Robust penalized inference for Gaussian Scale Mixtures

Karina Ashurbekova, Sophie Achard, Florence Forbes

Robust penalized inference for Gaussian Scale Mixtures

Karina Ashurbekova
GIPSA-lab, Inria
Univ. Grenoble Alpes
Grenoble, France
karina.ashurbekova@gipsa-lab.fr

Sophie Achard
CNRS, Grenoble INP, GIPSA-lab
Univ. Grenoble Alpes
Grenoble, France
sophie.achard@gipsa-lab.fr

Florence Forbes
Inria, CNRS, Grenoble INP, LJK
Univ. Grenoble Alpes
Grenoble, France
florence.forbes@inria.fr

I. Abstract

A. Brief introduction

The literature on sparse precision matrix estimation is rapidly growing and received significant attention from the research community [4], [3]. Many strong methods are valid only for Gaussian variables [8], [7]. One of the most commonly used approaches in this case is *glasso* [6] which aims to minimize the negative L1-penalized log-likelihood function:

$$\min_{\boldsymbol{\Theta}\succ 0} tr(\boldsymbol{\Theta}\boldsymbol{S}) - \log\ det(\boldsymbol{\Theta}) + \rho\|\boldsymbol{\Theta}\|_1 \tag{1}$$

$\boldsymbol{S} = \frac{1}{n}\sum_{i=1}^n(\boldsymbol{x}_i - \boldsymbol{\mu})(\boldsymbol{x}_i - \boldsymbol{\mu})^T$ is the sample covariance matrix of observed independent samples $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n$ from $N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, $\boldsymbol{\Theta} = \boldsymbol{\Sigma}^{-1}$ is the precision matrix, ρ is a regularization parameter.

In practice, data may deviate from normality in various ways, outliers and heavy tails frequently occur that can severely degrade the Gaussian models performance. A natural solution is to turn to heavier tailed distributions that remain tractable. For this purpose, we propose a penalized version of EM-algorithm for Gaussian Scale Mixtures. The proposition we state below allows us to design the penalized EM algorithm valid for the distributions that can be seen as Gaussian Scale Mixtures.

Definition 1 (Scale mixture of multivariate Gaussian distributions)**.** *If $\boldsymbol{\mu}$ is a p-dimensional vector, $\boldsymbol{\Sigma}$ is a $p\times p$ positive definite symmetric matrix and f_W is a probability distribution function of a univariate positive variable $W \in \mathbb{R}^+$, then the p-dimensional density given by*

$$p(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\theta}) = \int_0^\infty \mathcal{N}_p\left(\boldsymbol{x}; \boldsymbol{\mu}, \frac{\boldsymbol{\Sigma}}{w}\right)\ f_W(w; \boldsymbol{\theta})\ dw \tag{2}$$

is said to be a scale mixture of Gaussian densities with mixing distribution function f_W. If vector $\boldsymbol{X}$ has density (2), we write $\boldsymbol{X} \sim \mathcal{GSM}_p(\boldsymbol{\mu}, \boldsymbol{\Sigma}, f_W)$ and we refer to W as the mixing variable.

Proposition 1. *Let $\boldsymbol{X} \sim \mathcal{GSM}_p(\boldsymbol{\mu}, \boldsymbol{\Sigma}, f_W)$ and W denote the mixing variable. Then $\boldsymbol{X}$ has an elliptical distribution*

$$p(\boldsymbol{x} : \boldsymbol{\mu}, \boldsymbol{\Sigma}, f_W) = (2\pi)^{\frac{p}{2}}|\boldsymbol{\Sigma}|^{-\frac{1}{2}} g((\boldsymbol{x} - \boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x} - \boldsymbol{\mu}))$$

with generator $g(t) = g(t) = \int_0^\infty w^{\frac{p}{2}}\exp(-\frac{1}{2}tw)f_W(w)dw$. It follows that

$$E[W|\boldsymbol{x}, \boldsymbol{\Psi}] = -2\frac{g'((\boldsymbol{x} - \boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x} - \boldsymbol{\mu}))}{g((\boldsymbol{x} - \boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x} - \boldsymbol{\mu}))},$$

where g' is the derivative of g.

