Latent Gaussian Graphical Models with Golazo Penalty

Ignacio Echave-Sustaeta Rodríguez I.ECHAVE.SUSTAETA.RODRIGUEZ@TUE.NL
Frank Röttger F.ROTTGER@TUE.NL
Eindhoven University of Technology, Eindhoven, The Netherlands

Editors: J.H.P. Kwisthout & S. Renooij

Abstract

The existence of latent variables in practical problems is common, for example when some variables are difficult or expensive to measure, or simply unknown. When latent variables are unaccounted for, structure learning for Gaussian graphical models can be blurred by additional correlation between the observed variables that is incurred by the latent variables. A standard approach for this problem is a latent version of the graphical lasso that splits the inverse covariance matrix into a sparse and a low-rank part that are penalized separately. In this paper we propose a generalization of this via the flexible Golazo penalty. This allows us to introduce latent versions of for example the adaptive lasso, positive dependence constraints or predetermined sparsity patterns, and combinations of those. We develop an algorithm for the latent Gaussian graphical model with the Golazo penalty and demonstrate it on simulated and real data.

Keywords: graphical models; latent variables; sparse estimators; Golazo penalty.

1. Introduction

In many inference problems it is common to implicitly assume that all variables of interest are being observed and measured. This is however often not the case, for various reasons. For example, it is possible that there exist unknown factors that influence the observed variables. Alternatively, there may be variables which are too expensive or difficult to measure. When our interest is in structure learning for Gaussian graphical models, in particular in high-dimensional settings, a common approach is covariance estimation with the graphical lasso (Meinshausen and Bühlmann, 2006; Yuan and Lin, 2007) which can recover the zero pattern of the inverse covariance matrix $K = \Sigma^{-1}$. In the presence of latent variables however, a potentially sparse structure might be inaccessible. Let O denote the indices of the observed and H the indices of the latent (or hidden) variables of some Gaussian random vector $\mathbf{X}$. The inverse covariance matrix of $\mathbf{X}_O$ is the Schur complement

$$(\Sigma_{OO})^{-1} = K_{OO} - K_{OH}(K_{HH})^{-1}K_{HO}.$$

Here, even when the complete model is sparse, the subtrahend can blur the sparsity pattern in K_{OO}. In other words, the latent variables incur correlations in the observed system which can render attempts to estimate directly the dependence structure of the system unsuccessful.

For this setting Chandrasekaran et al. (2012) proposed to model the inverse observed covariance matrix as the difference of a sparse matrix $A = K_{OO}$ and a low-rank matrix $B =$

$K_{OH}(K_{HH})^{-1}K_{HO}$. They penalize sparsity in A (ℓ_1 norm) and rank in B (the nuclear norm is the trace for symmetric PSD matrices), resulting in the following optimization problem

$$(\widehat{A}, \widehat{B}) = \underset{A,B}{\operatorname{argmin}} - \ell(A - B; S_{OO}) + \lambda(\gamma\|A\|_1 + \operatorname{tr}(B)),$$

where ℓ is the Gaussian log-likelihood, S_{OO} is the observed sample covariance, A is required to be positive definite and B to be positive semidefinite, and λ and γ are non-negative scalars.

In structure learning for multivariate Gaussians some alternatives to the ℓ_1-penalty as in the graphical lasso have been proposed in the literature, for example the adaptive lasso (Fan et al., 2009) or positive dependence (Lauritzen et al., 2019). Recently, Lauritzen and Zwiernik (2022) introduced the Golazo penalty as a flexible generalization of many penalties. The Golazo penalty includes not only the adaptive lasso and positive dependence, but also allows for graphical model constraints or asymmetric penalties, and combinations of those.

In this paper we propose to modify the approach of Chandrasekaran et al. (2012) using the Golazo penalty to allow more flexible structure learning in latent Gaussian graphical models. This yields a convex optimization problem, which we tackle with an alternating direction method of multipliers (ADMM) algorithm (Chang et al., 2020). We demonstrate the application of our method on simulated and real data, obtained from Chang et al. (2020) but with the original source being Hughes et al. (2000). The code for this paper is publicly available on Github at https://github.com/iechave-tue/golazo-latent-gaussian.

1.1. Notation

Let $\mathcal{S}^d_>$ be the collection of all symmetric positive definite $d \times d$-matrices and $\mathcal{S}^d_\geq$ the cone of symmetric positive semidefinite $d \times d$-matrices. We abbreviate $M_{\mathcal{I},\mathcal{J}}$ to $M_{\mathcal{I}\mathcal{J}}$ for some matrix M and index sets $\mathcal{I}, \mathcal{J}$.

2. Preliminaries

2.1. Gaussian Graphical Model

Let $\mathbf{X} \sim N(\mu, \Sigma)$ be a multivariate Gaussian with mean $\mu \in \mathbb{R}^d$ and covariance $\Sigma \in \mathcal{S}^d_>$. We call $K = \Sigma^{-1}$ the concentration matrix. Let $G = (V, E)$ be a simple undirected graph with vertices $V = \{1, \ldots, d\}$ and edge set $E \subset V \times V$. A gaussian graphical model with respect to G is the collection of all multivariate Gaussian distributions that satisfy

$$\forall\, ij \notin E \Longrightarrow K_{ij} = 0. \tag{1}$$

As $K_{ij} = 0$ is equivalent to the conditional independence $X_i \perp\!\!\!\perp X_j \mid X_{V\setminus\{i,j\}}$, the graph G implies conditional independence constraints on $\mathbf{X}$. As a slight abuse of notation, we will refer to any multivariate Gaussian $\mathbf{X}$ that satisfies (1) with respect to some graph G as a Gaussian graphical model.

