

MBE, 17(6): 6557–6572. DOI: 10.3934/mbe.2020342 Received: 28 June 2020 Accepted: 20 September 2020 Published: 28 September 2020

http://www.aimspress.com/journal/MBE

Research article

Worry less about the algorithm, more about the sequence of events

Farrokh Alemi*

Health Informatics Program, Department of Health Administration and Policy, George Mason University, Fairfax VA, USA

* Correspondence: Email: falemi@gmu.edu; Tel: 7038933799.

Abstract: Background: Many algorithms exist for learning network structure and parameters from data. Some of these algorithms include Grow Shrink, Incremental Association Markov Blanket, IAMB, Fast IAMB, and Interleaved IAMB, Hill Climbing, Restricted Maximization and Maximum-Minimum Hill Climbing. These algorithms optimize the fit to the data, while ignoring the order of occurrences of the variables in the network structure. Objective: This paper examines if sequence information (i.e., one variable occurs before another) can make algorithms for learning directed acyclical graph networks more accurate. Methods: A 13variable network was simulated, where information on sequence of occurrence of some of the variables was assumed to be known. In each simulation 10,000 observations were generated. These observations were used by 4 conditional dependency and 4 search and score algorithms to discover the network from the simulated data. Partial sequence was used to prohibit a directed arc from a later variable to an earlier one. The Area under the Receiver Operating Curve (AROC) was used to compare the accuracy of the sequence-constrained and unconstrained algorithms in predicting the last node in the network. In addition, we examined the performance of sequence constrained algorithms in a real data set. We analyzed 1.3 million disability assessments done on 296,051 residents in Veterans Affairs nursing homes; where the sequence of occurrence of variables was inferred from the average age of occurrence of disabilities. We constructed three networks using Grow-Shrink algorithm, one without and the other two use two permutation of the observed sequence. The fit of these three models to data was examined using Bayesian Information Criterion (BIC). Results: In simulated data, algorithms that used sequenced constraints (AROC = 0.94, confidence intervals, C.I. = 0.86 to 1) were significantly more accurate than the same algorithm without use of sequence constraints (AROC = 0.74, C.I. = 0.65 to 0.83). The agreement between discovered and observed networks improved from range of 0.54 to 0.97 to range of 0.88 to 1.

Keywords: network models; grow shrink; incremental association Markov Blanket; IAMB; fast IAMB; interleaved IAMB; hill climbing; restricted maximization; maximum-minimum hill climbing; sequence constraints; reverse causality; causal networks; directed acyclical graph

1. Introduction

This paper examines if sequence information can make algorithms for learning directed acyclical graph networks more accurate. By sequence we mean whether one variable occurs before another. For example, we can assume that race of a patient is established before history of infection because race occurs at birth and infection later. Historically, sequence information is ignored, or used after the data has been modelled. For example, regression analysis fits a model to the data with no regard for sequence among the independent variables. This paper examines if the sequence information could help improve the accuracy of fit of the model to data.

There are three types of algorithms for learning network structures: search and score [1-13], conditional independency [14-20], and multivariate methods [21-26]. These algorithms must accomplish two tasks: Determine the structure of the network (i.e., the arcs that connect the nodes in the network) and orient the direction of the arcs (sequence). All network learning algorithms, with some exceptions [9,27], determine the direction of the arcs after establishing the network structure. Historically, in learning networks from data, sequence is determined after the network structure has been established. For example, Pearl suggested the collider test for directing the arcs in a network after establishing the network structure [28]. To do the reverse, to first establish the sequence and then estimate the network, goes against the practice of many established algorithms for learning networks. The aim of this study is to show the importance of changing this tradition.

In the published literature, there are sporadic reports of use of sequence in learning network structures [9,2931]. A good example is Shojaie and Michaildis method, where the full order among variables is used to learn network models by iteratively fitting multivariate regression models [27]. Elsewhere, we have also shown that sequence information could make some search and score algorithms more accurate [32,33]. These occasional reports in the literature have raised the possibility that sequence information may improve not only the few algorithms, where the effect has been reported, but all network-learning algorithms. In this paper, we check if sequence information changes the accuracy of different network-learning algorithms.

