KEN4258: Computational Statistics
Homework Assignment 1

Ayse Arslan, Kristian van Kuijk, Carlos Soto Garcia-Delgado, Philip Mühlenfeld, and Ali Alsaeedi

Department of Advanced Computing Sciences, Maastricht University, The Netherlands

1 Introduction

The goal of this assignment is to make a causal statement on how the variables in a small dataset are related to one another (11 variables). In this report, we apply the PC method [4], an algorithm to identify the causal structure in graphs from sample probabilities. This causal discovery method is used to discover causal relations by analyzing statistical properties of purely observational data. The PC algorithm under i.i.d. sampling assuming no latent confounders, uses a search architecture that involves plugging in statistical decision procedures, to determine conditional independence. The algorithm works by forming a complete undirected graph, eliminating unconditionally independent edges, and then using a series of rules to orient edges and recover the true Markov Equivalence Class of the directed acyclic graph. The output of the PC algorithm is typically more informative than the conditional independence graph [1], as it has a causal interpretation and may differ from the estimated conditional independence graph. The PC algorithm is feasible for sparse graphs with at least tens of thousands of variables, even though this is not the case for our dataset, assuming efficient conditional independence tests.
We also applied another popular method for causal discovery, namely the DirectLiNGAM [3], for the sake of comparison. Due to the page limit, we do not discuss the algorithm in the main body of this report. Nonetheless, the results can be found in the appendix A.

2 Causal Discovery

2.1 Assumptions

In this report, we make the following assumptions:

- We make no assumptions on the distribution of the data.
- We assume causal sufficiency: There are no unobserved confounders for any of the variables in the graph.
- We assume acyclicity: There are no cycles in the graph.
- We make a faithfulness assumption: Variables that are causally connected in a particular way in the graph are probabilistically dependent.

2.2 A Procedure to Estimate the DAG from the Data using our Assumptions

To explain the procedure of the PC method, we illustrate an example with 5 variables: A, B, C, D and E. The process from which the data was generated is represented in Fig. 1. The procedure is as follows:

1. We start with a complete undirected graph.
2. We identify the skeleton (the undirected edges in our graph).
3. We identify the graph inmoralities. A graph inmortality occurs when a child has two or more parents without dependencies between them.
4. We orient the rest of the edges using our discoveries from $Step\_2$. We orient edges that are incident in colliders.

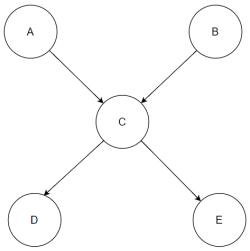

Fig. 1: Original true causal graph

Step 1: Complete undirected graph We start the procedure by constructing an undirected graph, where each variable is a node and all nodes are connected to each other.

Step 2: Identifying the skeleton of the graph We perform independence tests between variables in order to remove edges. More specifically, we start conditioning on the empty set at first and we perform independence tests between two variables X and Y. If X and Y are independent, we remove the edge from the graph. We start at first conditioning on the empty set, however we increase the set of the conditioning set later. We provide the following example to illustrate this step. We test whether A and B are independent $X \perp\!\!\!\perp Y|\{\}$. Let's assume this is the case. Hence, we remove the edge between A and B (Fig. 2). No other pair of nodes is independent given the empty set. Then, we condition on sets of nodes when testing for independence. For the sake of the explanation we condition on C. Test whether: $\forall$ other pairs $(X, Y), X \perp\!\!\!\perp |\{C\}$. If we condition on C, A and D are independent, same for A-E, B-D and B-E. Therefore, those edges are removed.

Step 3: Identifying the inmoralities In this step for any path X-Z-Y for which the following holds:

1. There is no edge between X and Y.
2. Z was not in the conditioning set that makes X and Y conditionally independent.

Then, X-Z-Y forms an immorality. A graph immorality occurs when a child has two (or more) parents without dependencies between them. In our example, we take the edge A and B. As explained in the previous step, there is no edge between A and B. C was not in the conditioning set that made A and B independent.

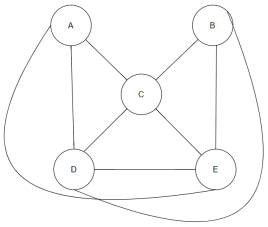

Fig. 2: A and B are independent

Step 4: Identifying the rest of the edges The idea is we use all discovered inmoralities in step 3 to orient the rest of the edges. In our example: If there was a directed edge from D to C, we would have discovered it in step 2. Hence, the edge needs to be directed from C to D. Same reasoning from C to E.

2.3 Results

We applied the procedure explained to our given dataset and obtained the results in Fig. 4. Our git repository can be found here.

2.4 Limitations of the procedure

In real life we may encounter violations of our assumption:

- No causal efficiency. This means some cofounder variables are not observable. This will be the case in many real life settings. Even if we can have access to all the cofounder variables, we will not know it.
- There may be no acyclicity in the data.

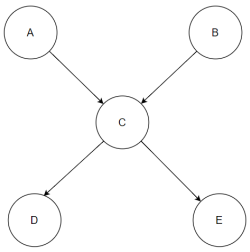

Fig. 3: Final graph

Furthermore, conditional independence testing is hard. The PC method relies on accurate conditional independence testing. If this is not accurate, then the method will not be as well. As a rule of thumb, the more data available, the more accurate is the conditional independence test. Hence this method may not be considered for *"small"* data problems. In the original paper [4], the authors report an error rate of 14% on the direction of the edges on the ALARM dataset with no prior ordering information. This method is fast, simple, scalable and does not make too many assumptions on the distribution of the data. However, depending on the use case, it is worth investing other methods that may lead to a lower error.

