Graphical Lasso and Its Applications

Xiaofei Zhou, Hengrui Luo, Nate Wilairat, Alex Wasielewski

Department of Statistics
The Ohio State University

April 27, 2018

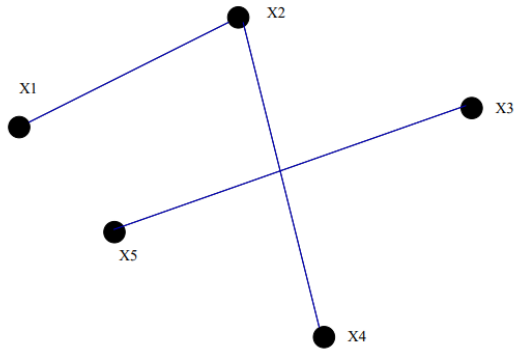
Overview

Graphical Models

- Used in many fields with applications in imaging, network analysis, Bayesian statistics, and gene analysis.
- Our focus will be on estimating the covariance matrix under a multivariate normal distribution assumption (for an undirected graph).
- As we will see, we can extend the lasso procedure in this setting allowing for more efficient computing and ways to deal with high-dimensional data which is becoming more common among the applications listed above.

What is a Graphical Model?

- Let $\boldsymbol{X} = (X_1, ..., X_p)^T \sim N_p(\mu, \Sigma)$, then we can represent each of the p random variables as a vertex in a graph. An example for $p = 5$ is shown below:



- The edges in this graph capture the conditional dependency relations between two random variables.
  - If there is no edge between two vertices, then the two corresponding random variables are conditionally independent given the rest of the random variables in the graph.

Initial Approaches to Estimating the Covariance Matrix

- All of these methods focus on estimating the precision matrix, $\Theta = \Sigma^{-1}$, the inverse covariance matrix.
  - The edges in an undirected graph can be regarded as the nonzero entries in the precision matrix. The zero entries indicate conditional independence between two variables.
- A naive approach is to maximize the likelihood function

$$\ell(\Theta) = logdet\Theta - Tr(S\Theta)$$

where S is the empirical covariance matrix of the data.

  - This results in the usual estimate, S, which is not very useful in our case since it does not consider the underlying dependence structure provided by the undirected graph.

Initial Approaches to Estimating the Covariance Matrix

Continued

- We can impose constraints on some of the parameters and then maximize the resulting log-likelihood, which after adding the Lagrange constants looks like

$$\ell_C(\Theta) = logdet\Theta - Tr(S\Theta) - \sum_{(j,k)\notin E} \gamma_{jk}\theta_{jk}$$

  where E is the set of edges in the graph.

- An iterative method can then be used to find $\hat{\Theta}$.

- These pre-defined constraints imply that some structure of the graph is known, which is not always the case. This led to authors proposing other means of penalization and in particular the use of L_1 regularization.

Use of the L_1 Norm

- Meinshausen and Buhlmann (2006) proposed a simpler approach to this precision matrix estimation problem.
- They only estimated the θ_{ij} that were nonzero by fitting a lasso regression on each random variable using the rest as the predictors.
- They then used the rule: if either the estimated coefficient of i on j is nonzero, or the estimated coefficient of variable j on i is nonzero, then θ_{ij} is estimated to be nonzero.
- Asymptotically, for fixed p, this procedure consistently estimates the nonzero elements of Θ [5].

The Graphical Lasso

- To complement the neighborhood detection procedure of Meinshausen and Buhlmann, Friedman et al. (2008) proposed modifying the likelihood directly.
- We then have the usual penalized likelihood that is associated with the graphical lasso:

$$\ell_\lambda(\Theta) = logdet\Theta - Tr(S\Theta) - \lambda||\Theta||_1$$

- In this case the L_1 norm of Θ is defined as the sum of the absolute values of the elements of the precision matrix.
- The algorithm used to solve this optimization problem (described on the next slide) is very fast, being able to solve a sparse problem with 1000 vertices in less than a minute [2].