2. Materials and methods

2.1. Methods of learning sequence

There are many methods for establishing sequence among a pair of variables. Pearl's collider test is one method, there are also other ways:

(a) Definition of variables. The easiest way to establish a sequence is to define the variables so that one occurs before another. Many statisticians already do so. They assume, or design studies, so that some variables are measuring events at baseline, other variables refer to treatment after baseline and outcomes are measured last. These assumptions about timing of events allow an easy way to establish a partial sequence. For example, by definition, race occurs at birth; it is established prior to medical history, which occurs before current conditions; which occurs prior to treatment, which also occurs prior to outcomes. These events by definition create a partial sequence among the variables. In this paper, in analysis of simulated data, we use the definition of variables to establish a partial sequence.

(b) Error reduction. Most statistical measures are symmetric measures and cannot be used to order a pair of variables. This is not the case for Goodman and Kruskal error reduction [34]. Vang used this method to examine sequence among predictors of drug abuse [35].

(c) Strength of causal reasoning. In human judgment, causes have a bigger impact on the effects than the reverse. Zargoush and colleagues used this method to examine sequence among disabilities [32].

(d) Age of patients at time of occurrence of the variables. For example, a number of cross-sectional studies show that feeding disabilities occur after walking disabilities because they occur at a later age [33,36–38]. In this paper, in analysis of real data, we use this method to establish a sequence of events/variables.

(e) Longitudinal order. This method is the gold standard for defining sequence among variables. These data are widely available in electronic health records, where events are time-stamped and therefore it is easy to examine which event has occurred before another.

Sequence information is widely available, multiple methods exist to extract it, and Pearl's collider test is just one among many methods. Bayesian probability networks have focused mostly on the collider test which requires knowledge of the network structure. Other methods of learning sequence do not require knowledge of the network structure and therefore can be implemented before the network structure is discovered.

It is important to point out that often sequence of occurrence among a pair of variables is not deterministic. Sequence between a pair of variables is an uncertain judgment that needs to be based on data. Naturally, these inferences could be erroneous. Sequences derived from data will almost never apply to all individuals in the sample. The majority of patients may experience one sequence of variables, while a minority may experience the reverse order. For example, consider substance abuse disorders, heart attack, and Alzheimer's diseases. Most patients will report substance abuse in younger age than heart attack. In addition, majority of patients will report Alzheimer's disease at a later age than heart attack. Despite these patterns, it is also possible that some patients have these diseases in reverse order; meaning that substance abuse can occur after heart attack or Alzheimer's disease could occur before heart attack. When we talk of sequence among a pair of variables, we are referring to the experience of majority of patients and not all patients. We assume that such estimates could be erroneous and require sensitivity analysis to check if small perturbation of the order could radically change the network. For example, we may require that one variable should occur on average at least 3 months before another. If variables are less than 3-months apart on average, then we may assume that their sequence is unknown leading to a partial knowledge of the sequence of variables.

2.2. Sequence wrapper

A wrapper is a code that repeatedly calls an algorithm and provides it with a subset of variables to analyze [19]. Many algorithms have a random starting point. In sequence wrappers, algorithms start with variables in the order provided by the sequence wrapper. In this method, only relationships that are found consistently across all time periods are used to establish the network structure. Figure 1 shows the procedure for sequence wrapping an algorithm.



Figure 1. Time-based forward-pass wrapper.

One way to implement sequence constraints is through the blacklist feature of existing algorithms' software. In the algorithm's software, the blacklist feature prohibits arc directions from being considered as candidates in learning the network structure. In this approach, arcs from later to earlier time periods are prohibited. The variables that occur in first time period have no prohibited directions. Variables that occur at the last time period have maximum number of

directions prohibited. Variables in between the first and last time periods have directed arcs prohibited proportional to how late they occur. The later a variable occurs the more restrictions it has. The procedure increases the probability that the network learning algorithm first analyzes the earlier variables, de facto implementing the main feature of sequence wrapper.

2.3. Methods of analysis

We test our hypothesis by examining the impact of partial sequence on the following eight algorithms:

- (a) Grow Shrink [19],
- (b) Incremental Association Markov Blanket [39,40],
- (c) Interleaved Incremental Association Markov Blanket [31]
- (d) Fast Incremental Association Markov Blanket [34]
- (e) Hill Climbing [28],
- (f) Maximum-Minimum Hill Climbing [33],
- (g) Restricted Maximization, and
- (h) Taboo [8,9].

