COPULA GAUSSIAN GRAPHICAL MODELS WITH HIDDEN VARIABLES

Hang Yu, Justin Dauwels, and Xueou Wang

School of Electrical and Electronics Engineering, School of Physical and Mathematical Sciences
Nanyang Technological University, Singapore, 639798

ABSTRACT

Gaussian hidden variable graphical models are powerful tools to describe high-dimensional data; they capture dependencies between observed (Gaussian) variables by introducing a suitable number of hidden variables. However, such models are only applicable to Gaussian data. Moreover, they are sensitive to the choice of certain regularization parameters. In this paper, (1) copula Gaussian hidden variable graphical models are introduced, which extend Gaussian hidden variable graphical models to non-Gaussian data; (2) the sparsity pattern of the hidden variable graphical model is learned via stability selection, which leads to more stable results than cross-validation and other methods to select the regularization parameters. The proposed methods are validated on synthetic and real data.

Index Terms— Gaussian copula, hidden variable graphical model, stability selection, bioinformatics

1. INTRODUCTION

Sparse graphical models (see, e.g., [1]) provide an effective way to capture statistical structure in high-dimensional data, such as gene expression data, multi-electrode brain recordings, and stock market data. A sparse graph displays the most significant interactions between variables (e.g., genes, brain areas, stocks), and may help to interpret the data.

In practice, it is quite common that data is unavailable for some relevant variables. For instance, one typically measures the expression of a limited subset of genes, chosen among the large number of genes of an organism; the observed genes may be strongly affected by genes that have not been measured. The latter may then be treated as hidden variables in a statistical model, providing a simple explanation for the statistical relations between the observed genes.

Therefore, sparse graphical models with hidden variables are quite powerful models for a large variety of real-life datasets. When the observed variables Z_o and hidden variables Z_h are jointly Gaussian distributed, the structure of the graphical model can be defined by the precision matrix (inverse covariance matrix) of the observed and hidden variables. Recently, Chandraskaran et al. [2] decomposed the (marginal) precision matrix of Z_o into a sparse matrix K_o (conditional precision matrix) and a low-rank matrix L, which describes the coupling between the observed and hidden variables. The conditional graphical model K_o and the number of hidden variables (rank of L) are inferred by solving a convex regularized maximum-likelihood problem. The conditional precision matrix K_o is represented as a graph, where nodes i and j are connected by an edge iff the corresponding element (i, j) in K_o is non-zero. That graph visualizes the dependence among the observed variables, conditioned on the hidden variables.

Obviously, Gaussian hidden variable graphical models (GHVGM) are limited to Gaussian data. In this paper, we extend GHVGM to non-Gaussian data by means of Gaussian copulas [3], referred to as copula Gaussian hidden variable graphical model (CGHVGM).

Another issue with the GHVGM is the selection of the regularization/penalty parameters, which determine the resulting sparsity pattern of K_o and rank of L. Standard approaches for regularization selection, including cross validation (cv), Akaike's information criterion (AIC), and Bayesian information criterion (BIC), are known to overfit the data, and they typically result in graphs that are too dense [4].

In this paper, we circumvent the delicate issue of regularization selection by first learning the graph structure and next inferring the parameters. Specifically, stability selection [5] is used to learn the structure (sparsity pattern) of K_o. We further specify K_o and L by solving a convex problem subject to the structure constraints. The estimated number of hidden variables equals to the rank of L.

We apply our model (CGHVGM) to non-Gaussian synthetic and real data (cell signaling data). The CGHVGM is able to recover the number of hidden variables and the conditional graph K_o, in contrast to other related models. Interestingly, the GHVGM dramatically failed for both datasets, presumably because it is not intended for non-Gaussian data.

This paper is organized as follows. In Section 2, we first review the copula Gaussian graphical model and the Gaussian hidden variable graphical model, and next we present the proposed copula Gaussian hidden variable graphical model. In Section 3, we explain how we learn the structure and parameters of the proposed model. In Section 4, we assess the proposed model and benchmark it with other models, by means of synthetic and real data. In Section 5, we offer concluding remarks.

