Multi-Task Transfer Learning for Bayesian Network Structures

Sarah Benikhlef, Philippe Leray, Guillaume Raschia, Ben Messaoud, Fayrouz Sakly

Multi-Task Transfer Learning for Bayesian Network Structures

Sarah Benikhlef[1], Philippe Leray[1][0000−0002−0207−9280], Guillaume Raschia[1][0000−0001−7968−262X], Montassar Ben Messaoud[2], and Fayrouz Sakly[2]

[1] LS2N, UMR CNRS 6004, University of Nantes, France
{sarah.benikhlef,philippe.leray,guillaume.raschia}@ls2n.fr
[2] LARODEC, ISG Sousse, Tunisia
montassar.benmessaoud@gmail.com,saklyfayrouz@outlook.fr

Abstract. We consider the interest of leveraging information between related tasks for learning Bayesian network structures. We propose a new algorithm called Multi-Task Max-Min Hill Climbing (MT-MMHC) that combines ideas from transfer learning, multi-task learning, constraint-based and search-and-score techniques. This approach consists in two main phases. The first one identifies the most similar tasks and uses their similarity to learn their corresponding undirected graphs. The second one directs the edges with a Greedy Search combined with a Branch-and-Bound algorithm. Empirical evaluation shows that MT-MMHC can yield better results than learning the structures individually or than the state-of-the-Art MT-GS algorithm in terms of structure learning accuracy and computational time.

Keywords: Bayesian Networks · Structure learning · Multi-task learning · Transfer learning.

1 Introduction

Learning reliable models from small datasets is difficult, therefore transfer learning (also known as domain adaptation [21]) can enhance the robustness of the discovered models by leveraging data from related tasks [12]. Transfer learning is a well-studied area. It has been successfully employed in a variety of machine learning fields, focusing mainly on neural networks [13, 21].

Multi-Task (MT) learning, also referred to as parallel transfer learning, is a learning paradigm that aims to leverage useful information contained in related tasks to help improve the generalization of all the tasks [20]. In order to considerably increase learning tasks performance, MT learning can be combined with other learning mechanisms [20].

Inductive transfer and MT learning are closely related. They both intend to leverage knowledge among similar problems [10]. The distinction between these two approaches lies in the transfer technique. In Transfer Learning (TL), the information is transferred from the source to the target task. Ultimately, the goal is to improve the performance of the target task with the support of source

tasks by affording additional information. We note here that the target task has a more significant role than source tasks. On the contrary, the tasks in MT learning are considered equally [17].

Bayesian Networks (BNs) have proven to be an efficient tool to capture conditional dependencies and independencies between random variables. They give an effective way to represent the structure of real-world applications and determine the effect of many observations on an outcome. They are frequently employed in decision support systems and machine learning applications, where it is generally assumed to have sufficient data from which a reliable model can be learned. However, in some fields, as manufacturing or medicine [7], data can be rare and usually gathered from different but closely related problems. In this situation, existing solutions have been proposed for transfer and multi-task learning of Bayesian networks [8, 11, 10]. These approaches are mainly adaptations of basic constraint-based or score-based BN structure learning algorithms.

In this paper, we propose an extension to multi-task learning of an efficient BN structure discovery algorithm called Max-Min Hill-Climbing (MMHC) [18] that takes advantage of both constraint based and score-based algorithms.

Our three main contributions are: (1) MT-MMHC, a hybrid transfer learning algorithm that can learn multiple BN structures simultaneously by inducing information between similar tasks to improve the performance of the constructed networks; (2) one procedure to generate MT benchmarks from any reference model, by controlling the similarity between tasks; and (3) one experimental validation of MT-MMHC by using such benchmarks compared to Single Task (ST) structure learning, but also to MT-GS reference algorithm (Multi Task Greedy Search).

Section 2 introduces background information and relevant related work about BN structure learning. Section 3 describes our MT-MMHC approach. Section 4 is dedicated to the generation of MT benchmarks and to the empirical evaluation of our proposition. Finally, section 5 gives a general conclusion and several research perspectives.

2 Bayesian Network Multi-task structure learning

A Bayesian Network (BN) $\mathcal{B} =< G, \Theta >$ represents the joint probability distribution of a set of n random variables $\{X_1, X_2, ..., X_n\}$ [4]. It is characterized by a Directed Acyclic Graph (DAG) G and a set Θ of Conditional Probability Tables (CPTs), also called parameters. The graph or structure G is given by a pair (V, E) where V is the set of nodes in the graph and E the set of edges between them. Each node corresponds to a random variable and each edge represents a direct dependency between the two variables that it connects. A strong property of BNs is that this representation is easy to interpret and can help in visualizing dependencies between variables.

BN Single-Task (ST) learning aims at discovering the structure of the network G and estimating the parameters Θ of the model from one dataset D.

In Multi-Task (MT) learning, we consider k tasks corresponding to k datasets $\mathcal{D} = \{D_1, D_2, ..., D_k\}$ from which we learn k corresponding BN graphs $\mathcal{G} = \{G_1, G_2, ..., G_k\}$ and the associated parameters. The objective is to learn all the models simultaneously as a multi-task problem while leveraging information between the tasks. It is worth to notice here that Azzimonti et al. [1] propose an alternative BN learning approach where one common BN structure is learned from related data sets.

MT parameter learning, when $\mathcal{G}$ is known, have been processed in the literature [4, 8]. For instance, Luis et al. [8] present a method of inductive transfer to determine the CPTs using aggregation functions from several sources.