Note that Grow Shrink and the three variations of Incremental Association Markov Blanket are all based on conditional independence tests and focus on constructing a Markov Blanket. The remaining four are search and score algorithms and do not rely on search for a Markov Blanket. All eight ignore sequence information during the learning of network structure.

We used the implementation of these algorithms in BNLearn R Package (version 4.0). The sequence information was used to blacklist the impact of later variables on earlier ones. For example, if one variable was judged to occur later than another, then we blacklisted the impact of the later variable on the earlier variable.

2.4. Methods of simulation

To test the performance of these algorithms, we simulated a 13-variable network using the Binomial distribution. This network describes how various cost overruns in a 90-day episode of treatment of hip fracture will affect CMS's bundled payment. Figure 1 shows the network used to simulate the data. In this network, Durable Medical Equipment cost (DME), Clinical Laboratory tests (CL), Physician bills (P) are assumed to affect Hospital (H) costs. Likewise, hospital costs are assumed to affect Long Term Hospital cost (LTH), Rehabilitation Facility cost (RF), Skilled Nursing Facility cost (SNF), Hospice cost (HOS) and eventually CMS's Bundled Payment cost (BP). Physician billing is expected to affect Home Health Agency cost (HHA), SNF, RF, and LTH. LTH and Part B Drug cost (PBD) are expected to affect Hospital Outpatient (HO) cost, and PBD is expected to affect HO and Outpatient Therapy cost (OT). OT is also

affected by RF, SNF, and HHA. Finally, BP is affected by seven variables including H, HOS, HO, OT, PBD, SNF, and HHA.



Figure 2. Predicting bundled payment cost overrun from related costs. Durable Medical Equipment (DME), Clinical Laboratory (CL), Physician (P) Hospital (H), Long Term Hospital (LTH), Rehabilitation Facility (RF), Skilled Nursing Facility (SNF), Home Health Agency (HHA), Part B Drug (PBD), Hospital Outpatient (HO), Outpatient Therapy (OT), Hospice (HOS), Bundled Payment (BP) cost overruns.

The simulation was done in two steps. First, a network structure was organized. We call this network the original structure (Figure 2). In the original network, the odds of any variable with a single parent was set to 2.33:1, namely a conditional probability of 70%. For each additional parent, the odds were added. For example, the presence of 2 parents increased the odds to 4.66:1 or a conditional probability of 82%. Also note that the original network in Figure 2 includes variables that affect not only the next time period but several time periods later. These variations in lag time of impact make learning the network structure more difficult and thus make the simulated data a reasonable environment to test model accuracy.

The network structure in Figure 2 was used to randomly generate 10,000 cases using the Netica software (5.24). We chose a large number of cases so that sample size was not a factor in our estimate of accuracy. It is well known that small sample sizes deteriorate the performance of all algorithms. This is well established and not central to our paper and therefore we did not manipulate this variable in our simulations.

The sequence among the variables in this simulated data can be inferred from the definition of the variables and the assumed continuity of care across locations for treatment of hip-fractured patients. The starting time for all patients is on admission to the hospital. Variables that describe treatment during hospitalization occur prior to discharge (first row in Figure 2). At discharge (second row), patients are typically discharged to institutional care for post-acute recovery (third row). Later, these patients may receive care in outpatient settings (fourth row). Bundled payment (fifth row) is made 90 days after admission. Each row in Figure 2 describes one time period. In total there are 5 time periods.

The accuracy of the network model was examined in two ways. First, we used Kappa statistic to examine the extent to which the discovered and the original network agreed. Second, we examined predictive accuracy of the network by examining Area under Receiver Operating Curves (AROC) for the last node in the network: Bundled Payment Cost Overrun.

We did not distinguish between arc errors or direction errors. Thus, to be considered a correct prediction, the algorithm had to be accurate in both the arc and the direction. An algorithm's predictions may differ from the original network in two ways: first it may indicate a directed arc where it does not exist (a commission error) or miss indicating a directed arc where it does exist (omission error). Based on omission and commission errors, an algorithm can have different levels of sensitivity and specificity. The Receiver Operating Curve indicates the tradeoffs between sensitivity and specificity achieved by the algorithm.

