Journal of the American Statistical Association

ISSN: 0162-1459 (Print) 1537-274X (Online) Journal homepage: http://amstat.tandfonline.com/loi/uasa20

An Equivalent Measure of Partial Correlation Coefficients for High-Dimensional Gaussian Graphical Models

Faming Liang, Qifan Song & Peihua Qiu

To cite this article: Faming Liang, Qifan Song & Peihua Qiu (2015) An Equivalent Measure of Partial Correlation Coefficients for High-Dimensional Gaussian Graphical Models, Journal of the American Statistical Association, 110:511, 1248-1265, DOI: 10.1080/01621459.2015.1012391

To link to this article: http://dx.doi.org/10.1080/01621459.2015.1012391

Accepted author version posted online: 01 Apr 2015.
Published online: 07 Nov 2015.

An Equivalent Measure of Partial Correlation Coefficients for High-Dimensional Gaussian Graphical Models

Faming LIANG, Qifan SONG, and Peihua QIU

Gaussian graphical models (GGMs) are frequently used to explore networks, such as gene regulatory networks, among a set of variables. Under the classical theory of GGMs, the construction of Gaussian graphical networks amounts to finding the pairs of variables with nonzero partial correlation coefficients. However, this is infeasible for high-dimensional problems for which the number of variables is larger than the sample size. In this article, we propose a new measure of partial correlation coefficient, which is evaluated with a reduced conditional set and thus feasible for high-dimensional problems. Under the Markov property and adjacency faithfulness conditions, the new measure of partial correlation coefficient is equivalent to the true partial correlation coefficient in construction of Gaussian graphical networks. Based on the new measure of partial correlation coefficient, we propose a multiple hypothesis test-based method for the construction of Gaussian graphical networks. Furthermore, we establish the consistency of the proposed method under mild conditions. The proposed method outperforms the existing methods, such as the PC, graphical Lasso, nodewise regression, and qp-average methods, especially for the problems for which a large number of indirect associations are present. The proposed method has a computational complexity of nearly $O(p^2)$, and is flexible in data integration, network comparison, and covariate adjustment.

KEY WORDS: Adjacency faithfulness; Gaussian graphical model; Markov property; Multiple hypothesis test; Partial correlation coefficient.

1. INTRODUCTION

Gaussian graphical models (GGMs) have recently become a popular tool to study association networks for a large number of variables, where the variables can refer to genes, proteins, molecules, stocks, or any other subjects depending on the problem under study. The idea underlying GGMs is to use the partial correlation coefficient as a measure of dependency for any two variables. A zero partial correlation coefficient indicates *conditional independence* of the two variables. An alternative measure for the dependency of two variables is correlation coefficient. Compared to the partial correlation coefficient, the correlation coefficient is much weaker as marginally, that is, directly or indirectly, all variables in a system are more or less correlated. Hence, the goal of GGM learning is to distinguish direct from indirect dependencies for all variables in a system. To be more precise, let $\boldsymbol{X} = (X^{(1)}, \ldots, X^{(p)})$ denote a p-dimensional random vector drawn from a multivariate Gaussian distribution $N_p(\boldsymbol{\mu}, \Sigma)$, where $\boldsymbol{\mu}$ and Σ denote the unknown mean and covariance matrix, respectively. A popular way to learn GGMs is covariance selection (Dempster 1972), which is to identify the nonzero entries of the concentration matrix (i.e., inverse of the covariance matrix) because those entries correspond to conditionally dependent variables. Furthermore, Lauritzen (1996) showed that the partial correlation coefficient between $X^{(i)}$ and $X^{(j)}$ given all other variables can be expressed as

$$\rho_{ij|\boldsymbol{V}\setminus\{i,j\}} = -\frac{C_{i,j}}{\sqrt{C_{i,i}C_{j,j}}}, \quad i, j = 1, \ldots, p, \tag{1}$$

where $C_{i,j}$ denotes the (i, j)-entry of the concentration matrix, and $\boldsymbol{V} = \{1, 2, \ldots, p\}$ denotes the set of indices of all variables of a system. Hence, construction of GGMs amounts to estimating their partial correlation coefficients or concentration matrices. However, this approach is not applicable to the problems with $p > n$. In this case, the sample covariance matrix is singular and thus the concentration matrix can no longer be directly estimated.

To tackle this problem, various methods have been proposed in the literature. According to the employed strategies, the existing methods can be roughly grouped into three categories, namely, limited order partial correlations, nodewise regression, and regularized GGMs.

The work belonging to the first category includes Magwene and Kim (2004), Wille and Bühlmann (2006), and Castelo and Roverato (2006, 2009), among others. Magwene and Kim (2004) and Wille and Bühlmann (2006), proposed to use the first-order partial correlation coefficient as a surrogate of the full-order partial correlation coefficient. Castelo and Roverato (2006) proposed a procedure, the so-called qp-procedure, to learn GGMs based on a quantity that they called the nonrejection rate. The nonrejection rate for vertices i and j is the probability of not rejecting the null hypothesis $\rho_{ij|Q} = 0$, where Q is a subset of q variables randomly selected from $\boldsymbol{V} \setminus \{i, j\}$ and q is prespecified by the user. A higher nonrejection rate provides more evidence that an edge is not present in $\boldsymbol{G}$. The qp-procedure proceeds by first estimating the nonrejection rate

Faming Liang is Professor, Department of Biostatistics, University of Florida, Gainesville, FL 32611 (E-mail: *faliang@ufl.edu*). Qifan Song is Assistant Professor, Department of Statistics, Purdue University, West Lafayette, IN 47906 (E-mail: *qfsong@purdue.edu*). Peihua Qiu is Professor, Department of Biostatistics, University of Florida, Gainesville, FL 32611 (E-mail: *pqiu@ufl.edu*).
Liang's research was partially supported by the National Science Foundation grants DMS-1106494 and DMS-1317131. The authors thank the editor, associate editor, and two referees for their constructive comments which have led to significant improvement of this article.

Color versions of one or more of the figures in the article can be found online at *www.tandfonline.com/r/jasa*.

DOI: 10.1080/01621459.2015.1012391

for each pair of vertices based on random samples of Q, and then removing those edges with the nonrejection rates exceeding a given threshold. It is obvious that the performance of the qp-procedure depends on the choice of $q = |Q|$. A larger value of q can lead to a better approximation of the nonrejection rate to the empirical partial correlation coefficient, but compromise the power of the statistical tests which depends on $n - q$. To robustify the qp-procedure, Castelo and Roverato (2009) further proposed an averaging qp-procedure, which is to estimate the nonrejection rate by averaging the estimates obtained from a sequence of q values. Due to their approximation nature, it is widely acknowledged that these methods can result in something in between the full GGM (with correlations conditioned on all $p - 2$ variables) and the correlation graph (with unconditioned correlations). A remarkable algorithm belonging to this category is the PC algorithm (Spirtes et al. 2000), which works in an iterative procedure: It starts with a full graph with edges between all variables, and then for each edge of the current graph, it searches for a subset Q (with $|Q| \le m$, m is prespecified and called the depth of the search) such that the two variables connected by the edge are independent conditional on Q. If such a subset Q is found, then the corresponding edge is removed. Since the PC algorithm essentially uses a $maxP$ statistic (i.e., maximum of a set of p-values) to test for the conditional independence, its power can be very low when p is large. Recently, this algorithm has been extended to the problem of variable selection for high-dimensional linear models (Bühlmann, Kalisch, and Maathuis 2010).

The nodewise regression method was proposed by Meinshausen and Bühlmann (2006) based on the relationship between the partial correlation coefficients and regression coefficients. Consider a linear regression

$$X^{(j)} = \beta_i^{(j)} X^{(i)} + \sum_{r \in \boldsymbol{V} \setminus \{j,i\}} \beta_r^{(j)} X^{(r)} + \epsilon^{(j)}, \tag{2}$$

where $\epsilon^{(j)}$ is a zero-mean Gaussian random error. Then $\beta_i^{(j)} = -C_{j,i}/C_{j,j}$ and $\beta_j^{(i)} = -C_{i,j}/C_{i,i}$; that is,

$$C_{ij} \neq 0 \Longleftrightarrow \rho_{ij|\boldsymbol{V}\setminus\{i,j\}} \neq 0 \Longleftrightarrow \beta_i^{(j)} \neq 0 \text{ and } \beta_j^{(i)} \neq 0. \tag{3}$$

Since p can be larger than n, Meinshausen and Bühlmann (2006) suggested to use the Lasso method (Tibshirani 1996) to identify the nonzero regression coefficients of (2). Let $S^{(j)} = \{r : \beta_r^{(j)} \neq 0\}$ denote the set of explanatory variables identified by Lasso for $X^{(j)}$. The GGM can then be constructed using the "or" rule: *estimate an edge between vertices i and $j \Longleftrightarrow i \in S^{(j)}$ or $j \in S^{(i)}$*, or the "and" rule: *estimate an edge between vertices i and $j \Longleftrightarrow i \in S^{(j)}$ and $j \in S^{(i)}$*. Although, as shown in Meinshausen and Bühlmann (2006), both the "or" and "and" rules are consistent, this method often results in a dense network. This is because the L_1 penalty employed in Lasso shrinks the parameters on true edges toward 0 and, hence, the unexplained signal will pull in other edges that would otherwise not be included.

Instead of working on nodewise regression, Yuan and Lin (2007) proposed to directly estimate the concentration matrix $\boldsymbol{C}$ by minimizing the penalized likelihood function

$$-\log(\det(\boldsymbol{C})) + \operatorname{trace}(\widehat{\Sigma}_{\mathrm{MLE}}\boldsymbol{C}) + \lambda\|\boldsymbol{C}\|, \tag{4}$$

where $\widehat{\Sigma}_{\mathrm{MLE}}$ denotes the maximum likelihood estimator of Σ, $\|\boldsymbol{C}\|$ denotes the norm of $\boldsymbol{C}$, and λ is the regularization parameter. If an L_1-penalty is used, that is, setting $\|\boldsymbol{C}\| = \sum_{i<j} |C_{i,j}|$, the minimization problem in (4) is convex and fast algorithms (Friedman, Hastie, and Tibshirani 2008; Banerjee, Ghaoui, and D'Aspremont 2008) have been developed. The regularization parameter λ can be determined via a cross-validation procedure. Recently, some other approaches, such as stability selection (Liu, Roeder, and Wasserman 2010) and the rotational information criterion (Lysen 2009; Zhao et al. 2012), have also been proposed for the determination of λ. This method is usually referred to as graphical Lasso or, in short, gLasso because of its use of the L_1-penalty. Like nodewise regression, gLasso may inherit the weakness of Lasso; it tends to produce a dense network. In addition, as pointed out by Mazumder and Hastie (2012), convergence of gLasso can be tricky; the converged concentration matrix might not be the inverse of the estimated covariance matrix.

In this article, we propose a new method to learn high-dimensional GGMs based on an equivalent measure of partial correlation coefficients. Let ψ_{ij} denote the equivalent measure of the partial correlation coefficient $\rho_{ij|\boldsymbol{V}\setminus\{i,j\}}$. They are equivalent in the sense that

$$\psi_{ij} = 0 \Longleftrightarrow \rho_{ij|\boldsymbol{V}\setminus\{i,j\}} = 0, \tag{5}$$

provided that the GGM satisfies the Markov property and the adjacency faithfulness condition (defined in Section 2). As a significant advantage of the equivalent measure, ψ_{ij} can be calculated for the case with $p > n$, whereas $\rho_{ij|\boldsymbol{V}\setminus\{i,j\}}$ cannot. Compared to the existing methods, the new method can provide more accurate inference for the underlying graph, especially for the problems for which a large number of indirect associations are present. The new method is computationally efficient, whose computational complexity is nearly $O(p^2)$ for $p > n$, whereas the existing methods have usually a computational complexity of $O(p^3)$ or higher. The new method is also flexible for data integration, network comparison, and covariate adjustment.

The remainder of this article is organized as follows. In Section 2, we give a brief review for the theory of GGMs. In Section 3, we describe the new method and establish its consistency. In Section 4, we illustrate the proposed method using simulated datasets. In Section 5, we apply the proposed method to two real data examples along with comparisons with gLasso and nodewise regression. In Section 6, we illustrate how the proposed method can be used for data integration and network comparison. In Section 7, we analyze the computational complexity of the proposed method. In Section 8, we conclude the article with a brief discussion.

2. GAUSSIAN GRAPHICAL MODELS

This section provides a brief review for the theory of GGMs required by the article. For a full account of graphical model theory, we refer to Lauritzen (1996) and Koller and Friedman (2009).

The GGM can be represented by an undirected graph $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$, where $\boldsymbol{V}$, with a slight abuse of notations, denotes the set of p vertices corresponding to the p variables $X^{(1)}, \ldots, X^{(p)}$, and $\boldsymbol{E} = (e_{ij})$ denotes the adjacency matrix. If two vertices $i, j \in \boldsymbol{V}$ form an edge, then we say that i and j are *adjacent* and set $e_{ij} = 1$. The *boundary set* of a vertex $v \in \boldsymbol{V}$, denoted by $b_{\boldsymbol{G}}$, is the set of vertices adjacent to v, that is, $b_{\boldsymbol{G}}(v) = \{j : e_{vj} = 1\}$.

The *boundary set* is also called the neighborhood or neighboring set in this article. A *path* of length $l > 0$ from v_0 to v_l is a sequence $v_0, v_1, \ldots, v_l$ of distinct vertices such that $e_{v_{k-1},v_k} = 1$ for all $k = 1, \ldots, l$. The subset $\boldsymbol{U} \subset \boldsymbol{V}$ is said to *separate* $\boldsymbol{I} \subset \boldsymbol{V}$ from $\boldsymbol{J} \subset \boldsymbol{V}$ if for every $i \in \boldsymbol{I}$ and $j \in \boldsymbol{J}$, all paths from i to j have at least one vertex in $\boldsymbol{U}$. For a pair of vertices $i \neq j$ with $e_{ij} = 0$, a set $\boldsymbol{U} \subset \boldsymbol{V}$ is called an $\{i, j\}$-*separator* if it separates $\{i\}$ and $\{j\}$ in $\boldsymbol{G}$. Let $\boldsymbol{G}_{ij}$ be a reduced graph of $\boldsymbol{G}$ with e_{ij} being set to zero. Then both the boundary sets $b_{\boldsymbol{G}_{ij}}(i)$ and $b_{\boldsymbol{G}_{ij}}(j)$ are $\{i, j\}$-separators in $\boldsymbol{G}_{ij}$.

Let $X_{\boldsymbol{V}}$ denote a random vector indexed by $\boldsymbol{V} = \{1, \ldots, p\}$ with probability distribution $P_{\boldsymbol{V}}$. Let $\boldsymbol{A} \subset \boldsymbol{V}$ be a subset of $\boldsymbol{V}$, and let $P_{\boldsymbol{A}}$ be the marginal distribution associated with the random vector indexed by $\boldsymbol{A}$. For a triplet $\boldsymbol{I}, \boldsymbol{J}, \boldsymbol{U} \subset \boldsymbol{V}$, we use $X_{\boldsymbol{I}} \perp X_{\boldsymbol{J}} | X_{\boldsymbol{U}}$ to denote that $X_{\boldsymbol{I}}$ is independent of $X_{\boldsymbol{J}}$ conditioned on $X_{\boldsymbol{U}}$.

