Discovery of Influence between Processes Represented by Hidden Markov Models

Ritesh Ajoodha
School of Computer Science and Applied Mathematics
The University of the Witwatersrand, Johannesburg
South Africa
Email: ritesh.ajoodha@wits.ac.za

Benjamin Rosman
School of Computer Science and Applied Mathematics
The University of the Witwatersrand, Johannesburg
South Africa
Email: benjamin.rosman@wits.ac.za

***Abstract*—Learning the underlying structure between processes is a common problem found in the sciences, however not much work is dedicated towards this problem. In this paper, we attempt to use the language of structure learning to address learning the dynamic influence network between partially observable processes represented as hidden Markov models (HMMs). The importance of learning an influence network is for knowledge discovery and to improve density estimation in temporal distributions. We learn the dynamic influence network, defined by this paper, by first learning the optimal distribution for each process using hidden Markov models, and thereafter apply redefined structure learning algorithms for temporal models to reveal influence relationships. This paper provides the following contributions: we (a) provide a definition of influence between stochastic processes represented by HMMs; and (b) expand on the conventional structure learning literature by providing a structure score and learning procedure to learn influence relationships between HMMs. We provide empirical evidence of the effectiveness of our method over several baselines.**

***Index Terms*—Learning influence networks, Structure learning, Hidden Markov models, Stochastic processes.**

I. Introduction

The problem of describing the interaction or influence between stochastic processes has received little scrutiny in the current literature, despite its developing importance. Solving this complex problem has large implications for density estimation and knowledge discovery.

Usually, the individual structure of each stochastic process is ignored and all are merged into one big process which is modelled by some probabilistic temporal model. This approach undermines the explanatory importance of the relations between these processes [1]. The core of the issue is that we lose the underlying structure of the relationships between the processes which is essential to learn how one process influences the other.

In this paper, we provide a complete method for learning the dynamic influence network between processes represented by hidden Markov models (HMMs). This paper also explores the case when we are learning the influence relationship between partially observable processes. This is a significantly harder problem since the likelihood of the temporal model to the data has multiple optima which is induced from the missing samples [1]. Unfortunately, given that learning parameters from missing data is also a NP-hard problem, heuristic approaches are needed to solve for a suitable local optimum to the likelihood function of the parameters [1].

The application of this research is broad. Influence networks for stochastic processes can capture the complex relationships of how processes impact others. For example, we can learn the influence of traffic in a network of roads to determine how the traffic condition of a road will impact on another road. In educational data-mining we may want to determine the influence of participants in a lecture environment to encourage student success. We may wish to learn the influence between an IoT network [2]–[4]; influence between musical pieces [5], [6]; or influence between the skills of learners which impacts on their attrition [7]–[9].

The following contributions is made by this paper: (a) The concept of dynamic influence networks (DINs) which represents the influence (relationships) between partially observable stochastic processes. (b) The extension of the Bayesian information Criterion (BIC) for for dynamic models (d-BIC score). (c) The concept of a structural assemble which is able to relate dynamic models statistically. Finally (d), a greedy structure learning procedure for learning DINs between these HMMs.

II. Related Work

Numerous statistical procedures have been used to identify influence between variables [10]–[13]. These statistical procedures have been extended to the temporal environment to learn relationships between processes (variables over time). A significant contribution is the use of hidden Markov models (HMMs) which is defined as a set of parameters and conditional independence assumptions which together make up an acyclic structure between variables defined using factors [14]–[16]. The values in these factors are referred to as the parameters, and the list of conditional independence assumptions between variables are referred to as the structure of the model.

Learning the independence assertions of a dynamic Bayesian network can be used to make conditional independence inferences over time (density estimation) or to simply learn the relationships between variables (knowledge discovery). [17]–[20]. On the one hand leaning a sparse graph structure may have more generalisability for density

estimation, but on the other hand, having a more dense graph can reveal unknown relationships for knowledge discovery. Care must be taken when considering for what purpose is the network required (more on this in the discussion) [16].

A successful approach to structure learning is using score-based structure learning [16], [21]. In score-based structure learning we develop a set of hypothesis structures which are evaluated using a score-based function that computes the likelihood of the data to the hypothesised structure. The likelihood is usually expressed as the information gain (mutual information) of the structure and parameters of the distribution to the data.

A search algorithm is then performed to identify the highest (possible) structure based on the structure score [22]–[25]. Viewing this problem as an optimisation problem allows us to adopt the already established literature on search methods in this super-exponential space to find the optimal structure given the data [26]–[29].

