On Perspectives of Causal Networks Reconstruction by Independence-Based Methods

Olexandr S. Balabanov

Institute of Software Systems of NASU, Glushkov prosp.40, Kiev 03680, Ukraine

bas@isofts.kiev.ua

Abstract. *We concern in independence-based approach to recovery a causal nets and dependency structures from data. It is demonstrated how an efficiency of independence-based methods can be enhanced by means of reducing a search space during model inference. Such search-space reducing is attainable by using separation rules of inductive inference acceleration, which reflect (convey) the necessary requirements on bodies of separators required. The necessary requirements on separator's body follow from graphical properties of locally-minimal d-separators in digraph. We examine to what extent the efficiency of the acceleration rules may be depreciated in cases when temporal order of variables is known. Experimental results on efficiency of new version of our algorithm (Razor-1.2) are presented.*

Keywords

Inference of causal networks from data, independence-based algorithms, locally-minimal d-separator, conditional independence.

1 Introduction

We consider a problem of model's structure reconstruction in observational setting. So, methods which rely on randomized experiments are out of our scope. A various kinds of statistical models (which can be inferred or learned from empirical data) sometimes are divided into generative and discriminative (targeted) ones. In discriminative model a target variable is given. A generative modeling is aimed to describe a process of data generation. By now the probabilistic graphical models of dependency and causal networks became very attractive ones among all generative models. Causal networks not merely describe a joint probability distribution of variables, but pretend to reflect a genuine causal structure of the underlying process [1, 2]. A causal network can also be viewed as an integrated collection of local discriminative models. The Granger (non-)causality technique can be applied to a time series data. But one may have setting where temporal order of variables (in available data set) is unknown. There are two main traditional approaches to govern a model choosing: 1) model validity checking by quasi-prediction (like cross-validation and GMDH); 2) evaluation by some criterion which balances a likelihood and complexity of the model. An independence-based (or constraint-based) approach [2, 3] to a model inference differs from the two aforementioned ones: it resorts to considering of links individually instead of entire model or even fragments of a model. In what follow we address the problem of improving the performance of independence-based algorithms for inference of dependency networks from data. We argue a new way to enhance independence-based algorithms by equipping them with special rules, called separation rules of inductive inference acceleration (SIIA-rules) [4, 5]. In terms of d-separation, these rules are proved to be correct. For justification of empirical counterparts of SIIA-rules also appropriate versions of the Causal faithfulness assumption are needed. SIIA-rules can radically reduce a run-time of structural inference due to cutting branches when searching for separators. It is demonstrated by experiments for Bayesian networks of moderate density, that our algorithm ('Razor') outperforms the well-known PC algorithm.

2 Basics of ADG-models and Independence-based Methods

Among all graphical models perhaps the most widespread are those based on acyclic directed graphs (ADG) [1–4]. If there is an arc (directed edge) $X \rightarrow Y$ in a graph, then vertex (variable) X is said to be a (immediate) cause of vertex Y. ADG is a digraph which contains no one cyclone (directed cycle) $X \rightarrow \cdots \rightarrow X$. Sometimes it is reasonable to ignore the directions of some arc $X \rightarrow Y$ (calling it an edge X—Y). Vertices connected by arc (edge) are said to be adjacent. We denote by $\mathrm{F}(X)$ a set of causes of vertex X. When ignoring the directions of all arcs in a digraph G, we get the skeleton of G. ADG-model M is defined as pair (G, J), with G being ADG, and J being attributed parameters. (There is one-to-one correspondence between vertices in G and variables in J.) A parameters of ADG-model are defined as $p(X|\mathrm{F}(X))$. Bayesian networks are ADG-models with variables of nominal type; Gaussian networks are ADG-models with linear dependencies and normal disturbances. Markov properties of ADG-model are formalized through the criterion of d-separation [1]. This criterion is defined purely in graphical terms. If a set **Z** of vertices d-separates vertices X and Y, then **Z** is said to be a separator for pair (X, Y), denoting this by $\mathrm{Ds}(X; \mathbf{Z}; Y)$, where $X,Y \notin \mathbf{Z}$. If a set **Z** not d-separates vertices X and Y, we will denote this $\sim\mathrm{Ds}(X; \mathbf{Z}; Y)$. Apparently, there no one d-separator exists for adjacent vertices in a ADG.