Definition 1. (Markov property) We say that $P_{\boldsymbol{V}}$ satisfies the *Markov property* with respect to $\boldsymbol{G}$ if for every triple of disjoint sets $\boldsymbol{I}, \boldsymbol{J}, \boldsymbol{U} \subset \boldsymbol{V}$, it holds that $X_{\boldsymbol{I}} \perp X_{\boldsymbol{J}} | X_{\boldsymbol{U}}$ whenever $\boldsymbol{U}$ separates $\boldsymbol{I}$ and $\boldsymbol{J}$ in $\boldsymbol{G}$.

In particular, if $X^{(i)}$ and $X^{(j)}$ are not adjacent in $\boldsymbol{G}$, that is, $e_{ij} = 0$, then $X^{(i)} \perp X^{(j)} | X_{\boldsymbol{V}\setminus\{i,j\}}$.

Definition 2. (Adjacency faithfulness) We say that $P_{\boldsymbol{V}}$ satisfies the adjacency faithfulness condition with respect to $\boldsymbol{G}$: If two variables $X^{(i)}$ and $X^{(j)}$ are adjacent in $\boldsymbol{G}$, then they are dependent conditioned on any subset of $X_{\boldsymbol{V}\setminus\{i,j\}}$.

The adjacency faithfulness condition implies that if there exists a subset $\boldsymbol{U} \subseteq \boldsymbol{V} \setminus \{i, j\}$ such that $X^{(i)} \perp X^{(j)} | X_{\boldsymbol{U}}$, then $X^{(i)}$ and $X^{(j)}$ are not adjacent in $\boldsymbol{G}$. Furthermore, by the Markov property, we have

$$X^{(i)} \perp X^{(j)} | X_{\boldsymbol{U}} \Longrightarrow X^{(i)} \perp X^{(j)} | X_{\boldsymbol{V}\setminus\{i,j\}}, \quad \text{for any } \boldsymbol{U} \subseteq \boldsymbol{V} \setminus \{i, j\}. \tag{6}$$

In particular, if $\boldsymbol{U} = \emptyset$, we have

$$X^{(i)} \text{ and } X^{(j)} \text{ are marginally independent} \Longrightarrow X^{(i)} \perp X^{(j)} | X_{\boldsymbol{V}\setminus\{i,j\}}, \tag{7}$$

or, equivalently,

$$\rho_{ij|\boldsymbol{V}\setminus\{i,j\}} \neq 0 \Longrightarrow \operatorname{corr}\{X^{(i)}, X^{(j)}\} \neq 0. \tag{8}$$

Hence, to infer the conditional independence structure for a GGM, one may perform a correlation screening which can often reduce the dimensionality of the problem by a substantial amount. In this article, we go one step further to show that the correlation screening can be done with a positive threshold value instead of 0, while ensuring (5) holds with ψ_{ij}'s being calculated based on the screened correlation graph.

The adjacency faithfulness yields a graph with fewer edges than without requiring adjacency faithfulness. With adjacency faithfulness, there is no edge between vertices i and j as long as $\rho_{ij|\boldsymbol{U}} = 0$ for some $\boldsymbol{U} \subset \boldsymbol{V} \setminus \{i, j\}$, whereas without adjacency faithfulness, the edge does not exist if and only if $\rho_{ij|\boldsymbol{V}\setminus\{i,j\}} = 0$.

The adjacency faithfulness condition is strictly weaker than the faithfulness condition on which the PC algorithm is based (Ramsey, Zhang, and Spirtes 2006). The latter states that for every triple of disjoint sets $\boldsymbol{I}, \boldsymbol{J}, \boldsymbol{U} \subset \boldsymbol{V}$, it holds $\boldsymbol{U}$ separates $\boldsymbol{I}$ and $\boldsymbol{J} \Longleftrightarrow X_{\boldsymbol{I}} \perp X_{\boldsymbol{J}} | X_{\boldsymbol{U}}$. Zhang and Spirtes (2008) discuss the relationship between faithfulness and adjacency faithfulness. The faithfulness consists of two components, adjacency faithfulness and orientation faithfulness (but they together do not imply the faithfulness). The correctness of the PC algorithm only depends on the truth of the adjacency faithfulness and orientation faithfulness conditions; the former is used for inferring adjacencies and the latter for inferring edge orientations. Note that the PC algorithm is originally designed for inference of directed acyclic graphs (DAGs) for related Gaussian random variables. The validity of the faithfulness condition is supported by the Lebesgue measure zero argument (Meek 1995); that is, the problems that violate the faithfulness condition usually correspond to some particular parameter values that form a zero measure set in the space of all possible parameterizations. Assuming the Markov property and the adjacency faithfulness condition, any violation of the orientation faithfulness condition is detectable. Recently, Lemeire et al. (2012) studied the cases of adjacency faithfulness violations, and gave a remedial strategy for certain types of detectable violations. In general, undetectable violations of adjacency faithfulness are due to cancellations of multiple causal paths, see Zhang and Spirtes (2008) for some examples. Since the exact cancellation rarely occurs, adjacency faithfulness is plausible and this is particularly true when the underlying causal structure is sparse.

In this article, we assume that the Markov property and the adjacency faithfulness condition hold for the GGMs under study as they are undirected. These assumptions are essentially the same as those used in the PC algorithm for inferring adjacencies of the Gaussian DAG.

3. AN EQUIVALENT MEASURE OF PARTIAL CORRELATION COEFFICIENTS

In this section, we first propose an equivalent measure of partial correlation coefficients for GGMs under the assumption of the Markov property and adjacency faithfulness, then describe a multiple hypothesis testing-based algorithm for learning GGMs based on the equivalent measure of partial correlation coefficients, and finally establish the consistency of the proposed algorithm.

3.1 The Equivalent Measure of Partial Correlation Coefficients

Let r_{ij} denote the correlation coefficient of variables $X^{(i)}$ and $X^{(j)}$. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ denote the correlation graph of $X^{(1)}, \ldots, X^{(p)}$, where $\mathcal{E} = (\tilde{e}_{ij})$ is the adjacency matrix with $\tilde{e}_{ij} = 1$ if $|r_{ij}| > 0$ and 0 otherwise. Let $\hat{r}_{ij}$ denote the empirical correlation coefficient of $X^{(i)}$ and $X^{(j)}$, let γ_i denote a threshold value, and let $\hat{\mathcal{E}}_{\gamma_i,i} = \{v : |\hat{r}_{iv}| > \gamma_i\}$ denote a reduced neighborhood of node i in the empirical correlation graph. For convenience, we define $\hat{\mathcal{E}}_{\gamma_j,j} = \{v : |\hat{r}_{jv}| > \gamma_j\}$, $\hat{\mathcal{E}}_{\gamma_i,i,-j} = \{v : |\hat{r}_{iv}| > \gamma_i\} \setminus \{j\}$, and $\hat{\mathcal{E}}_{\gamma_j,j,-i} = \{v : |\hat{r}_{jv}| > \gamma_j\} \setminus \{i\}$. For any pair of vertices i and j, we define the partial correlation coefficient ψ_{ij} by

$$\psi_{ij} = \rho_{ij|S_{ij}}, \tag{9}$$

where $S_{ij} = \hat{\mathcal{E}}_{\gamma_i,i,-j}$ if $|\hat{\mathcal{E}}_{\gamma_i,i,-j}| < |\hat{\mathcal{E}}_{\gamma_j,j,-i}|$ and $S_{ij} = \hat{\mathcal{E}}_{\gamma_i,i,-j}$ otherwise, and $|D|$ denotes the cardinality of the set D. To dis-

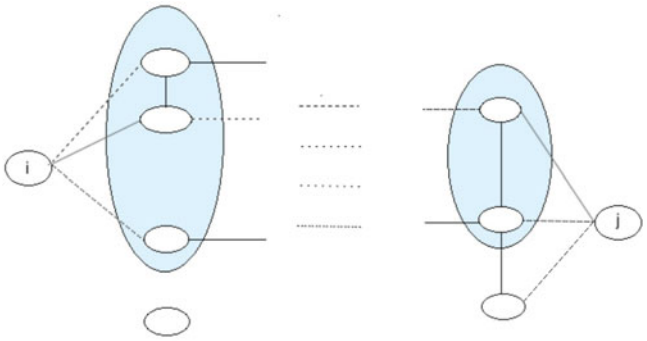

Figure 1. Illustrative plot for the calculation of ψ-partial correlation coefficients, where the solid and dotted edges indicate the direct and indirect associations, respectively. The left and right shaded ellipses cover, respectively, the reduced neighborhoods of node i and node j in the correlation graph.

tinguish ψ_{ij} from conventional partial correlation coefficients, we call it the ψ-partial correlation coefficient in this article. Figure 1 illustrates the calculation of ψ_{ij}.

Let $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ denote the true conditional independence graph, and let $b_{\boldsymbol{G}}(i)$ denote the neighborhood of node i in $\boldsymbol{G}$. Theorem 1 shows that ψ_{ij} is an equivalent measure of the partial correlation coefficient $\rho_{ij|\boldsymbol{V}\setminus\{i,j\}}$ under appropriate conditions.

Theorem 1. Suppose that a GGM $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ satisfies the conditions of Markov property and adjacency faithfulness, and $b_{\boldsymbol{G}}(i) \subseteq \hat{\mathcal{E}}_{\gamma_i,i}$ holds for each node i. Then, ψ_{ij} defined in (9) is an equivalent measure of the partial correlation coefficient $\rho_{ij|\boldsymbol{V}\setminus\{i,j\}}$ in the sense that

$$\psi_{ij} = 0 \Longleftrightarrow \rho_{ij|\boldsymbol{V}\setminus\{i,j\}} = 0.$$

Proof. ($\Longleftarrow$) If $\rho_{ij|\boldsymbol{V}\setminus\{i,j\}} = 0$, then $e_{ij} = 0$. In this case, by the assumption $b_{\boldsymbol{G}}(i) \subseteq \hat{\mathcal{E}}_{\gamma_i,i}$, S_{ij} forms a separator for i and j in $\boldsymbol{G}_{ij}$. Then, by the Markov property, we have $\psi_{ij} = 0$.

($\Longrightarrow$) If $\psi_{ij} = 0$, then by (6), we have $\rho_{ij|\boldsymbol{V}\setminus\{i,j\}} = 0$. The proof is completed. □

Note that we have many ways to specify the separator S_{ij}. For example, we can set S_{ij} to $\hat{\mathcal{E}}_{\gamma_i,i,-j}$ or $\hat{\mathcal{E}}_{\gamma_j,j,-i}$ for which the cardinality is larger, or even set $S_{ij} = \hat{\mathcal{E}}_{\gamma_i,i,-j} \cup \hat{\mathcal{E}}_{\gamma_j,j,-i}$. Then, it is easy to show that ψ_{ij} is still an equivalent measure of $\rho_{ij|\boldsymbol{V}\setminus\{i,j\}}$ under these settings. However, since ψ_{ij} has an approximate variance of $1/(n - |S_{ij}| - 3)$, we prefer to set S_{ij} to $\hat{\mathcal{E}}_{\gamma_i,i,-j}$ or $\hat{\mathcal{E}}_{\gamma_j,j,-i}$ for which the cardinality is smaller.

3.2 Learning the Structure of GGMs

Based on the equivalent measure of partial correlation coefficients, the GGMs can be learned in the following algorithm, the so-called ψ-learning algorithm in this article.

Algorithm 1. (ψ-Learning)

(a) (Correlation screening) Determine the reduced neighborhood $\hat{\mathcal{E}}_{\gamma_i,i}$ for each variable $X^{(i)}$.
   (i) Conduct a multiple hypothesis test to identify the pairs of vertices for which the empirical correlation coefficient is significantly different from zero. This step results in a so-called empirical correlation network.
   (ii) For each variable $X^{(i)}$, identify its neighborhood in the empirical correlation network, and reduce the size of the neighborhood to $O(n/\log(n))$ by removing the variables having a lower correlation (in absolute value) with $X^{(i)}$. This step results in a so-called reduced correlation network.

(b) (ψ-calculation) For each pair of vertices i and j, identify the separator S_{ij} based on the reduced correlation network resulted in step (a) and calculate ψ_{ij} by inverting the subsample covariance matrix indexed by the variables in $S_{ij} \cup \{i, j\}$.

(c) (ψ-screening) Conduct a multiple hypothesis test to identify the pairs of vertices for which ψ_{ij} is significantly different from zero, and set the corresponding elements of $\boldsymbol{E}$ to be 1.

On the bound of the neighborhood size, we suggest to set it as $n/[\xi_n \log(n)]$, where ξ_n is a tunable parameter and has a default value of 1. For some problems, if n is too large, one may set $\xi_n > 1$, say, 2 or 3; if n is too small, one may set $\xi_n < 1$, say, 1/2 or 1/3, while ensuring the condition $n/[\xi_n \log(n)] < n - 4$ holds. The ψ-learning algorithm is very convenient for incorporating our prior knowledge into network construction. For example, if we know some pair of variables, say, $X^{(i)}$ and $X^{(k)}$, are correlated, then we can always include $X^{(k)}$ into the set $\hat{\mathcal{E}}_{\gamma_i,i}$ and include $X^{(i)}$ into the set $\hat{\mathcal{E}}_{\gamma_k,k}$, even if the empirical correlation between $X^{(i)}$ and $X^{(k)}$ is not strong enough.

The multiple hypothesis tests required in Steps (a) and (c) can be done as follows. First, we apply Fisher's transformation to $\hat{r}_{ij}$ to get

$$z_{ij} = \frac{1}{2}\log\left[\frac{1+\hat{r}_{ij}}{1-\hat{r}_{ij}}\right],$$

whose distribution, under the null hypothesis $H_0 : r_{ij} = 0$, is well approximated by a normal distribution with mean 0 and variance $1/(n-3)$. Based on this asymptotic result, we calculate the p-value for the test $H_0 : r_{ij} = 0 \leftrightarrow H_1 : r_{ij} \neq 0$, and then apply the probit transformation to the p-value to get

$$\begin{aligned}\tilde{z}_{ij} &= \Phi^{-1}\left(1 - 2\left[1 - \Phi\left(\sqrt{n-3}|z_{ij}|\right)\right]\right) \\ &= \Phi^{-1}\left(2\Phi\left(\sqrt{n-3}|z_{ij}|\right) - 1\right), \end{aligned} \tag{10}$$

where $\Phi(\cdot)$ denotes the cumulative distribution function (CDF) of the standard normal distribution. For convenience, we call $\tilde{z}_{ij}$ a correlation score in this article. Due to the monotonicity of the transformation (10) in $|z_{ij}|$, it converts a double-sided test $H_0 : z_{ij} = 0 \leftrightarrow H_1 : z_{ij} \neq 0$ to a single-sided test $H_0 : \tilde{z}_{ij} = 0 \leftrightarrow H_1 : \tilde{z}_{ij} > 0$. This would be more convenient for the multiple hypothesis testing procedure employed in this article.