2.5. Result for simulated data

Figure 3 indicates the performance of the algorithms before and after sequence constraints. For all algorithms, the AROC of the sequenced algorithms (dashed lines) was significantly higher than AROC of the un-sequenced algorithms (solid lines). The space between the two lines shows how much the sequence information improved the accuracy of the algorithm. Note also that the improvement in the Score-Based algorithms (i.e., Taboo, Maximum-Minimum Hill Climbing, and Restricted Maximization) was not as large as the algorithms that relied on conditional dependency tests (i.e., Grow Shrink, and IAMB variations).

Table 1 shows the AROC for the eight algorithms. The first column of data shows the performance of these algorithms before using sequence constraints. The last column shows the performance after using sequence constraints. Table 1 also presents the 95% confidence intervals for the AROCs [34]. The differences between two sets of performance are not statistically significant if the two confidence intervals overlap. For example, before used of sequence information the Grow Shrink algorithm had AROC of 0.71 and afterwards it had AROC of 0.98. The confidence intervals for these two estimates are not overlapping and therefore the differences are statistically significant. In fact, for all eight algorithms, the sequence information improved the performance of the algorithm significantly. The magnitude of improvement was large, in some instances the improved AROC was near 1.





	Area under receiver operating curve to predict bundled payment cost overrun		Kappa between discovered and original network	
Algorithm	Not Sequenced	Sequenced	Not Sequenced	Sequenced
Grow Shrink	0.71 (0.65 to 0.77)	0.98 (0.96 to 0.99)	0.70 (0.57 to 0.83)	0.94 (0.88 to 1)
IAMB	0.73 (0.67 to 0.78)	0.99 (0.98 to 1)	0.70 (0.57 to 0.83)	0.94 (0.88 to 1)
Fast IAMB	0.75 (0.68 to 0.80)	0.99 (0.98 to 1)	0.70 (0.57 to 0.83)	0.94 (0.88 to 1)
Interleaved IAMB	0.75 (0.69 to 0.80)	0.99 (0.99 to 1)	0.70 (0.57 to 0.83)	0.94 (0.88 to 1)
Hill Climb	0.75 (0.69 to 0.81)	0.91 (0.87 to 0.94)	0.68 (0.54 to 0.83)	0.94 (0.88 to 1)
Max Min Hill Climb	0.74 (0.69 to 0.80)	0.90 (0.87 to 0.94)	0.88 (0.79 to 0.97)	0.94 (0.88 to 1)
Restricted Max	0.74 (0.68 to 0.79)	0.90 (0.86 to 0.94)	0.88 (0.79 to 0.97)	0.94 (0.88 to 1)
Taboo	0.78 (0.72 to 0.83)	0.91 (0.87 to 0.94)	0.88 (0.79 to 0.97)	0.94 (0.88 to 1)

Table 1. Predictive and construct validity of learned networks.

Values in parentheses indicate 95% confidence intervals.

2.6. Methods for real data

We test the value of sequence in establishing a model for predicting prognosis of nursing home patients. In choosing to join hospice program, clinicians, patients and their caregivers would benefit from accurate information on when and how the patient will die. Often how a person dies translates into knowing how disabled the patient will be before dying. A network model can predict both mortality as well as disabilities that nursing home patients may experience. Nursing home clinicians and residents may use the model to identify what disability may occur next and take actions to prevent it. They may also use the model to assess the value of continuing as is or seeking hospice care. To accurately model the transition among disabilities and death, we examined data on 296,051 residents in Veterans Affairs nursing homes called Community Living Centers (CLCs). The period of study was from 1/1/2000 through 9/10/2012. These data included a comprehensive assessment of residents' disabilities, including ability to walk, bathe, groom, dress, sit (transfer), toilet, eat, bowel, and urinate independently. The data also included the resident's age and whether they had died within 6-months of assessment. By policy, assessments were supposed to be done within 14 days of admission, at least quarterly, or sooner when there was an event such as hospitalization, or when the nursing home staff identified changes in the residents' status. In our data, there were two peaks in the distribution of assessments; one for residents assessed monthly (75,994 residents) and the other for resident assessed every 3 months (42,904 residents). The average time between assessments was 115 days and the standard deviation was 235 days. More than 1.3 million assessments were available for analysis.