An assertion of conditional independence of variables X and Y given a set of variables **Z** will be denoted by predicate $\mathrm{Pr}(X; \mathbf{Z}; Y)$. This independence means that $p(XY|\mathbf{Z})= p(X|\mathbf{Z})\cdot p(Y|\mathbf{Z})$. If $\mathrm{Pr}(X; \mathbf{Z}; Y)$ holds, we call **Z** an empirical separator for (X, Y). Unconditional independence (independence with empty condition) is denoted by $\mathrm{Pr}(X;;Y)$. It is known [1, 2], that the fact of d-separation in G entails corresponding probabilistic conditional independence in $M= (G, J)$:

$$\mathrm{Ds}(X; \mathbf{Z}; Y) \Rightarrow \mathrm{Pr}(X; \mathbf{Z}; Y).$$

Such independencies express Markov property of the model. Class of ADG-models naturally generalizes to the class of non-recursive Causal networks which allow bi-directed edges (to reflect latent variables) [1, 2, 3, 6].

The two main groups of methods are known for inference a structure of causal network from data. The first group comprises score-based methods, which learn the structure by executing a search in the space of (all possible) structures in an attempt to find the model with maximal score. This is computationally expensive task due to the enormous size of the space of possible structures with many variables. The score metric is usually a penalized likelihood (for example, BIC or MDL) or a posteriori probability etc. Algorithms of the second group rely on revealing a set of conditional independencies of variables. These algorithms use the outcomes of a number of conditional independence tests to infer a consistent structure. (It is possible to identify a structure up to equivalence class only.) Among all independence-based algorithms the most known seems to be the PC-algorithm [2]. It performs inference in three phases: 1) inference of graph’s skeleton; 2) orienting edges; 3) evaluating model’s parameters. The first phase is point of our attention. An independence-based algorithm deletes an edge upon finding an empirical separator for corresponding pair of variables. Independence-based algorithms rely on heavy assumptions, such as the Causal faithfulness assumption [2–4], which asserts that all independencies in a model are Markov. Known inference algorithms are combinatorial in nature since they search for separator for every pair of variables, examining numerous subsets of variables as tentative separator. When variables are discrete and number of variables goes beyond a few tens, the algorithm requires overly many runtime. (For Gaussian networks the algorithms are more scalable.) The task is especially hard when temporal ordering of variables is not given.

3 Idea and Tools for Improving Model Induction

The PC algorithm starts from a complete, undirected graph and deletes edges having obtained independence facts. Rank of tests (cardinality of tentative separators) iteratively grows up until getting independence or until a search exhausts. The key invention of PC algorithm is the following principle: to form a tentative separator for pair (X, Y) taking only those variables which are supposedly adjacent to X or to Y. This considerably reduces a search space and speeds up the inference. But searching for separators in networks of high or moderate density still remains very complex and expensive. Besides the problems of combinatorial complexity, important drawback of the algorithm is low reliability in edge identification. This comes from unreliability of independence testing under sample bias. The larger rank of test is the less reliable edge identification turns out to be. So, it is desirable to be satisfied with tests of low rank whenever possible. Perhaps the worst peculiarity of PC algorithm strategy manifests itself in typical situations when there an edge X—Y exists in the model, but PC continues an attempts to find a separator for (X, Y) and executes useless work. Therefore, of especial impotance is to recognize an edge presence as early as possible.

Our aim was to focus searching for separators and to render an algorithm more clever and efficient. The key idea to achieve the aim came from perceiving that there must hold some implications among d-separation facts in ADG model. As well known, when a model meets certain set of conditional independencies, this implies some other certain conditional independencies. Similarly, when certain d-separations are satisfied in digraph G, this would bind a set of other d-separations in G or even can render some edges prohibited and some other edges necessary. So, having obtained a pattern of dependencies/independencies, we can constraint a space of possible separators or even immediately identify some edges. Outlined idea became productive only after focusing attention on special subset of d-separators. So, a starting point for our elaboration is notion of locally-minimal separator [4–6].

Definition 1. A d-separator **Z** for pair (X, Y) is said to be locally-minimal d-separator (LoMS), iff for any $W \in \mathbf{Z}$ it is satisfied $\sim\mathrm{Ds}(X; \mathbf{Z}\backslash\{W\}; Y)$.

In words, removing any member of locally-minimal d-separator from the separator's body destroys the d-separation. For ADG-models the notion of LoMS is known to be equivalent to the non-redundant d-separator [6]. A d-separator **Z*** for pair (*X*, *Y*) is said to be minimal in *G*, iff there is no one d-separator **Z** for pair (*X*, *Y*) in *G* such that |**Z**| < |**Z***|.