Example 1 *Let $d = 4$ and let G be the graph in Figure 1. The graph G implies zeros in K as follows:*

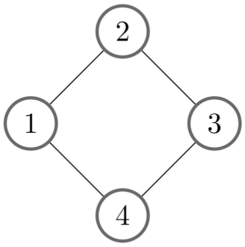

Figure 1: Example of a Gaussian graphical model.

$$K = \begin{pmatrix} K_{11} & K_{12} & 0 & K_{14} \\ K_{12} & K_{22} & K_{23} & 0 \\ 0 & K_{23} & K_{33} & K_{34} \\ K_{14} & 0 & K_{34} & K_{44} \end{pmatrix}.$$

This is equivalent to conditional independence statements $X_1 \perp\!\!\!\perp X_3 \mid X_{\{2,4\}}$ *and* $X_2 \perp\!\!\!\perp X_4 \mid X_{\{1,3\}}$.

2.2. Multivariate Gaussians with Hidden Variables

Let $\mathbf{X}$ be a multivariate Gaussian. We assume to observe only the subvector of variables $\mathbf{X}_O$ with $O \subset [d] := \{1, \ldots, d\}$, and consider the remaining variables H as hidden, where $[d] = O \cup H$ and $O \cap H = \emptyset$. Given i.i.d. (centered) observations $\{\mathbf{x}_O^1, \ldots, \mathbf{x}_O^n\}$ of $\mathbf{X}_O \sim N(\mathbf{0}, \Sigma_{OO})$, we define the sample covariance matrix $S_{OO} = \frac{1}{n}\sum_{i=1}^n \mathbf{x}_O^i (\mathbf{x}_O^i)^T$. The inverse covariance (concentration) matrix of $\mathbf{X}_O$ can be expressed in terms of the full concentration matrix K, such that

$$(\Sigma_{OO})^{-1} = K_{OO} - K_{OH}(K_{HH})^{-1}K_{HO}. \tag{2}$$

Here, the right hand side is the Schur complement $K/K_{H,H}$.

If the complete vector $\mathbf{X}$ satisfies certain constraints, e.g. a sparsity pattern in K as imposed by a Gaussian graphical model, the subset of observed variables $\mathbf{X}_O$ would by default not show the same constraints. For example, the inverse covariance matrix $(\Sigma_{OO})^{-1}$ of the observed variables would typically be a dense matrix even when K is sparse. We illustrate this behavior with an example:

Example 2 *Let* $\mathbf{X}$ *be a 5-variate Gaussian vector that is Markov to the graph in Figure 2. Therefore its concentration matrix* K *satisfies*

$$K = \begin{pmatrix} K_{11} & 0 & 0 & 0 & K_{15} \\ 0 & K_{22} & 0 & 0 & K_{25} \\ 0 & 0 & K_{33} & 0 & K_{35} \\ 0 & 0 & 0 & K_{44} & K_{45} \\ K_{15} & K_{25} & K_{35} & K_{45} & K_{55} \end{pmatrix},$$

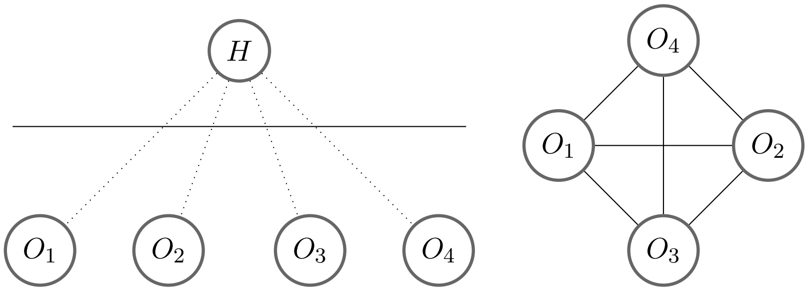

Figure 2: Graph with four observed variables and one hidden (left), and completely connected graph with four observed variables (right).

Here, we can see that the hidden variable is connected with all of the observed variables, while there are no edges between observed variables. The observed subset of variables $\mathbf{X}_O$ *has a dense concentration matrix*

$$(\Sigma_{OO})^{-1} = \begin{pmatrix} K_{11} - \frac{K_{15}^2}{K_{55}} & -\frac{K_{15}K_{25}}{K_{55}} & -\frac{K_{15}K_{35}}{K_{55}} & -\frac{K_{15}K_{45}}{K_{55}} \\ -\frac{K_{15}K_{25}}{K_{55}} & K_{22} - \frac{K_{25}^2}{K_{55}} & -\frac{K_{25}K_{35}}{K_{55}} & -\frac{K_{25}K_{45}}{K_{55}} \\ -\frac{K_{15}K_{35}}{K_{55}} & -\frac{K_{25}K_{35}}{K_{55}} & K_{33} - \frac{K_{35}^2}{K_{55}} & -\frac{K_{35}K_{45}}{K_{55}} \\ -\frac{K_{15}K_{45}}{K_{55}} & -\frac{K_{25}K_{45}}{K_{55}} & -\frac{K_{35}K_{45}}{K_{55}} & K_{44} - \frac{K_{45}^2}{K_{55}} \end{pmatrix},$$

such that the corresponding graphical model is completely connected.