We used the Grow-Shrink algorithm to learn the network structure from the data. We first learned the network without using sequence information and then used the same algorithm now with sequence information. We established the sequence among the variables by examining the age with which the patient reported the disability. Two disabilities were judged to occur at different times, if the average age at which one disability occurred was significantly different (at alpha levels less than 0.001) from the age at which the other occurred. The use of age to determine the sequence among events is common in health care literature [33,36–38]. Nevertheless, because of the large size of the data, it is possible that small differences in age may be statistically significant but clinically meaningless. One solution is to require a minimum effect size, e.g., an average age difference of more than 3 months. We examined the sensitivity of our conclusions to requirement of at least 3-month difference in average age of occurrence of the variables. To establish the networks, we use the implementation of Grow Shrink algorithm in BNLearn R package. The sequence information was used to create a blacklist, where later variables were not allowed to point to earlier variables.

2.7. Results for real data

Table 2 provides a summary of the findings. These data suggest that on average disabilities occur in the following order: Unable to Walk, Unable to Sit, Unable to Bathe, Unable to Eat, Unable to Groom, Unable to Toilet, Unable to Dress, Unable to Bowel, Unable to Urinate, and Death. The differences in average age of these disabilities are all statistically significant at alpha levels less than 0.001. To test the sensitivity of network structure to the estimated rank orders, we also required that the two variables should differ in average age by at least 3 months. Some variables, such as bathing and eating disabilities were reversed. Walking disability occurs at least 3 months prior to sitting disability, so the order does not need to be reversed. Note that by definition death occurs last, on average death occurs at 73.71, although some patients survive beyond 85 years of age.

Using the BNLearn package, we created a blacklist which prohibited arcs from diagnosis with later average ages to diagnoses with earlier ages. For example, using the observed rank order we prohibited an arc going from unable to eat to:

- unable to bathe
- unable to sit
- unable to walk
- and age of 65–74 years

In the alternative rank order, we prohibited an arc going from unable to eat to:

- unable to sit
- and unable to walk
- and age of 65–74 years

Variables	Average Age of Incidence	t-test (p-value) with Next Variable	Observed Order	Alternative Order
65–75 Years	70.30	By definition	1	1
Unable to Walk	71.75	526.49 (< 0.00001)	2	2
Unable to Sit	72.31	836.15 (< 0.00001)	3	3
Unable to Bathe	72.68	880.15 (< 0.00001)	4	5
Unable to Eat	72.70	675.46 (< 0.00001)	5	4
Unable to Groom	72.99	280.05 (< 0.00001)	6	8
Unable to Toilet	73.02	11.14 (< 0.00001)	7	7
Unable to Dress	73.09	179.04 (< 0.00001)	8	6
Unable to Bowel	73.24	35.39 (< 0.00001)	9	10
Unable to Urinate	73.27	394.05 (< 0.00001)	10	9
75–85 Years	79.93	By definition	11	11
85+ Years	88.08	By definition	12	12
Death	73.71	By definition	13	13

Table 2. Timing of various events.

Figure 4a shows the network structure without use of sequence information. The age indicators did not affect the network, so they are absent in the networks in Figure 4. This is not surprisingly as chronological age may not be as important in predicting mortality as history of functional decline. Among the functional abilities, a complex set of relationships emerged. On average, each disability had the other 8 disabilities in its Markov Blanket. On average, there were 5.8 arcs to, or from, each node. There was a total of 29 arcs in the network, 3 of which were not directed. Undirected arcs are not allowed in DAG networks and Pearl has suggested that when one cannot determine the direction of an arc that it can be set randomly, without creating cycles or new colliders in the network. Obviously, such random assignment of direction could be wrong and may affect network predictions.

Surprisingly, the variable "Dead in 6 months post assessment" in the network in Figure 4 pointed to the variables "Unable to Eat", "Unable to Urinate", and "Unable to Sit". This, of course, makes little sense, since by definition death cannot be followed by a change in the resident's functional disability.

We also used the observed sequence information in Table 2 to prohibit arcs from later variables to earlier ones. Again 29 arcs were identified. This time all arcs were directed. Furthermore, the directions of the arcs from death to other disabilities were reversed. In total 18 out of the 29 arcs (62%) changed. We show the arcs in which the sequenced networked differed from the non-sequenced network in red in Figure 4. Each red line in Figure 4a shows how sequence information changed the network. The goodness of fit to the data of the two networks

was assessed through Bayesian Information Criterion score. For large data, the Bayesian Information Criterion is -2 times natural log of the maximum likelihood of observing the data given the model. The model with the lowest Bayesian Information Criterion should be preferred. The sequenced network had a score of -4,105,657 and the non-sequenced network had a higher score of -3,873,261. Sequenced network had 6% better fit (lower Bayesian Information Criterion score) to the data.

