with sample size 40). So the likelihood measures are informative. Further, for a given matrix and algorithm the weighted likelihoods did not vary much across sample sizes. Finally, generally gips operates comparably with RIDGE estimation, when the true matrix has some structure. In other case, it is often outperformed or simply does not produce a valid estimate. Both methods usually perform better than LASSO estimation, even in settings more preferable for it.

Now, to the spectral distance to ground truth (Figure 6). Here we observe more variability in a given setting and algorithm, for gips and RIDGE estimation. LASSO, in this measure, is strongly consistent (again, except for matrix `large_nozeros` with sample size 40). The variability decreases as sample size increases, which is expected - the more observations, the

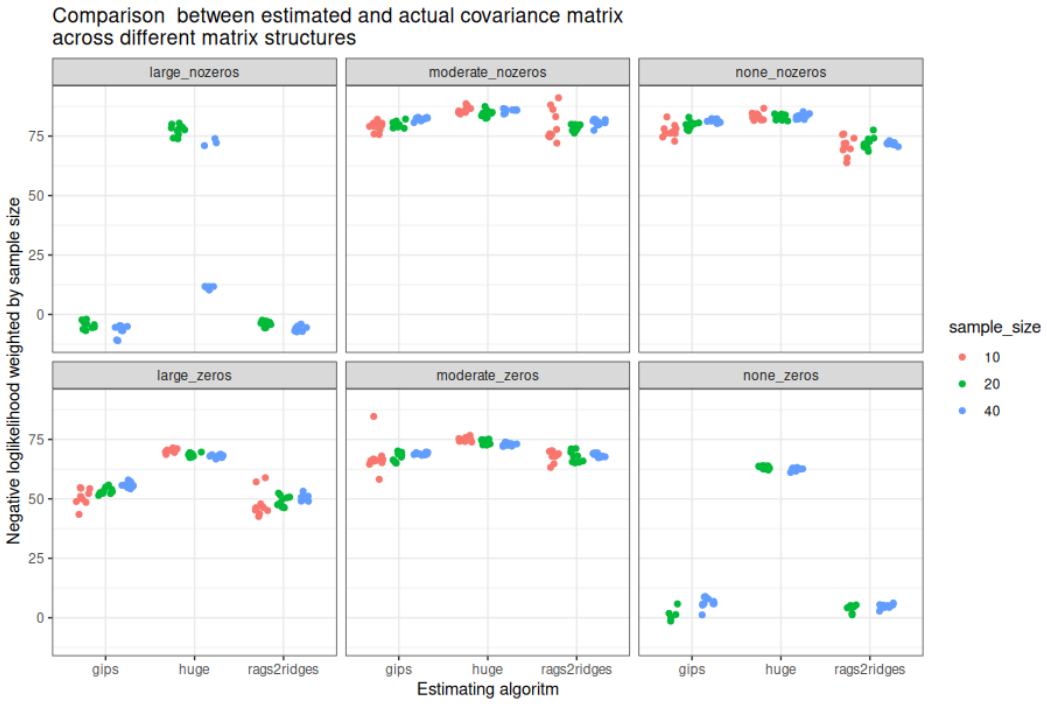

Figure 5: Negative loglikelihoods of covariance matrix estimations. Configurations for sample size 10 for matrices codelarge_nozeros and codenone_zeros are not presented, because gips often did not provide positive definite matrices; see Table 1. 10 runs for each configuration. gips MH runs were done with 300 000 iterations and default parameters.

more information about true covariance matrix and the better the estimation quality. Finally, for matrices with large structures gips method operates on similar or better level than other methods. However, for matrices with moderate or without structure, it is worse and more variable than others. This is not surprising.

In general, the results act in support of the theory: gips method is a viable choice if we suspect, that the true matrix has some structure. However, it is difficult to recognise it post-hoc by comparing the method's performance using loglikelihood (or possibly other measures) usable in real world cases, when the true matrix is unknown.

From a practical perspective, it worth repeating, that the gips method's output provides not only a projected covariance matrix, but also some interpretation of the structure of the data. On the other hand, MH used in gips method is much more expensive computationally than solvers for RIDGE and LASSO methods. For 300 000 iterations the difference was a matter of hours against seconds.