In this setting, we would be interested in being able to estimate K_{OO}, since it gives us information about the sparsity of the full model, and also to estimate $K_{OH}(K_{HH})^{-1}K_{HO}$, since this matrix tells us information about the hidden variables. For instance, if $|H|$ is small, then it will have low rank, since its rank is bounded above by $|H|$. In particular, we can use an estimate of this matrix to estimate the number of hidden variables via the rank.

To tackle this problem, Chandrasekaran et al. (2012) proposed to penalize the two components K_{OO} and $K_{OH}(K_{HH})^{-1}K_{HO}$ that form $(\Sigma_{OO})^{-1}$ separately. To facilitate notation, we define $A := K_{OO}$ and $B := K_{OH}(K_{HH})^{-1}K_{HO}$. Let $\ell(K;S) = \log\det(K) - \operatorname{tr}(KS)$ be the Gaussian log-likelihood for some concentration matrix K and sample covariance S as seen in Chandrasekaran et al. (2012). They introduce the following optimization problem:

$$(\widehat{A}, \widehat{B}) = \operatorname*{argmin}_{A\in\mathcal{S}^d_{>}, B\in\mathcal{S}^d_{\geq}} -\ell(A - B; S_{OO}) + \lambda(\gamma\|A\|_1 + \operatorname{tr}(B)). \tag{3}$$

Here, the ℓ_1-norm penalty $\|A\|_1$ promotes the assumed sparsity, and the trace penalty term $\operatorname{tr}(B)$ the low-rank constraint for B, allowing us to try to estimate this hidden variable component without prior knowledge about it.

Chandrasekaran et al. (2012, Theorem 4.1) provide a theoretical analysis of the convergence of the estimation above. Under a number of assumptions related with the tangent spaces of the sparse and low-rank matrices (please refer to Chandrasekaran et al. (2012) for details), the signs in A and the rank of B are estimated accurately with high probability.

Theorem 2.1 *(Chandrasekaran et al., 2012, Theorem 4.1) Let A and B denote the ground-truth sparse and low-rank components. Let*

$$g_\gamma(A,B) := \max\{\frac{\|A\|_\infty}{\gamma}, \|B\|_2\}$$

and given a matrix M and its tangent space $T(M)$, let

$$\xi(T(M)) := \max_{N \in T(M), \|N\|_2 \leq 1} \|N\|_\infty.$$

Under the assumptions of Chandrasekaran et al. (2012, Proposition 3.3 and Theorem 4.1), we have that the probability of having simultaneously

- $\operatorname{sign}(A^*) = \operatorname{sign}(\widehat{A})$.
- $\operatorname{rank}(B^*) = \operatorname{rank}(\widehat{B})$.
- $g_\gamma(A^* - \widehat{A}, B^* - \widehat{B}) \lesssim \frac{1}{\xi(T(B^*))}\sqrt{\frac{|O|}{n}}$

is at least $1 - 2\exp(-|O|)$.

This result does not give us exactly consistency, since although we have error bounds depending on the sample size n, this does not happen with probability 1 as n goes to infinity. Instead this only happens with probability at least $1 - 2\exp(-|O|)$, which is however close to one with large enough $|O|$.

2.3. Golazo Constraints

Lauritzen and Zwiernik (2022) introduce the Golazo penalty function:

$$\|K\|_{LU} = \sum_{i \neq j} \max\{L_{ij}K_{ij}, U_{ij}K_{ij}\}.$$

Here, L, U are matrices with entries in $\mathbb{R} \cup \{\infty, -\infty\}$ such that $L_{ij} \leq 0 \leq U_{ij}$ for all $i, j \in [d]$ and $\operatorname{diag}(L) = \operatorname{diag}(U) = \mathbf{0}$. Adding the Golazo penalty to the negative Gaussian log-likelihood gives rise to a flexible penalized estimation procedure

$$\widehat{K} = \operatorname*{argmin}_{K \succeq 0} -\ell(K; S) + \|K\|_{LU},$$

that generalizes the standard ℓ_1-penalty as in the graphical lasso. Among the possible constraints that can be enforced with the Golazo penalty are the following:

- **Asymmetric adaptive graphical lasso**: Let $L_{ij} = l_{ij} < 0$ and $U_{ij} = u_{ij} > 0$ for all $i \neq j$. With this, it is possible to penalize differently positive and negative entries. When $L_{ij} = -U_{ij}$ for all $i \neq j$ we are in the adaptive graphical lasso framework, see Fan et al. (2009) for details. If $-l_{ij} = u_{ij} = \lambda$ for all $i \neq j$ for some scalar λ, we have the usual symmetric graphical lasso.