We repeated the analysis, this time using the alternative sequence in Table 2. Again, 29 arcs were identified and the sequence information corrected illogical arcs from death to disabilities. In total 9 out of 29 arcs (31%) changed and the Bayesian Information Criterion score of the model increased to -5,107,770. The alternative sequence improved the fit to the data by 24.40% compared to un-sequenced network.



Figure 4. Changes due to sequence. N is no disability, W is unable to walk, B is unable to bathe, E is unable to eat, S is unable to sit, G is unable to groom, T is unable to toilet, R is unable to dress, L is unable to bowel, U is unable to urinate, and D is death in 6 months post assessment. Red arcs show changes between unsequenced and sequenced networks. Sequence is shown counter clockwise starting at N and ending in D.

3. Limitations

This study was limited in several ways. First, we focused on eight algorithms available for learning network structures. Future researchers should examine the effect of sequence-wrappers, or sequence constraints, in improving other algorithms.

Second, in the simulation we assumed that a partial sequence can be established accurately. Obviously, errors in establishing the sequence could reduce the accuracy of sequenceconstrained algorithms. In the real data, we made small changes in order of the variables that were less than 3-month apart from each other. Both the original, and the permutated, order improved the fit of the model to the data. Study findings were not sensitive to small changes in the order of variables, highlighting that analyst need to consider only large, and perhaps obvious, order among the variables.

Finally, and perhaps most important limitation, we simulated only one data structure. We randomly simulated many sets of data from this one original network. It is possible that sequence information is not as helpful in other types of networks. We did not examine how the number of variables, intercorrelation among the variables, and discrete/continuous variables might affect the impact of sequence. Additional research is needed to test the performance of sequence constraints in multiple simulated environments. In the real data, we also examined the impact of sequence in only one data set. Here again, it is possible that the results may be different in other data sets.

4. Discussions

This study has shown that information on sequence improved accuracy of network structure for eight widely used network discovery algorithms. On average, the predictive accuracy improved by 27%. The Area under Operating Characteristic Curve improved by 20%, from 0.75 to 0.95. The agreement between discovered and the original network increased by 22%. These are large jumps in various measures of accuracy. In contrast, changing from one algorithm to another improved the accuracy of predictions by only 6.4%. Therefore, the advice to a scientist might be: never mind the algorithm, focus on getting accurate sequence information. Most of the differences among the 8 algorithms disappeared, when sequence information was used.

In both simulated and real data, the sequenced-constrained network was more accurate than unconstrained networks. There are many possible reasons for this finding. Sequence removes a large number of possible arcs from further consideration; and thus, it reduces the occasions the model can make an error. For example, in the simulated data, there were 156 possible directed arcs. Sequence constraints removed 61 possible directed arcs. Thus, the model had 40% less occasions in which it could make an error. The algorithms were more accurate simply because they had fewer chances of making an error. While this explanation seems reasonable to us, it does not explain the differential impact of sequence on search and score as opposed to conditional dependency algorithms.

We also speculate that sequence information may have been more helpful to algorithms that use Markov Blankets because it reduced errors that occur because of stratification of common effects. When a common effect is stratified, two or more previously independent causes become dependent and show spurious and non-causal associations. To have an intuition about how this might occur consider that later variables are more likely to be an effect than a cause. An algorithm, like Grow Shrink, first assembles candidates for Markov Blanket (Grow Phase) and then verifies them (Shrink phase). This algorithm starts randomly with one variable. The use of blacklists changed the order of considering variables. Despite randomization, variables that occurred earlier in time were more likely to enter the Grow phase than variables that occurred later in time. Furthermore, variables that occurred later in time were more likely to be dropped in the shrink phase. De facto, these two steps reduce stratification on effects and thus improve accuracy of the algorithm.

We hypothesize that sequence data improves accuracy of algorithms because it removes the possibility of "reverse causality". In the real data, the sequence information corrected obvious errors in the network (death pointing to other disabilities), it re-oriented 62% of the arcs, and improved the fit between the network and the data by 6%. For some time, it has been known that expert opinions and algorithms may make the resulting model more accurate [41]. Experts are aware of causal direction of events and thus may correct models that have reverse causality. To an expert, it is clear that "the sun does not rise because the rooster crows". To the computer algorithm, this is not clear at all. Sequence may be playing a similar role to what experts do: It is avoiding reverse causal findings.