We acknowledge, that this method is not systematic enough to draw conclusions in full generality.

4. Summary and discussion

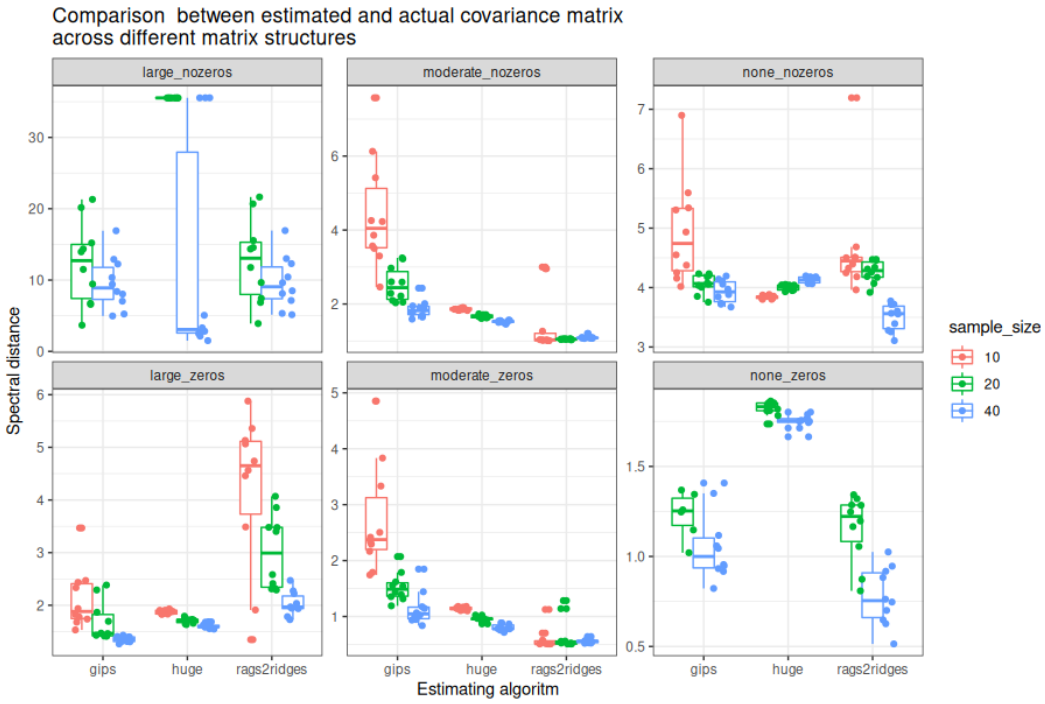

Figure 6: Spectral distances from covariance matrix estimations to ground truth. Note the difference in scales of y axis. Configurations for sample size 10 for matrices codelarge_nozeros and codenone_zeros are not presented, because gips often did not provide positive definite matrices; see Table 1. 10 runs for each configuration. gips MH runs were done with 300 000 iterations and default parameters.

Computational details

The results in this paper were obtained using R 3.4.1. R itself and all packages used are available from the Comprehensive R Archive Network (CRAN) at `https://CRAN.R-project.org/`.

Acknowledgments

Research was funded by (POB Cybersecurity and Data Science) of Warsaw University of Technology within the Excellence Initiative: Research University (IDUB) programme.

References

Andersson S (1975). "Invariant normal models." *The Annals of Statistics*, **3**, 132–154.

Andersson S, Madsen J (1998). "Symmetry and lattice conditional independence in a multivariate normal distribution." *The Annals of Statistics*, **26**(2), 525–572. `doi:10.1214/aos/1028144848`.

Diaconis P, Ylvisaker D (1979). "Conjugate priors for exponential families." *The Annals of Statistics*, **7**(2), 269–281.

Gao X, Massam H (2015). "Estimation of symmetry-constrained Gaussian graphical models: application to clustered dense networks." *Journal of Computational and Graphical Statistics*, **24**(4), 909–929. doi:10.1080/10618600.2014.937811.

Gehrmann H (2011). "Lattices of graphical Gaussian models with symmetries." *Symmetry*, **3**(3), 653–679.