- **Positive lasso**: If we only want to penalize positive entries, we set $L_{ij} = 0$ and $U_{ij} = \lambda > 0$.
- **MTP$_2$ distributions**: A multivariate Gaussian is multivariate totally positive of order two (MTP$_2$) if and only if $K_{ij} \leq 0$ for all $i \neq j$ (Lauritzen et al., 2019). Setting $L_{ij} = 0$ and $U_{ij} = \infty$ for all $i \neq j$ yields the Gaussian MLE under MTP$_2$ when $\|K\|_{LU}$ penalizes the log-likelihood.
- **Positivity and sparsity**: It is possible to constrain for MTP$_2$ and additionally enforce sparsity by setting $L_{ij} = -\lambda < 0$ and $U_{ij} = \infty$ for all $i \neq j$.
- **Gaussian graphical models**: If by assumption / practical knowledge we wish to set the entry K_{ij} to 0, it is possible to enforce this by setting $-L_{ij} = U_{ij} = \infty$, under the convention that $0 \cdot \pm\infty = 0$. This recovers maximum likelihood estimation in Gaussian graphical models when $\|K\|_{LU}$ penalizes the log-likelihood.

3. Learning Gaussian Latent Variable Models under Golazo Constraints

The main idea of this paper is to introduce more flexible latent variable modeling for multivariate Gaussians. For this we propose to substitute the ℓ_1-penalty in the latent optimization problem (3) with the Golazo penalty. This allows to incorporate custom constraints for the dependence structure of $A = K_{OO}$, see Section 2.3 for a list of examples. We thus propose the following optimization problem:

$$\big(\widehat{A}, \widehat{B}\big) = \underset{A \in \mathcal{S}^d_{>}, B \in \mathcal{S}^d_{\geq}}{\operatorname{argmin}} \; -\ell(A - B; S_{OO}) + \|A\|_{LU} + \lambda \operatorname{tr}(B). \tag{4}$$

Note that here the regularization constants can be absorbed into the L, U parameters of the Golazo penalty, so we don't include them explicitly. The log-likelihood $\ell(K; S)$ is a strictly concave function in K. The Golazo penalty is convex (Lauritzen and Zwiernik, 2022). Thus the optimization problem (4) is convex.

Chandrasekaran et al. (2012) provide an asymptotic result (see Theorem 2.1) for the latent Gaussian graphical lasso. The following corollary of Theorem 2.1 extends their result to certain asymmetric Golazo constraints. We believe that a similar result should hold for arbitrary Golazo constraints.

Corollary 3.1 *Let K be the true inverse covariance matrix and define A, B as before. Let all the assumptions of Theorem 2.1 be satisfied, including the choice of λ and γ. Then, define the Golazo parameters L, U such that*

- *if $A^*_{ij} > 0$, choose $L_{ij} \in [-\infty, -\lambda\gamma]$ and let $U_{ij} = \lambda\gamma$,*
- *if $A^*_{ij} < 0$, let $L_{ij} = -\lambda\gamma$ and choose $U_{ij} \in [\lambda\gamma, \infty]$.*
- *if $A^*_{ij} = 0$, choose $L_{ij} \in [-\infty, -\lambda\gamma]$ and $U_{ij} \in [\lambda\gamma, \infty]$.*

In this case we recover the correct sign pattern of A^* and rank of B^* with probability greater than $1 - 2\exp(-|O|)$.

Proof The original statement (when $U_{ij} = -L_{ij} = \lambda\gamma$) tells us that with probability larger than $1 - 2\exp(-|O|)$, the sign of the estimate $\widehat{A}$ is equal to that of A^*, and the rank of $\widehat{B}$ is the same as that of B^*. This means that with that probability, the optimal point of the problem in Equation (4) has the correct signs and rank.

In general, if we add a larger positive penalty to any non-optimal points, the optimal point will stay the same. Here, if $A^*_{ij} > 0$ is positive, we can increase the penalty on the negative values by making L_{ij} smaller. Similarly, if $A^*_{ij} < 0$, we can increase the penalty on positive points by increasing U_{ij}. Finally, if $A^*_{ij} = 0$, then we can increase both penalties simultaneously while maintaining the same optimal point. This proves that the statement about sign and rank is still satisfied. ■

Corollary 3.1 implies that any sign constraints (such as enforcing positivity in an entry, or enforcing sparsity) can be added without losing guarantees if such an assumption is accurate in the specific practical setting. A positive entry in the matrix is enforced by fixing the corresponding entry of L to $-\infty$, a negative entry is enforced by fixing the corresponding entry of U to ∞, and a zero is enforced by doing both simultaneously. Thus, Corollary 3.1 extends the result of Chandrasekaran et al. (2012) to any setting where the ground truth satisfies such constraints.

4. ADMM Algorithm

To tackle the convex optimization problem (4) it is possible to use a general convex solver. For this paper we will employ a multi-block ADMM algorithm that is often used for solving similar problems in the machine learning context, given that this methods can give better time performance than a general convex solver by taking advantage of separable problems in terms of the blocks of variables. Here, we are adapting the algorithm studied in Chang et al. (2020), which is a good reference for the details on the general idea of the algorithm. A further reference is Li et al. (2023), where a similar setting under graph Laplacian constraints was studied. We rewrite (4) in terms of three blocks of variables as follows:

$$(\widehat{M}, \widehat{A}, \widehat{B}) = \underset{M,A\in\mathcal{S}^d_>,B\in\mathcal{S}^d_\geq}{\operatorname{argmin}} \; -\ell(M; S_{OO}) + \|A\|_{LU} + \lambda\,\mathrm{tr}(B) \qquad \text{s.t. } M = A - B. \tag{5}$$