The Graphical LASSO Algorithm

Friedman et al. 2001

- Initialize $W = S + \lambda I$ where S is the empirical covariance matrix of the data. The diagonal of W remains unchanged in what follows because the w_{22} is never involved in the derivation above.
- Repeat for $j = 1, 2, ...p$ until certain convergence criterion is met.
  - Partition the matrix W into
    - W_{11} = all but the jth row and column are zeros.
    - W_{-11} = only the jth row and column are nonzero.
  - Solve the estimating equations $W_{11}\beta_{12} - s_{12} + \lambda \cdot sgn(\beta_{12}) = 0$ above using the *cyclical coordinate-descent algorithm* [7] for β_{12} and obtain $\hat{\beta}$. Update $w_{12} = W_{11}\hat{\beta}$ and then obtain $\hat{\theta}_{12}$ from the regression function.
- In the final cycle (for each j) solve for $\hat{\theta}_{12} = -\hat{\beta} \cdot \hat{\theta}_{22}$ with $\frac{1}{\hat{\theta}_{22}} = w_{22} - w_{12}^T\hat{\beta}$.

Selection of λ

Cross-Validation

Our simulation study focused on selection of λ using cross-validation, BIC, and extended BIC for three different scenarios with fixed $p = 20$ and from $n = 15$ to $n = 1250$. The original paper recommends cross-validation. A common rule of thumb to select a range of feasible λ is $(0, \max(|S^{-1}|_{ij}))$.

A k-fold cross validation procedure would proceed as follows:

- Partition the dataset into k subsets. Let A_s represent the s^{th} subset and A_s^C be the complement of the s^{th} subset.
- For $s = 1, ..., k$, train the Graphical LASSO model on A_s^C .
- Evaluate the log-likelihood using the likelihood based on A_s^C, $\hat{\Theta}_{-s}$, and the empirical covariance from the test subset S_{A_s} ,
- Set the estimated cross-validated likelihood $\hat{\alpha}_k(\lambda) = \frac{1}{k} \sum_{s=1}^{k} \ell(\hat{\Sigma}_{-s}, S_{A_s})$.
- Select the value of λ that maximizes the log-likelihood.

Selection of λ

BIC

An alternative approach is to use Bayes Information Criterion (BIC), to select an optimal λ. This has been shown to lead to "consistent graphical model selection" (Gao et al. 2012). The effective degrees of freedom parameter is the sum of non-zero off-diagonal entries in the upper triangular part of the estimated precision matrix $\hat{\Theta}$.

$$BIC_\lambda = -n\log|\hat{\Theta}| + nTr(S\hat{\Theta}) + \log(n)\sum_{1\le i<j\le p} I(\theta_{ij} \neq 0)$$