The structure of this section is as follows. In section II-A we introduce the well established BIC score which offers a way to trade-off the fit to data vs model complexity (the amount of independence assumptions between variables in the data). Finally, in section II-B we introduce a greedy search method to find the an optimal graph structure.

A. The BIC score

The BIC score models the structural fit to data verses the complexity of the conditional independence assumptions between variables. That is, the amount of independence assumptions made on the structure [30]. This makes it a popular choice for structure learning methods since the model complexity has a direct impact on the performance to do inference tasks. This is because the amount of conditional independence assumptions on a particular variable increases the factor size of that variable exponentially. The mathematical expression of the BIC score comprises of two terms: the first term models the fit to data; and the second term penalises the fit to data based on the complexity of the structure considered. The complete BIC score is as follows:

$$score_{BIC} = \ell(\hat{\theta}_{\mathcal{G}} : \mathcal{D}) - \frac{\log M}{2} DIM[\mathcal{G}],$$

where the count of instances is denoted by M and the count of independent parameters is denoted by $DIM[\mathcal{G}]$ in the Bayesian network. $\ell(\hat{\theta}_{\mathcal{G}} : \mathcal{D})$ represents the likelihood fit to the data.

The intuition of the Bayesian score is that as the amount of samples increase (ie. M) the score is willing to consider more complicated structures if enough evidence (samples, ie. M) is considered [31], [32]. The BIC core is particularly effective since the likelihood score (one without a penalty to complexity) will always prefer the most complicated network. However, the most complex networks also impose the risk of fragmentation, which is the exponential increase to the size of the factors caused by the increase of the in-degree of a variable. Penalty-based structure scores allows us to explore the opportunity to adopt more complicated structures if there is enough justification that the likelihood of the structure and parameters to the data is high-enough to compromise on the models speed to perform inference tasks caused by fragmentation.

There has been much contributions in the literature on the properties of the BIC score [30], [33], [34]. Key constitutions include a proof the it is consistent and is score equivalent which are necessary for efficient search procedures [35]–[37].

B. Learning General Graph-structured Networks

Since the search space for the optimal Bayesian structure is super-exponential, the difficulty of learning a graph structure for a Bayesian network is NP-hard. More specifically, for any $d \geq 2$, the problem of finding a structure with a maximum score with d parents is NP-hard [26]–[29]. See [38]–[40] for a detailed proof.

Despite this, there have been many contributions to learning an optimal structure. A key contribution is using heuristic search procedures to find an optimal acyclic graph structure [41]. These heuristic search procedures make use of search operators (changes to the graph structure) and a search algorithm (e.g. greedy search, best first search, simulated annealing e.t.c.) [42]. The intuition of this approach is to find an optimal acyclic structure by gradually improving the choice of the structure using the various search operators [43]–[47].

III. Influence between Hidden Markov Models

In this paper we consider a structure learning procedure which evaluates candidate dynamic influence networks (DINs) using scoring metrics. We provide evidence for the effectiveness of our structure learning procedure over the standard benchmarks selected.

An overview of the proposed structure learning procedure is given by the below instructions relating to Figure 1.

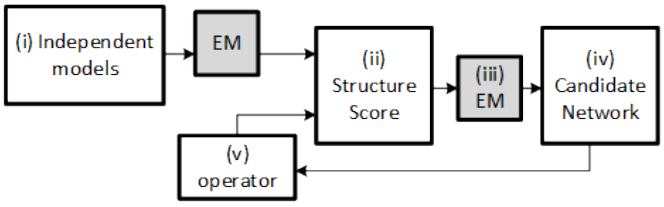

Fig. 1. An overview of the proposed structure learning procedure in this paper.

(i) The stochastic processes are given as input. The parameters for a HMM is learned for each stochastic process. This is the input.
(ii) A structure is imposed between the HMMs (using a relation function called an assemble). This gives us a dynamic influence network (DIN). The observable parameters are relearned in the model. The structure score for the DIN is computed.
(iii) Expectation maximisation is performed to learn the latent parameters of the DIN.
(iv) A candidate DIN is presented as output.

(v) The resulting DIN is evaluated and the score is recorded. If the score converges or a threshold is reached then the learning procedure is terminated. If not, we apply a structural operator (edge addition, deletion, reversal) and move back to step (iii).

Figure 2 illustrates an examples of a DIN between a set of HMMs. Each HMM is represented as a node in the acyclic graph structure and is denoted as a tuple, $\langle \mathcal{H}_0^i, \mathcal{H}_{\rightarrow}^i \rangle$, where $\mathcal{H}_0^i$ is the starting state of the HMM, and $\mathcal{H}_{\rightarrow}^i$ is the unrolled state for HMM i. A DIN structure is the output of the structure learning procedure.