We define the augmented Lagrangian of the optimization problem

$$\mathcal{L}_\sigma(M, A, B, \Lambda) := -\ell(M; S_{OO}) + \|A\|_{LU} + \lambda\,\mathrm{tr}(B) - \langle \Lambda, M - A + B\rangle + \frac{\sigma}{2}\|M - A + B\|^2,$$

where $\Lambda \in \mathbb{R}^{d\times d}$ are the Lagrange multipliers. This algorithm used this augmented Lagrangian since the additional penalty helps enforce the constraints between the blocks of variables. Here, σ denotes the hyperparameter that tunes how strongly the constraints between the blocks of variables are enforced. The $k+1$ iteration of the algorithm will be as follows:

$$\begin{cases} M^{k+1} := \underset{M\in\mathbb{R}^{d\times d}}{\operatorname{argmin}}\, \mathcal{L}_\sigma(M, A^k, B^k, \Lambda^k) + \frac{\rho\sigma}{2}\|M - M^k\|^2, \\ \Lambda^{k+\frac{1}{2}} := \Lambda^k - \alpha\sigma(M^{k+1} - A^k + B^k), \\ A^{k+1} := \underset{A\in\mathbb{R}^{d\times d}}{\operatorname{argmin}}\, \|A\|_{LU} + \frac{\tau r_1}{2}\left\|A - A^k + \frac{\Lambda^{k+\frac{1}{2}}}{\tau r_1}\right\|^2, \\ B^{k+1} := \underset{B\in\mathbb{B}^{d\times d}\ B\succeq 0}{\operatorname{argmin}}\, \lambda_n \operatorname{tr}(B) + \frac{\tau r_2}{2}\left\|B - B^k + \frac{\Lambda^{k+\frac{1}{2}}}{\tau r_2}\right\|^2, \\ \Lambda^{k+1} := \Lambda^{k+\frac{1}{2}} + \sigma(A^{k+1} - A^k) - \sigma(B^{k+1} - B^k). \end{cases}$$

The Lagrange multiplier is updated two times in each iteration given the multi-block nature of the problem. For details about the procedure see Bai et al. (2017). As shown in Chang et al. (2020), the conditions $\tau \in (\frac{2+\alpha}{2}, +\infty), \rho \in [0, +\infty), r_1 > \sigma, r_2 > \sigma$ are sufficient conditions for convergence. Here α is the step size of the half-update of the Lagrange multiplier. It is suggested by them to fix for practical reasons $\rho = 0$, $\tau = \varsigma\frac{2+\alpha}{2}$ and $r_1 = r_2 = \varsigma\sigma$, where $\varsigma > 1$. Here, ρ is a parameter than can help speed up convergence of the method but we do not worry about this in our paper.

The three subproblems that we have after the considerations about the parameters have simple closed form solutions, which we briefly summarize here. Firstly, the subproblem for M^{k+1} has a first order condition

$$S_{OO} - M^{-1} + \sigma\big(M - A^k + B^k - \frac{\Lambda^k}{\sigma}\big) + \rho\sigma(M - M^k) = 0.$$

By multiplying by M, this is converted into a quadratic equation on M:

$$(\rho+1)\sigma M^2 + \big(S_{OO} + \sigma(B^k - A^k) - \Lambda^k - \rho\sigma M^k\big)M - I = 0.$$

If we consider the eigendecomposition $C\operatorname{diag}(\mathbf{v})C^T = S_{OO} + \sigma(B^k - A^k) - \Lambda^k - \rho\sigma M^k$ and define a new vector of eigenvalues $\mathbf{x}$ such that

$$x_i := \frac{-v_i + \sqrt{v_i^2 + 4(\rho+1)\sigma}}{2(\rho+1)\sigma},$$

then the closed form solution to the problem is $M^{k+1} = C\operatorname{diag}(\mathbf{x})C^T$.

For the second subproblem, let $\mathbf{0}$ denote the zero matrix and let max denote here the entry-wise maximum. Then, the solution is

$$A^{k+1} = \min\left\{A^k - \frac{\Lambda^{k+\frac{1}{2}} + L}{\tau r_1}, \mathbf{0}\right\} + \max\left\{A^k - \frac{\Lambda^{k+\frac{1}{2}} - U}{\tau r_1}, \mathbf{0}\right\}.$$

Finally, the third subproblem also has a simple closed form solution. Consider the eigendecomposition $D\operatorname{diag}(\beta)D^T = B^k + \frac{\Lambda^{k+\frac{1}{2}} - \lambda I}{\tau r_2}$. Then, the closed form solution is given by $B^{k+1} = D\operatorname{diag}(\max(\beta, \mathbf{0}))D^T$, where again the max is taken entry-wise.

Therefore, it is straightforward to solve this problem iteratively. Let N denote the maximum number of iterations that we allow in a practical setting, and let $\epsilon_1, \epsilon_2 \in \mathbb{R}_{\geq 0}$ be parameters such that the algorithm stops if we have that both of the following conditions are satisfied:

$$\begin{aligned} \text{RelChg} :&= \max\left\{ \frac{\|M^{k+1}-M^k\|_F}{1+\|M^k\|_F}, \frac{\|A^{k+1}-A^k\|_F}{1+\|A^k\|_F}, \frac{\|B^{k+1}-B^k\|_F}{1+\|B^k\|_F} \right\} < \epsilon_1 \\ \text{IER} :&= \|M^k - A^k + B^k\|_F < \epsilon_2. \end{aligned}$$

The algorithm stops only after the maximum number of iterations are performed or when the previous criterion is satisfied. We show pseudocode for the algorithm in Algorithm 1.

