LEARNING GAUSSIAN GRAPHICAL MODELS USING DISCRIMINATED HUB GRAPHICAL LASSO

Zhen Li, Jingtian Bai, Weilian Zhou
Department of Statistics, North Carolina State University

Objectives

Learning underlying stochastic dependency structures among different factors for the data:

- n observations and p variables, p > n possible
- Assuming \( x_1, \ldots, x_n \sim N_p(0, \Sigma) \)
- Domain knowledge of some dependency relationships incorporated

Hub Gaussian Graphical Model

Graphical Model: A set of multivariate joint distributions associated with a graph \( G = (V, E) \)
- \( V \): vertex set, representing variables
- \( E \): edge set, representing conditional dependency; \( X \) satisfies the pairwise Markov property if \( x_v \) and \( x_u \) are independent given \( x_{V \setminus \{v,u\}} \) whenever \( \{v,u\} \notin E \)

Gaussian Graphical Model: Further assuming \( x_1, \ldots, x_n \sim N_p(0, \Sigma) \), and \( \Theta = \Sigma^{-1} \) is the precision matrix
- The MLE maximizes \( \ell(X, \Theta) = -\log \det \Theta + \text{trace}(S\Theta) \)
- \( S \) is the empirical covariance matrix of \( X \)
- \( V_v \) and \( V_u \) are conditionally independent iff \( \Theta_{vu} = 0 \)

Graphical Model with Hubs: Nodes that are connected to a very high number of other nodes in a graph

\[ \Theta = V + V^T + Z \]

Discriminated Hub Graphical Lasso (DHGL)

\[
\begin{align*}
\min_{\Theta, V, Z} & \ell(X, \Theta) + \lambda_1 \|Z - \text{diag}(Z)\|_1 + \lambda_2 \sum_{j \notin D} \|V - \text{diag}(V_j)\|_1 + \lambda_3 \sum_{j \in D} \|V - \text{diag}(V_j)\|_q \\
\text{subject to} & \quad \Theta = V + V^T + Z, S = \{\Theta : \Theta > 0 \text{ and } \Theta = \Theta^T\}
\end{align*}
\]

Computation

- Give “loose conditions” \((\lambda_2 \leq \lambda_3 \leq \lambda_4)\) to nodes in \( D \)
- Reduce to HGL in Tan et al. (2014) when \( D = \emptyset \)
- Use Alternating Direction Methods of Multipliers (ADMM) to solve the convex problem
- Computational complexity: \( O(p^3) \) per iteration
- Select tuning parameters by minimizing a BIC-type quantity

DHGL with Known Hub Nodes

- Use HGL to get the estimated hubs \( \hat{H}_{\text{HGL}} \)
- Set \( D = K \setminus \hat{H}_{\text{HGL}} \) where \( K \) is set of known hubs.
- If \( D \neq \emptyset \), use DHGL to estimate \( \Theta \) and get the estimated hubs \( \hat{H}_{\text{DHGL}} \) where \( \lambda_1, \lambda_2, \lambda_3 \) remain the same values as in HGL and \( \lambda_4 \) is selected using the BIC-type quantity. Then, set \( \hat{H} = \hat{H}_{\text{HGL}} \cup \hat{H}_{\text{DHGL}} \) as the set of estimated hubs. If \( D = \emptyset \), use the estimation in HGL directly.

DHGL without Known Hub Nodes

- Use HGL to get the estimated hubs \( \hat{H}_{\text{HGL}} \)
- Adjust regularization parameter \( \lambda \) of GL from large to small until \( |\hat{H}_{\text{GL}}\setminus\hat{H}_{\text{HGL}}| > 0 \) and \( |\hat{H}_{\text{GL}}\setminus\hat{H}_{\text{HGL}}| \leq \max\{|\hat{H}_{\text{GL}}|, a, b|\hat{H}_{\text{HGL}}|\} \), where \( a \in N, b > 1 \) but \( b \approx 1 \). \( \hat{H}_{\text{GL}} \) is the set of estimated hubs by GL with parameter \( \lambda \).
- Set \( D = \hat{H}_{\text{GL}} \setminus \hat{H}_{\text{HGL}} \) which is non-empty.
- Use DHGL to estimate \( \Theta \), where \( \lambda_1, \lambda_2, \lambda_3, \lambda_4 \) remain the same values as in HGL and \( \lambda_2 \) is selected using the BIC-type quantity.

Conclusion

- With some hubs known, DHGL outperforms HGL in estimating the precision matrix.
- Without known hubs, DHGL outperforms HGL given correct prior information, and rarely degenerates even if the prior information is incorrect.