Causal Inference Programming Homework #2
Sepehr Torab Parhiz
93100774

PC-Algorithm

For this assignment, I studied *Estimating High-Dimensional Directed Acyclic Graphs with the PC-Algorithm* by Kalisch et al. In this paper, partial correlation is computed using the recursive formula below:

$$\rho_{i,j|\mathbf{k}} = \frac{\rho_{i,j|\mathbf{k}\setminus h} - \rho_{i,h|\mathbf{k}\setminus h}\rho_{j,h|\mathbf{k}\setminus h}}{\sqrt{(1-\rho^2_{i,h|\mathbf{k}\setminus h})(1-\rho^2_{j,h|\mathbf{k}\setminus h})}}.$$

Then, the decision whether two nodes are conditionally independent or not is made by following instruction which replaces line 11 in the pseudocode for the PC-Pop algorithm:

$$\textbf{if } \sqrt{n-|\mathbf{k}|-3}\,|Z(i,j|\mathbf{k})| \leq \Phi^{-1}(1-\alpha/2) \textbf{ then}.$$

In the partial_correlation function, I also implemented another way to compute partial correlation with regression. In this method, two linear models are fit to the each of the variables i and j, using data of variables in k as training data. Next, the residuals for i and j are computed. The correlation coefficient of the residuals is the partial correlation. This method takes much longer to complete and produced different skeletons with fewer remaining edges compared to the recursive method.

Both PC-Pop and PC-Stable finish after 7 levels.

The random graphs are generated with the *generate_random_graph* function as follows:

The function receives three arguments; number of nodes, the probability that each node can have a directed edge to a node before it (in the topological order) and the probability of that a node has a directed edge to any of the edges after it.

First, an adjacency matrix is created, where all the diagonal elements are 1. Then, for each ordered pair of nodes, an edge is added by sampling a bit according to the probabilities given to us.

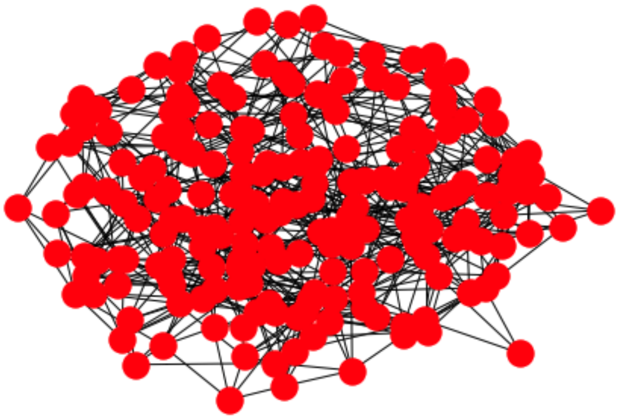
In data generation, for each sample a standard normal variable is sampled for each variable as the noise of the sample. Next, the values of parents of each node is added to it by using the adjacency matrix as a mask.

The recall and missing are computed as follows:

$$\text{Recall} = \frac{\text{Shared edges in the estimated and the actual graph}}{\text{Number of edges in the actual graph}}$$

$$\text{Missing} = \frac{\text{Number of edges different between the estimated and the actual graph}}{\text{Number of edges in the actual graph}}$$

Recall and missing in PC-Pop can be lower or higher compared to PC-Stable based on the random graph that has been generated. Overall, PC-Stable seems to work better, with higher recall and lower missing.



Skeleton generated by PC-Pop algorithm on the given data.