In this article, we adopted a generalized empirical Bayesian method developed by Liang and Zhang (2008) for conducting the multiple hypothesis tests required in Steps (a) and (c) of the ψ-learning algorithm. Appendix A.1 gives a brief review of this method. An attractive feature of this method is that it allows for the general dependence between test statistics. Other multiple hypothesis testing methods which account for the dependence

between test statistics, for example, Benjamini, Krieger, and Yekutieli (2006), can also be applied here.

Similarly, we let $\hat{\psi}_{ij}$ denote the empirical value of ψ_{ij}. Applying Fisher's transformation to $\hat{\psi}_{ij}$, we get

$$z'_{ij} = \frac{1}{2}\log\left[\frac{1+\hat{\psi}_{ij}}{1-\hat{\psi}_{ij}}\right], \tag{11}$$

which, under the hypothesis $H_0 : \psi_{ij} = 0$, is approximately distributed as a normal distribution with mean 0 and variance $1/(n - |S_{ij}| - 3)$. Then, we calculate the p-value for the corresponding test and apply the probit transformation to the p-value to get

$$\tilde{z}'_{ij} = \Phi^{-1}\left(2\Phi\left(\sqrt{n-|S_{ij}|-3}|z'_{ij}|\right)-1\right), \tag{12}$$

which, for convenience, is called a ψ-score in this article. Furthermore, we call $n - |S_{ij}| - 3$ the effective sample size of the ψ-score. The multiple testing procedure described in Appendix A.1 can then be applied to the ψ-scores to identify the partial correlations that are significantly different from zero.

As described in Appendix A.1, the multiple hypothesis testing procedure used in this article employs a stochastic approximation procedure to fit a mixture exponential power distribution to the test scores (correlation scores or ψ-scores). To achieve convergence of the stochastic approximation procedure, it usually needs 100∼500 iterations, with the whole set of test scores scanned once for each iteration. We set 100 as the default value, but we do urge the user to conduct a diagnosis for the convergence of the stochastic approximation procedure, for example, by visualizing the model fitting to the histogram of test scores, as shown in Figure 6. This is because the stochastic approximation, like the Newton–Raphson algorithm, will converge to a local optimal solution near the starting point. When p is very large, say, greater than 10,000, the computation can be very slow. Note that the total number of test scores increases in p^2. To accelerate the computation, we propose to work on a subset of test scores for determining the cutoff value of the multiple hypothesis test. For example, the subset can be drawn from the whole set of test scores using a stratified sampling scheme. The details of the sampling scheme are given in Appendix A.2. With the subset samples, the computational time can be reduced substantially.

One important issue related to the implementation of the multiple hypothesis tests is the choice of significance levels, which are denoted by α_1 and α_2 for those used in the correlation screening and ψ-screening steps, respectively. As described in Appendix A.1, the significance levels can be set in Storey's q-value (Storey 2002). For correlation screening, we generally recommend a moderate significance level, for example, 0.05, 0.1, or 0.2. A large value of α_1 will increase the size of the neighborhood $\hat{\mathcal{E}}_{\gamma_i,i}$ for each node i, and so does the chance $b_{\boldsymbol{G}}(i) \subseteq \hat{\mathcal{E}}_{\gamma_i,i}$ as required by Theorem 1. On the other hand, an excessively large set of $\hat{\mathcal{E}}_{\gamma_i,i}$ will reduce the effective sample size $n - |\hat{\mathcal{E}}_{\gamma_i,i}| - 3$ of the conditional independence test and thus lower its power. In this article, we set 0.05 as the default value of α_1. Otherwise, it will be stated in the context. For ψ-screening, we generally recommend a low significance level, for example, 0.01, but it is very much user-dependent. With different values of α_2, a network path can be constructed. This is similar to gLasso for which a network path can be constructed with different values of the regularization parameter.

3.3 Consistency of the ψ-Learning Algorithm

Assume that the observations $X_1, \ldots, X_n$ are independently identically distributed (iid) with $X_i \in \mathbb{R}^p$ having a probability distribution P. To indicate that the dimension p can grow as a function of the sample size n, we will rewrite the dimension p as p_n, the distribution P as $P^{(n)}$, and the true conditional independence graph $\boldsymbol{G}$ as $\boldsymbol{G}^{(n)} = (\boldsymbol{V}^{(n)}, \boldsymbol{E}^{(n)})$. Let $\mathcal{G}^{(n)} = (\mathcal{V}^{(n)}, \mathcal{E}^{(n)})$ denote the true correlation graph, where $\mathcal{V}^{(n)} = \boldsymbol{V}^{(n)}$. Let γ_n denote a threshold value of the empirical correlation coefficient. With a slight abuse of notation, we let $\hat{\mathcal{E}}_{\gamma_n}$ denote the edge set of the network obtained through correlation thresholding at γ_n, and let $\hat{\mathcal{E}}_{\gamma_n,i}$ denote the neighborhood of node i in $\hat{\mathcal{E}}_{\gamma_n}$. That is, we define

$$\begin{aligned} \hat{\mathcal{E}}_{\gamma_n} &= \{(i,j) : |\hat{r}_{ij}| > \gamma_n\}, \\ \text{and} \quad \hat{\mathcal{E}}_{\gamma_n,i} &= \{j : j \neq i, |\hat{r}_{ij}| > \gamma_n\}. \end{aligned} \tag{13}$$

For convenience, we call the network with the edge set $\hat{\mathcal{E}}_{\gamma_n}$ the thresholding correlation network. Similar to (13), we define

$$\begin{aligned} \widetilde{\boldsymbol{E}}^{(n)} &= \{(i,j) : \rho_{ij|V\setminus\{i,j\}} \neq 0,\ i,j = 1,\ldots,p_n\}, \\ \widetilde{\mathcal{E}}^{(n)} &= \{(i,j) : r_{ij} \neq 0,\ i,j = 1,\ldots,p_n\}, \end{aligned} \tag{14}$$

as the edge sets of $\boldsymbol{G}^{(n)}$ and $\mathcal{G}^{(n)}$, respectively.

To establish the consistency of the ψ-learning method, we first make the following assumptions:

(A_1) The distribution $P^{(n)}$ satisfies the following conditions:

(i) $P^{(n)}$ is multivariate Gaussian.
(ii) $P^{(n)}$ satisfies the Markov property and adjacency faithfulness condition with respect to the undirected graph $\boldsymbol{G}^{(n)}$ for all $n \in \mathbb{N}$.

(A_2) The dimension $p_n = O(\exp(n^\delta))$ for some constant $0 \leq \delta < 1$.

(A_3) The correlation satisfies

$$\min\{|r_{ij}|; r_{ij} \neq 0,\ i,j = 1,2,\ldots,p_n,\ i \neq j\} \geq c_0 n^{-\kappa}, \tag{15}$$

for some constants $c_0 > 0$ and $0 < \kappa < (1-\delta)/2$, and

$$\max\{|r_{ij}|; i,j = 1,\ldots,p_n, i \neq j\} \leq M_r < 1, \tag{16}$$

for some constant $0 < M_r < 1$.

Assumption (A_1) is often used in graphical modeling, although the adjacency faithfulness condition slightly restricts the class of probability distributions. Assumption (A_2) allows an exponential growth of dimension as a function of sample size. Assumption (A_3) ensures detectability of nonzero correlation, and restricts the linear dependency among the variables by requiring an upper bound $0 < M_r < 1$. It follows from (8) that $\widetilde{\boldsymbol{E}}^{(n)} \subseteq \widetilde{\mathcal{E}}^{(n)}$. Furthermore, it follows from (15) that there exist constants $c_1 > 0$ and $0 < \kappa' \leq \kappa$ such that

$$\min\{|r_{ij}|; (i,j) \in \widetilde{\boldsymbol{E}}^{(n)}, i,j = 1,\ldots,p_n\} \geq c_1 n^{-\kappa'}. \tag{17}$$

This result is quite understandable, as the directly dependent variables, that is, those connected by edges in $\boldsymbol{G}^{(n)}$, tend to have higher correlations than the indirectly dependent variables.

See, for example, Witten, Friedman, and Simon (2011) and Mazumder and Hastie (2012) for more exploration of this issue.

Lemma 1 concerns the sure screening property of the thresholding correlation network, which is slightly modified from Luo, Song, and Witten (2015). In Luo, Song, and Witten (2015), the proof of the lemma is based on (A_1)-(i), (A_2), and (17) without assuming the adjacency faithfulness condition.

Lemma 1. Assume (A_1), (A_2), and (A_3) hold. Let $\gamma_n = 2/3c_1n^{-\kappa'}$. Then there exist constants c_2 and c_3 such that

$$P\left(\widetilde{\boldsymbol{E}}^{(n)} \subseteq \hat{\mathcal{E}}_{\gamma_n}\right) \geq 1 - c_2\exp\left(-c_3n^{1-2\kappa'}\right),$$
$$P\left(b_{\boldsymbol{G}^{(n)}}(i) \subseteq \hat{\mathcal{E}}_{\gamma_n,i}\right) \geq 1 - c_2\exp\left(-c_3n^{1-2\kappa'}\right).$$

Lemma 1 shows that if the correlation between any two directly dependent random variables is detectable, then as the sample size n becomes large, the edge set $\widetilde{\boldsymbol{E}}^{(n)}$ is almost surely contained in the edge set of the thresholding correlation network. This implies that the ψ-partial correlation coefficient can be evaluated based on the thresholding correlation network, while ensuring its equivalence to the full conditional partial correlation coefficient.

Based on an additional assumption on the covariance matrix Σ, Lemma 2 gives a probabilistic upper bound for the neighborhood size of the thresholding correlation network. Lemma 2 is slightly modified from Theorem 2 of Luo, Song, and Witten (2015). Our modification ensures sparsity of the thresholding correlation network.

(A_4) There exist constants $c_4 > 0$ and $0 \leq \tau < 1 - 2\kappa'$ such that $\lambda_{\max}(\Sigma) \leq c_4n^{\tau}$, where Σ denotes the covariance matrix of X_i and $\lambda_{\max}(\Sigma)$ is the largest eigenvalue of Σ.

Lemma 2. Assume (A_1), (A_2), (A_3), and (A_4) hold. Let $\gamma_n = 2/3c_1n^{-\kappa'}$. Then, for each node i,

$$P\left[|\hat{\mathcal{E}}_{\gamma_n,i}| \leq O(n^{2\kappa'+\tau})\right] \geq 1 - c_2\exp\left(-c_3n^{1-2\kappa'}\right),$$

where c_2 and c_3 are as given in Lemma 1.

Assumption (A_4) indicates that the largest eigenvalue of Σ is allowed to grow with n, but the growth rate should be restricted. Otherwise, the resulting thresholding correlation network can be dense. With Lemma 2, the ψ-learning method has successfully reduced the problem of GGM learning from a high-dimensional setting (with $n < p$) to a low-dimensional setting (with $n > |\hat{\mathcal{E}}_{\gamma_n,i}|$). Since the exact value of $2\kappa' + \tau$ is unknown, we may bound the neighborhood size by $O(n/\log(n))$ in practice. However, when n is large, $n/\log(n)$ can be too large. An excessively large neighborhood will reduce the effective sample size of the ψ-partial correlation coefficient, and thus affect the performance of the algorithm adversely. To address this issue, we propose a multiple hypothesis test-based procedure, that is, step (a)-(i) of Algorithm 1, for preidentification of the nonzero correlation coefficients. To justify this procedure, we have the following lemmas.

Lemma 3 concerns uniform consistency of the estimated correlation coefficient, which is adopted from Bühlmann and van de Geer (2011).

Lemma 3. (Lemma 13.1, Bühlmann and van de Geer 2011) Assume (A_1)-(i) and condition (16) in (A_3). Then, for any $0 < \gamma < 2$,

$$\sup_{i,j\in\{1,\ldots,p_n\}} P[|\hat{r}_{ij} - r_{ij}| > \gamma] \leq c_5(n-2) \times \exp\left\{(n-4)\log\left(\frac{4-\gamma^2}{4+\gamma^2}\right)\right\},$$

for some constant $0 < c_5 < \infty$ depending on M_r in (A_2) only.

Lemma 4 shows that $\hat{\mathcal{E}}_{\eta_n}$ can be a consistent estimator of $\widetilde{\mathcal{E}}^{(n)}$ with an appropriate value of η_n.

Lemma 4. Assume (A_1)-(i), (A_2), and (A_3). If $\eta_n = 1/2c_0n^{-\kappa}$, then

$$P[\hat{\mathcal{E}}_{\eta_n} = \widetilde{\mathcal{E}}^{(n)}] = 1 - o(1), \quad \text{as } n \to \infty.$$

Proof. Let A_{ij} denote that an error event occurs when testing the hypotheses $H_0 : r_{ij} = 0$ versus $H_1 : r_{ij} \neq 0$ for variables i and j. Thus,

$$P[\text{an error occurs in } \hat{\mathcal{E}}_{\eta_n}] = P\left[\cup_{i\neq j}A_{ij}\right] \leq O(p_n^2)\sup_{i\neq j}P(A_{ij}). \tag{18}$$

Let A_{ij}^I and A_{ij}^{II} denote the false positive and false negative errors, respectively. Then,

$$A_{ij} = A_{ij}^I \cup A_{ij}^{II}, \tag{19}$$

where

$$\begin{cases} \text{false positive error } A_{ij}^I : \ |\hat{r}_{ij}| > \dfrac{c_0}{2}n^{-\kappa} \quad \text{and } r_{ij} = 0, \\ \text{false negative error } A_{ij}^{II} : \ |\hat{r}_{ij}| \leq \dfrac{c_0}{2}n^{-\kappa} \quad \text{and } r_{ij} \neq 0. \end{cases} \tag{20}$$

Then, there exists some constant $0 < C < \infty$,

$$\sup_{ij} P\left(A_{ij}^I\right) = \sup_{ij} P\left(|\hat{r}_{ij} - r_{ij}| > \frac{c_0}{2}n^{-\kappa}\right) \leq O(n)\exp\left(-Cn^{1-2\kappa}\right), \tag{21}$$

using Lemma 3 and the fact that $\log((4-a^2)/(4+a^2)) \sim -a^2/2$ as $a \to 0$. Furthermore,

$$\sup_{ij} P\left(A_{ij}^{II}\right) = \sup_{ij} P\left(|\hat{r}_{ij}| \leq \frac{c_0}{2}n^{-\kappa}\right) \leq \sup_{ij} P\left(|\hat{r}_{ij} - r_{ij}| > \frac{c_0}{2}n^{-\kappa}\right), \tag{22}$$

by (15) that $\min_{ij}|r_{ij}| \geq c_0n^{-\kappa}$ in this case. By Lemma 3, we have

$$\sup_{ij} P\left(E_{ij}^{II}\right) \leq O(n)\exp\left(-Cn^{1-2\kappa}\right), \tag{23}$$

for some $0 < C < \infty$. As a summary of (18)–(23), we have

$$P[\text{an error occurs in } \hat{\mathcal{E}}_{\eta_n}] \leq O\left(p_n^2n\right)\exp\left(-Cn^{1-2\kappa}\right) = o(1), \tag{24}$$

because it is assumed in (A_3) that $0 < \kappa < (1-\delta)/2$, and in (A_2) that $\log(p_n) = n^{\delta}$ for some $0 < \delta < 1$. This concludes the proof. □

It follows from (8) that $\widetilde{\boldsymbol{E}}^{(n)} \subseteq \widetilde{\mathcal{E}}^{(n)}$. Furthermore, it follows from Lemma 4 that

$$P[\widetilde{\boldsymbol{E}}^{(n)} \subseteq \hat{\mathcal{E}}_{\eta_n}] = 1 - o(1). \tag{25}$$

Therefore, based on Lemma 1, Lemma 2, and (25), we propose to restrict the neighborhood size of each node to be

$$\min\left\{|\hat{\mathcal{E}}_{\eta_n,i}|, \frac{n}{\xi_n \log(n)}\right\}, \tag{26}$$

where ξ_n is as defined in Section 3.2. The value of η_n can be determined through a simultaneous test for the hypotheses $H_0 : r_{ij} = 0 \leftrightarrow H_1 : r_{ij} \neq 0$, $1 \leq i < j \leq p_n$, at a significance level of α_1. Our experience shows that the rule (26) can perform much better than the rule $n/[\xi_n \log(n)]$, especially when n is large.