B. Penalized EM algorithm for Gaussian Scale Mixtures

Like in the Gaussian case, we put a L1-norm penalty on the elements of the matrix $\boldsymbol{\Theta}$ and wish to maximize the penalized log-likelihood function. Let us consider n *i.i.d.* variables $\boldsymbol{x}_i$ following a $\mathcal{GSM}_p(\boldsymbol{\mu}, \boldsymbol{\Sigma}, f_W)$ distribution for $i = 1 : n$, the model parameters to estimate are $\boldsymbol{\Psi} = \{\boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\theta}\}$. Introducing the corresponding latent variables $\{W_i, i = 1 : n\}$, we can consider an EM algorithm. The expected complete likelihood $Q(\boldsymbol{\Psi}, \boldsymbol{\Psi}^{(r-1)})$ at iteration (r) of the algorithm takes the form:

$$Q(\boldsymbol{\Psi}, \boldsymbol{\Psi}^{(r-1)}) = \sum_{i=1}^n E\left[\log p(\boldsymbol{x}_i|W_i; \boldsymbol{\mu}, \boldsymbol{\Sigma})|\boldsymbol{x}_i, \boldsymbol{\Psi}^{(r-1)}\right] + \\ + E\left[\log f_W(W_i)|\boldsymbol{x}_i, \boldsymbol{\Psi}^{(r-1)}\right] - \rho\|\boldsymbol{\Sigma}^{-1}\|_1 \tag{3}$$

By definition the distribution of $(\boldsymbol{x}_i|W_i = w_i)$ is $\mathcal{N}(\ .\ ; \boldsymbol{\mu}, \frac{\boldsymbol{\Sigma}}{w_i})$ and does not depend on $\boldsymbol{\theta}$.

The **E-step** is the same as for non-penalized version. For $i = 1 : n$, compute $E[W_i|\boldsymbol{x}_i, \boldsymbol{\Psi}^{(r-1)}]$ according to Proposition 1 we get: $E[W_i|\boldsymbol{x}_i, \boldsymbol{\Psi}^{(r-1)}] = \bar{w}_i^{(r)} = -2\frac{g'(\delta(\boldsymbol{x}, \boldsymbol{\mu}^{(r-1)}, \boldsymbol{\Sigma}^{(r-1)}))}{g(\delta(\boldsymbol{x}, \boldsymbol{\mu}^{(r-1)}, \boldsymbol{\Sigma}^{(r-1)}))}$, where $\delta(\boldsymbol{x}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = (\boldsymbol{x} - \boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x} - \boldsymbol{\mu})$ is the Mahalanobis distance between $\boldsymbol{x}$ and $\boldsymbol{\mu}$.

For the updating of $\boldsymbol{\Psi}$, the **M-step** consists of two independent steps for $\{\boldsymbol{\mu}, \boldsymbol{\Sigma}\}$ and $\boldsymbol{\theta}$ respectively. The update for the vector $\boldsymbol{\mu}$ as well as for parameter $\boldsymbol{\theta}$ has the same form as for non-penalized version of EM. In contrast with the non-penalized version, the update value $\boldsymbol{\Theta}^{(r)}$ is found by solving the following optimization problem:

$$\min_{\boldsymbol{\Theta}\succ 0}\ tr(\boldsymbol{\Theta}\boldsymbol{S}^{(r)}) - \log det(\boldsymbol{\Theta}) + \rho\|\boldsymbol{\Theta}\|_1, \tag{4}$$

where a "weighted sample covariance matrix" $\boldsymbol{S}^{(r)}$ and vector $\boldsymbol{\mu}$ are computed as $\boldsymbol{S}^{(r)} = \frac{1}{n}\sum_{i=1}^n \bar{w}_i^{(r)}(\boldsymbol{x}_i - \boldsymbol{\mu}^{(r)})(\boldsymbol{x}_i - \boldsymbol{\mu}^{(r)})^T$ and $\boldsymbol{\mu}^{(r)} = \sum_{i=1}^n(\bar{w}_i^{(r)}\boldsymbol{x}_i)/\sum_{i=1}^n \bar{w}_i^{(r)}$. Note that (4) is a similar objective function minimized by *glasso*, so that the proposed penalized algorithm reduces to solving at each iteration a *glasso* optimization problem. There is no need to inverse the matrix $\boldsymbol{\Sigma}$ on each step since for all $i = 1 : n$ the variables $\bar{w}_i$ depend on $\boldsymbol{\Theta}^{(r-1)}$ and not on $\boldsymbol{\Sigma}^{(r-1)}$. Moreover, it is shown in [1] that the optimization problem 4 can be replaced by more efficient CLIME algorithm [2].