Algorithm 1: Multi-block ADMM for GGM estimation

```
Input: $S_{OO}, L, U, P, \{\sigma, \alpha, r_1, r_2, \tau, \lambda_n, \rho\}, \{\epsilon_1, \epsilon_2\}, N, k = 0$
Output: $\hat{M}_n, \hat{A}_n, \hat{B}_n$
Starting point: $M^0 \leftarrow I, A^0 \leftarrow I, B^0 \leftarrow \mathbf{0}$
while $k < N$ and (RelChg $\geq \epsilon_1$ or IER $\geq \epsilon_2$) do
    Compute eigendecomposition $C\text{diag}(\alpha)C^T$ of $S_{OO} + \sigma(B^k - A^k) - \Lambda^k - \rho\sigma M^k$
    $x_i \leftarrow \frac{-\alpha_i + \sqrt{\alpha_i^2 + 4(\rho+1)\sigma}}{2(\rho+1)\sigma}$
    $M^{k+1} \leftarrow C\text{diag}(\mathbf{x})C^T$
    $\Lambda^{k+\frac{1}{2}} = \Lambda^k - \alpha\sigma(M^{k+1} - A^k + B^k)$
    $A^{k+1} \leftarrow \min\left\{A^k - \frac{\Lambda^{k+\frac{1}{2}} + L}{\tau r_1}, \mathbf{0}\right\} + \max\left\{A^k - \frac{\Lambda^{k+\frac{1}{2}} - U}{\tau r_1}, \mathbf{0}\right\}$
    Compute eigendecomposition $D\text{diag}(\beta)D^T$ of $B^k + \frac{\Lambda^{k+\frac{1}{2}} - \lambda_n I}{\tau r_2}$
    $B^{k+1} \leftarrow D\text{diag}(\max(\beta, \mathbf{0}))D^T$
    $\Lambda^{k+1} \leftarrow \Lambda^{k+\frac{1}{2}} + \sigma(A^{k+1} - A^k) - \sigma(B^{k+1} - B^k)$
    $k = k + 1$
end
return $\hat{M}_n \leftarrow M^k, \hat{A}_n \leftarrow B^k, \hat{B}_n \leftarrow B^k$
```

5. Application

Note that during our experiments we will fix the values of the ADMM parameters following the practical choices made in the paper of Chang et al. (2020), that is, we do not tune these values for speed, we only pick values that guarantee convergence of the method.

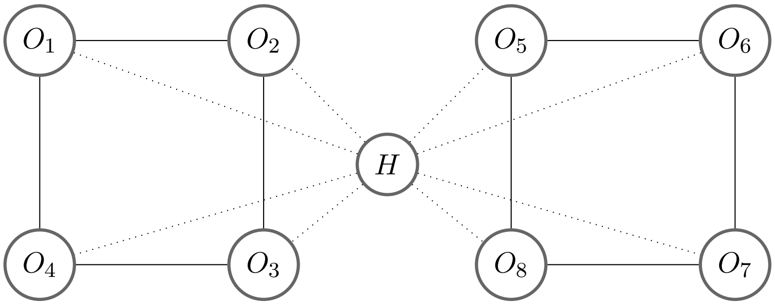

Figure 3: Two disconnected 4-cycles with one hidden variable.

5.1. Simulated Data

Taking inspiration from Engelke and Taeb (2024), we consider a graph $G = (V, E)$ consisting of two disconnected (except for edges going through the hidden variable) cycles with 25 observed nodes each, and one hidden variable. We set $K_{ii} = 5$ for all $i \in V = \{1, \dots, 51\}$ and $K_{ij} = -2$ for all $1 \leq i, j \leq p = 50$ with $ij \in E$, and $K_{ij} = 0$ otherwise. The hidden variable is connected to all of the observed variables, with $K_{ih} = K_{hi} = 5/p$ for all $i \neq h = 51$.

In this study we compare the standard ℓ_1-penalty with a positive dependence constraint. To showcase the flexibility of the Golazo approach, we further include two modified versions of these penalties that incorporate partial graphical model constraints (i.e. partial sparsity in K). To simplify notation, let us call $O_1 = \{1, \dots, 25\}, O_2 = \{26, \dots 50\}$, $H = \{51\}$, where O_1 denotes the indices of the nodes of the first cycle, O_2 the nodes of the second cycle and H the hidden variable. The constraints that we are going to test are the following:

1. $L_{ij} = -\lambda\gamma$ and $U_{ij} = \lambda\gamma$ for all $i \neq j$, that is, the standard ℓ_1-penalty.
2. $L_{ij} = -\lambda\gamma$ and $U_{ij} = \lambda\gamma$ for all $i \neq j$ where i, j are both either in O_1 or O_2. For i, j where each node is in a different subcycle, $L_{ij} = -\infty$ and $U_{ij} = \infty$, that is, we assume that O_1 or O_2 are not connected by an edge.
3. $L_{ij} = 0$ and $U_{ij} = \infty$ for all $i \neq j$, that is, the MTP_2 constraint.
4. $L_{ij} = 0$ and $U_{ij} = \infty$ for all $i \neq j$ where i, j are both either in O_1 or O_2. For i, j where each node is in a different subcycle, $L_{ij} = -\infty$ and $U_{ij} = \infty$, that is, the MTP_2 constraint with the additional assumption that O_1 or O_2 are not connected by an edge.

