SIMoNe: Statistical Inference for MOdular NEtworks.

Julien Chiquet, Alexander Smith, Gilles Grasseau, Catherine Matias, Christophe Ambroise

BIOINFORMATICS

Vol. 00 no. 00 2008
Pages 1–2

SIMoNe: Statistical Inference for MOdular NEtworks

Julien Chiquet*, Alexander Smith, Gilles Grasseau, Catherine Matias and Christophe Ambroise

UMR CNRS 8071 Statistique et Génome, 523, place des Terrasses, F-91000 Évry, FRANCE

Received on X; revised on X; accepted on X

ABSTRACT

Summary: The `R` package `SIMoNe` enables inference of gene-regulatory networks based on partial correlation coefficients from microarray experiments. Modelling gene expression data with a Gaussian Graphical Model (hereafter GGM), the algorithm estimates nonzero entries of the concentration matrix, in a sparse and possibly high-dimensional setting. Its originality lies in the fact that it searches for a latent modular structure to drive the inference procedure through adaptive penalization of the concentration matrix.

Availability: Under the GNU General Public Licence at `http://cran.r-project.org/web/packages/simone/`

Contact: `julien.chiquet@genopole.cnrs.fr`

1 INTRODUCTION

The uncovering of gene-regulatory networks, which play a central role in the cell's system, has received much attention recently. High throughput data from molecular biology offers a way of peering into this mechanism by inferring the dependencies between the expression levels of a large number of genes.

The `SIMoNe` (Statistical Inference for MOdular NEtworks) algorithm addresses this problem by combining the estimation of sparse undirected graphs with mixture model-based network clustering. Under the assumption that genes involved in a same cellular process are more likely inter-connected than genes involved in different processes, it is intuitive to search preferentially for links between genes belonging to the same group.

2 MODEL

Conditional dependency between the expression levels of two genes may reveal a regulation link between these genes. Thus, we infer a graph for which an edge is *not* present between two genes if the corresponding expression levels, considered as random variables, are independent conditional on the remaining variables. Such a graphical representation may be interpreted as a co-expression graph between genes, thus describing an influence network rather than the regulations themselves.

GGMs (Lauritzen, 1996) provide an appropriate and commonly-used framework, where the repeated microarray measurements are considered as i.i.d. occurrences of a Gaussian multivariate random vector. The conditional dependencies are encoded in the partial correlations between the variables, which may be computed using the inverse covariance matrix (hereafter concentration matrix). Detecting nonzero entries in this matrix is in fact equivalent to reconstructing the graph of conditional dependencies.

Ideally, the concentration matrix can be estimated by inverting the empirical covariance matrix. However, in the high-dimensional setting the latter is not invertible. Moreover, such a procedure does not lead to a sparse estimate, whereas biological evidence advocates for sparse networks. Sparsity means that the concentration matrix has a large number of zero entries. In this context, several estimation methods have been proposed based on ℓ_1 penalization (Meinshausen and Bühlmann, 2006; Friedman *et al.*, 2007; Banerjee *et al.*, 2008).

Our method can be seen as a mixture version of the *graphical lasso* (hereafter `GLasso`) by Friedman *et al.* (2007). Like `GLasso`, `SIMoNe` algorithm favors the network sparsity. Moreover, it may take into account a latent network structure in order to improve estimation accuracy (Ambroise *et al*., 2008). The latent structure relies on a mixture model for random graphs (Daudin *et al.*, 2006), which assumes that each node belongs to some unobserved group. Conditional on the node groups, the (weighted) edges are *i.i.d.* random variables, whose distribution depends on the groups of the nodes to be connected. This latent structure is further used to drive a penalization procedure towards the inference of a modular network. More precisely, let $\mathbf{Z}_i = (Z_{i1}, \ldots, Z_{iQ}) \sim \mathcal{M}(1, \boldsymbol{\alpha})$ be a random vector denoting which class node i belongs to. Here, $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_Q)$ is a vector of cluster proportions, so that $\sum_q \alpha_q = 1$. Conditional on the node clusters, the entries K_{ij} of the concentration matrix $\mathbf{K}$ are independent and follow a Laplace distribution, that is, $K_{ij} | \{Z_{iq} Z_{j\ell} = 1\} \sim f(\cdot)$ where

$$f(x) = \frac{1}{2\lambda_{q\ell}} \exp\left\{ -\frac{|x|}{\lambda_{q\ell}} \right\},$$

which mimics the role of an ℓ_1 penalization in a maximum likelihood framework. The data is assumed to be sampled from a multivariate Gaussian distribution with covariance matrix $\mathbf{K}^{-1}$.

3 ALGORITHM

The `SIMoNe` algorithm consists in a global EM-like strategy that alternates inference of the network latent structure and inference of the network's edges:

The E*-step: cluster inference.* Assuming that the concentration matrix $\mathbf{K}$ (i.e. the edges) is known, the parameters of the mixture model (cluster proportions $\boldsymbol{\alpha}$, cluster distribution parameters $\lambda_{q\ell}$ and node clustering $\mathbf{Z}_i$) are estimated by a variational algorithm initialized

*to whom correspondence should be addressed

with spectral clustering. The user must specify how many classes Q the algorithm should search for.

The M*-step: edge inference.* The edge inference is based on a modified version of the `GLasso` offering the possibility to apply different types of penalties to the concentration matrix entries. Each entry K_{ij} (or edge) is penalized according to the latent classes of the corresponding Gaussian variables (gene expression levels or nodes). In this way, when assuming for instance an affiliation structure, we penalize more heavily inter-class than intra-class edges.