Let ζ_n denote the threshold value of ψ-partial correlation used in the ψ-screening step, and let $\hat{\boldsymbol{E}}_{\zeta_n}$ denote the partial correlation network obtained through thresholding ψ-partial correlation. That is, we define

$$\hat{\boldsymbol{E}}_{\zeta_n} = \{(i, j) : |\hat{\psi}_{ij}| > \zeta_n,\ i, j = 1, 2, \ldots, p_n\}.$$

To establish the consistency of $\hat{\boldsymbol{E}}_{\zeta_n}$, we make the following assumption, which ensures detectability of nonzero ψ-partial correlations and restricts the linear dependence by requiring an upper bound $0 < M_\psi < 1$.

(A_5) The ψ-partial correlation coefficients satisfy

$$\inf\{\psi_{ij}; \psi_{ij} \neq 0,\ i, j = 1, \ldots, p_n,\ i \neq j,\ |S_{ij}| \leq q_n\} \geq c_6 n^{-d},$$

where $q_n = O(n^{2\kappa'+\tau})$, $0 < c_6 < \infty$ and $0 < d < (1 - \delta)/2$ are some constants. In addition,

$$\sup\{\psi_{ij}; i, j = 1, \ldots, p_n,\ i \neq j, |S_{ij}| \leq q_n\} \leq M_\psi < 1,$$

for some constant $0 < M_\psi < 1$.

Lemma 5 concerns uniform consistency of the estimated ψ-correlation coefficient, whose proof follows from the proof of Lemma 13.1 of Bühlmann and van de Geer (2011) for the correlation coefficient.

Lemma 5. Assume (A_1)-(i) and (A_5) hold, and $q_n < n - 4$. Let $q_0 = \min_{(i,j)} |S_{ij}|$. Then, for any $0 < \zeta < 2$,

$$\sup_{i,j\in\{1,\ldots,p_n\}} P[|\hat{\psi}_{ij} - \psi_{ij}| > \zeta] \leq c_7(n - q_0 - 2) \times \exp\left\{(n - q_0 - 4)\log\left(\frac{4 - \zeta^2}{4 + \zeta^2}\right)\right\},$$

for some constant $0 < c_7 < \infty$ depending on M_ψ in (A_5) only.

Let $\hat{\mathcal{E}}_*$ denote the edge set of a correlation network for which each node has a degree of $O(n/\log(n))$, adjacent with $O(n/\log(n))$ highest correlated nodes. It follows from Lemma 2 and (A_2) that

$$P[\widetilde{\boldsymbol{E}}^{(n)} \subseteq \hat{\mathcal{E}}_*] \geq 1 - c_2 p_n \exp(-c_3 n^{1-2\kappa'}) = 1 - o(1). \tag{27}$$

Lemma 6 establishes the consistency of $\hat{\boldsymbol{E}}_{\zeta_n}$ conditioned on $\widetilde{\boldsymbol{E}}^{(n)} \subseteq \hat{\mathcal{E}}_* \cap \hat{\mathcal{E}}_{\eta_n}$. Its proof follows closely the proof of Lemma 4 and is thus omitted here.

Lemma 6. Assume (A_1)–(A_5) hold and $\widetilde{\boldsymbol{E}}^{(n)} \subseteq \hat{\mathcal{E}}_* \cap \hat{\mathcal{E}}_{\eta_n}$ is true. Let $\zeta_n = \frac{1}{2}c_6 n^{-d}$, then

$$P\left[\hat{\boldsymbol{E}}_{\zeta_n} = \widetilde{\boldsymbol{E}}^{(n)} | \widetilde{\boldsymbol{E}}^{(n)} \subseteq \hat{\mathcal{E}}_* \cap \hat{\mathcal{E}}_{\eta_n}\right] = 1 - o(1), \quad \text{as } n \to 1.$$

Theorem 2 establishes the consistency of the ψ-learning algorithm.

Theorem 2. Consider a GGM with distribution $P^{(n)}$ and underlying conditional independence graph $\boldsymbol{G}^{(n)}$. Assume (A_1)–(A_5) hold. Then

$$P[\hat{\boldsymbol{E}}_{\zeta_n} = \widetilde{\boldsymbol{E}}^{(n)}] \geq 1 - o(1), \quad \text{as } n \to \infty.$$

Proof. By invoking (25), (27), and Lemma 6, we have

$$\begin{aligned} P\left[\hat{\boldsymbol{E}}_{\zeta_n} = \widetilde{\boldsymbol{E}}^{(n)}\right] &\geq P\left[\hat{\boldsymbol{E}}_{\zeta_n} = \widetilde{\boldsymbol{E}}^{(n)} | \widetilde{\boldsymbol{E}}^{(n)} \subseteq \hat{\mathcal{E}}_* \cap \hat{\mathcal{E}}_{\eta_n}\right] \\ &\quad \times P\left[\widetilde{\boldsymbol{E}}^{(n)} \subseteq \hat{\mathcal{E}}_* \cap \hat{\mathcal{E}}_{\eta_n}\right] \\ &\geq [1 - o(1)][1 - o(1) + 1 - o(1) - 1] \\ &= 1 - o(1), \end{aligned}$$

which concludes the proof. □

Theorem 2 is proved with a constant threshold value of ζ_n. However, since each ψ_{ij} has a different effective sample size, that is, $n - |S_{ij}| - 3$ varies with the size of the separator S_{ij}, we may vary the threshold value for different ψ_{ij} by adjusting with its effective sample size. In our implementation, the adjustment is done via ψ-score transformation as described in (11) and (12). This adjustment can generally improve the performance of the ψ-learning algorithm. Finally, we note that the strategy of the proof of Theorem 2 is very much inspired by the analysis in Kalisch and Bühlmann (2007), which also appeared in the book by Bühlmann and van de Geer (2011), for the PC algorithm.

4. AN ILLUSTRATIVE EXAMPLE

In this section, we compare the performance of the ψ-learning algorithm with the PC (Spirtes, Glymour, and Scheines 2000), gLasso (Friedman, Hastie, and Tibshirani 2008), nodewise regression (Meinshausen and Bühlmann 2006), and qp-average (Castelo and Roverato 2009) methods on some simulated datasets. The latter four methods represent the state-of-the-art methods for GGM learning. For gLasso and nodewise regression, we adopted their implementations in the R package *huge* (Zhao et al. 2012), which are known to be efficient and automatic for the determination of the regularization parameter. For qp-average, we adopted its implementation in the R package *qpgraph* (Castelo and Roverato 2009). For this method, we set the q-sequence to be (1, 10, 20, 30, 40, 50) for all datasets studied in this section. Under this setting, the nonrejection rate is calculated by averaging the estimates obtained with $q = 1$, $10, \ldots, 50$. For the PC algorithm, we adopted its implementation in the R package *pcalg*. This algorithm includes a free parameter, denoted by α_{pc}, which represents the Type I error of each individual conditional independence test. To distinguish GGMs from correlation networks, we have also included the correlation method for comparison.

The simulated example is an autoregressive process of order two with the concentration matrix given by

$$C_{ij} = \begin{cases} 0.5, & \text{if } |j - i| = 1, i = 2, \ldots, (p - 1), \\ 0.25, & \text{if } |j - i| = 2, i = 3, \ldots, (p - 2), \\ 1, & \text{if } i = j, i = 1, \ldots, p, \\ 0, & \text{otherwise.} \end{cases} \tag{28}$$

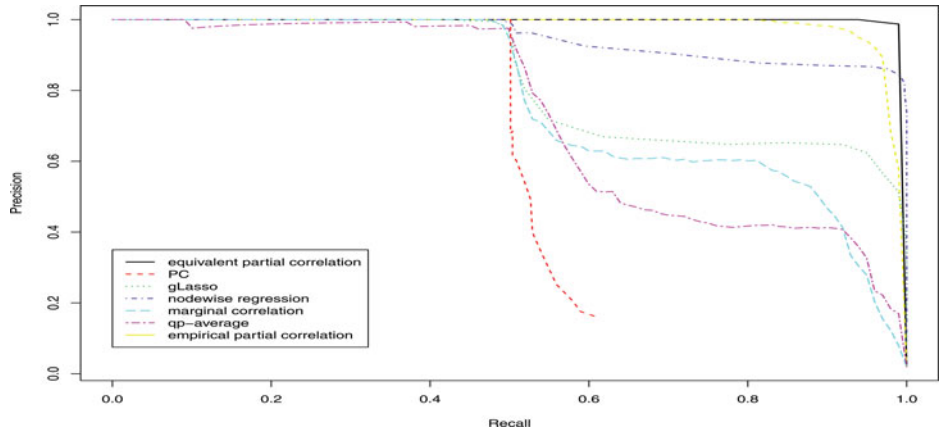

Figure 2. PR curves of the ψ-learning, PC, gLasso, nodewise regression, qp-average, empirical partial correlation, and marginal correlation methods for one autoregressive dataset simulated under the setting $(n, p) = (500, 200)$.

This example was used by Yuan and Lin (2007) and Mazumder and Hastie (2012) to illustrate their gLasso algorithms. We are interested in it because for which all the variables are dependent, either directly or indirectly, and thus it can serve as a good example for testing whether or not the ψ-learning method can distinguish direct dependencies from indirect ones. As previously defined, two variables are called directly dependent if their partial coefficient is nonzero, and indirectly dependent or independent otherwise.

We consider two settings of this example with $(n, p) = (500, 200)$ and $(100, 200)$, respectively. Under each setting, we simulated 10 datasets independently. For the setting $(n, p) = (500, 200)$, the empirical partial correlation coefficients can be directly estimated by inverting the sample covariance matrix. Hence, this method is also included for comparison under this setting.

Figure 2 shows the precision–recall (PR) curves of the ψ-learning, gLasso, nodewise regression, PC, qp-average, partial correlation, and correlation methods for one dataset simulated with $(n, p) = (500, 200)$. The PR curve is often used in information retrieval for comparison of performances of different binary decision algorithms. Let us define an experiment from P positive instances and N negative instances under some conditions. The four outcomes can be summarized in Table 1. Then, precision and recall are defined as follows:

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \quad \text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}.$$

Figure 2 indicates that the ψ-learning method performs much better than the gLasso, nodewise regression, PC, and qp-average methods for this example. For the ψ-learning method, Figure 2 is drawn by varying the significance level α_2, while fixing $\alpha_1 = 0.05$ and $\xi_n = 1$. The other four methods fail to distinguish direct dependencies from indirect ones for many instances. As mentioned in the Introduction, the failures of gLasso and nodewise regression may be due to the intrinsic weakness of the shrinkage methods: More variables need to be selected to compensate the effect of true parameter shrinkage, so the precision is low in the high-recall region. To plot Figure 2, their regularization parameters have been varied over a sequence of different values automatically selected by the package *huge*.

Table 1. Outcomes of a binary decision

	Actual positive (P)	Actual negative (N)
Predicted positive	True positive (TP)	False positive (FP)
Predicted negative	False negative (FN)	True negative (TN)

For the PC algorithm, Figure 2 is drawn by varying the value of α_{pc} from 10^{-8} to 0.6. This algorithm is extremely slow for a large value of α_{pc}, as in this case it is difficult to have an edge removed and thus the separator set Q considered by the algorithm becomes larger and larger. Just for the single value $\alpha_{\text{pc}} = 0.6$, it took over 1000 CPU minutes on a Dell Precision T7610 workstation (with a processor of 2.7 GHz). As implied by Figure 2, it is very difficult to improve the recall of the PC algorithm. Increasing α_{pc} from 10^{-8} to 0.6, the recall only increases from 0.50 to 0.61, while the precision drops from 1.0 to 0.159. This phenomenon can be explained as follows: The PC algorithm essentially employs a *maxP* statistic for testing the conditional independence, where the *maxP* statistic is defined, for each pair of variables $(X^{(i)}, X^{(j)})$, by

$$\max P(i, j) = \max_{Q \subset V \setminus \{i,j\}, |Q| \le m} p\left(X_i \perp X_j | X_Q\right),$$

where $p(\cdot)$ denotes the p-value of the test for the relationship $X_i \perp X_j | X_Q$ and m is the depth of the search. (We note that based on the adjacency faithfulness condition, the PC algorithm avoids the exhaustive scan for the subset Q by running with an iterative procedure, where Q is only taken from the neighborhood of X_i and X_j in the current graph.) The maxP statistic implies that to claim the existence of an edge, the p-values in all related tests must be small. Hence, the power of the PC algorithm can be very low when p is large. For this example, it is very hard for it to detect the second-order dependencies even with a large value of α_{pc}, although the first-order dependencies can be easily detected with a small value of α_{pc}. Reflected on Figure 2, we can see that its PR curve drops very fast within a narrow interval of recall.