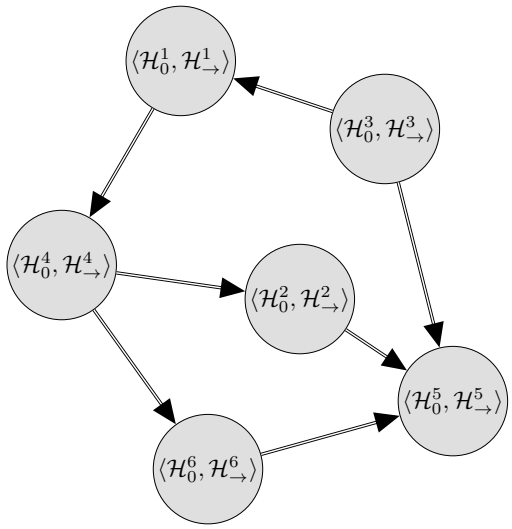

Fig. 2. A dynamic influence network (DIN) whose nodes represent six HMMs. Each HMM is represented as $\langle \mathcal{H}_0^i, \mathcal{H}_{\rightarrow}^i \rangle$, where $\mathcal{H}_0^i$ is the initial network and $\mathcal{H}_{\rightarrow}^i$ is the unrolled network. The double edges between each network represents the structure assemble (subsection III-B).

We will begin by providing a brief introduction to the hidden Markov model (HMM) which is used to represent the stochastic processes. An HMM is a dynamic Bayesian network (DBN). The likelihood function (mutual information) for a HMM, as illustrated by Figure 3, decomposes as:

$$L(\Theta : X^{0:T}, O^{0:T}) = \prod_{i,j} \theta_{X^i \rightarrow X^j}^{M[X^i \rightarrow X^j]} \prod_{i,k} \theta_{O^k|X^i}^{M[X^i, O^k]},$$

where the parameters correspond to the observable value k in the state i to the exponent of the number of times we observe both X^i and O^k. We will often refer to an HMM as a tuple as we did in Figure 2. [16], [48]–[50] provide excellent introduction to DBNs and HMMs.

In the section III-A we will extend the current literature of structure scores for Bayesian networks to scores for DINs; and finally, in section III-B we will introduce the notion of an assemble to relate HMMs in our DIN.

A. Structure Scores for DINs

In Step (ii) of Figure 1, we needed to calculate the score of the influence structure. In this paper we adapt the celebrated

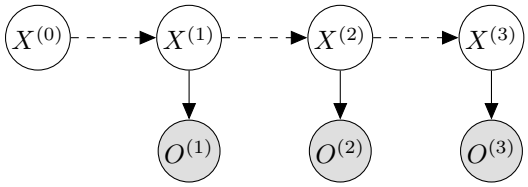

Fig. 3. An illustration of a hidden Markov model (HMM) with 4 time-slices. The dotted lines indicate the inter-time-slice edges for the persistent variable $X^{(t)}$. The solid line indicate the intra-time-slice edges for each respective time-slice.

BIC score to a dynamic BIC (d-BIC) for our dynamic influence networks. The d-BIC score make the same trade-off between model complexity and fit to the data, only the d-BIC can be applied to dynamic networks.

The d-BIC score is as follows:

$$\text{score}_{BIC}(\mathcal{H}_0 : \mathcal{D}) = M \sum_{k=1}^{K} (\sum_{t=1}^{T} (\sum_{i=1}^{N} (\mathbf{I}_{\hat{P}}(X_i^{\langle \mathcal{H}_0^k, \mathcal{H}_{\rightarrow}^k \rangle^{(t)}}; \mathbf{Pa}_{X_i^{\langle \mathcal{H}_0^k, \mathcal{H}_{\rightarrow}^k \rangle^{(t)}}}^{\mathcal{G}}))) - \frac{\log M}{c} DIM[\mathcal{G}],$$

where the amount of samples is given by M; the amount of dependency models is given by K; the amount of time-slices is given by T for any dependency model; the amount of variables in each time-slice is given by N; $\mathbf{I}_{\hat{P}}$ denotes the information gain in terms of the empirical distribution; and $DIM[\mathcal{G}]$ is the amount of independent parameters in the entire DIN.

The d-BIC score is designed to exchange the complexity of the DIN, $\frac{\log M}{c} DIM[\mathcal{G}]$, for the fit to the data, $\mathcal{D}$. As the amount of samples increases, the information gain term grows linearly, and the model complexity part grows logarithmically. The intuition of the d-BIC score is that we will be willing to consider more complicated structures, if we have more data that justifies the need for a more complex structure (i.e. more conditional independence assumptions).