2. GRAPHICAL MODELS

In the following, we briefly describe the copula Gaussian graphical model, the Gaussian hidden variable graphical model, and the proposed copula Gaussian hidden variable graphical model.

2.1. Copula Gaussian Graphical Model

We denote the observed non-Gaussian variables and hidden Gaussian variables as $Y_1, \ldots, Y_P$ and $Z_1, \ldots, Z_P$ respectively. A Gaussian copula graphical model is defined as [3]:

$$Z \sim \mathcal{N}(0, K^{-1}) \tag{1}$$

$$Y_k = F_k^{-1}(\Phi(Z_k)), \tag{2}$$

where K is the precision matrix whose inverse (the covariance matrix) has normalized diagonal, Φ is the CDF (cumulative distribution function) of the standard Gaussian distribution, and F_k is the CDF of Y_k. The latter is often approximated by the empirical distributions $\hat{F}_k$. Note that F_k^{-1} is the pseudo-inverse of F_k, which is defined as:

$$F^{-1}(y) = \inf_{x \in \mathcal{X}} \{F(x) \geq y\}. \tag{3}$$

2.2. Gaussian Hidden Variable Graphical Model

Suppose we have Gaussian distributed observed variables Z_o and hidden variables Z_h. The joint precision matrix $K_{(o\,h)}$ associated with these variables is given by:

$$K_{(o\,h)} = \begin{bmatrix} K_o & K_{o,h} \\ K_{h,o} & K_h \end{bmatrix}. \tag{4}$$

According to Schur complement, the marginalized precision matrix $\tilde{K}_o$ of Z_o can be written as:

$$\tilde{K}_o = K_o - K_{o,h} K_h^{-1} K_{h,o} = K_o - L, \tag{5}$$

with product matrix $L = K_{o,h} K_h^{-1} K_{h,o}$. Those two components have their own properties [2]: K_o is the supposedly sparse conditional precision matrix of Z_o, conditioned on Z_h; the product matrix L summarizes the effect of marginalization over the hidden variables. The rank of that matrix (equal to the number of hidden variables Z_h) is low, since the number of hidden variables is supposed to be small.

Given i.i.d. samples of Z_o, our objective is to estimate K_o and L; we are especially interested in the rank of L, since it equals the number of hidden variables Z_h. Those matrices may be recovered by solving the convex relaxation [2]:

$$\begin{aligned}(\hat{K}_o, \hat{L}) = \underset{K_o, L}{\operatorname{argmin}}\, &\operatorname{trace}((K_o - L)\Sigma_o) - \log\det(K_o - L) \\ &+ \lambda(\gamma \|K_o\|_1 + \operatorname{trace}(L)),\end{aligned} \tag{6}$$

where $\hat{K}_o$ and $\hat{L}$ are the estimates of K_o and L respectively, and Σ_o is the empirical marginal covariance of Z_o. The convex problem (6) can be solved efficiently by the Newton-CG primal proximal point algorithm [6]. To recover the correct matrices K_o and L, the parameters λ and γ need to be chosen appropriately, which is a critical issue that will be addressed in Section 3.

2.3. Copula Gaussian Hidden Variable Graphical Model

The observed (continuous) variables Y are non-Gaussian, and each of them is associated with a Gaussian distributed hidden variable Z_o, as in the copula Gaussian model. However, besides the hidden variables Z_o, there exist several hidden variables Z_h that are not associated with observed variables. In the graphical model, the nodes Z_h are only connected to hidden variables; they are not connected to observed variables Y. In other words, the variables Y and Z_o constitute a Gaussian copula graphical model, while the variables Z_o and Z_h form a Gaussian hidden variable graphical model; together, the variables (Y, Z_o, Z_h) form a copula Gaussian hidden variable graphical model with associated conditional precision matrix K_o and product matrix L (cf. (5)).

Given i.i.d. samples of the non-Gaussian variables Y, we wish to infer the conditional precision matrix K_o of Z_o (conditioned on Z_h), and the product matrix L.