Although the ψ-learning method is also based on the adjacency faithfulness condition, it employs a different test statistic and performs very differently from the PC algorithm. For the ψ-learning method, the correlation screening step is the key,

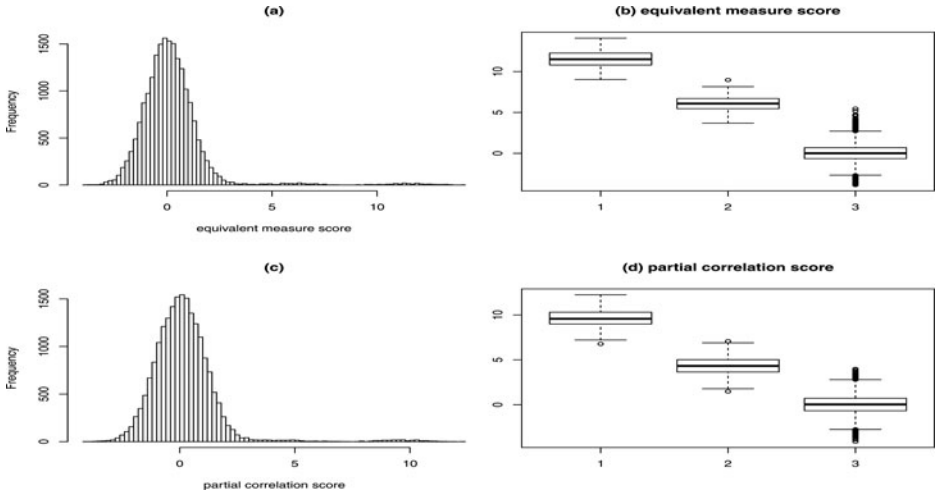

Figure 3. Comparison of ψ-scores and empirical partial correlation scores. Plot (a) and (c) show the histograms of ψ-scores and empirical partial correlation scores, respectively; and plot (b) and (d) show, respectively, the boxplots of ψ-scores and empirical partial correlation scores for the first-order dependent nodes (labeled by 1), the second-order dependent nodes (labeled by 2), and other nodes (labeled by 3).

which reduces the dimension of the problem and renders an equivalent measure of the partial correlation coefficient being used for the conditional independent test. The use of the equivalent measure of partial correlation coefficients improves the power of the conditional independence tests.

It is remarkable that the ψ-learning method even outperforms the empirical partial correlation method for this example. To make a further exploration of this issue, we compare the empirical partial correlation scores with ψ-scores in Figure 3. The empirical partial correlation score, denoted by $\tilde{z}^*_{ij}$, is defined as follows:

$$z^*_{ij} = \frac{1}{2}\log\left[\frac{1+\rho_{ij|V\setminus\{i,j\}}}{1-\rho_{ij|V\setminus\{i,j\}}}\right],$$

$$\tilde{z}^* = \Phi^{-1}\left(2\Phi\left(\sqrt{n-p-1}|z^*_{ij}|\right)-1\right).$$

Similar to the ψ-score, we call $n-p-1$ the effective sample size of the empirical partial correlation score. A comparison of Figures 3(b) and (d) shows that the ψ-score makes a better separation of the direct dependencies (labeled by 1 and 2) from the indirect ones (labeled by 3) than the empirical partial correlation score: Boxplot 2 and boxplot 3 have more overlaps in plot (d) than in plot (b). This is because the ψ-score has a larger effective sample size than the empirical partial correlation score.

Table 2 summarizes the performance of different methods for the 10 simulated datasets by calculating the average areas under their respective PR curves. The results indicate that the ψ-learning method works stably for different datasets and significantly outperforms all other methods.

The ψ-learning method was also compared with other methods under the setting $(n, p) = (100, 200)$, which represents the scenario of $p > n$. Note that under this setting the empirical partial correlation method is no longer available. Figure 4 shows the PR curves resulted from different methods for one dataset, and Table 2 shows the average areas under their respective PR curves for 10 datasets. The comparison indicates that the ψ-learning method significantly outperforms the other methods. For this example, we set the significance level $\alpha_1 = 0.2$ in the step of correlation screening. We have also tried different values of α_1, for example, 0.05 and 0.1. Under all these settings, the ψ-learning method outperforms the other methods.

Figure 4 shows that the ψ-learning method outperforms the other methods in the high-precision region, while this is reversed in the low-precision region. This shift in performance is due to the phase transition phenomenon of partial correlation screening. As pointed out by Hero and Rajaratnam (2011, 2012), the correlation and partial correlation screening suffers from a phase transition phenomenon; as the threshold decreases, the number of discoveries increases abruptly. Hence, in the high-recall region, the ψ-learning method tends to have a low precision. We note that the performance of the ψ-learning method in this region can be partially improved using other information of the

Table 2. Average areas under the PR curves resulted from different methods: ψ-learning, empirical partial correlation, gLasso, nodewise regression, PC algorithm, qp-average, and marginal correlation

(n, p)	ψ-learning	Empirical	gLasso	Nodewise	qp-average	PC	Correlation
	0.9940	0.9831	0.8259	0.9466	0.7268	0.5285	0.7696
(500, 200)	(0.0002)	(0.0011)	(0.0010)	(0.0019)	(0.0025)	(0.0040)	(0.0017)
	0.7925	—	0.5336	0.6207	0.5819	0.4945	0.5215
(100, 200)	(0.0086)	—	(0.0030)	(0.0024)	(0.0030)	(0.0026)	(0.0029)

NOTE: The numbers in parentheses represent the standard deviation of the average area.

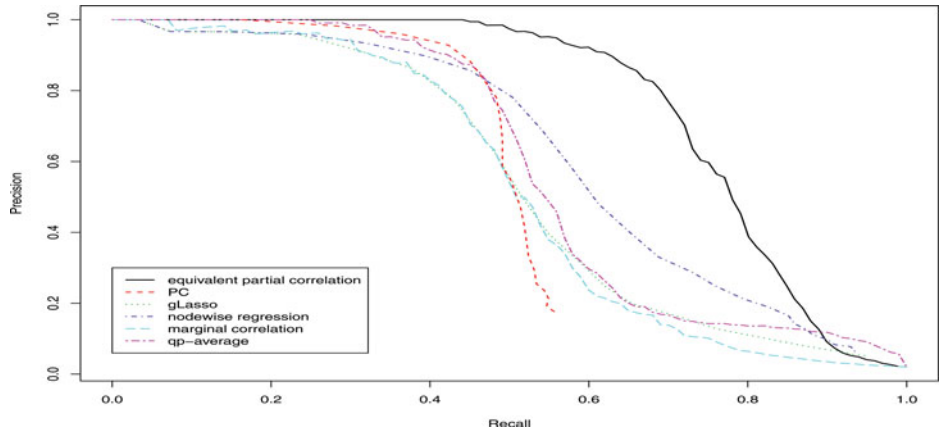

Figure 4. PR curves of the ψ-learning, PC, gLasso, nodewise regression, qp-average, and marginal correlation methods for one autoregressive dataset simulated under the setting $(n, p) = (100, 200)$.

data, for example, the sign of partial correlations. For this example, if we know the partial correlations are all positive, then the ψ-screening step can be done based on the Z-score calculated in (11), with only the positive Z-scores being considered for potential edges. Figure 5 compares the performance of ψ-learning for this example under the two scenarios, with and without sign information of partial correlations. As expected, the sign information improves the performance of the ψ-learning method. In practice, even the ψ-scores calculated in (12) are used in ψ-screening, we can still remove some identified edges to improve the precision by using the known sign information.

5. REAL DATA EXAMPLES

5.1 T-Cell Data

This dataset results from one experiment investigating the expression response of human T-cells to phorbol myristate acetate (PMA). It contains the temporal expression levels of 58 genes for 10 unequally spaced time points. At each time point there are 34 separate measurements. The data have been log-transformed and quantile-normalized and are available at the R package *longitudinal* (Opgen-Rhein and Strimmer 2008). See Rangel et al. (2004) for more descriptions of the dataset. In analyzing the data, we ignore its longitudinal structure and treat the observations as independent. This is acceptable as we are mainly interested in the relation between different genes and the dependence between different observations will not affect much the estimation of correlation between different genes.

The ψ-learning method was first applied to this dataset. Figure 6 shows the histogram of ψ-scores together with the cutoff values for significant ψ-scores. The two bars correspond to the cutoff values with $\alpha_2 = 0.01$ and 0.0001, respectively. Figures 7(a) and (b) show the networks identified at the test levels of 0.01 and 0.0001, respectively. The two networks consist of 95 edges and 46 edges, respectively. Note that Figure 7(b) is a subgraph of Figure 7(a), although this cannot be clearly viewed from the plots. Figure 7(b) indicates that there may consist of two functional groups among the 58 genes. In practice, we can vary the value of α_2 to see how the network grows as more genes are added to the network. This forms a network path as in gLasso for which the path can be formed by varying the regularization parameter.

For comparison, the gLasso and nodewise regression methods were also applied to this dataset. Both methods were implemented using the R package *huge*, where the regularization parameters were automatically determined using a stability approach (Liu, Roeder, and Wasserman 2010). The stability approach seeks the setting of regularization parameters that leads

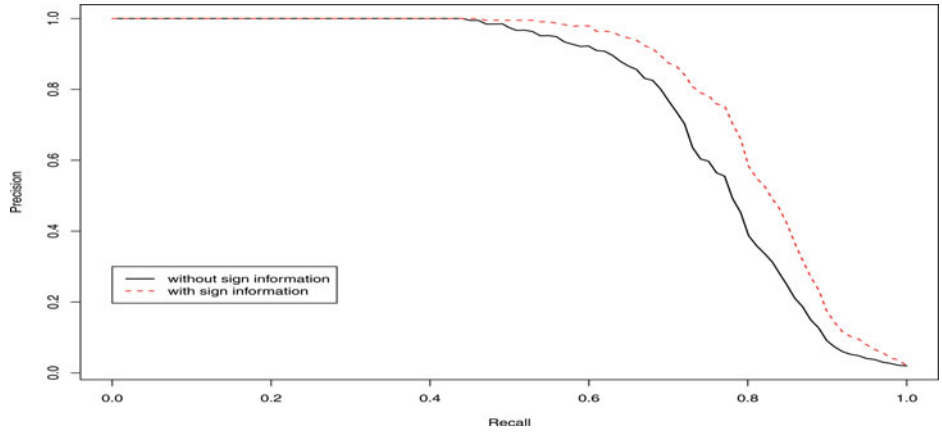

Figure 5. Comparison of the performance of the ψ-learning method under the two scenarios, with (dotted line) and without (solid line) sign information of partial correlations, for one autoregressive dataset simulated with $(n, p) = (100, 200)$.

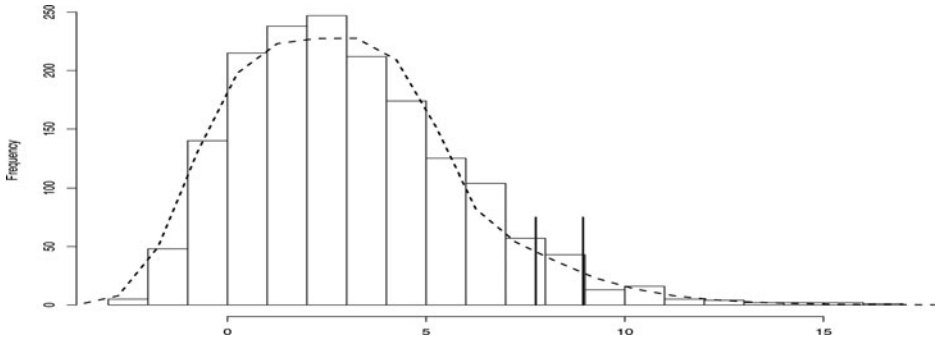

Figure 6. Histogram of ψ-scores for the T-cell data: The dashed curve shows the fitted density function of the mixture exponential power distribution, and the left and right bars show the cutoff values of ψ-score for $\alpha_2 = 0.01$ and .0001, respectively.

to the most stable set of edges. In brief, stability selection subsamples the data and estimates a separate graph for each subsample, and the "optimal" value of the regularization parameter controls the average variance over the edges of the subsampled graphs. Figures 7(b) and (c) show the networks resulted from the two methods. The two networks consist of 305 and 214 edges, respectively, and both are much denser than the networks resulted from the ψ-learning method. The qp-average method was not applied to this example as for which it is unclear how to choose the cutoff value for the nonrejection rate.

To assess the quality of the networks resulted from different methods, we fit the power law (see, e.g., Kolaczyk 2009, pp. 80–85) to them. A nonnegative random variable X is said to have a power-law distribution if

$$P(X = x) \propto x^{-\nu},$$

for some positive constant ν. The power law states that the majority of vertices are of very low degree, although some are of much higher degree—two or three orders of magnitude higher in some cases. A network whose degree distribution follows the power law is called a scale-free network. Many biological networks, such as gene expression networks, protein–protein interaction networks, and metabolic networks, have been observed to be scale-free (Barabási and Albert 1999). Figure 8 plots the log-probability $\log P(X = x)$ versus the log-degree $\log(x)$ for the four networks shown in Figure 7, where the degree x refers to the number of edges incident to a node. For a scale-free network, the plot should show a linear, decreasing pattern. Figure 8 indicates that the networks resulted from the ψ-learning method approximately follow the power law, while those resulted from the gLasso and nodewise regression do not. The latter two networks contain too many high-degree vertices; that is, these methods fail to distinguish direct dependencies from indirect ones. Therefore, we may conclude that the networks resulted from the ψ-learning method are closer to the true than those resulted from gLasso and nodewise regression.

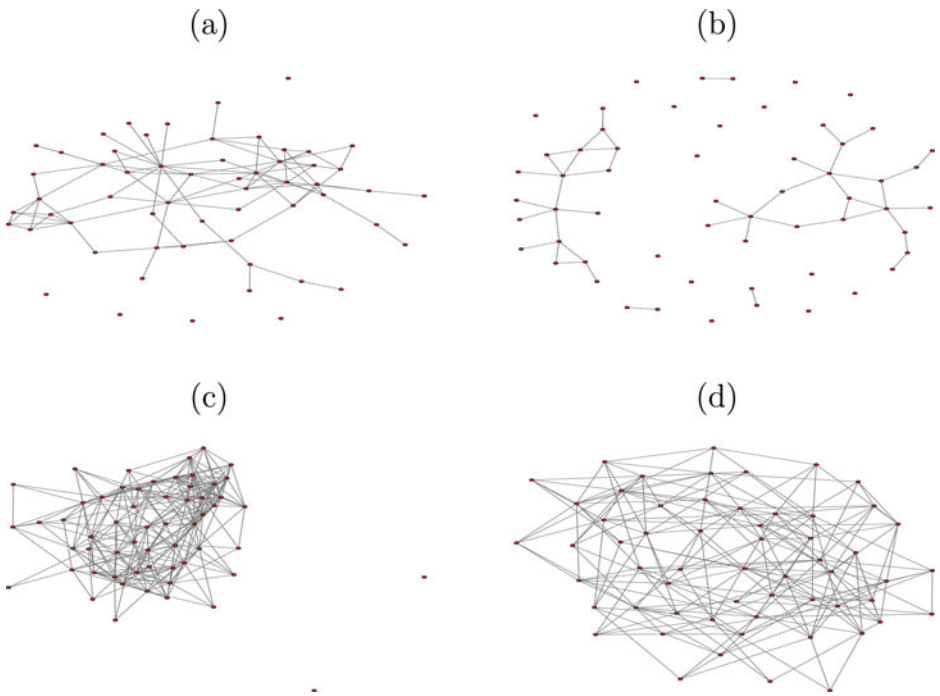

Figure 7. Networks identified for the T-cell data by (a) the ψ-learning method with $\alpha_2 = 0.01$, (b) the ψ-learning method with $\alpha_2 = 0.0001$, (c) gLasso, and (d) nodewise regression.

