Learning Sparse Graphs with a Core-periphery Structure

Sravanthi Gurugubelli and Sundeep Prabhakar Chepuri
Indian Institute of Science, Bangalore, India

Core-Periphery Structure

- Densely connected core vertices and sparsely connected periphery vertices
- Identifying the core and peripheral vertices helps in analyzing the central processes in complex networks

Gaussian Graphical Model with a Core-periphery Structure

We model the dependence of the node attributes on the core scores through a latent graph structure

\[
p(X|\Theta) = \det \Theta \exp(-\text{tr}(\Theta S))
\]

where

\[
w_{ij} = 1 - c_i - c_j + e \log(d_{ij})
\]

Prior Art

Existing algorithms estimate core scores given the network topology
- We often have access only to node attributes
- Underlying graph structure may not always be available

Conventional approaches to network topology inference do not readily incorporate a core-periphery structure

Gaussian Graphical Model

Feature matrix of graph \( G \): \( X = [x_1, x_2, \ldots, x_d] \)

\( x_i \sim \mathcal{N}(\mu_i, \Theta_i^{-1}), \forall i = 1, \ldots, d \)

Graphical lasso learns the sparsity pattern in \( \Theta \) by solving

maximize \( \log \det \Theta - \text{tr}(\Sigma \Theta) - \lambda \left\| \Theta \right\|_1 \)

subject to \( w_{ij} > 0, \quad i, j = 1, 2, \ldots, N \)

Model Evaluation

The proposed algorithm converges in about 10 iterations

Updating the Graph

This is a convex program that can be solved using existing solvers, e.g., QUIC

Updating the Core Scores

maximize \( \sum_{i,j=1}^{N} |\Theta_{ij}|(c_i + c_j) \)

subject to \( \sum_{i=1}^{N} c_i = M, c_i \in [0,1], \quad i, j = 1, \ldots, N \)

This is a linear program and can be solved using standard off-the-shelf solvers

Brain Network Analysis

The regions with a large difference in the core scores of the two groups coincide with the regions that have differences in activation for healthy individuals and patients with ADHD

sravanthig@iisc.ac.in, spchepuri@iisc.ac.in
https://ece.iisc.ac.in/~spchepuri/