Graczyk P, Ishi H, Kołodziejek B, Massam H (2022a). "Model selection in the space of Gaussian models invariant by symmetry." *The Annals of Statistics*, **50**(3), 1747–1774. doi:10.1214/22-aos2174.

Graczyk P, Ishi H, Kołodziejek B, Massam H (2022b). "Supplement to "Model selection in the space of Gaussian models invariant by symmetry"." *The Annals of Statistics*, **50**(3), 1747–1774. doi:10.1214/22-aos2174.

Hastie T, Tibshirani R, Friedman J (2009). *The elements of statistical learning*. Springer Series in Statistics, second edition. Springer, New York. ISBN 978-0-387-84857-0. doi:10.1007/978-0-387-84858-7. Data mining, inference, and prediction.

Højsgaard S, Lauritzen SL (2008). "Graphical Gaussian models with edge and vertex symmetries." *Journal of the Royal Statistical Society. Series B. Statistical Methodology*, **70**(5), 1005–1027. doi:10.1111/j.1467-9868.2008.00666.x.

Kotiang S, Eslami A (2020). "A probabilistic graphical model for system-wide analysis of gene regulatory networks." *Bioinformatics*, **36**(10), 3192–3199. doi:https://doi.org/10.1093/bioinformatics/btaa122.

Li Q, Gao X, Massam H (2020). "Bayesian model selection approach for coloured graphical Gaussian models." *Journal of Statistical Computation and Simulation*, **90**(14), 2631–2654. doi:10.1080/00949655.2020.1784175.

Li Q, Sun X, Wang N, Gao X (2021). "Penalized composite likelihood for colored graphical Gaussian models." *Statistical Analysis and Data Mining*, **14**(4), 366–378. doi:10.1002/sam.11530.

Massam H, Li Q, Gao X (2018). "Bayesian precision and covariance matrix estimation for graphical Gaussian models with edge and vertex symmetries." *Biometrika*, **105**(2), 371–388. doi:10.1093/biomet/asx084.

Ranciati S, Roverato A, Luati A (2021). "Fused graphical lasso for brain networks with symmetries." *Journal of the Royal Statistical Society. Series C. Applied Statistics*, **70**(5), 1299–1322. doi:10.1111/rssc.12514.

Toh H, Horimoto K (2002). "Inference of a genetic network by a combined approach of cluster analysis and graphical Gaussian modeling ." *Bioinformatics*, **18**(2), 287–297. doi:10.1093/bioinformatics/18.2.287.