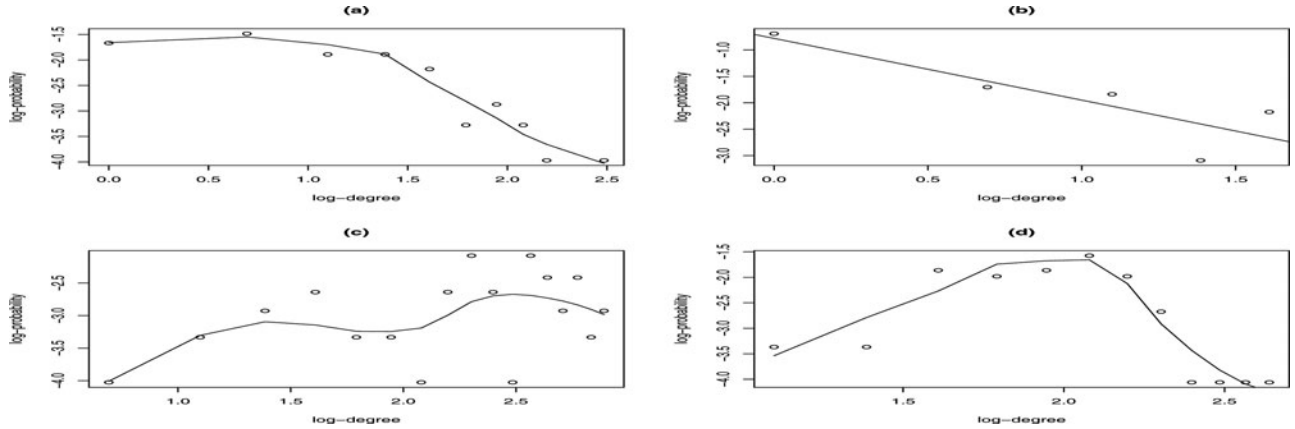

Figure 8. Log–log plots of the degree distributions of the four networks shown in Figure 7: (a) ψ-learning with $\alpha_2 = 0.01$; (b) ψ-learning with $\alpha_2 = 0.0001$; (c) gLasso; and (d) nodewise regression. The curves shown in (a), (c), and (d) are fitted by the loess function in R, and the lines shown in (b) are fitted by linear regression.

5.2 Breast Cancer Data

This dataset contains 49 breast cancer samples and 3883 genes arising from a study of molecular phenotypes for clinical prediction (West et al. 2001). The cleaned data are available at *http://strimmerlab.org/data.html*.

The ψ-learning method was first applied to this example. The full graph consists of 7,536,903 edges, and thus there are 7,536,903 correlation scores to screen in step (a) and the same number of ψ-scores to screen in step (c). To avoid a complete scan of such a large number of test scores at each iteration of the stochastic approximation algorithm, we work on subsamples. The subsamples are obtained via stratified sampling as described in Appendix A.2. For this example, we set the sampling rate to be 1%, which results in a subsample of size 75,370. Working with such a small subsample accelerates the computation significantly.

For this example, since the sample size is small, we set $\xi_n = 1/3$, which increases the maximum neighborhood size from the default value $13(\approx n/\log(n))$ to $38(\approx 3n/\log(n))$. Figure 9 shows the histogram of ψ-scores together with the cutoff values for the significant ψ-scores. As indicated by the histogram, the GGM consists of some edges for which the corresponding partial correlation coefficients are significantly different from zero. Figure 10(a) shows the network identified by the ψ-learning method at a significance level of 0.01 (for the ψ-screening step), which consists of 398 edges. Figure 11 shows all the edges of the network. From the network, we can identify a few hub genes. A gene is called a hub gene if it has strong connectivity to other genes. As noted by many researchers (see, e.g., Langfelder, Mischel, and Horvath 2013), hub genes are expected to play an important role in biology. If we set the cutoff value of connectivity at 5, then 13 hub genes can be identified from the network. The top four hub genes are CD44, IGFBP-5, HLA, and STARD3, which are all related to breast cancer. It is known that CD44 is involved in many essential biological functions associated with the pathology activities of cancer cells, and CD44 expression is commonly used as a marker for breast cancer stem cells (Louderbough and Schroeder 2011); IGFBP-5 plays an important role in the molecular mechanism of breast cancer, especially in metastasis (Akkiprik et al. 2008), and has been considered as a potential therapeutic target for breast cancer; HLA expression is associated with the genetic susceptibility (Chaudhuri et al. 2000), tumor progression and recurrent risk of breast cancer (Kaneko et al. 2011); and STARD3 (also named as MLN64 and CAB1) is in the 17q11-q21 region in which amplification is found in about 25% of primary breast carcinomas, and its gene expression is associated with poor clinical outcome, such as increasing risk of relapse and poor prognosis (*http://atlasgeneticsoncology.org/*). We have also examined the degree distribution of the network and found that it fits to the power-law distribution very well.

For comparison, gLasso and nodewise regression were also applied to this example. With the stability approach, both

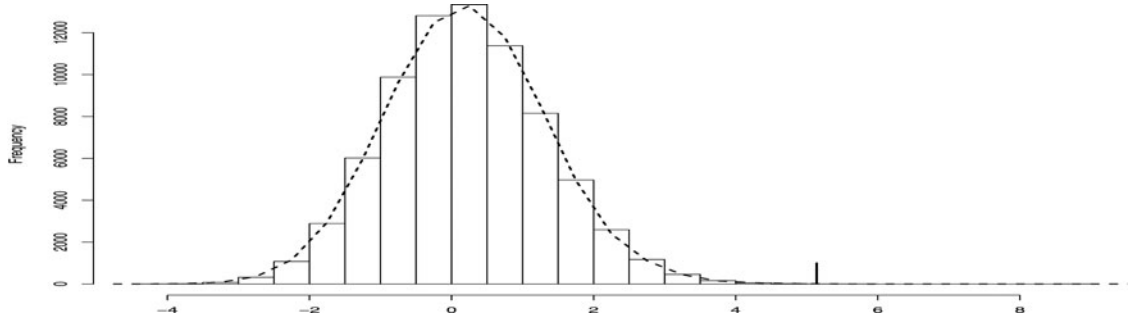

Figure 9. Histogram of ψ-scores for the breast cancer data: The dashed curve shows the fitted density function of the mixture exponential power distribution and the bar shows the cutoff value of ψ-score for $\alpha_2 = 0.01$.

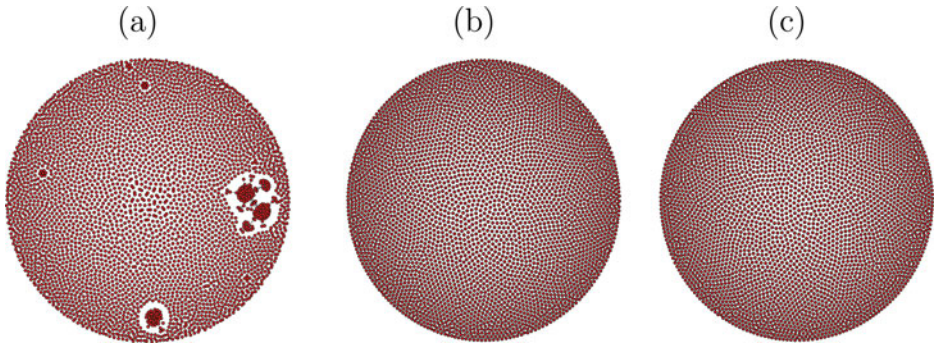

Figure 10. Networks identified for the breast cancer data by (a) ψ-learning with $\alpha_2 = 0.01$, (b) gLasso, and (c) nodewise regression. The network (a) consists of 398 edges, and the networks (b) and (c) are both empty.

methods selected 0.993 as their optimal regularization parameter value from a sequence of 30 values computed by the package under the default setting. As shown in Figures 10(b) and (c), both methods produced empty graphs for this example. In addition to the stability approach, we have tried the rotational information criterion, which has also been implemented in the R package *Huge*, for the determination of the regularization parameter. With the rotational information criterion, both methods still produced empty graphs for this example. A possible reason for their failure is that the sample size-to-dimension ratio n/p is too small, which makes the concentration matrix estimation and nodewise regression estimation highly unreliable. From this perspective, we can see that the ψ-learning method is more robust to the ratio of the sample size to dimension, as in which the equivalent partial correlation coefficients are calculated with a reduced conditional set.

6. ψ-LEARNING WITH DATA INTEGRATION AND NETWORK COMPARISON

The ψ-learning method provides an equivalent measure of the partial correlation coefficient for each pair of variables. This feature makes it very flexible for data integration, covariate adjustment, and network comparison. In what follows, we discuss how to make data integration and network comparison using the ψ-learning method. The discussion for covariate adjustment is postponed to Section 8.

6.1 ψ-Learning With Data Integration

In practice, we often need to integrate multiple sources of data to improve the construction of the Gaussian graphical network. For example, The Cancer Genome Atlas (TCGA) have collected genome, transcriptome, and epigenome data for more than 20 types of cancer from thousands of patients. The availability of such a wealth of data makes integrative methods essential for inferring the structure of the true genetic network as well as the heterogeneity of biological processes and phenotypes.

To integrate multiple sources of data, we first note that the ψ-partial correlation coefficient can be transformed to a Z-score via Fisher's transformation:

$$\psi_{z_{ij}} = \frac{\sqrt{n - |S_{ij}| - 3}}{2} \log\left[\frac{1 + \hat{\psi}_{ij}}{1 - \hat{\psi}_{ij}}\right], \quad i, j = 1, 2, \ldots, p. \tag{29}$$

For convenience, we call the Z-score a ψ_z-score to indicate its relationship with ψ-learning. Then, ψ_z-scores from different sources of data can be combined using Stouffer's meta-analysis method (Stouffer et al. 1949; Mosteller and Bush 1954) in the

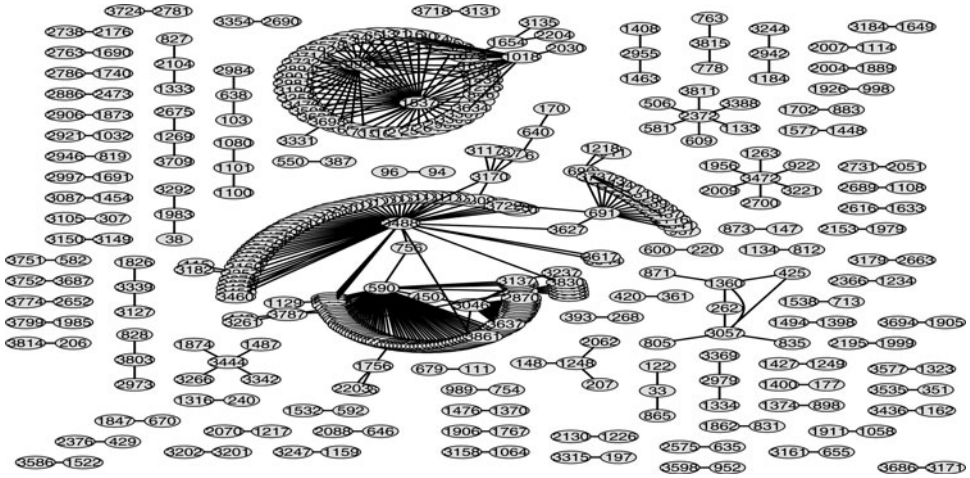

Figure 11. Edges of the network identified by the ψ-learning method for the breast cancer data, where the genes numbered 590, 3488, 1537, and 992 are CD44, IGFBP-5, HLA, and STARD3, respectively.

following way:

$$\psi_{c_{ij}} = \frac{\sum_{k=1}^{K} w_k \psi_{z_{ij}}^{(k)}}{\sqrt{\sum_{k=1}^{K} w_k^2}}, \quad i, j = 1, 2, \ldots, p, \tag{30}$$

where K denotes the total number of data sources, $\psi_{z_{ij}}^{(k)}$ denotes the ψ_z-score from source k, and w_k denotes the weight assigned on source k. The assignment of w_k's may depend on the sample size or data quality of different sources. It is easy to see that $\psi_{c_{ij}}$ follows a standard normal distribution under the null hypothesis H_0: $e_{ij} = 0$. Then, a multiple hypothesis test can be done on $\psi_{c_{ij}}$'s to identify the pairs of vertices for which $\psi_{c_{ij}}$ is differentially distributed from the standard normal $N(0, 1)$. To ease this test, we make a further transformation:

$$\psi'_{c_{ij}} = \Phi^{-1}(2\Phi(|\psi_{c_{ij}}|) - 1), \tag{31}$$

under which the two-sided test for $\psi_{c_{ij}}$'s is converted to a one-sided test for $\psi'_{c_{ij}}$. This facilitates the use of the multiple hypothesis test procedure of Liang and Zhang (2008). In summary, we have the following algorithm.

Algorithm 2. (ψ-Learning with data integration)

(i) (ψ-correlation calculation) Perform steps (a) and (b) of the ψ-learning algorithm independently for each source of data.
(ii) (ψ_z-score combination) Calculate combined ψ_c-scores in (29) and (30).
(iii) (ψ_c-score screening) Conduct a multiple hypothesis test to identify the pairs of vertices for which $\psi_{c_{ij}}$ is differentially distributed from the standard normal $N(0, 1)$.

Alternative to Stouffer's meta-analysis method, other combined probability test methods, such as Fisher's method and Pearson's method (see, e.g., Owen 2009), can also be used here with minor modifications. In case the data of different sources are dependent, they can be integrated using the method of Kost and McDermott (2002).

6.2 Network Comparison Under Distinct Conditions

In genetic studies, one often wants to compare networks obtained under distinct but related conditions, such as cancer and normal tissue. Danaher, Wang, and Witten (2014) proposed to estimate two graphical models jointly using a regularization method. Although the joint approach improves network estimation through borrowing information from each other, it fails to address the uncertainty issue related to the difference of the networks because the regularization method only results in a point estimate.

Under the framework of ψ-learning, the uncertainty issue can be easily addressed. Let $\{\psi_{z_{ij}}^{(1)}\}$ and $\{\psi_{z_{ij}}^{(2)}\}$ denote the ψ_z-scores [refer to Equation (29)] obtained under conditions 1 and 2, respectively. Then, the problem of network comparison is reduced to a multiple testing problem; simultaneously testing $H_0^{(i,j)} : e_{ij}^{(1)} = e_{ij}^{(2)} \leftrightarrow H_1^{(i,j)} : e_{ij}^{(1)} \neq e_{ij}^{(2)}$ for all $1 \le i < j \le p$, where $e_{ij}^{(k)}$ is the indicator of the edge between node i and node j for the network constructed under condition k. The corresponding test statistic can be

$$\psi_{d_{ij}} = \left[\psi_{z_{ij}}^{(1)} - \psi_{z_{ij}}^{(2)}\right] / \sqrt{2}, \tag{32}$$

which follows a standard normal distribution under the null hypothesis. Again, to facilitate the use of the multiple hypothesis testing procedure of Liang and Zhang (2008), we may transform $\psi_{d_{ij}}$ to $\psi'_{d_{ij}}$ via (31). In summary, we have the following algorithm.