B. Structure Assembles

Choosing the set of parent variables in a DIN establishes the notion of a structural assemble. A structural assemble is a template which relates temporal models. The structural assemble defines the parent sets for variables to construct a DIN. More specifically, the assemble relation is defined as follows:

Consider a family of hidden Markov models (H), where $\langle H_0^0, H_{\rightarrow}^0 \rangle$ represents the child with the parent set $\mathbf{Pa}_{\langle H_0^0, H_{\rightarrow}^0 \rangle}^{\mathcal{G}} = \{\langle H_0^1, H_{\rightarrow}^1 \rangle, \ldots, \langle H_0^k, H_{\rightarrow}^k \rangle\}$. Further assume that $\mathcal{I}(\langle H_0^j, H_{\rightarrow}^j \rangle)$ is the same for all $j = 0, \ldots, k$. Then the *delayed* dynamic influence network, denoted by $\langle \mathcal{A}_0, \mathcal{A}_{\rightarrow} \rangle$, will satisfy all the independence assumptions in $\mathcal{I}(\langle H_0^i, H_{\rightarrow}^i \rangle)$ $\forall$ $i = 0, \ldots, k$. In addition, $\forall$ j and $\forall$ t, $\langle \mathcal{A}_0, \mathcal{A}_{\rightarrow} \rangle^{(t)}$ also satisfies the following independence assumptions for each hidden or latent variable denoted L_i and some $t > \alpha \in \mathbb{Z}^+$:

$$\forall\ L_i^{\langle H_0^0, H_\rightarrow^0\rangle^{(t)}} : (L_i^{\langle H_0^0, H_\rightarrow^0\rangle^{(t)}} \perp\!\!\!\perp \text{NonDescendants}_{L_i^{\langle H_0^0, H_\rightarrow^0\rangle^{(t)}}} | L_i^{\langle H_0^k, H_\rightarrow^k\rangle^{(t)}}, L_i^{\langle H_0^k, H_\rightarrow^k\rangle^{(t)-1}}, \ldots, L_i^{\langle H_0^k, H_\rightarrow^k\rangle^{(t)-\alpha}}, Pa_{L_i}^{\langle H_0^0, H_\rightarrow^0\rangle^{(t)}}).$$

The assemble is an expressive representation to capture influence relationships that persist through time between temporal models in this case HMMs. However, the choice of α is important since choosing a large α will render many dependencies on variables. This causes a fragmentation bottleneck which causes a larger computational burden for learning and inference tasks.

To illustrate an example of using an assemble relation between two HMMs, $\langle \mathcal{A}_0, \mathcal{A}_\rightarrow \rangle$ and $\langle \mathcal{B}_0, \mathcal{B}_\rightarrow \rangle$, consider Figure 4. Figure 4 unrolls two HMMs, $\langle \mathcal{A}_0, \mathcal{A}_\rightarrow \rangle$ and $\langle \mathcal{B}_0, \mathcal{B}_\rightarrow \rangle$, using a structural assemble with $\alpha = 0$.

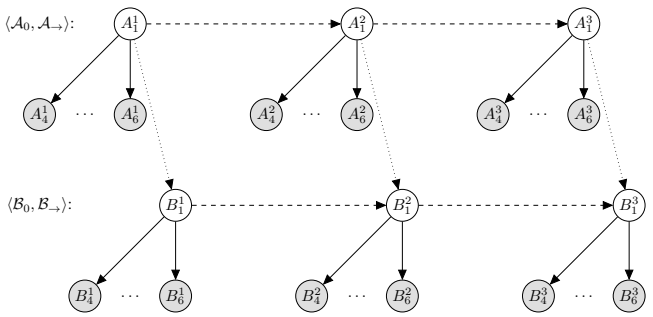

Fig. 4. Two unrolled HMMs, $\langle \mathcal{A}_0, \mathcal{A}_\rightarrow \rangle$ and $\langle \mathcal{B}_0, \mathcal{B}_\rightarrow \rangle$, as represented with 3 time-slices. The HMMs are connected by a structural assemble with $\alpha = 0$.