As demonstrated in [4–6], any member of some (locally-)minimal separator must meet a certain necessary conditions. This was formalized by appropriate statements and rules, which were derived from the criterion of d-separation and topological properties of digraph. Knowing of some (minimal) separators allows to narrow a search for "contiguous" separators (in neighborhood). The developed rules assist to filter out 'false' (or unnecessary) candidates to a body of a LoMS. All proposed rules may be classified (grouped) accordingly to the roles (functions) they perform. There are the following four roles of the rules: 1) recognizing of edge presence; 2) recognizing of edge absence; 3)deleting ('filtering out') some vertex from list of candidates to body of respective separator; 4) recognizing of some vertex as obligate member of supposed separator. One can find a collection of the rules in [4–6]. (Those rules apply patterns of (non-)separation of zero- and first-rank only.) Perhaps the most important rule is the following one.

Rule of 'placing aside'. If there Ds(*Z*; *X*; *Y*) holds in *G*, then vertex *Z* is not a member of any locally-minimal d-separator for pair (*X*, *Y*) in *G*.

It has been demonstrated, that it is sufficient to augment algorithm with the two rules – the presented just above one and the "lack of separator's pivot" rule – to provide ability to identify any structure in a class of forest or poly-forest (poly-tree) by test of zero- and first-rank only [7, 8]. For an ADG-model the following rule also may be useful.

Rule of 'alien gene'. If for some *Z* there exists *W* in a digraph *G* such that ~Ds(*W*;;*Z*) & ~Ds(*Z*;;*X*) & ~Ds(*Z*;;*Y*) & Ds(*W*;;*X*) & Ds(*W*;;*Y*) holds in *G*, then vertex *Z* is not a member of any locally-minimal d-separator for pair (*X*, *Y*) in *G*.

One can suspect that in a situation with known temporal order of vertices (variables) the SIIA-rules would lose their effectiveness and utility. Indeed, if vertex *Z* is later than both *X* and *Y*, obviously *Z* is not a member of any LoMS for pair (*X*, *Y*). Then placing *Z* aside of pair (*X*, *Y*) would be needless and useless. But the rule of 'placing aside' can still work (retaining the same utility) in the cases where vertex *Z* is earlier of both *X* and *Y*, and where *Z* is between *X* and *Y*. The latter case is illustrated in figure 1(a). So, we can expect that effectiveness of the rule would be near 67% of that for setting with unknown temporal order. Further, it may seem that the rule of 'alien gene' would be absolutely useless under known temporal order of vertices, because a vertex *Z* we are going to 'filter out' is collider vertex → *Z* ← on a path between *X* and *Y*, and hence "must be" later than (*X*, *Y*). But this is illusive guess. One can see in figure 1(b) a structure where the rule of 'alien gene' can effectively work for *Z* which precedes (*X*, *Y*). To perceive this example, one should imagine that (to prevent the picture overloading) not all paths are depicted in figure 1(b), and at current state of inference this *Z* is in set of adjacent to *X* or (and) to *Y* because respective separator(s) yet hasn't been found.

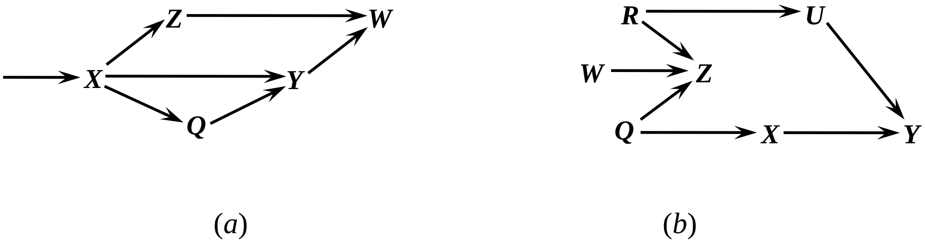

Fig. 1. Examples where SIIA-rules can work under known temporal order:
(*a*) a structure for the rule of 'placing aside'; (*b*) a structure for the rule of 'alien gene'.

To obtain empirical counterparts of SIIA-rules it is needed to replace a graphical predicates Ds(*) in the rule's bodies by isomorphic empirical predicates Pr(*). Such conversion is principally justified by the Causal faithfulness assumption. Appropriate weakened versions of the Causal faithfulness assumption are presented in [4, 5, 9]. Of course, applying empirical SIIA-rules for inference from real data sample brings additional risk of mistakes.