Initialization of this EM-like algorithm relies on the `GLasso`. This first structure is used in the clustering E-step. Then, the node clustering information is used to build a new structured penalty matrix, allowing an updated estimation (M-step) of the regulation network topology. These two steps are repeated iteratively until convergence of the node clustering and network topology.

4 FEATURES AND EXAMPLES

The package is built around a core of necessary functions, interfacing `R` with `C/C++` sub-routines. The previously described two steps of the algorithm are implemented separately in the functions `InferEdges` and `InferClasses`. These two are combined in the function `simone`. The `InferEdges` function implements three alternative estimation procedures based on the ℓ_1 penalization: the two `Lasso` estimation strategies of Meinshausen and Bühlmann (2006), with either an AND or an OR rule; and the `GLasso` estimation strategy of Friedman *et al.* (2007).

Two different representations of the inferred network are available via the functions `Mplot` and `Gplot` (see Figure 1). The `Mplot` function plots the adjacency matrix of the network, whose columns may be rearranged according to an optional node classification vector (Figure 1 left). The `Gplot` proposes a more classical display of the graph (Figure 1 right).

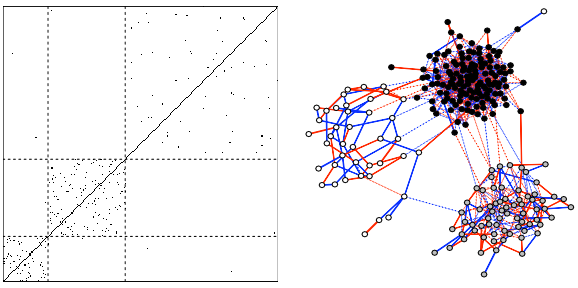

Figure 1. A simulated graph with modular structure: adjacency matrix with rows and columns reorganized according to the affiliation structure (using `Mplot`) and the corresponding graph (using `Gplot`).

Let us for instance investigate the gene expression data set provided by Hess et al. (2006), concerning 133 patients with stage $I - III$ breast cancer. The patients were treated with chemotherapy prior to surgery. Patient response to the treatment is classified as either a pathologic complete response (pCR) or a residual disease (not-pCR). We apply our algorithm on each class of patients: two distinct gene-regulatory networks are inferred from the execution of the following `R` code:

```
library(simone)
## load the data set
data(cancer)
## SIMoNe inference
pcr <- simone(cancer.pcr, Q=2, rho=5000)
not <- simone(cancer.notpcr,Q=2, rho=20000)
## Plot the results
par(mfrow=c(1,2))
Gplot(pcr$K.hat, pcr$cl, labels=gene.names, main="pCR")
Gplot(not$K.hat, not$cl, labels=gene.names, main="not pCR")
```

The inferred networks plotted on Figure 2 exhibit very different structures according to the class of patients. Gene regulation differs with respect to the presence or absence of a pCR and this stresses the fact that the whole dataset cannot be considered as i.i.d.

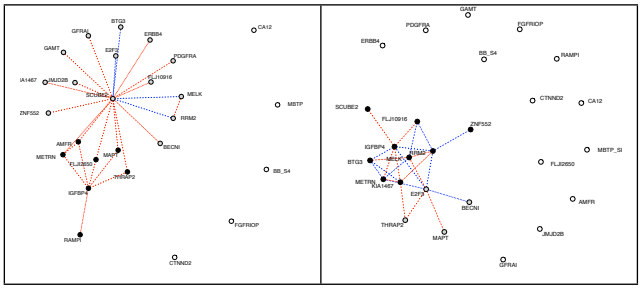

Figure 2. Inferred graphs for Hess dataset (not-pCR and pCR).

The computational complexity of `SIMoNe` is $\mathcal{O}(p^3)$, where p is the number of genes. Inferring a network with a thousand genes requires about a minute on a dual core computer for sparse networks, and about a hour for dense networks.

REFERENCES

Ambroise, C., Chiquet, J. and Matias, C. Inferring Gaussian Graphical Models with Latent Structure, *preprint available at* `http://arxiv.org/abs/0810.3177`.

Banerjee, O., El Ghaoui, L. and d'Aspremont, A. Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data, *Journal of Machine Learning Research*, **9**, 485–516, 2008.

Daudin, J.-J., Picard, F. and Robin, S. A mixture model for random graphs . *Statistics and Computing*, **18**(2), 173–183, 2008.

Friedman, J., Hastie, T. and Tibshirani, R. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, **9**(3), 432–441, 2007.

Hess, K.R., Anderson, K., Symmans, W.F., Valero, V., Ibrahim, N., Mejia, J.A., Booser, D., Theriault, R.L., Buzdar, U., Dempsey, P.J., Rouzier, R., Sneige, N., Ross, J.S., Vidaurre, T., Gómez, H.L., Hortobagyi, G.N. and Pustzai, L. Pharmacogenomic predictor of sensitivity to preoperative chemotherapy with paclitaxel and fluorouracil, doxorubicin, and cyclophosphamide in breast cancer. *Journal of Clinical Oncology*, **24**(26), 4236–4244, 2006.

Lauritzen, S.L. Graphical models, *Oxford University Press*, 1996.

Meinshausen, N. and Bühlmann, P. High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, **34**, 1436–1462, 2006.