As a first step, we transform the non-Gaussian observed variables Y into Gaussian distributed hidden variables Z_o (associated with the observed variables Y):

$$Z_{ok} = \Phi^{-1}(\hat{F}_k(Y_k)), \tag{7}$$

where Φ is the CDF of the standard Gaussian distribution and $\hat{F}_k$ is the empirical CDF of Y_k. As a result, we are dealing with Gaussian variables Z_o which together with Z_h constitute a GHVGM.

In the second step, we follow the procedure of (6) to infer the sparse conditional precision matrix K_o of Z_o and the low-rank product matrix L. Also here, of course, we need to pay special attention to the parameters λ and γ, which will be the subject of Section 3.

3. LEARNING AND INFERENCE

A suitable choice of regularization parameters λ and γ in (6) can produce the graphical model with true sparsity pattern of K_o. However, standard procedures for selecting λ and γ are known to overfit the data and result in graphs that are too dense [4]. As an alternative, we employ a two-step procedure of structure learning and parameter learning. Stability selection [5] is used for structure learning, resulting in the sparsity pattern of K_o; constrained by this inferred sparsity pattern, we then infer the parameters by solving a problem similar to (6).

3.1. Structure Learning

We use the stability selection procedure [5] to infer the sparsity pattern of the conditional precision matrix K_o, from N i.i.d. samples S of $Y \in \mathbb{R}^P$ (or $Z_o \in \mathbb{R}^P$).

First, M subsets S_1, S_2, $\ldots$, S_M are randomly drawn without replacement from the dataset, each of size $\lfloor N/2 \rfloor$.

Second, we select a range of λ and γ (cf. (6)). Now focus on one pair of parameters (λ, γ) in that range. For each subset S_m (for $m = 1, \ldots, M$), we estimate one precision matrix K_o using (6), resulting in M precision matrices K_1, $\ldots$, K_M. For each element (i, j) in the matrix K_m, the number of times it is non-zero ($K_m(i, j) \neq 0$) among the M matrices is counted and divided by M; as a result, we obtain the probability (stability) that this edge exists in the graphical model associated with (λ, γ). By varying λ and γ through the chosen range, we can draw a surface of the stability for each edge.

At last, we include edge (i, j) in the graphical model associated with K_o, if the probability of that edge, for at least one pair (λ, γ) in the selected range, is larger than threshold π_{thr} [5]:

$$\pi_{thr} = \frac{\overline{p}^2}{P(P-1)E} + 0.5. \tag{8}$$

The parameter $\overline{p}$ is the *average* number of edges in the graphs associated with each pair (λ, γ) in the selected range, inferred from the entire dataset S through (6). E is the expected number of falsely selected edges.

We also applied a randomized approach suggested in [5]. When inferring the matrices K_m, we divide the parameter pair (λ, γ) by a random pair (α_1, α_2) (different for each subset), where α_1 and α_2 are uniformly distributed on $[0.2, 1]$. In the following, we will only report results for the randomized approach, since it yields the best results.

3.2. Parameter Learning

The structure (sparsity pattern) of K_o has now been inferred, and that helps us to estimate K_o and L; specifically, we solve a problem similar to (6), where the l_1 term is removed, and the sparsity pattern is encoded by a large penalty σ on the absolute value of zero elements in K_o:

$$\begin{aligned}(\hat{K}_o, \hat{L}) = \operatorname*{argmin}_{K_o, L} \operatorname{trace}((K_o - L)\Sigma_o) - \log\det(K_o - L) \\ + \lambda \operatorname{trace}(L) + \sigma \sum_{K_o(i,j)=0} |K_o(i,j)|. \end{aligned} \tag{9}$$

The parameter λ is selected as the mean of the λs in the chosen range (the second step of stability selection) that generate the same structure as K_o estimated by stability selection. The number of hidden variables can be estimated easily by computing the rank of L.

4. NUMERICAL RESULTS

We test our proposed graphical model and existing graphical models on a synthetic and real data set.