C. Structure Search

At this point we have the following well-defined optimisation problem:

1) A training set $\mathcal{H}_{\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^\mathcal{G}} = \{\mathcal{H}_{\langle H_0^1, H_\rightarrow^1 \rangle}, \ldots, \mathcal{H}_{\langle H_0^k, H_\rightarrow^k \rangle}\}$, where $\mathcal{H}_{\langle H_0^i, H_\rightarrow^i \rangle} = \{\xi_1, \ldots, \xi_M\}$ is a set of M instances from underlying ground-truth HMM $\langle H_0^i, H_\rightarrow^i \rangle$;
2) a structure score: score$(\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle : \mathcal{H}_{\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^\mathcal{G}})$;
3) and, finally, we have an array of L distinct candidate structures, $\mathcal{G} = \{\mathcal{G}^1, \ldots, \mathcal{G}^L\}$, where each structure $\mathcal{G}^l$ represents a unique list of condition independence assertions $\mathcal{I}(\mathcal{G}) = \mathcal{I}(\mathcal{G}^\mathbb{I} \cup \mathcal{G}^\mathbf{B})$.

Our objective of this optimisation problem is to output the DIN which produces the maximum score. We present the following influence structure learning procedure in Algorithm 1, where $\mathcal{S} = \{\mathcal{S}_\rightarrow^1, \ldots, \mathcal{S}_\rightarrow^P\}$ represents the set of stochastic processes; $assemble$, is the option of the parameters for a structure assemble; and $score$, which is the selected scoring function used by the search procedure.

Algorithm 1 Influence structure search

1: **procedure** STRUCSEARCH($\mathcal{S} = \{\mathcal{S}_\rightarrow^1, \ldots, \mathcal{S}_\rightarrow^P\}$, $assemble$, $score$)
2: for each process we learn a temporal model ($H = \{\langle H_0^1, H_\rightarrow^1 \rangle, \ldots, \langle H_0^P, H_\rightarrow^P \rangle\}$)
3: Using the models in H we generate a search space (ie. $\mathbf{G} = \{\mathcal{G}_1, \ldots, \mathcal{G}_n\}$)
4: Find the structure $\mathcal{G}_i$ which produces the highest $score$ (w.r.t. $assemble$) in $\mathbf{G}$
5: **return** $\mathcal{G}_i$
6: **end procedure**

The dynamic influence network, $\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^\mathcal{G}$, holds a distribution between a set of HMMs, denoted $\langle H_0^1, H_\rightarrow^1 \rangle$, $\ldots$, $\langle H_0^k, H_\rightarrow^k \rangle$, with the conditional independence assumptions listed by $\mathcal{I}(\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^\mathcal{G})$. We further assume that $P^*(\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^\mathcal{G})$ is induced by another model, $\mathcal{G}^*(\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^\mathcal{G})$, we will refer to this model as the underlying ground-truth model. The model is evaluated by recovering the set of local independence assertions in $\mathcal{G}^*(\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^\mathcal{G})$, denoted $\mathcal{I}(\mathcal{G}^*(\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^\mathcal{G}))$, by only observing $\mathcal{H}_{\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^\mathcal{G}}$. This structure learning procedure is referred to by this paper as the greedy structure search (GESS).

D. Computational Complexity and Savings

The overall computational complexity of the above structure search algorithm is given by [1]. In order to allow for notable computational savings we suggest using a cache to store sufficient statistics and using a max priority queue (implemented using heaps) to arrange contending structures using their scores as keys. Random restarts and Tabu lists are also used to manage the structure search procedure.

IV. Empirical Results

This sections presents the performance of modelling influence between partially observable stochastic processes represented by HMMs using DINs. We evaluate the performance of our model aside several benchmarks.

The experimental setup is as follows. We constructed a ground-truth DIN which was used to sample sequential data. To simulate a partially observed process, several variables were removed from the sequential data sample. Algorithm 1 was used to learn candidate networks. Several variations of the algorithm was also used, such as using the d-AIC (dynamic Akaike Information Criterion) score instead of the d-BIC; using prior knowledge of the ground-truth structure such as the maximum in-degree used in the generative distribution; using a random structure; and even using no structure.

The parameters for the ground-truth DIN distribution is summarised by Table I. The ground-truth DIN distribution described the influences between 10 processes, each represented using HMMs with 5 time-slices, 2 hidden layers, 5 observable variables and 3 latent variables per time-slice. Each variable could take 3 discrete values. The overall ground-truth DIN had a max in-degree of 2 for any variable; and finally, the number of conditional independence assumptions (CIA) between processes was limited to 15.

The results of the experiment is summarised by Figure 5, which shows the relative entropy to the generative ground-truth DIN over the number of training samples used.

The results are averaged over 10 trials for various structure learning tasks:

TABLE I
A TABLE SUMMARISING THE PARAMETERS FOR THE GROUND-TRUTH DIN DISTRIBUTION.