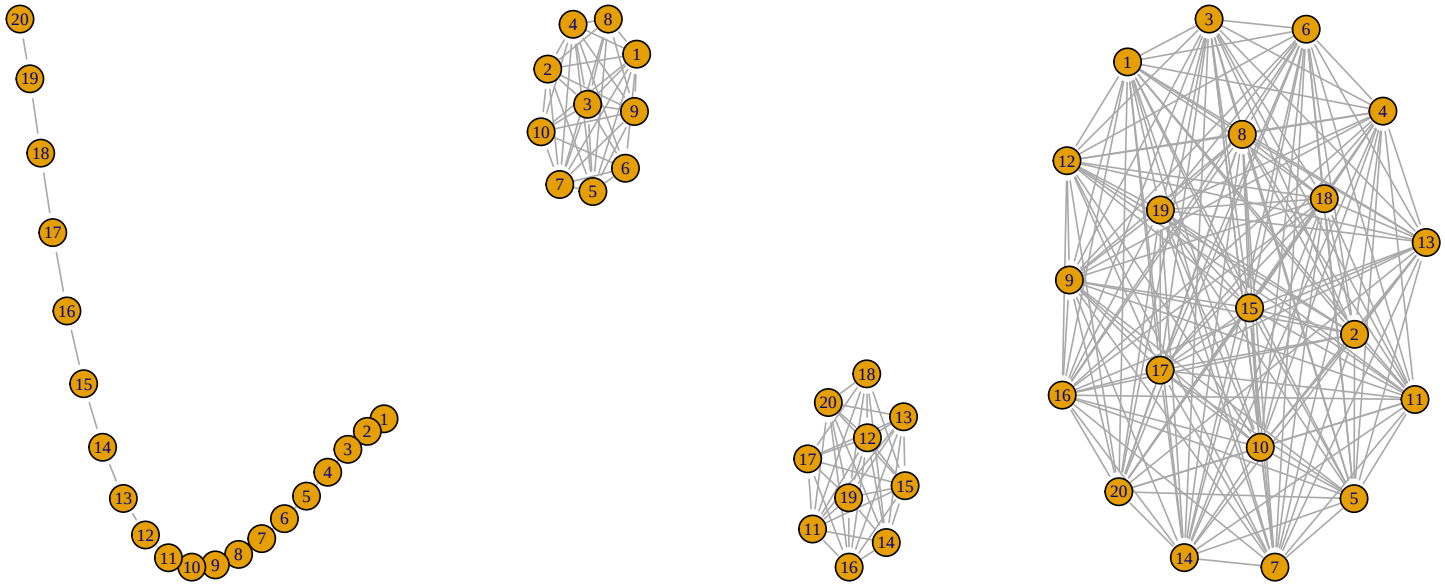
The extended BIC adds an additional term and is consistent when $p \to \infty$ and $n \to \infty$. Note that $\gamma \in [0,1]$ where $\gamma = 1$ represents maximum penalization of large graphs. Simulation studies suggest extended BIC performs well when p is close to n (Foygel & Drton 2010) .

$$BIC_{\gamma,\lambda} = BIC_\lambda + 4\gamma\log(p)\sum_{1\le i<j\le p} I(\theta_{ij} \neq 0)$$

Simulation Results

Models

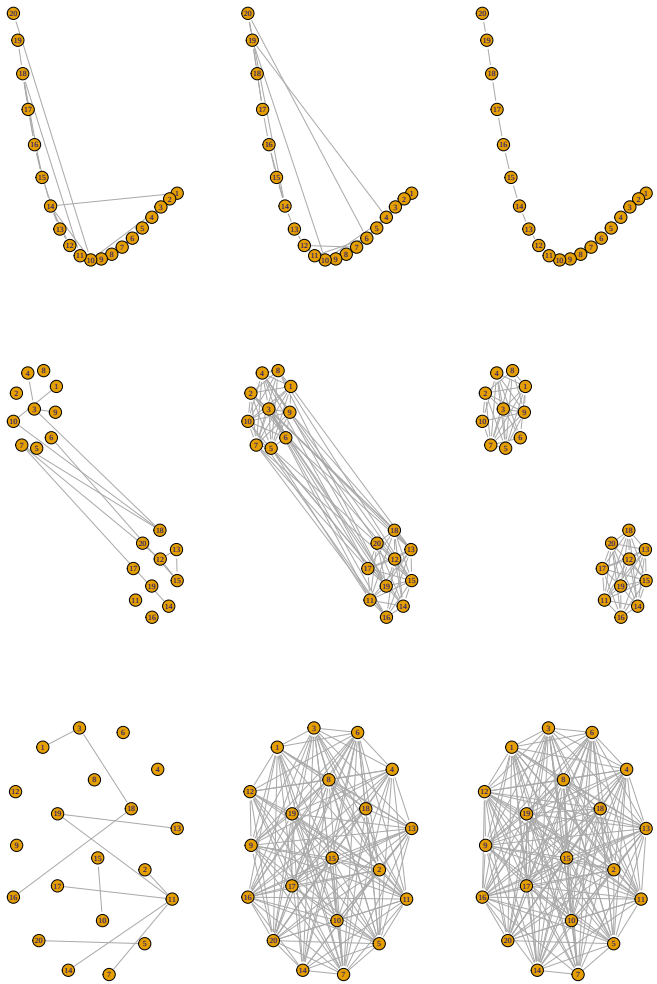
We simulated models using a sparse (AR1), grouped, and dense precision matrix.