We generate two samples of size $n = 100$ in $N = 20$ different trials. We train the model using the first sample and then evaluate the Gaussian log-likelihood on the second one. This could also be done using the ground truth covariance. We fix $\gamma = 0.5$ for the constraints 1 and 2, after testing various values and noticing that the overall behavior is stable for a range of values of γ (compare also the discussions in Chandrasekaran et al. (2012, 2011); Engelke and Taeb (2024)). Note that constraints 3 and 4 are independent of γ. We select values for λ from 10^{-8} to 1, with 50 values evaluated in total. We perform the simulation,

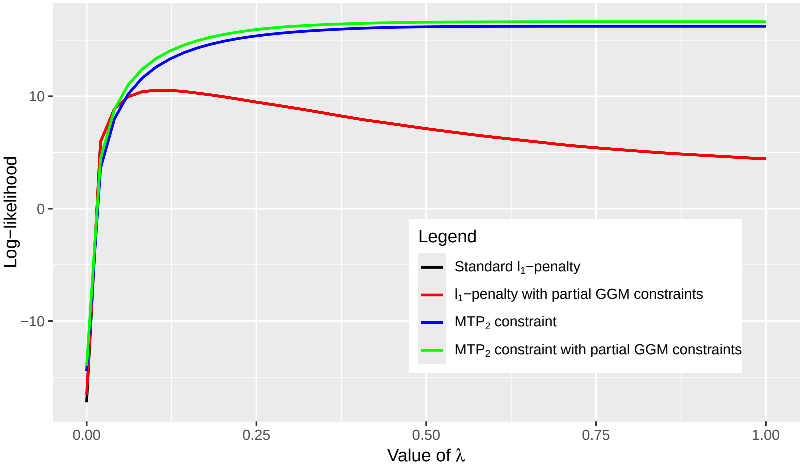

Figure 4: Results for the two cycles (red and black line become equal).

calculation and validation steps for each constraint and value of λ and compute an average of the log-likelihoods over the different trials. Figure 4 visualizes the results of this study. We observe that the MTP$_2$ constraints provide a robust method that outperform the ℓ_1-penalty. Furthermore, there is a small improvement when the partial graphical model constraints are added.

5.2. Real-world Data

For this real-world data application we will use gene data from the Rosetta dataset (see Hughes et al. (2000) for the original source), which has 301 samples from 6316 variables. We obtained the dataset from the code of Chang et al. (2020). The way to process this data to obtain a sample covariance matrix (which is the data input to our algorithm) is described in Ma et al. (2013). Here, the idea is to compute the sample variances of each variable, and then pick the p variables with the largest sample variance, resulting in $p = 25$ observed variables for the latent Gaussian graphical model.

During these experiments, we fix $\gamma = 0.1$, after testing various values and seeing that this one gave near optimal result for the lasso-based methods. We select it in this way since the positivity-based methods optimal performance is not affected by this parameter. Then we explore how the behavior of the estimates depend on the value of λ and the type of Golazo constraint selected. We select a large enough interval for λ so that the general behavior of each constraint can be appreciated. Here λ takes values from 10^{-8} to 0.4, with 30 values evaluated in total.

We use 5-fold cross-validation to evaluate how well each of the methods generalizes better, and we will use as the score the log-likelihood with respect to the validation set. We show the results for four different Golazo constraints:

1. $L_{ij} = -\lambda\gamma$ and $U_{ij} = \lambda\gamma$ for all $i \neq j$, that is, the standard ℓ_1-penalty.

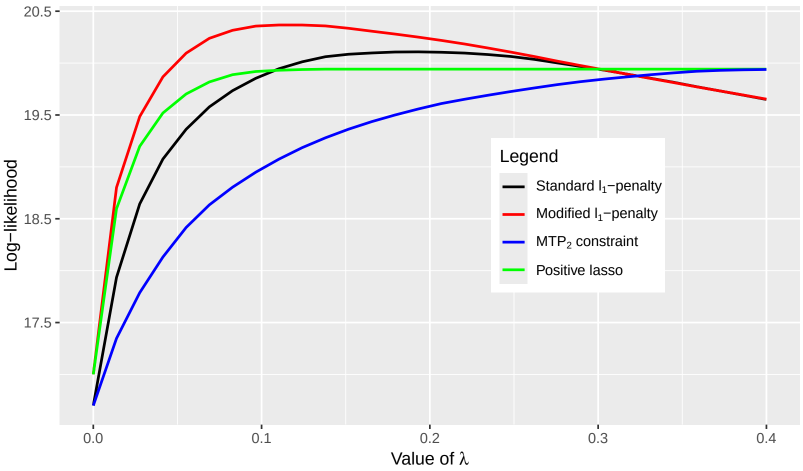

Figure 5: Results for the gene data (red and black line become equal).

2. $L_{ij} = -\lambda\gamma$ and $U_{ij} = \infty$ for all $i \neq j$, that is, a modified MTP$_2$ constraint.
3. $L_{ij} = 0$ and $U_{ij} = \infty$ for all $i \neq j$, that is, the MTP$_2$ constraint.
4. $L_{ij} = 0$ and $U_{ij} = \lambda\gamma$ for all $i \neq j$, that is, the positive lasso constraint.