Finally, sequence data can radically improve face validity of a model. In our effort to model real data, the face validity of the network was radically improved when obvious errors such as death causing changes in functional disability was avoided. In our experience, when a model makes an obvious error, then no one will trust it any further. In most occasions, reverse causality is obvious error. Any observer may be able to point to these errors. If a model misses something so obvious, then no one will trust it. No statistic can replace the fall from grace of a model that misses obvious relationships. By reducing occasions of reverse causality, sequence improves face validity and increases trust in algorithms.

Acknowledgements

Amr Elrafey completed the data analysis reported in this paper.

Conflict of interest

The author has no conflict of interest to report.

References

- 1. D. Heckerman, D. Geiger, D. Chickering, Learning Bayesian networks: the combination of knowledge and statistical data, *Mach. Learn.*, **20** (1995), 197–234.
- 2. G. Cooper, E. Herskovits, A Bayesian method for the induction of probabilistic networks from data, *Mach. Learn.*, **7** (1992), 299–347.
- 3. W. Buntine, Theory refinement on Bayesian networks, in *Uncertainty in Artificial Intelligence: Proceedings of the Seventh Conference (1991)*, **26** (1991), 52–60.
- 4. W. Lam, F. Bacchus, Theory refinement on Bayesian networks, *Comput. Intell.*, **10** (1994), 269–293.
- 5. C. Wallace, K. Korb, H. Dai, Causal discovery via MML, Int. Confer. Mach. Learn., (1996), 516–524.
- 6. H. Akaike, A new look at the statistical model identification, *IEEE Trans. Autom. Contr.* **19** (1974), 716–723.
- 7. G. Schwarz, Estimating the dimension of a model, *Annals. Stat.*, **6** (1978), 461–464.

- 8. D. Chickering, D. Geiger, D. Heckerman, Learning Bayesian networks: Search methods and experimental results, in *Fifth International Workshop on Artificial Intelligence and Statistics*, (1995), 112–128.
- 9. N. Friedman, I. Nachman, D. Pe'er, Learning Bayesian network structure from massive datasets: The "sparse candidate" algorithm, in *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence (UAI1999)*, 1999.
- 10. P. Na m, P. H. Wuillemin, P. Leray, O. Pourret, A. Becker, *R éseaux Bay ésiens*, Eyrolles, Paris, 2011.
- 11. P. Munteanu, M. Bendou, The EQ framework for learning equivalence classes of Bayesian networks, in *Proceedings of the 2001 IEEE International Conference on Data Mining*, (2001), 417–424.
- 12. W. Lam, F. Bacchus, Learning Bayesian belief networks: An approach based on the MDL principle, *Comput. Intell.*, **10** (1994), 269–293.
- 13. L. Jouffe, P. Munteanu, New search strategies for learning Bayesian networks, in Proceedings of the 10th International Symposium on Applied Stochastic Models and Data Analysis, Compiègne, France, (2001), 591–596.
- 14. J. Cheng, R. Greiner, J. Kelly, D. A. Bell, E. Liu, Learning Bayesian networks from data: an information-theory based approach, *Artif. Intell. J.*, **137** (2002), 43–90.
- 15. P. Spirtes, C. Glymour, R. Scheines, *Causation, Prediction, and Search*, Springer-Verlag, 2000.
- 16. J. Pearl, Probabilistic Reasoning in Intelligent Systems, Morgan Kaufmann, 1988.
- 17. T. Verma, J. Pearl, Equivalence and synthesis of causal models, in UAI '90: Proceedings of the Sixth Annual Conference on Uncertainty in Artificial Intelligence, (1990), 255–270.
- 18. D. Margaritis, *Learning Bayesian Network Model Structure from Data*, Ph.D thesis, Carnegie Mellon University, 2003.
- 19. C. F. Aliferis, I. Tsamardinos, A. Statnikov, HITON: A novel Markov Blanket algorithm for optimal variable selection, *AMIA Annu. Symp. Proc.*, **2003** (2003), 21–25.
- 20. P. Spirtes, An anytime algorithm for causal inference, AISTATS, 2001.
- 21. F. Fu, Q. Zhou, Learning sparse causal Gaussian networks with experimental intervention: regularization and coordinate descent, *J. Am. Stat. Assoc.*, **108** (2013), 288–300.
- 22. B. Aragam, Q. Zhou, Concave penalized estimation of sparse Gaussian Bayesian networks, *J. Mach. Learn. Res.*, **16** (2015), 2273–2328.
- 23. G. I. Allen, Z. A. Liu, A Local Poisson Graphical Model for inferring networks from sequencing data, *IEEE Trans. Nanobiosci.*, **12** (2013), 189–198.
- 24. A. Agresti, Categorical Data Analysis, Wiley-Interscience, 2002.
- 25. G. Park, G. Raskutti, Learning large-scale poisson DAG models based on over dispersion scoring, in *Advances in Neural Information Processing Systems*, (2015), 631–639.
- 26. S. W. Han, H. Zhong, Estimation of sparse directed acyclical graphs for multivariate counts, *Biometrics*, **72** (2016), 791–803.
- 27. A. Shojaie, G. Michaildis, Analysis of gene sets based on the underlying regulatory network, *J. Comp. Biol.*, **16** (2009), 407–426.
- 28. J. Pearl, Causality: Models, Reasoning, and Inference, Cambridge University Press, 2000.