Algorithm 3. (ψ-Learning for network comparison)

(i) (ψ-correlation calculation) Perform steps (a) and (b) of the ψ-learning algorithm independently for each source of data.
(ii) (ψ_d-score calculation) Calculate the difference of ψ-scores in (32).
(iii) (ψ_d-score screening) Conduct a multiple hypothesis test to identify the pairs of vertices for which $\psi_{d_{ij}}$ is differentially distributed from the standard normal $N(0, 1)$.

Extending this algorithm to the case of multiple conditions is simple based on a multiple comparison procedure, such as Fisher's least significance difference test or Tukey's W-procedure (see, e.g., Ott and Longnecker 2010).

6.3 An Illustrative Example

To illustrate the above two algorithms, we conducted the following experiment: Generate two datasets from $N(0, \frac{1}{2}C^{-1})$ and $N(0, C^{-1})$, respectively, where C is as specified in (28)

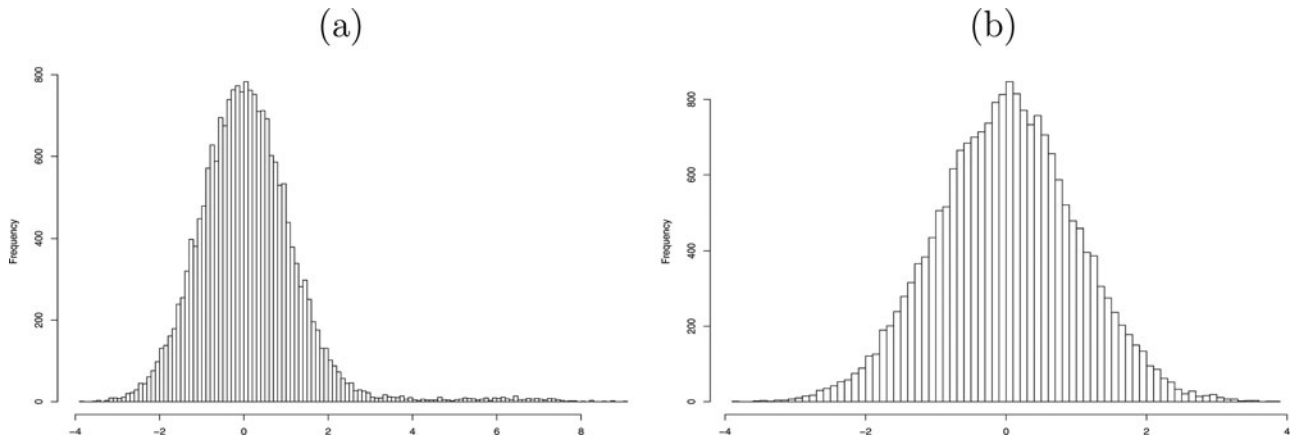

Figure 12. Histograms of $\psi'_{c_{ij}}$ (left) and $\psi'_{d_{ij}}$ (right) for the two simulated datasets.

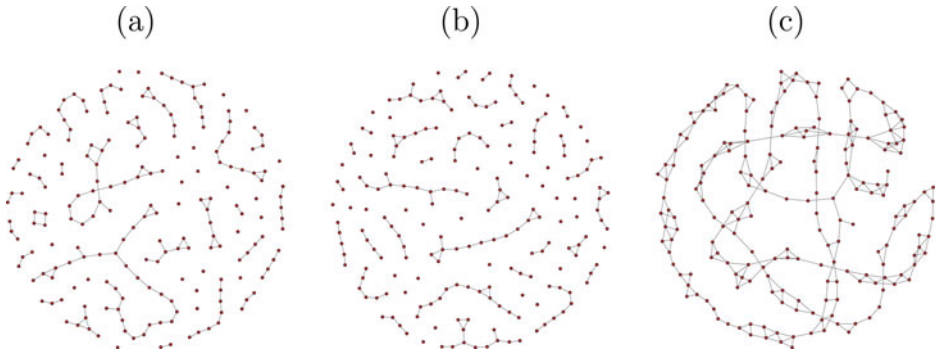

Figure 13. Networks identified by the ψ-learning method for the simulation experiment: (a) network for dataset 1, (b) network for dataset 2, and (c) integrated network of dataset 1 and dataset 2.

with $p = 200$. Both datasets have a sample size of $n = 100$. Figures 13(a) and (b) show the networks identified by the ψ-learning algorithm for the two datasets, respectively. Here the ψ-learning algorithm was run under its default setting with $\alpha_1 = 0.05, \alpha_2 = 0.01$, and $\xi_n = 1$. The network shown in Figure 13(a) has a precision of 0.963 and a recall of 0.388, and the network shown in Figure 13(b) has a precision of 0.958 and a recall of 0.343. This result is reasonable, as the data source of network (a) has a smaller variation than that of network (b). To get denser networks, one may apply a large value of α_2.

Figure 12(b) shows the histogram of $\psi'_{d_{ij}}$-scores for the two simulated datasets. A multiple hypothesis test on the $\psi'_{d_{ij}}$-scores shows that there are no differential edges between the two underlying networks even at a very high significance level $\alpha_2 = 0.5$. This result suggests the validity of our approach for network comparison.

Figure 12(a) shows the histogram of $\psi'_{c_{ij}}$-scores for the two simulated datasets, and Figure 13(c) shows the integrated network. In the ψ_c-score screening step, the default significance level $\alpha_2 = 0.01$ was applied here. The integrated network has a precision of 0.934 and a recall of 0.602, and it is much closer to the true network than any of the networks constructed from a single source of data.

In summary, the ψ-learning algorithm has provided a very flexible and sensible framework for data integration and network comparison. However, these problems are usually difficult to handle by other methods, such as the PC and regularization methods.

7. COMPUTATIONAL COMPLEXITY

In terms of p, the computational complexity of the ψ-learning algorithm is bounded by $O(p^2(\log p)^b)$, where $b = 3(2\kappa' + \tau)/\delta$, $O(p^2)$ is for the total number of ψ-scores that need to be calculated, and $O((\log p)^b)$ is for inverting a neighboring matrix of size $O(n^{2\kappa'+\tau})$. When $\delta > 0$, the computational complexity of the algorithm is nearly $O(p^2)$. The case $\delta = 0$, that is, p is fixed as n increases, can be treated as a low-dimensional case, and the computational complexity is bounded by $O(p^3)$ in this case.

Compared to the existing algorithms, the computational complexity of the ψ-learning algorithm is favorable. The gLasso algorithm (Friedman, Hastie, and Tibshirani 2008) is known to have a computational complexity of $O(p^3)$. With its fast implementation (Witten, Friedman, and Simon 2011; Mazumder and Hastie 2012), which makes use of the block diagonal structure in the graphical lasso solution, the computational complexity can be reduced to $O(p^{2+\nu})$, where $0 < \nu \leq 1$ may depend on the number of blocks and the size of each block. Refer to Witten, Friedman, and Simon (2011) for detailed complexity analysis of the algorithm. Since the ordinary Lasso has a computational complexity of $O(np \min(n, p))$ (Meinshausen 2007), the nodewise regression has a complexity of $O(p^3(\log p)^{2/\delta})$ if $p > n$ and $O(p^4(\log p)^{1/\delta})$ otherwise. The computational complexity of qp-average is difficult to analyze. Since the algorithm needs to estimate the nonrejection rate for each pair of vertices by randomly drawing some conditional sets from a collection of $\binom{p-2}{q}$ conditional sets for a sequence of q values, the computational complexity of the algorithm is at least $O(p^3)$ by assuming that $q = 1$ and the number of conditional sets that one has sampled is proportional to p. In practice, a larger value of q is often used, such as $q = n/2$; the computational complexity of the algorithm can be much higher than $O(p^3)$. The PC algorithm has a computational complexity bounded by $O(p^{2+m})$, where m is the depth of the search. When m is large, the algorithm can be very slow.

We have compared the CPU times of ψ-learning, gLasso, and nodewise regression for the breast cancer example studied in Section 5. On a 2.9 GHz computer, ψ-learning costs 61.1 min, gLasso costs 6,993.2 min, and nodewise regression costs 24.2 min. Both gLasso and nodewise regression were run in the package *huge* under their default setting with the regularization parameter being determined using the stability approach. However, as shown in Section 5, both gLasso and nodewise regression failed for this example. This example shows that not only in network construction, ψ-learning is also attractive in computational time compared to the other existing algorithms.

8. CONCLUSION

In this article, we have proposed a new measure of partial correlation coefficient for high-dimensional GGMs. Under the Markov property and adjacency faithfulness conditions, the new measure of partial correlation coefficient is equivalent to the partial correlation coefficient evaluated on the full conditional set in the construction of Gaussian graphical networks. Based on the new measure of partial correlation coefficient, we proposed the ψ-learning method for the construction of Gaussian graphical networks. Under mild conditions, we established the consistency

of the ψ-learning method. The numerical examples indicate that the ψ-learning method outperforms the existing methods, such as graphical Lasso, nodewise regression, PC, and qp-average, especially for the problems for which a large number of indirect associations are present.

The key to the success of the ψ-learning method is correlation screening, which, as shown in Lemma 4, reduces the size of estimated neighborhoods. Here we would like to point out that in a strict sense, as implied by Assumption (A_3), the proof of Lemma 4 is based on a setting where the graph is made up of disconnected components. By Lemma 2, the size of each component is bounded by $O(n^{1-b})$ for some $0 < b < 1$. In practice, however, we do not have direct access to the true population correlation coefficients, and need to do statistical inference based on finite samples. By appropriately thresholding sample correlation coefficients, the ψ-learning method can also work well for the connected graphs, for example, the autoregressive example studied in Section 4. In this example, the correlation coefficients of nodes i and j tend to be treated as zero when $|i - j|$ becomes large.

The ψ-learning method is developed based on a multiple hypothesis testing procedure where the distributions of the null and alternative effects are estimated using the stochastic approximation algorithm. As shown in Equation (A.3) of Appendix A.1, the false discovery rate (FDR) can be calculated based on the estimator of the null effect distribution only. An alternative approach for estimating the null effect distribution is developed by Jin and Cai (2007), which works based on the empirical characteristic function and Fourier analysis. We expect that this approach can work equally well here.

The ψ-learning method provides a very flexible framework for GGM learning with different types of data. In Section 6, we discuss how to make data integration and network comparison under this framework. In what follows, we discuss how to adjust covariate effects and extend the method to non-Gaussian variables.

Covariate effect adjustment is important for GGM learning, as the relationship of the variables $X^{(1)}, \ldots, X^{(p)}$ is often affected by external variables. Let $W_1, \ldots, W_q$ denote the external variables. To adjust for their effects, we can replace the empirical correlation coefficient used in step (a) of the ψ-learning algorithm by the p-value obtained in testing the hypotheses $H_0 : \beta_{q+1} = 0 \leftrightarrow H_1 : \beta_{q+1} \neq 0$ for the regression

$$X^{(i)} = \beta_0 + \beta_1 W_1 + \cdots + \beta_q W_q + \beta_{q+1} X^{(j)} + \boldsymbol{\epsilon}, \quad (33)$$

where $\boldsymbol{\epsilon}$ denotes a vector of Gaussian random errors. Note that reversing the role of $X^{(i)}$ and $X^{(j)}$ in (33) will not affect the p-value of the test. Similarly, we replace the ψ-correlation coefficient used in step (c) of Algorithm 1 by the p-value obtained in testing the hypotheses $H_0 : \beta_{q+1} = 0 \leftrightarrow H_1 : \beta_{q+1} \neq 0$ for the regression

$$X^{(i)} = \beta_0 + \beta_1 W_1 + \cdots + \beta_q W_q + \beta_{q+1} X^{(j)} + \sum_{k \in S_{ij}} \gamma_k X^{(k)} + \boldsymbol{\epsilon}, \quad (34)$$

where S_{ij}, as defined previously, denotes the selected separator of $X^{(i)}$ and $X^{(j)}$. In the literature, one often formulates the covariate adjustment problem as a problem of joint estimation of the multiple regression coefficients and precision matrix, and then solves the problem using a regularization method, see for example, Yin and Li (2011) and Cai et al. (2013). Compared to the joint estimation approach, the ψ-learning method can be at least more efficient as analyzed in Section 7.

Ravikumar, Wainwright, and Lafferty (2010) showed that the nodewise regression method developed in Meinshausen and Bühlmann (2006) for GGMs can be extended to binary graphical models by replacing the linear regression with the logistic regression. With the same reasoning, the ψ-learning algorithm can be extended to binary graphical models. In this case, some new theory for sure screening in logistic regression may be required. For non-Gaussian continuous random variables, one may first apply the semiparametric Gaussian copula transformation suggested by Liu, Lafferty, and Wasserman (2009) to render the data normal, and then apply the ψ-learning algorithm to infer the GGM structure.

APPENDIX A: MULTIPLE HYPOTHESIS TESTS

A.1 The Multiple Hypothesis Testing Procedure

This procedure is adopted from Liang and Zhang (2008). Let $z_1, z_2, \ldots, z_N$ denote a set of test scores, where for GGMs $N = p(p-1)/2$ and the test score refers to the correlation score or ψ-score. To identify the test scores that are significantly larger than zero, we model them by an m-component mixture exponential power distribution, for which the most-left component corresponds to the null effects, and the other components correspond to the alternative effects. How to determine the value of m will be discussed later.

For an m-component mixture exponential power distribution, the density function is given by

$$g(z|\vartheta_m) = \sum_{i=1}^{m} \varpi_i \psi(z|\nu_i, \sigma_i, \alpha_i), \quad (A.1)$$

where $\vartheta_m = (\varpi_1, \nu_1, \sigma_1, \alpha_1, \ldots, \varpi_m, \nu_m, \sigma_m, \alpha_m)$ contains all parameters of the distribution, ϖ_i is the weight of the ith component with $0 < \varpi_i < 1$ and $\sum_{i=1}^{m} \varpi_i = 1$, and

$$\psi(z|\nu_i, \sigma_i, \alpha_i) = \frac{\alpha_i}{2\sigma_i \Gamma(1/\alpha_i)} \exp\{-(|z - \nu_i|/\sigma_i)^{\alpha_i}\}, \quad -\infty < \nu_i < \infty,\ \sigma_i > 0,\ \alpha_i > 1, \quad (A.2)$$

where the parameters ν_i, σ_i, and α_i represent the center, dispersion, and decay rate of the distribution, respectively. For $\alpha_i = 2$, the distribution (A.2) is reduced to $N(\nu_i, \sigma_i^2/2)$; for $1 < \alpha_i < 2$, the distribution is heavy-tailed; and for $\alpha_i > 2$, the distribution is light-tailed.

The identifiability of (A.1) has been established in Holzmann, Munk, and Gneiting (2006).