4.1. Synthetic Data

We generate non-Gaussian synthetic data using the following method:

1. Generate a random precision matrix by using the method of [7], which mimics characteristics of real-world biological networks. More specifically, we first uniformly sample $x_1, ..., x_n$ from a unit square. The precision matrix is initialized as a unit matrix. Next, we set the element $K(i, j) = K(j, i)$ of precision matrix equal to $\rho = 0.245$ with probability $(\sqrt{2\pi})^{-1} \exp(-4\|x_i - x_j\|^2)$, and equal to zero otherwise.
2. Add a few variables and connect each of them to at least 80% of other variables (corresponding elements in precision matrix are non-zero).
3. Generate a Gaussian dataset corresponding to the above precision matrix and discard all the samples of variables added in Step 2 (hidden variables).
4. Apply different types of copula to each variable (including beta, exponential, chi-square copula), transforming the Gaussian variables to (continuous) non-Gaussian variables.

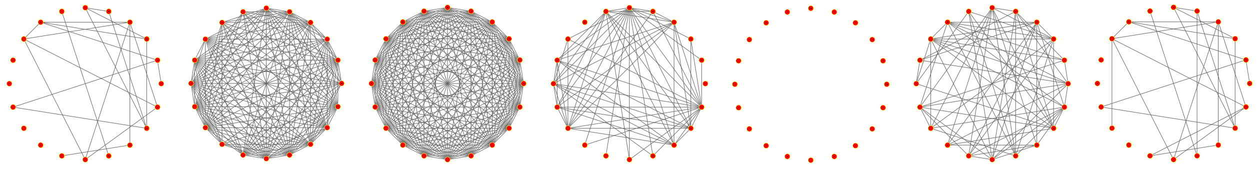
We apply our proposed graphical model to that data, in particular, copula Gaussian hidden variable graphical model selection with stability selection (CGHVGM with ss). We also consider 6 other approaches including glasso [1], copula glasso [7], Gaussian hidden variable graphical model inferred by cross validation (GHVGM with cv), Gaussian hidden variable graphical model with stability selection (GHVGM with ss), and copula Gaussian hidden variable graphical model selection with cross validation (CGHVGM with cv). We evaluate those methods through various criteria including precision, recall, F_1-score, and number of parameters (Prm No.). Precision is defined as the proportion of correctly estimated edges to all the edges in the estimated graph; recall is defined as the proportion of successfully estimated edges to all the edges in the true graph; F_1-score is defined as 2·precision·recall/(precision+recall), which is a weighted average of the precision and recall.

The results for a 20-dimensional dataset are summarized in Fig. 1 and Table 1.

Table 1. Quantitative comparison of different methods

Methods	Criteria			
	Precision	Recall	F_1-score	Prm No.
glasso	0.1195	0.9048	0.2111	179
copula glasso	0.1105	1.0000	0.1990	210
GHVGM with cv	0.1324	0.4286	0.2023	488
GHVGM with ss	0.0000	0.0000	0.0000	0
CGHVGM with cv	0.2877	1.0000	0.4468	173
CGHVGM with ss	0.8750	1.0000	0.9333	64
glasso(m)	1.0000	0.6737	0.8050	179
copula glasso(m)	1.0000	1.0000	1.0000	210

The results show that CGHVGM with ss achieves the best performance with the least number of parameters. The CGHVGM approach with cv generates a dense graph, whereas GHVGM with ss produces graph without edges.



(a) True graph (b) Glasso (c) Copula glasso (d) GHVGM(cv) (e) GHVGM(ss) (f) CGHVGM(cv) (g) CGHVGM(ss)

Fig. 1. Results of different methods on the 20-dimensional synthetic dataset

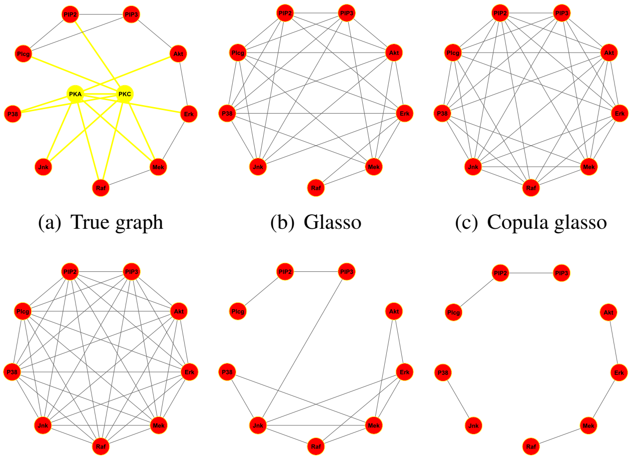
The rank of the inferred L in CGHVGM with ss equals to the number of hidden variables. However, the rank estimated using CGHVGM with cv is four times larger than the true value, while GHVGM with cv yields a full-rank matrix L.