- 29. N. Friedman, D. Koller, Being Bayesian about network structure: A Bayesian approach to structure discovery in Bayesian networks, *Mach. Learn.*, **50** (2003), 95–125.
- 30. P. Larrafiaga, M. Poza, Y. Yurramendi, R. H. Murga, C. M. H. Kuijpers, Structure learning of Bayesian networks by genetic algorithms: A performance analysis of control parameters, *IEEE Trans. Pattern Anal. Mach. Intell.*, **18** (1996).
- 31. M. Teyssier, D. Koller, Ordering-based search: A simple and effective algorithm for learning Bayesian networks, in *Proceedings of the 21st conference on Uncertainty in AI* (2005), 584.
- 32. M. Zargoush, F. Alemi, V. E. Vinzi, J. Vang, R. Kheirbek, A psychological approach to learning causal networks, *Health Care Manage. Sci.*, **17** (2014), 194–201.
- C. R. Levy, M. Zargoush, A. E. Williams, A. R. Williams, P. Giang, J. Wojtusiak, et al., Sequence of functional loss and recovery in nursing homes, *Gerontologist*, 56 (2016), 52–61.
- 34. L. A. Goodman, W. H. Kruskal, Measures of association for cross classifications, Part I, J. *Am. Stat. Assoc.*, **49** (1954), 732–764.
- 35. J. Vang, Using a model of human cognition of causality to orient arcs in structural learning of Bayesian networks, Ph.D thesis, George Mason University, 2008.
- 36. D. D. Dunlop, S. L. Hughes, L. M. Manheim, Disability in activities of daily living: patterns of change and a hierarchy of disability, *Am. J. Public Health.*, **87** (1997), 378–383.
- 37. L. Ferrucci, J. M. Guralnik, F. Cecchi, N. Marchionni, B. Salani, J. Kasper, et al., Constant hierarchic patterns of physical functioning across seven populations in five countries, *Gerontologist*, **38** (2008), 286–294.
- 38. S. Katz, A. B. Ford, R. W. Moskowitz, B. A. Jackson, M. W. Jaffe, Studies of illness in the aged, the index of ADL: A standardized measure of biological and psychological function, *JAMA*, **185** (1963), 914–919.
- 39. I. Tsamardinos, C. F. Aliferis, A. Statnikov, Algorithms for large scale Markov blanket discovery, in *Proceedings of the Sixteenth International Florida Artificial Intelligence Research Society Conference*, (2003), 376–381.
- 40. I. Tsamardinos, L. E. Brown, C. F. Aliferis, The max-min hill-climbing Bayesian network structure learning algorithm, *Mach. Learn.*, **65** (2006), 31–78.
- 41. H. S. Sousa, F. Prieto-Castrillo, J. C. Matos, J. M. Branco, P. B. Louren ço, Combination of expert decision and learned based Bayesian networks for multi-scale mechanical analysis of timber elements, *Expert Syst. Appl.*, **93** (2018), 156–168.



©2020 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0)