4 Experimental Evaluation of New Version of Algorithm (Razor-1.2)

We have developed a series of new independence-based algorithms (which we call 'Razor'). We have kept the main principles of the PC algorithm, but have enforced our algorithms and augment they with SIIA-rules proposed above. Results of logical simulation (when d-separation facts are given in the input of the algorithm) have confirmed the correctness of our algorithms. Evaluation of the first version of algorithm (Razor-1.1) has revealed a great gain in runtime. Razor-1.1 performs a model inference 3-10 times faster than PC does [8, 9]. But Razor-1.1 often performs worse in accuracy (it commits more extra-edges). In aiming to overcome this drawback we have developed a more cautious and conservative version (Razor-1.2) of algorithm. The Razor-1.2 reduces a search space less radically, also orients edges more carefully and can operate with bi-directed edges. (Note that most of the rules of inductive inference acceleration may be extended to the case of Causal networks with latent variables [6]).

To examine the developed algorithm, a series of Bayesian networks were randomly generated with binary and ternary variables. Each generated structure contains 30 variables; number of edges was 60, 75, 90 and 120. Temporal order of variables hasn't been given. Results of experiments with data samples of size 20000 are presented just below. To evaluate an accuracy of inference, we have concentrated attention on causal arcs only (i.e. arcs that should be perfectly identified as directed from one variables to another). So, the accuracy of algorithms was measured by causal productivity (a percentage and accuracy of recovery of causal edges). Absolutely all accomplished experiments have demonstrated better performance of Razor-1.2 algorithm relative to that of PC. Causal productivity of Razor-1.2 varies between 6% and 45% (vs. 1%–21% for PC) for different models. Totally, algorithm Razor-1.2 has correctly recovered 3.9 times more causal edges than PC. At the same time the number of tests executed by algorithm Razor-1.2, was 1.5 times less than that of PC. So, algorithm Razor-1.2 performs inferring Bayesian networks (of moderate and rather high density) significantly faster and more accurately than PC algorithm does.

5 Conclusion

We have demonstrated a new principled framework to enforce independence-based algorithms for inferring Bayesian networks and causal networks. Novelty of our contribution comes from implementing the rules of inductive inference acceleration, which can radically reduce a space of search for separators, thus reducing computational complexity. The rules express a subset of Markov properties concerning (locally-)minimal d-separating sets. The rules remain to be effective even when a temporal ordering of vertices (variables) is given (although the gain in effectiveness may decrease). Improving of model inference from data is achived by equipping the algorithm with empirical versions of the rules. Experiments have confirmed that our algorithm Razor-1.2 performs inference of Bayesian nets (of moderate density) significantly faster and more accurately then well-known PC algorithm does. Although applying the empirical rules may result in missing a separator (and committing extra-edges), better accuracy is still attained due to cutting a risky regions of search space where misleading independence may likely be encountered. We are confident in perspectives to farther improve 'Razor' algorithms and to extend their ability to more general settings.

References

[1] Pearl J. CAUSALITY: models, reasoning, and inference. – Cambridge Univ. Press, 2000. – 526 p.

[2] Spirtes P., C. Glymour, and R. Scheines. Causation, prediction and search. (2nd Ed.), New York: MIT Press, 2001. – 543 p.

[3] Learning high-dimensional directed acyclic graphs with latent and selection variables / D. Colombo, M.H. Maathuis, M. Kalisch, T. S. Richardson. *Annals of Statistics*. – (2012). – Vol. 40. – N 1. – P. 294–321.

[4] Balabanov A.S. Minimal separators in dependency structures: Properties and identification. *Cybernetics and Systems Analysis*. – (2008). – Vol. 44. – No 6. – P. 803–815. – Springer N.Y.

[5] Balabanov A.S. Construction of minimal d-separators in a dependency system. *Cybernetics and Systems Analysis*. – (2009). – Vol. 45. – No 5. – P. 703–713.

[6] Balabanov O.S. Logic of minimal separation in causal networks. *Cybernetics and Systems Analysis*. – (2013). – Vol. 49. – No 2. – P. 191–200.

[7] Balabanov O.S. Accelerating algorithms for Bayesian network recovery. Adaptation to structures without cycles (in Ukrainian). *Problems in programming journal*. – (2011). – No 1. – P.63–69. – Kiev, Ukraine, ISBN 1727-4907.

[8] Balabanov O.S. Acceleration of Inductive Inference of Causal Diagrams. In: *Proc. of the Intern. Workshop on Inductive Modeling* (IWIM-2011), July 4-11, 2011. – P.16–21. Kyiv, Ukraine 2011. – ISBN 978-966-02-6078-8. (Revised version can be found at http://eprints.isofts.kiev.ua/634/)

[9] Fast algorithm for learning the Bayesian networks from data / A.S. Balabanov, A.S. Gapyeyev, A.M. Gupal, S.S. Rzhepetskiy. *Journal of Automation and Information Sciences*. – (2011). – Vol. 43. – Issue 10. – P. 1–9.