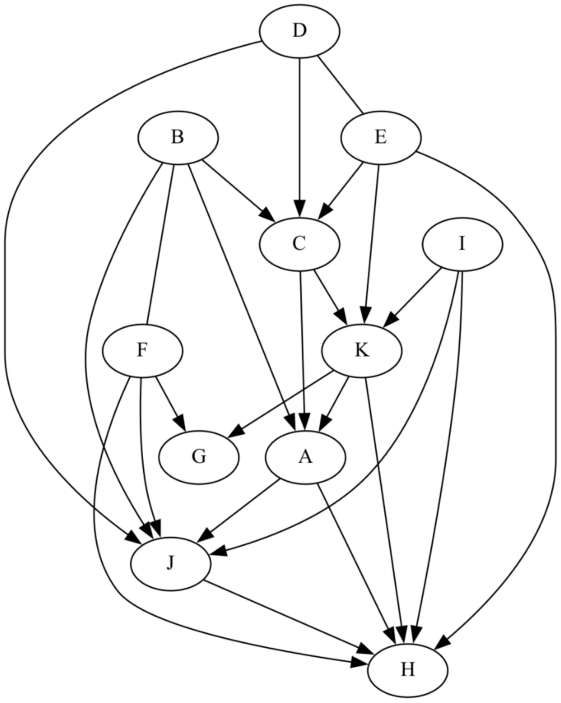

Fig. 4: DAG obtained on the dataset

References

1. Lauritzen, S.: Graphical models. Oxford: Clarendon Press. (1996)
2. Shimizu, S., Hoyer, P.O., Hyvärinen, A., Kerminen, A., Jordan, M.: A linear non-gaussian acyclic model for causal discovery. Journal of Machine Learning Research **7**(10) (2006)
3. Shimizu, S., Inazumi, T., Sogawa, Y., Hyvarinen, A., Kawahara, Y., Washio, T., Hoyer, P., Bollen, K.: Directlingam: A direct method for learning a linear non-gaussian structural equation model. Journal of Machine Learning Research **12** (01 2011)
4. Spirtes, P., Glymour, C.: An algorithm for fast recovery of sparse causal graphs. Social Science Computer Review - SOC SCI COMPUT REV **9**, 62–72 (04 1991). https://doi.org/10.1177/089443939100900106

A Appendix

A.1 DirectLINGAM

As a comparison, we also implemented the DirectLiNGAM algorithm. The detailed procedure can be found in the original paper [2]. In this paper, we want to go through the steps in a simplified version. The steps are as follows:

1. Standardize the data to have a mean of 0 and a variance of 1.
2. Use independence test to estimate the casual ordering. Statistical independence tests such as PC algorithm could be used for the estimation. An example of casual ordering of 5 variables will be: A -> B -> C -> D -> E.
3. Direct edge estimation: Estimate the direct casual effects for each variable using the other variables in the estimated casual order.(i.e. estimate the coefficients of the linear regression models between every variable and the parents of that variable in the casual graph). An example would be estimating the direct casual effect of variable B on C, which would be by regressing C on both A and B: C = aA + bP + e. Then we subtract the effect of A on C which will give us the direct casual effect of B on C. The direct casual effects is estimated for A on B, C on D, B on D and finally for C on E.

4. Indirect edge estimation: Estimate the indirect casual effects for each variable using the other variables in the estimated casual order. (i.e. estimate the coefficients of the linear regression models between every variable and its non-parents of that variable in the casual graph). An example would be estimating the indirect casual effect of variable A on C through B, we regress C on B and D which will give us: C = aB + dD + e. Then we subtract the effect of B on C which results in the indirect casual effect of A on C. We repeat the process and estimate the indirect casual effects of A on C through D, B on D through C, C and E through D.
5. Model selection: The final step will be choosing the best model which is the one that minimizes some criterion value.

The assumptions this procedure takes are as follows:

1. The variables form a DAG
2. The model assumes linearity and that the probability distributions of the error variables are not from a Gaussian distribution
3. Independence between the error variables

Furthermore, these are the limitations for this procedure:

1. The algorithm struggles to distinguish between causality and correlation.
2. If the second assumption is violated, the accuracy of the algorithm can be affected (linearity violation), and the causal estimates can be incorrect (non-Gaussian violation). If the third assumption is violated, it can cause biased estimates.

After running this procedure on our dataset, the result is as shown in 5.

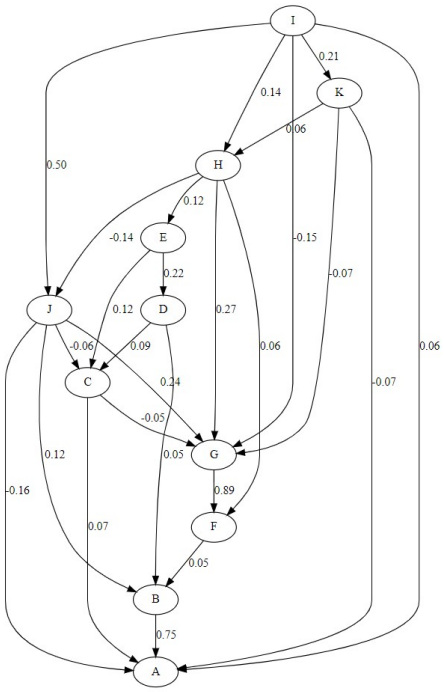

Fig. 5: DirectLINGAM result

As mentioned before, the DirectLiNGAM method assumes linear non-Gaussian causal relationships, whereas the PC algorithm assumes a sparse directed acyclic graph and uses conditional independence tests to estimate the causal graph. Both algorithms have the same goal but as their procedure are different, they have their own strengths and weaknesses, which makes the choice of the correct algorithm depend on the data. Normally, when having a fair number of features (less than 30), and a sparse causal relationship, it would be a better decision to choose the PC algorithm.