Simulation Results

Graphs

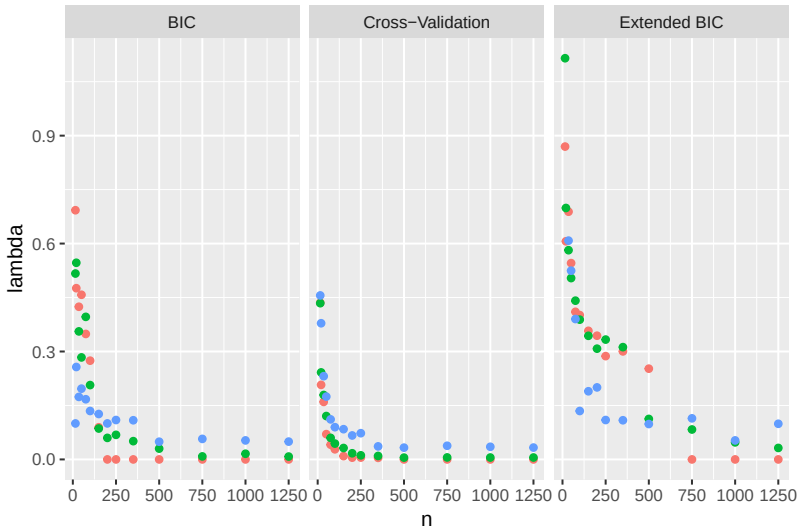
For $n = 15$, $n = 150$, and $n = 1250$ under cross-validation*.

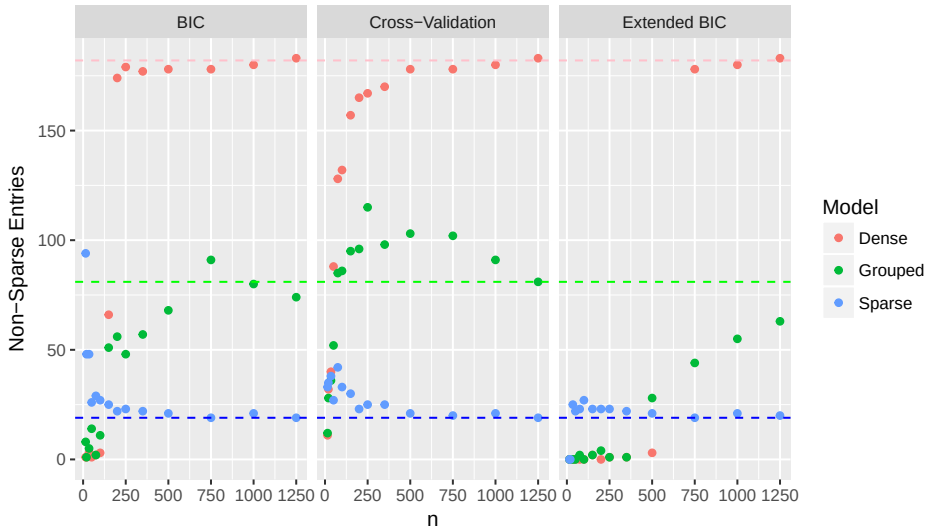


Simulation Results

Likelihood and Sparsity

As $n \to \infty$, the likelihoods using $\hat{\Theta}(\lambda_{CV})$ converge to the true likelihoods as expected. We also have $\lambda \to 0$ as $n \to \infty$ for the dense and grouped models. When $\lambda = 0$, then $\hat{\Theta} = S^{-1}$, and we must round $\hat{\Theta}$ to induce sparsity (and for λ close to 0). The BIC criteria perform better for the sparse model but poorly on the dense model (especially extended BIC).