The glasso and copula glasso methods infer the marginal precision matrix $\tilde{K}_o$ in (5), instead of the conditional precision matrix K_o. The results for that inference problem are listed under glasso(m) and copula glasso(m) in Table 1, where "m" refers to marginal. The performance of copula glasso is the best, since glasso is only effective for Gaussian data. However, the marginal precision matrix $\tilde{K}_o$ is much denser than the conditional precision matrix K_o, and therefore, is a more complicated model (more parameters involved).

4.2. Real Data

The dataset consists of the expression level of 11 proteins in 7466 cells [8]. Two proteins (PKA and PCA) seem to interact with most of the 9 other proteins, and they can be considered as "hubs" (see Fig. 2(a), indicated in yellow). We remove those proteins from the dataset. The graphical models should infer those two proteins as hidden variables.

We have verified that the data is non-Gaussian, and therefore, one would expect the copula Gaussian model with hidden variables to perform well on this dataset.



(a) True graph (b) Glasso (c) Copula glasso

(d) GHVGM with cv (e) CGHVGM with cv (f) CGHVGM with ss

Fig. 2. Results of different methods on the cell signaling data

The results are shown in Fig. 2. The undirected version of theoretical analysis in [8] is regarded as the "true" graph. The CGHVGM approach with cv overfits the data. Both glasso and copula glasso are trying to estimate the marginalized dense graph. GHVGM with cv generates an incorrect full graph, and GHVGM with ss leads to a fully disconnected graph. In contrast, the proposed method yields the most accurate conditional graph. Moreover, the rank of the inferred matrix L equals 2, which is indeed the true number of hidden variables (PKA and PCA).

5. CONCLUSION

In this paper, we introduced the copula Gaussian graphical model with hidden variables; such model can provide a simple description of high-dimensional non-Gaussian data, where correlations can be captured through a few hidden variables. We used stability selection to learn the structure of the model and inferred the parameters of model and the number of hidden variables by solving a convex problem.

6. REFERENCES

[1] J. Friedman, T. Hastie, and R. Tibshirani, "Sparse inverse covariance estimation with the graphical lasso," Biostatistics 9:3, pp. 432–441, 2008.

[2] V. Chandrasekaran, P. A. Parrilo, and A. S. Willsky, "Latent Variable Graphical Model Selection via Convex Optimization," *Technical report, Massachusetts Institute of Technology,* 2010.

[3] A. Dobra and A. Lenkoski, "Copula Gaussian graphical models and their application to modeling functional disability data," *Annals of Applied Statistics,* vol. 5, No. 2A, pp. 969-993, 2011.

[4] H. Liu, K. Roeder, and L. Wasserman, "Stability Approach to Regularization Selection (StARS) for High Dimensional Graphical Models," *Advances in Neural Information Processing Systems,* 2010.

[5] N. Meinshausen, P. Bühlmann, "Stability Selection," *Journal of the Royal Statistical Society,* vol. 72, Series B, pp. 417-473, 2010.

[6] C. Wang, D. Sun, and K. C. Toh, "Solving log-determinant optimization problems by a Newton-CG primal proximal point algorithm," *Society for Industrial and Applied Mathematics,* vol. 20, pp. 2994-3013, 2009.

[7] H. Liu, J. Lafferty, and L. Wasserman, "The Nonparanormal: Semiparametric Estimation of High Dimensional Undirected Graphs," *Journal of Machine Learning Research,* pp. 2295-2328, 2010.

[8] K. Sachs, O. Perez, D. Peer, D. A. Lauffenburger, and G. P. Nolan "Causal Protein-Signaling Networks Derived from Multiparameter Single-Cell Data," *Science Magazine* vol. 308, pp. 523–529, April 2005.