Ground-truth DIN Distribution	
No. HMMs	10
Random variable values	3
No. time-slices	5
No. layers	2
No. CIAs between HMMs	15
max in-degree	2
No. Obs	5 p.t.
No. Latent	3 p.t.

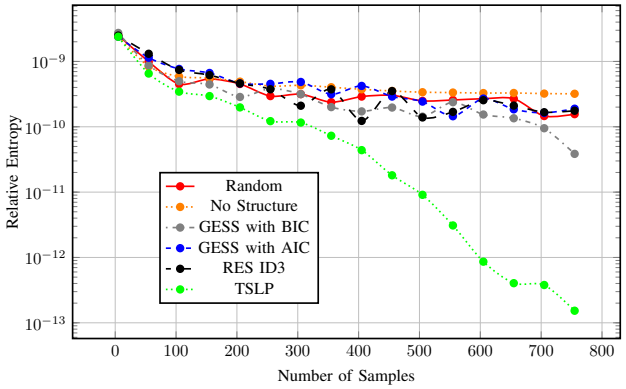

Fig. 5. The performance of parameter and structure learning tasks.

- Random, which used a randomly generated structure for a DIN and learned the missing and observable parameters;
- No structure, which modelled each HMM as mutually independent to others and learned parameters;
- d-BIC with GESS, which used the d-BIC score with GESS and learned missing and observable parameters;
- d-AIC with GESS;
- RES ID3, which used the likelihood score but restricted recovered structures with an in-degree greater than 3 and also learned parameters;
- and finally, TSLP (Learning Procedure with the True structure), which used the ground-truth DIN structure, but relearned missing and observable parameters.

The parameters of the experiment were as follows: 20 EM iterations were used for learning and latent variables. There were 50 structure search iterations used to recover each model (5 random restarts when reached local optima, and used a tabu list of length 10). All learned variables used a Dirichlet Prior of 5. To allow for a manageable use of memory all DIN with over 5000 independent parameters were heavily penalised.

The reported results suggest that, on the one hand, when we have fewer samples we are better-off not using any structure, since fewer parameters allow us to generalise better. On the other hand, when we have a sufficient amount of data, then a random structure gives us more information than no structure at all. The reason the random structure does better is because the likelihood to the data of a structure with more conditional independence assumptions rather than fewer will also be greater.

The three penalty-based score methods do better than both random and no structure. However we find that the sensitivity of the d-BIC score to judge when to constrain the structure (based on the number of training sample) guides the selection of the independence assumptions and outperforms the d-AIC score and the restricting the in-degree method. As expected, knowing the true structure gives us the most information and thus outperforms all the methods as the number of observations increase.

V. CONCLUSION AND IMPLICATIONS

In this paper we empirically demonstrated a score-based structure learning procedure to learn a DIN to represent the influence relationships between partially observable stochastic processes.

Why we would want to learn a DIN depends on what it will be used for. On the one hand, if we are trying to identify the original DIN for knowledge discovery, then we will need to identify each of the original conditional independence assumptions of the ground-truth network. This means we will need to find the set $\mathcal{I}(\mathcal{G}^*(\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^{\mathcal{G}}))$. This is not a pragmatic task since there are many perfect maps for $P^*(\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^{\mathcal{G}})$ that can be derived from $\mathcal{D}_{\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^{\mathcal{G}}}$.

Recognising $\mathcal{I}(\mathcal{G}^*(\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^{\mathcal{G}}))$ from the set of structures from $\mathcal{G}^*(\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^{\mathcal{G}})$ will yield the same fit to the data. Therefore identifying the original ground-truth structure is not identifiable from $\mathcal{H}_{\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^{\mathcal{G}}}$. This is because the structures in the I-equivalent structure set all produce the same numeric likelihood (mutual information) for $\mathcal{H}_{\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^{\mathcal{G}}}$. Therefore, we should rather try to learn a set of structures that are I-equivalent to $\mathcal{G}^*$.

On the other hand, if instead we are trying to learn a DIN for density estimation (i.e. to draw probabilistic inferences), then we are interested in capturing the distribution $P^*(\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^{\mathcal{G}})$. If we can successfully construct such a distribution then we can reason about or sample new data instances.

There are two implications when learning a structure or density estimation: Firstly, Although capturing more independence assertions than specified in $\mathcal{I}(\mathcal{G}^*(\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^{\mathcal{G}}))$ may still allow us to capture $P^*(\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^{\mathcal{G}}$, our selection of more independence assumptions could result in *data fragmentation*. Secondly, selecting very sparse structures can restrict us to never being able to learn the true distribution $P^*(\langle \mathbb{I}_0, \mathbb{I}_\rightarrow \rangle^{\mathcal{G}}$ no matter how we change the parameters. However, often sparse DIN structures can be used to promote computational complexity savings [16].