Graphical Lasso on Malaria Infection Analysis

Malaria Background

- Malaria is a mosquito-borne disease caused by parasites which affects humans and other animals.
- Patients with malaria often experience flu-like sickness, but the effects can go to long-term or fatal.
- In 2016, about 216 million people were infected with malaria worldwide and 445,000 died because of it. In the US, 1,500-2,000 cases of malaria are diagnosed each year.

Graphical Lasso on Malaria Infection Analysis

Malaria Data

- Gene Expression Omnibus (GEO) is a public database that provides gene expression data with easy user access.
- Among the 4,349 datasets available in this database, we found gene expression data in malaria infection (GSE 5418) that provides pairwise transcription readings (experimental malaria-infected versus baseline uninfected) for 22 human subjects on 22,283 different genes and genomic segments.
- The first few steps of data reading and raw processing are implemented with libraries from the Bioconductor website.

Graphical Lasso on Malaria Infection Analysis

Differentially Expressed Genes

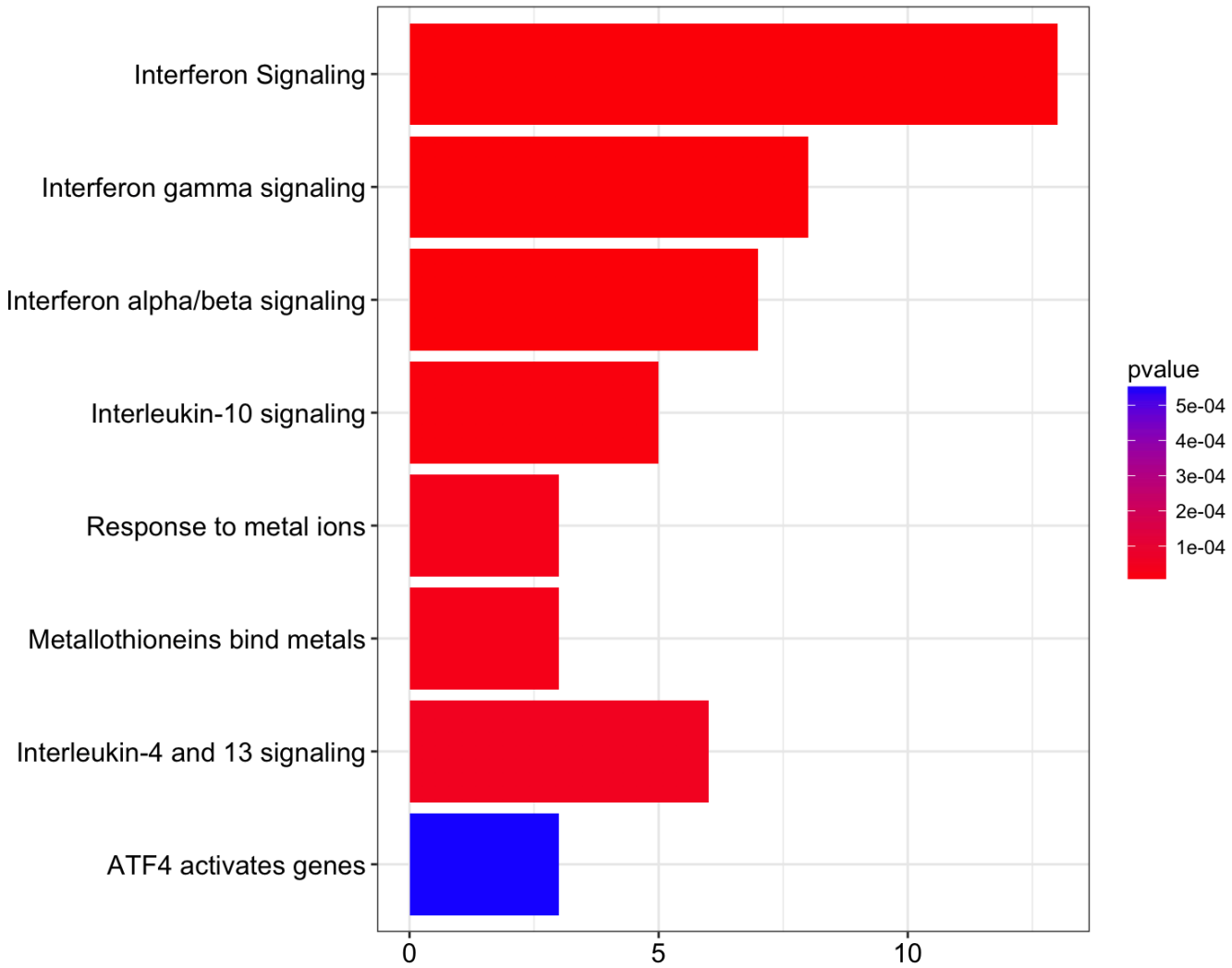
To identify the differentially expressed genes, we utilize the DIME software that considers log infection/normal ratio as an ensemble of finite mixture models.