The parameters ϑ_m can be estimated as in Liang and Zhang (2008) by minimizing the Kullback–Leibler divergence

$$\mathrm{KL}(g_{\vartheta_m}, g) = -\int \log\left\{\frac{g(z|\vartheta_m)}{g(z)}\right\} g(z)dz,$$

where $g(z)$ denotes the unknown true density of z_i's. For a given value of m, the minimization can be done using the stochastic approximation algorithm; refer to Liang and Zhang (2008) for more details. One significant advantage of this algorithm is that it permits the general dependence between z_i's. A proof of convergence for this algorithm can be found in Zhang and Liang (2008). The cutoff value z_c can be chosen by controlling the FDR of significant test scores at a prespecified test level. Figure 14 depicts such a rule. For a given rule $\Lambda_c = \{Z_i \geq z_c\}$,

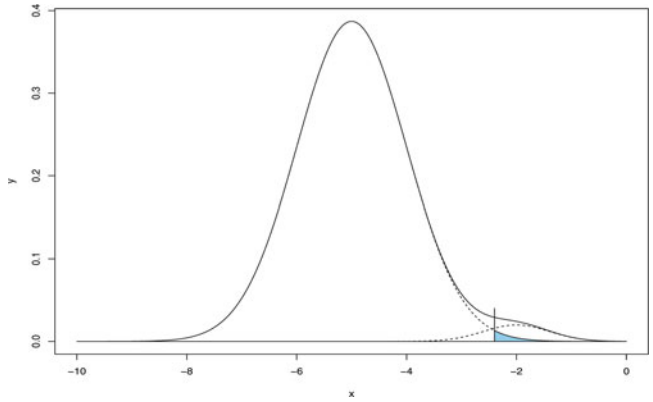

Figure 14. Illustrative plot for the multiple hypothesis testing procedure: Test scores are fitted by a two-component mixture exponential power distribution. The vertical line shows the cutoff z_c, and the shaded area shows the portion of falsely identified correlations or partial correlations.

the FDR can be estimated by

$$\text{FDR}(\Lambda_c) = \frac{N\hat{\varpi}_1[1 - F(z_c|\hat{\nu}_1, \hat{\sigma}_1, \hat{\alpha}_1)]}{\#\{z_i : z_i \geq z_c\}}, \tag{A.3}$$

where $\#\{z_i : z_i \geq z_c\}$ denotes the number of test scores larger than z_c and $F(\cdot)$ denotes the CDF of the exponential power distribution (A.2). Define the q-value (Storey 2002) as

$$q_c(z) = \inf_{\{\Lambda_c : z \in \Lambda_c\}} \text{FDR}(\Lambda_c), \tag{A.4}$$

which can be used as the reference quantity for the decision of multiple hypothesis tests. For example, we can set the test level to be 0.01, that is, choosing z_c such that $q_c(z) \leq 0.01$ for all $z \geq z_c$.

The value of m can be determined according to the fitness, which is measured by the Kullback–Leibler divergence, of the corresponding models. Refer to Liang and Zhang (2008) for the detail. In this article, we restrict m to 2 or 3.

A.2 A Stratified Sampling Scheme for Test Scores

Let $\{z_1, \ldots, z_N\}$ denote the whole set of test scores. Let K denote the inverse of the sampling rate. Then, the stratified sampling can be conducted as follows:

- Order the test scores to get $z_{(1)}, \ldots, z_{(N)}$.
- Group the ordered test scores into $L = \lceil N/K \rceil$ groups: $\{z_{(1)}, \ldots, z_{(K)}\}$, $\{z_{(K+1)}, \ldots, z_{(2K)}\}$, …, $\{z_{((L-1)K+1)}, \ldots, z_{(N)}\}$, where L is the minimum integer greater than or equal to N/K.
- Randomly pick one test score from each group.

Let $\{z_1^*, \ldots, z_L^*\}$ denote the sampled test scores. It is obvious that when p is large, the cutoff value z_c determined based on the sampled test scores is approximately unbiased. In practice, we often set K to be 50 or 100.

[Received March 2014. Revised January 2015.]

REFERENCES

Akkiprik, M., Feng, Y., Wang, H., Chen, K., Hu, L., Sahin, A., Krishnamurthy, S., Ozer, A., Hao, X., and Zhang, W. (2008), "Multifunctional Roles of Insulin-Like Growth Factor Binding Protein 5 in Breast Cancer," *Breast Cancer Research*, 10, 212. [1259]

Banerjee, O., Ghaoui, L. E., and D'Aspremont, A. (2008), "Model Selection Through Sparse Maximum Likelihood Estimation for Multivariate Gaussian or Binary Data," *Journal of Machine Learning Research*, 9, 485–516. [1249]

Barabási, A., and Albert, R. (1999), "Emergence of Scaling in Random Networks," *Science*, 286, 509–512. [1258]

Benjamini, Y., Krieger, A. M., and Yekutieli, D. (2006), "Adaptive Linear Step-Up Procedures That Control the False Discovery Rate," *Biometrika*, 93, 491–507. [1252]

Bühlmann, P., Kalisch, M., and Maathuis, M. H. (2010), "Variable Selection in High-Dimensional Linear Models: Partially Faithful Distribution and the PC-simple Algorithm," *Biometrika*, 97, 261–278. [1249]

Bühlmann, P., and van de Geer, S. (2011), *Statistics for High-Dimensional Data: Methods, Theory and Applications*, Berlin: Springer-Verlag. [1253,1254]

Cai, T. T., Li, H., Liu, W., and Xie, J. (2013), "Covariate-Adjusted Precision Matrix Estimation With an Application in Genetic Genomics," *Biometrika*, 100, 139–157. [1263]

Castelo, R., and Roverato, A. (2006), "A Robust Procedure for Gaussian Graphical Model Search From Microarray Data With p Larger Than n," *Journal of Machine Learning Research*, 7, 2621–2650. [1248]

——— (2009), "Reverse Engineering Molecular Regulatory Networks From Microarray Data With qp-Graph," *Journal of Computational Biology*, 16, 213–227. [1248,1254]

Chaudhuri, S., Cariappa, A., Tang, M., Bell, D., Haber, D. A., Isselbacher, K. J., Finkelstein, D., Forcione, D., and Pillai, S. (2000), "Genetic Susceptibility to Breast Cancer: HLA DQB*03032 and HLA DRB1*11 May Represent Protective Alleles," *Proceedings of the National Academy of Sciences USA*, 97, 11451–11454. [1259]

Danaher, P., Wang, P., and Witten, D. M. (2014), "The Joint Graphical Lasso for Inverse Covariance Estimation Across Multiple Classes," *Journal of the Royal Statistical Society,* Series B, 76, 373–397. [1261]

Dempster, A. P. (1972), "Covariance Selection," *Biometrics*, 28, 157–175. [1248]

Friedman, J., Hastie, T., and Tibshirani, R. (2008), "Sparse Inverse Covariance Estimation With the Graphical Lasso," *Biostatistics*, 9, 432–441. [1249,1254,1262]

Hero, A., and Rajaratnam, B. (2011), "Large-Scale Correlation Screening," *Journal of the American Statistical Association*, 106, 1540–1552. [1256]

——— (2012), "Hub Discovery in Partial Correlation Graphs," *IEEE Transactions on Information Theory*, 58, 6064–6078. [1256]

Holzmann, H., Munk, A., and Gneiting, T. (2006), "Identifiability of Finite Mixtures of Elliptical Distributions," *Scandinavian Journal of Statistics*, 33, 753–763. [1263]

Jin, J., and Cai, T. T. (2007), "Estimating the Null and the Proportion of Nonnull Effects in Large-scale Multiple Comparisons," *Journal of the American Statistical Association*, 102, 495–506. [1263]

Kalisch, M., and Bühlmann, P. (2007), "Estimating High-dimensional Directed Acyclic Graphs With the PC-algorithm," *Journal of Machine Learning Research*, 8, 613–636. [1254]

Kaneko, K., Ishigami, S., Kijima, Y., Funasako, Y., Hirata, M., Okumura, H., Shinchi, H., Koriyama, C., Ueno, S., Yoshinaka, H., and Natsugoe, S. (2011), "Clinical Implication of HLA Class I Expression in Breast Cancer," *BMC Cancer*, 11, 454. [1259]

Kolaczyk, E. D. (2009), *Statistical Analysis of Network Data: Methods and Models*, New York: Springer. [1258]

Koller, D., and Friedman, N. (2009), *Probabilistic Graphical Models: Principles and Techniques*, Cambridge, MA: The MIT Press. [1249]

Kost, J., and McDermott, M. (2002), "Combining Dependent p-Values," *Statistics & Probability Letters*, 60, 183–190. [1261]

Langfelder, P., Mischel, P. S., and Horvath, S. (2013), "When is Hub Gene Selection Better Than Standard Meta Analysis?" *PLoS One* [on-line], 8, e61505. [1259]

Lauritzen, S. (1996), *Graphical Models*, Oxford: Oxford University Press. [1248,1249]

Lemeire, J., Meganck, S., Cartella, F., and Liu, T. (2012), "Conservative Independence-Based Causal Structure Learning in Absence of Adjacency Faithfulness," *International Journal of Approximate Reasoning*, 53, 1305–1325. [1250]

Liang, F., and Zhang, J. (2008), "Estimating the False Discovery Rate Using the Stochastic Approximation Algorithm," *Biometrika*, 95, 961–977. [1251,1261,1263,1264]

Liu, H., Lafferty, J., and Wasserman, L. (2009), "The Nonparanormal: Semiparametric Estimation of High Dimensional Undirected Graphs," *Journal of Machine Learning Research*, 10, 2295–2328. [1263]

Liu, H., Roeder, K., and Wasserman, L. (2010), "Stability Approach to Regularization Selection (StARS) for High Dimensional Graphical Models," *Advances in Neural Information Processing Systems*, 23, 1432–1440. [1249,1257]

Louderbough, J. M., and Schroeder, J. A. (2011), "Understanding the Dual Nature of CD44 in Breast Cancer Progression," *Molecular Cancer Research*, 9, 1573–1586. [1259]

Luo, S., Song, R., and Witten, D. (2015), "Sure Screening for Gaussian Graphical Models," arXiv:1407.7819v1. [1253]

Lysen, S. (2009), "Permuted Inclusion Criterion: A Variable Selection Technique," PhD thesis, University of Pennsylvania. [1249]

Magwene, P. M., and Kim, J. (2004), "Estimating Genomic Coexpression Networks Using First-Order Conditional Independence," *Genome Biology*, 5, R100. [1248]

Mazumder, R., and Hastie, T. (2012), "The Graphical Lasso: New Insights and Alternatives," *Electronic Journal of Statistics*, 6, 2125–2149. [1249,1253,1255,1262]

Meek, C. (1995), "Strong Completeness and Faithfulness in Bayesian Networks," in *Proceedings of UAI-1995*, pp. 411–418. [1250]

Meinshausen, N. (2007), "Relaxed Lasso," *Computational Statistics & Data Analysis*, 52, 374–393. [1262]

Meinshausen, N., and Bühlmann, P. (2006), "High-Dimensional Graphs and Variable Selection with the Lasso," *Annals of Statistics*, 34, 1436–1462. [1249,1254,1263]

Mosteller, F., and Bush, R. R. (1954), "Selected Quantitative Techniques," in *Handbook of Social Psychology* (Vol. 1), ed. G. Lindzey, Cambridge, MA: Addison Wesley, pp. 289–334. [1260]

Opgen-Rhein, R., and Strimmer, K. (2008), "Longitudinal: Analysis of Multiple Time Course Data," R Package version 1.1.4. [1257]

Ott, R. L., and Longnecker, M. (2010), *An Introduction to Statistical Methods and Data Analysis* (6th ed.), Belmont, CA: Brooks/Cole Cengage Learning. [1261]

Owen, A. B. (2009), "Karl Pearson's Meta-Analysis Revisited," *The Annals of Statistics*, 37, 3867–3892. [1261]

Ramsey, J., Zhang, J., and Spirtes, P. (2006), "Adjacency-Faithfulness and Conservative Causal Inference," in *Proceedings of UAI-2006*, pp. 401–408. [1250]

Rangel, C., Angus, J., Ghahramani, Z., Lioumi, M., Sotheran, E., Gaiba, A., Wild, D., and Falciani, F. (2004), "Modeling T-Cell Activation Using Gene Expression Profiling and State-space Models," *Bioinformatics*, 20, 1361–1372. [1257]

Ravikumar, P., Wainwright, M., and Lafferty, J. (2010), "High-Dimensional Ising Model Selection Using L1-Regularized Logistic Regression," *The Annals of Statistics*, 38, 1287–1319. [1263]

Spirtes, P., Glymour, C., and Scheines, R. (2000), *Causation, Prediction, and Search* (2nd ed.), Cambridge: The MIT Press. [1249,1254]

Storey, J. D. (2002), "A Direct Approach to False Discovery Rates," *Journal of the Royal Statistical Society,* Series B, 64, 479–498. [1252,1264]

Stouffer, S. A., Suchman, E. A., DeVinney, L. C., Star, S. A., Jr Williams, R. M., (1949), *The American Soldier, Vol. 1: Adjustment During Army Life*. Princeton, NJ: Princeton University Press. [1260]

Tibshirani, R. (1996), "Regression Analysis and Selection Via the Lasso," *Journal of the Royal Statistical Society,* Series B, 58, 267–288. [1249]

West, M., Blanchette, C., Dressman, H., Huang, E., Ishida, S., Spang, R., Zuzan, H., Olson, J., Jr Marks, J., and Nevins, J. (2001), "Predicting the Clinical Status of Human Breast Cancer by Using Gene Expression Profiles," *Proceedings of the National Academy of Sciences USA*, 98, 11462–11467. [1259]

Wille, A., and Bühlmann, P. (2006), "Low-Order Conditional Independence Graphs for Inferring Genetic Networks," *Statistical Applications in Genetics and Molecular Biology*, 4, 32. [1248]

Witten, D. M., Friedman, J. H., and Simon, N. (2011), "New Insights and Faster Computations for the Graphical Lasso," *Journal of Computational and Graphical Statistics*, 20, 892–900. [1253,1262]

Yin, J., and Li, H. (2011), "A Sparse Conditional Gaussian Graphical Model for Analysis of Genetic Genomics Data," *Annals of Applied Statistics*, 5, 2630–2650. [1263]

Yuan, M., and Lin, Y. (2007), "Model Selection and Estimation in the Gaussian Graphical Model," *Biometrika*, 94, 19–35. [1249,1255]

Zhang, J., and Liang, F. (2008), "Convergence of Stochastic Approximation Under Irregular Conditions," *Statistica Neerlandica*, 62, 393–403. [1263]

Zhang, J., and Spirtes, P. (2008), "Detection of Unfaithfulness and Robust Causal Inference," *Minds & Machines*, 18, 239–271. [1250]

Zhao, T., Liu, H., Roeder, K., Lafferty, J., and Wasserman, L. (2012), "The *huge* Package for High-dimensional Undirected Graph Estimation in R," *Journal of Machine Learning Research*, 13, 1059–1062. [1249,1254]