ACKNOWLEDGEMENTS

This work is based on the research supported in part by the National Research Foundation of South Africa (Grant number: 121835).

REFERENCES

[1] R. Ajoodha, "Influence modelling and learning between dynamic bayesian networks using score-based structure learning," *The University of the Witwatersrand, Johannesburg (wirespace)*, 2019.

[2] R. Ajoodha and B. Rosman, "Learning the influence structure between partially observed stochastic processes using iot sensor data," in *Workshops at the Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[3] R. Ajoodha and B. Rosman, "Tracking influence between naïve bayes models using score-based structure learning," in *2017 Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech)*. IEEE, 2017, pp. 122–127.

[4] R. Ajoodha and B. Rosman, "Learning the influence between partially observable processes using score-based structure learning," *Advances in Science, Technology and Engineering Systems Journal. Special Issue on Multidisciplinary Sciences and Engineering*, 2020.

[5] A. Anshel and D. A. Kipper, "The influence of group singing on trust and cooperation," *Journal of Music Therapy*, vol. 25, no. 3, pp. 145–155, 1988.

[6] R. Ajoodha, R. Klein, and B. Rosman, "Single-labelled music genre classification using content-based features," in *IEEE proceedings, Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech), 2015*, Nov 2015, pp. 66–71.

[7] R. Ajoodha, A. Jadhav, and S. Dukhan, "Forecasting learner attrition for student success at a south african university," in *In Conference of the South African Institute of Computer Scientists and Information Technologists 2020 (SAICSIT '20), September 14-16, 2020, Cape Town, South Africa. ACM, New York, NY, USA, 10 pages.* ACM, 2020.

[8] R. Ajoodha, S. Dukhan, and A. Jadhav, "Data-driven student support for academic successby developing student skill profiles," in *International Multidisciplinary Information Technology and Engineering Conference. ISBN: 978-1-7281-9519-9.* IEEE, 2020.

[9] T. Abed, R. Ajoodha, and A. Jadhav, "A prediction model to improve student placement at a south african higher education institution," in *2020 International SAUPEC/RobMech/PRASA Conference*. IEEE, 2020, pp. 1–6.

[10] J. Hatfield, G. J. Faunce, and R. Job, "Avoiding confusion surrounding the phrase "correlation does not imply causation"," *Teaching of Psychology*, vol. 33, no. 1, pp. 49–51, 2006.

[11] R. Opgen-Rhein and K. Strimmer, "From correlation to causation networks: a simple approximate learning algorithm and its application to high-dimensional plant gene expression data," *BMC systems biology*, vol. 1, no. 1, p. 37, 2007.

[12] T. Grinthal and N. Berkeley Heights, "Correlation vs. causation," *AMERICAN SCIENTIST*, vol. 103, no. 2, pp. 84–84, 2015.

[13] D. Commenges and A. Gégout-Petit, "A general dynamical statistical model with causal interpretation," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 71, no. 3, pp. 719–736, 2009.

[14] M. Bunge, *Causality and modern science*. Routledge, 2017.

[15] W. Salmon, "Scientific explanation and the causal structure of the world," 1984.

[16] D. Koller and N. Friedman, *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[17] D. Heckerman and D. Geiger, "Learning bayesian networks: a unification for discrete and gaussian domains," in *Proceedings of the Eleventh conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 1995, pp. 274–284.

[18] A. Mohammadi and E. C. Wit, "Bayesian structure learning in sparse gaussian graphical models," *Bayesian Analysis*, vol. 10, no. 1, pp. 109–138, 2015.

[19] A. L. Madsen, F. Jensen, A. Salmerón, H. Langseth, and T. D. Nielsen, "A parallel algorithm for bayesian network structure learning from large data sets," *Knowledge-Based Systems*, vol. 117, pp. 46–55, 2017.

[20] X. Fan, C. Yuan, and B. M. Malone, "Tightening bounds for bayesian network structure learning." in *AAAI*, 2014, pp. 2439–2445.

[21] C. P. d. Campos and Q. Ji, "Efficient structure learning of bayesian networks using constraints," *Journal of Machine Learning Research*, vol. 12, no. Mar, pp. 663–689, 2011.

[22] S. Kok and P. Domingos, "Learning the structure of markov logic networks," in *Proceedings of the 22nd international conference on Machine learning*. ACM, 2005, pp. 441–448.

[23] J. B. Tenenbaum, C. Kemp, T. L. Griffiths, and N. D. Goodman, "How to grow a mind: Statistics, structure, and abstraction," *science*, vol. 331, no. 6022, pp. 1279–1285, 2011.