Table: Top 10 DE Genes from DIME

Gene	FDR	Class	Log Ratio	Log Intensity
CCL2	1.5E-45	1	2.737	5.258
GBP1	1.8E-37	1	2.479	7.148
GBP1	7.6E-35	1	2.390	7.908
FFAR2	4.5E-34	1	2.364	5.078
SERPING1	1.5E-31	1	2.273	6.176
FCGR1B	9.7E-30	1	2.207	7.266
IFI44L	1.1E-28	1	2.167	6.472
CXCL10	3.5E-28	1	2.148	7.049
WARS	2.4E-26	1	2.076	8.743
STX11	6.2E-26	1	2.060	5.707

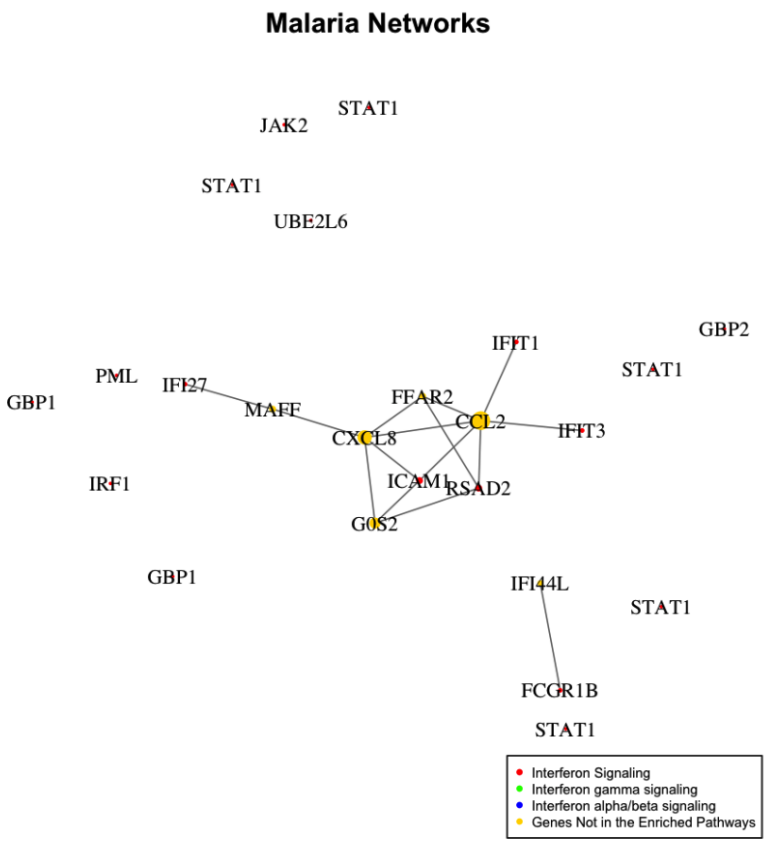
Graphical Lasso on Malaria Infection Analysis