A. More plots

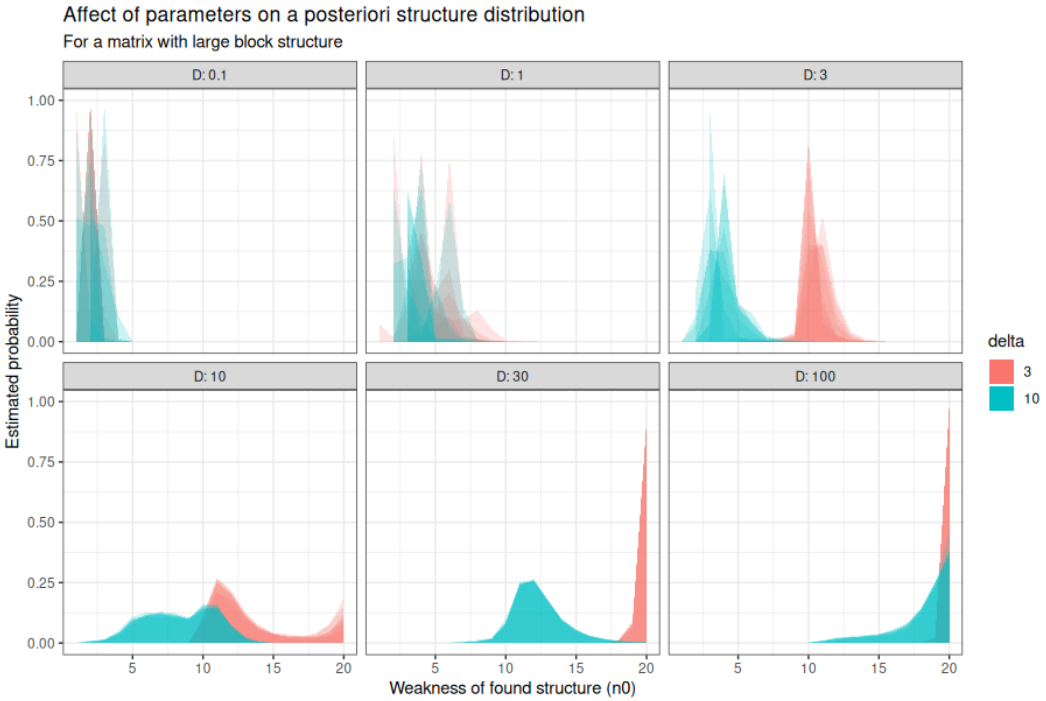

Figure 7: A posteriori distribution on structure strength estimated by MH algorihm. On the plot, the distributions found across 7 runs for each configuration are overlayed. Runs were done in 150 000 iterations.

Affiliation:

Adam Chojecki, Paweł Morgen, Bartosz Kołodziejek
Warsaw University of Technology
Faculty of Mathematics and Information Science
Koszykowa 75
00-662 Warsaw, Poland
E-mail: premysl.choj@gmail.com, seriousmorgen@protonmail.com,
b.kolodziejek@mini.pw.edu.pl

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
MMMMMM YYYY, Volume VV, Issue II *Submitted:* yyyy-mm-dd
doi:10.18637/jss.v000.i00 *Accepted:* yyyy-mm-dd

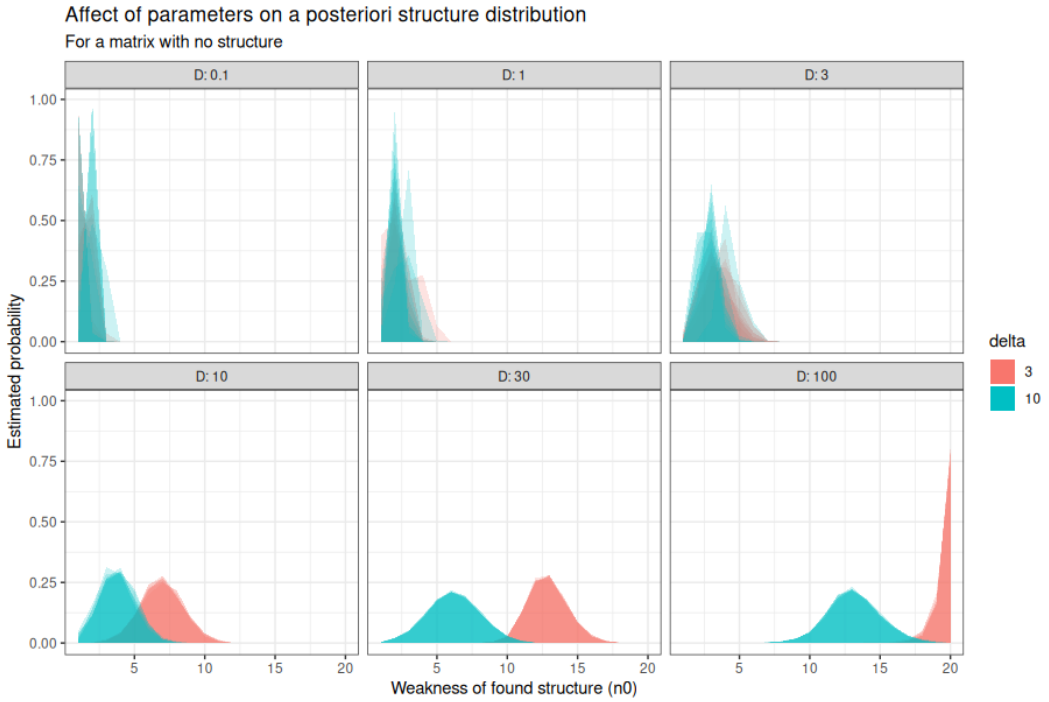

Figure 8: A posteriori distribution on structure strength estimated by MH algorihm. On the plot, the distributions found across 7 runs for each configuration are overlayed. Runs were done in 150 000 iterations.