[24] I. Tsamardinos, L. E. Brown, and C. F. Aliferis, "The max-min hill-climbing bayesian network structure learning algorithm," *Machine learning*, vol. 65, no. 1, pp. 31–78, 2006.

[25] S.-I. Lee, V. Ganapathi, and D. Koller, "Efficient structure learning of markov networks using l_1-regularization," in *Advances in neural Information processing systems*, 2007, pp. 817–824.

[26] D. M. Chickering, D. Geiger, and D. Heckerman, "Learning bayesian networks is np-hard," Technical Report MSR-TR-94-17, Microsoft Research, Tech. Rep., 1994.

[27] D. M. Chickering, "Learning bayesian networks is np-complete," in *Learning from data*. Springer, 1996, pp. 121–130.

[28] D. M. Chickering, D. Heckerman, and C. Meek, "Large-sample learning of bayesian networks is np-hard," *Journal of Machine Learning Research*, vol. 5, no. Oct, pp. 1287–1330, 2004.

[29] J. Suzuki, "An efficient bayesian network structure learning strategy," *New Generation Computing*, vol. 35, no. 1, pp. 105–124, 2017.

[30] G. Schwarz *et al.*, "Estimating the dimension of a model," *The annals of statistics*, vol. 6, no. 2, pp. 461–464, 1978.

[31] S. Chen and P. Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the bayesian information criterion," in *Proc. darpa broadcast news transcription and understanding workshop*, vol. 8. Virginia, USA, 1998, pp. 127–132.

[32] Y. Tamura, T. Sato, M. Ooe, and M. Ishiguro, "A procedure for tidal analysis with a bayesian information criterion," *Geophysical Journal International*, vol. 104, no. 3, pp. 507–516, 1991.

[33] J. Rissanen, "Stochastic complexity," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 223–239, 1987.

[34] A. Barron, J. Rissanen, and B. Yu, "The minimum description length principle in coding and modeling," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2743–2760, 1998.

[35] D. Geiger, D. Heckerman, H. King, and C. Meek, "Stratified exponential families: graphical models and model selection," *Annals of statistics*, pp. 505–529, 2001.

[36] D. Rusakov and D. Geiger, "Asymptotic model selection for naive bayesian networks," *Journal of Machine Learning Research*, vol. 6, no. Jan, pp. 1–35, 2005.

[37] R. Settimi and J. Q. Smith, "On the geometry of bayesian graphical models with hidden variables," in *Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 1998, pp. 472–479.

[38] M. Koivisto and K. Sood, "Exact bayesian structure discovery in bayesian networks," *Journal of Machine Learning Research*, vol. 5, no. May, pp. 549–573, 2004.

[39] A. P. Singh and A. W. Moore, "Finding optimal bayesian networks by dynamic programming," 2005.

[40] T. Silander and P. Myllymaki, "A simple approach for finding the globally optimal bayesian network structure," *arXiv preprint arXiv:1206.6875*, 2012.

[41] D. Chickering, D. Geiger, and D. Heckerman, "Learning bayesian networks: Search methods and experimental results," in *proceedings of fifth conference on artificial intelligence and statistics*, 1995, pp. 112–128.

[42] W. Buntine, "Theory refinement on bayesian networks," in *Proceedings of the Seventh conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann Publishers Inc., 1991, pp. 52–60.

[43] A. Moore and M. S. Lee, "Cached sufficient statistics for efficient machine learning with large datasets," *Journal of Artificial Intelligence Research*, vol. 8, no. 3, pp. 67–91, 1998.

[44] K. Deng and A. W. Moore, "Multiresolution instance-based learning," in *IJCAI*, vol. 95, 1995, pp. 1233–1239.

[45] A. W. Moore, "The anchors hierarchy: Using the triangle inequality to survive high dimensional data," in *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 2000, pp. 397–405.

[46] P. Komarek and A. W. Moore, "A dynamic adaptation of ad-trees for efficient machine learning on large data sets." in *ICML*, 2000, pp. 495–502.

[47] P. Indyk, "Nearest neighbors in high-dimensional spaces," 2004.

[48] K. P. Murphy, "Dynamic bayesian networks: representation, inference and learning," Ph.D. dissertation, University of California, Berkeley, 2002.

[49] L. Rabiner and B. Juang, "An introduction to hidden markov models," *ieee assp magazine*, vol. 3, no. 1, pp. 4–16, 1986.

[50] L. R. Rabiner, “A tutorial on hidden markov models and selected applications in speech recognition,” *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.