C. Results

As a particular example of the described approach, Finegold and Drton [5] proposed a *tlasso* procedure for multivariate t-distribution. We compare the results of *tlasso*, tCLIME [1] which a modified version of *tlasso*, CLIME [3] and EPIC method [9] designed for elliptical distributions. To measure how well the sparsity of the true precision matrix is recovered, we plot ROC curves presented in Figure 1. For Gaussian data, the tCLIME and EPIC performance is similar and better than that of *tlasso* and CLIME. When data is generated from a t-distribution, tCLIME significantly outperforms CLIME and also shows better results than EPIC and *tlasso*.

***Index Terms*—Structure learning, Gaussian Scale Mixtures, Gaussian graphical model, t-distribution, sparse precision matrix estimation, robust estimation**

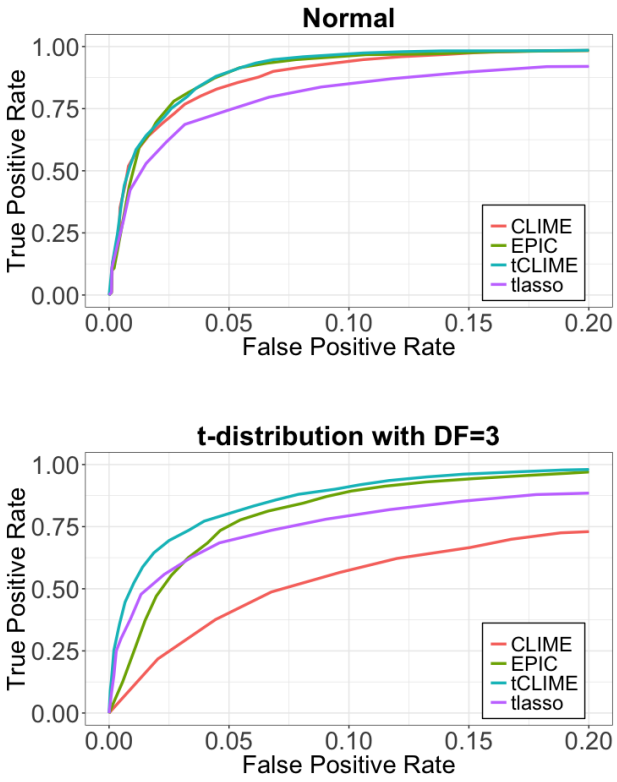

Fig. 1: ROC curves illustrating the performance of *tlasso*, CLIME, tCLIME and EPIC methods on 2 different data sets. A random 100×100 sparse precision matrix $\mathbf{\Theta}$ is generated according to the procedure described in [5]; n=150 observations are simulated from $t_{100}(0, \mathbf{\Theta}^{-1}, 3)$ and $N_{100}(0, \mathbf{\Theta}^{-1})$. The four algorithms are then run with different values of ρ and the whole process is repeated 50 times. The tuning parameter ρ is chosen in the range $[0.1, 2.5]$ with stepsize 0.05 for *tlasso* and $[0.01, 0.4]$ with stepsize 0.01 for CLIME, tCLIME anf EPIC.

REFERENCES

[1] K. Ashurbekova, S. Achard, and F. Forbes. Robust structure learning using multivariate t-distributions. In *50e Journées de la Statistique de la SFdS*, 2018.

[2] T. Cai, W. Liu, and X. Luo. A constrained l-1 minimization approach to sparse precision matrix estimation. *J. Am. Stat. Assoc.*, 106(494):594–607, 2011.

[3] T.T. Cai, Z. Ren, and H.H. Zhou. Estimating structured high-dimensional covariance and precision matrices: Optimal rates and adaptive estimation. *Electron. J. Stat.*, 10(1):1–59, 2016.

[4] J. Fan, Y. Liao, and H. Liu. An overview of the estimation of large covariance and precision matrices. *Econom. J.*, 19(1), 2016.

[5] M. Finegold and M. Drton. Robust graphical modeling of gene networks using classical and alternative t-distributions. *Ann. Appl. Stat.*, pages 1057–1080, 2011.

[6] J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008.

[7] L. Han and W. Lie. Tiger: A tuning-insensitive approach for optimally estimating gaussian graphical models. *arXiv preprint arXiv:1209.2437*, 2012.

[8] T. Sun and C.H. Zhang. Sparse matrix inversion with scaled lasso. *J. Mach. Learn. Res.*, 14(1):3385–3418, 2013.

[9] T. Zhao and H. Liu. Calibrated precision matrix estimation for high-dimensional elliptical distributions. *IEEE Trans. Inf. Theory*, 60(12):7874–7887, 2014.