We can see in Figure 5 that the best overall validation log-likelihood occurs when using constraint 2, which shows that combining MTP$_2$ and an ℓ_1-penalty can yield improved performance over either of them. We see as in the simulation study that the MTP$_2$ constraint seems to be relatively robust with respect to the choice of λ and performs comparably well, although not optimal in this case.

6. Discussion

In this paper we propose generalized latent Gaussian graphical model learning via the Golazo penalty function. We provide an ADMM algorithm that we apply to simulated and real data, and discuss various flexible penalization choices in comparison to the standard ℓ_1-penalty. In particular, the robustness of the MTP$_2$ constraint with respect to the hyperparameters provides an attractive alternative to settings when hyperparameter tuning is not possible (for instance, when training is too expensive). For future research, a main question beyond the scope of this paper is an extension of Corollary 3.1, as well as an application of other Golazo penalties. Furthermore, one could explore whether some kind of ensemble of such estimators can improve performance over one estimator alone. This would be an interesting practical improvement, since if a model is trained over multiple hyperparameters to obtain an optimal choice, then suboptimal models could still be used as part of such an ensemble. We would also like to investigate sparsity, ground truths with more than 1 hidden variable and a larger number of graph topologies in simulations in future work.

References

J. Bai, J. Li, F. Xu, and H. Zhang. Generalized symmetric admm for separable convex optimization. *Computational Optimization and Applications*, 70(1):129–170, Nov. 2017. ISSN 1573-2894. doi: 10.1007/s10589-017-9971-0. URL http://dx.doi.org/10.1007/s10589-017-9971-0.

V. Chandrasekaran, S. Sanghavi, P. A. Parrilo, and A. S. Willsky. Rank-sparsity incoherence for matrix decomposition. *SIAM Journal on Optimization*, 21(2):572–596, 2011. doi: 10.1137/090761793. URL https://doi.org/10.1137/090761793.

V. Chandrasekaran, P. A. Parrilo, and A. S. Willsky. Latent variable graphical model selection via convex optimization. *The Annals of Statistics*, 40(4):1935–1967, 2012. ISSN 00905364, 21688966. URL http://www.jstor.org/stable/41806519.

X. Chang, J. Bai, D. Song, and S. Liu. Linearized symmetric multi-block ADMM with indefinite proximal regularization and optimal proximal parameter. *Calcolo*, 57(4), Nov. 2020. ISSN 1126-5434. doi: 10.1007/s10092-020-00387-1. URL http://dx.doi.org/10.1007/s10092-020-00387-1.

S. Engelke and A. Taeb. Extremal graphical modeling with latent variables. 2024. doi: 10.48550/arXiv.2403.09604. URL https://doi.org/10.48550/arXiv.2403.09604.

J. Fan, Y. Feng, and Y. Wu. Network exploration via the adaptive LASSO and SCAD penalties. *The Annals of Applied Statistics*, 3(2):521 – 541, 2009. doi: 10.1214/08-AOAS215. URL https://doi.org/10.1214/08-AOAS215.

T. R. Hughes, M. J. Marton, A. R. Jones, C. J. Roberts, R. Stoughton, C. D. Armour, H. A. Bennett, E. Coffey, H. Dai, Y. D. He, M. J. Kidd, A. M. King, M. R. Meyer, D. Slade, P. Y. Lum, S. B. Stepaniants, D. D. Shoemaker, D. Gachotte, K. Chakraburtty, J. Simon, M. Bard, and S. H. Friend. Functional discovery via a compendium of expression profiles. *Cell*, 102(1):109–126, July 2000. ISSN 0092-8674. doi: 10.1016/s0092-8674(00)00015-5. URL http://dx.doi.org/10.1016/s0092-8674(00)00015-5.

S. Lauritzen and P. Zwiernik. Locally associated graphical models and mixed convex exponential families. *The Annals of Statistics*, 50(5):3009 – 3038, 2022. doi: 10.1214/22-AOS2219. URL https://doi.org/10.1214/22-AOS2219.

S. Lauritzen, C. Uhler, and P. Zwiernik. Maximum likelihood estimation in Gaussian models under total positivity. *The Annals of Statistics*, 47(4):1835–1863, 2019. ISSN 0090-5364,2168-8966. doi: 10.1214/17-AOS1668. URL https://doi.org/10.1214/17-AOS1668.

R. Li, J. Lin, H. Qiu, W. Zhang, and J. Wang. Graph learning for latent-variable Gaussian graphical models under Laplacian constraints. *Neurocomputing*, 532:67–76, 2023. ISSN 0925-2312. doi: 10.1016/j.neucom.2023.02.007. URL https://www.sciencedirect.com/science/article/pii/S092523122300139X.

S. Ma, L. Xue, and H. Zou. Alternating Direction Methods for Latent Variable Gaussian Graphical Model Selection. *Neural Computation*, 25(8):2172–2198, 08 2013. ISSN 0899-7667. doi: 10.1162/NECO_a_00379. URL https://doi.org/10.1162/NECO_a_00379.

N. Meinshausen and P. Bühlmann. High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, 34(3):1436–1462, 2006. ISSN 0090-5364,2168-8966. doi: 10.1214/009053606000000281. URL https://doi.org/10.1214/009053606000000281.

M. Yuan and Y. Lin. Model selection and estimation in the Gaussian graphical model. *Biometrika*, 94(1):19–35, 2007. ISSN 0006-3444,1464-3510. doi: 10.1093/biomet/asm018. URL https://doi.org/10.1093/biomet/asm018.