Enriched Gene Pathways Based on Top DE Genes

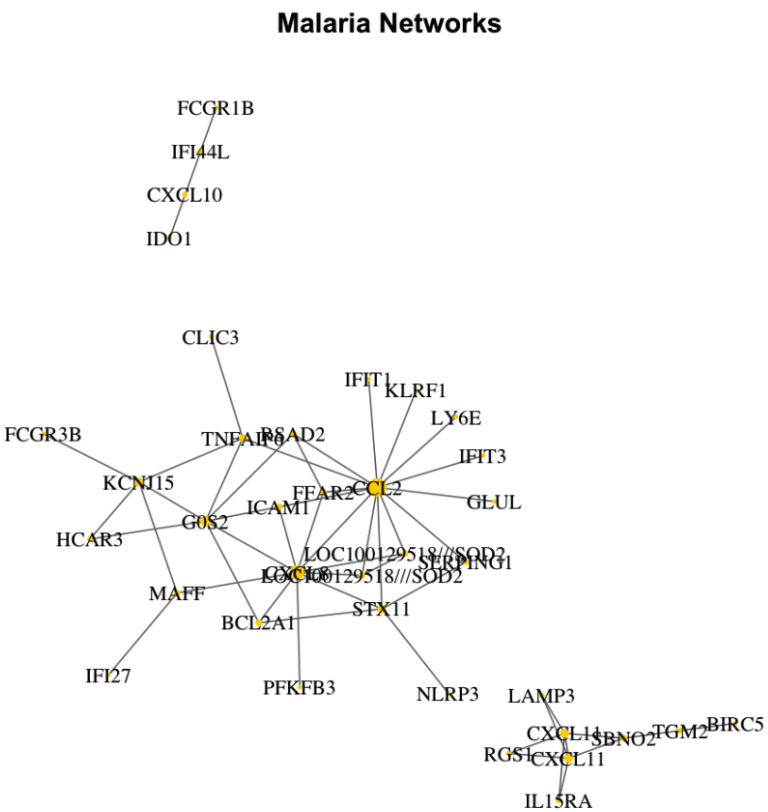


Graphical Lasso on Malaria Infection Analysis

Network Plots based on gLasso



(A) Interferon Signaling Estimate



(B) No Pathway (Top 100) Estimate

References I

[1] Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. "Sparse inverse covariance estimation with the graphical lasso." Biostatistics 9.3 (2008): 432-441.

[2] Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. The elements of statistical learning. Vol. 1. New York: Springer series in statistics, 2001.

[3] Foygel, R., & Drton, M. "Extended Bayesian information criteria for Gaussian graphical models." In Advances in Neural Information Processing Systems 23: 24th Annual Conference on Neural Information Processing Systems 2010, NIPS 2010.

[4] Gao, Xin, Daniel Pu, Yuehua Wu, and Hong Xu. "Tuning parameter selection for penalized likelihood estimation of Gaussian graphical model." Statistica Sinica 22 (2012): 1123-1146.

[5] Meinshausen, Nicolai, and Peter . "High-dimensional graphs and variable selection with the lasso." The annals of statistics (2006): 1436-1462.

References II

[6] Mazumder, Rahul, and Trevor Hastie. "The graphical lasso: New insights and alternatives." Electronic journal of statistics 6 (2012): 2125.

[7] Wright, Stephen J. "Coordinate descent algorithms." Mathematical Programming 151.1 (2015): 3-34.