In this paper, we are mainly interested in Bayesian network MT structure learning. In a single task context, discovering the structure of a Bayesian network can be seen as a problem of selecting a probabilistic model that fits and explains a given dataset [4]. Three main approaches are generally adopted [4] : (1) search and score based methods exploring the space of structures; (2) constraint-based methods using conditional independencies; and (3) hybrid methods combining the two previous ones.

Search and score approaches One of the most widely known methods of learning Bayesian network structures is the use of search and score techniques [4]. These approaches perform an exact or a heuristic search within the space of network structures and evaluate the best candidate structure to fit the data using a given scoring metric. Generally speaking, these algorithms require : (i) a search space of allowable states of the problem, each state represents a Bayesian network structure; (ii) a scoring function to evaluate a state and see how well it matches the data; (iii) a mechanism to explore this space in an exact way when the dimension of the search space is limited, or in a heuristic way.

Greedy Search (GS) [14], for example, is a ST structure learning algorithm that starts from some initial structure and explores the DAG space by selecting at each iteration the neighbor graph with the highest score. The search stops when the current structure has a better score than all its neighbors, and may be repeated several times starting from different initial states to avoid local maxima. A neighborhood of a structure G is commonly defined as all the DAGs obtained by removing or reversing an existing edge in G, or by adding a new one. For states evaluation, several scores have been settled in the literature such as AIC, BIC or BDeu [3] or more recent ones such as qNML [15].

In [10], Niculescu-Mizil and Caruana extend this score-based search in a MT context. Let us denote this algorithm MT-GS. They consider a configuration of k structures $\{G_1, G_2, ..., G_k\}$ and then follow a greedy search procedure in the space of DAGs to find the k graphs that fit the best the k respective datasets. The neighborhood of a configuration is defined as the set of all configurations obtained by considering all possible subsets of the graphs and applying an add, remove or reverse operation to the same edge in each graph of each subset. The score to maximise is the posterior probability of a configuration given the data defined in Eq. 1.

$$P(\mathcal{G}|\mathcal{D}) = P(G_1, ..., G_k|D_1, ..., D_k) \propto P(\mathcal{G}) \prod_{a=1}^{k} P(D_a|G_a) \tag{1}$$

They propose two different priors (i) an edit prior which considers the minimum number of updates needed to make an edge similar in every structure and (ii) a paired prior that considers the differences among each pair of structures as defined in Eq. 2.

$$P(\mathcal{G}) = Z_{\delta,k} \prod_{1 \leq a \leq k} P(G_a)^{\frac{1}{1+(k-1)\delta}} \prod_{1 \leq a < b \leq k} (1-\delta)^{\frac{d(G_a,G_b)}{k-1}} \tag{2}$$

where $\delta \in [0, 1]$ is a parameter that penalizes every difference between the models' structure when calculating the prior, $Z_{\delta,k}$ is a normalization constant and $d(G_a, G_b)$ is the number of edges in the symmetric difference between G_a and G_b. Thus, the scoring function takes into account data from all the tasks and leverages information between them.

As the search space can get large for large k and n (the number of variables in datasets), they provide a computational optimization based on a Branch-and-Bound strategy to find at each step the best configuration in the neighborhood of the current one. To this end, they define a partial configuration of order ℓ, $\mathcal{C}_\ell = (G_1, ..., G_\ell)$, as a configuration where only the first ℓ structures are specified and the other $k-\ell$ structures are not. This exploration goes through a search tree of depth k to reach the best scoring configuration. At each level $\ell < k$, only the score of the partial configuration $\mathcal{C}_\ell = (G_1, ..., G_\ell)$ is computed and compared to the current best score. The score of a partial configuration is defined as an upper bound to the scores of all complete configurations C that match it. By this way, each sub-tree rooted at a partial configuration whose score is lower than the current best score can be pruned.

Constraint-based approaches Algorithms following this approach test conditional independencies between variables in the data, and progressively identify the graph that describes these dependencies and independencies discovered in the data [4].

In a transfer learning context, Jia et al. [5] address the problem of constraint-based learning with inductive transfer. Luis et al. [8] propose a constraint-based structure learning for BNs in a MT setting. The general outline of the algorithm is inspired from the (single task) PC algorithm [16]. It starts with a fully connected undirected graph, and measures the association between variables to decide if an edge should be removed from the graph or not. The major difference is the way the independence tests are evaluated. It is replaced by a linear combination of independence measures from the target task with the closest auxiliary task, where closeness is determined by the combination of two metrics: a global similarity Sg defined in Eq. 3 and a local similarity Sl defined in Eq. 4.

The global similarity measure Sg_{ab} computes the number of common dependencies and independencies between every possible pair of variables (X, Y) in

task a and task b (i.e. in their corresponding datasets D_a and D_b)[8].

$$Sg_{ab} = \sum_{X<Y} \mathbb{1}(I_a(X,Y) - I_b(X,Y)) \tag{3}$$

where $I_a(X,Y)$ and $I_b(X,Y)$ are respectively the result of an independence test between variables X and Y performed on datasets D_a and D_b.

The local similarity measure $Sl_{ab}(X,Y|S)$ compares independencies between two variables X and Y given a subset of variables S [8].

$$Sl_{ab}(X,Y|S) = \begin{cases} 1, & \text{if } I_a(X,Y|S) = I_b(X,Y|S). \\ 0.5, & \text{otherwise.} \end{cases} \tag{4}$$

where $I_a(X,Y|S)$ and $I_b(X,Y|S)$ are respectively the result of the conditional independence test between variables X and Y given S performed on datasets D_a and D_b. Based on the two previous metrics, Luis et al. [8] define the combined similarity measure $Sc_{ab}(X,Y|S)$ as:

$$Sc_{ab}(X,Y|S) = Sg_{ab} \times Sl_{ab}(X,Y|S) \tag{5}$$

For a given task a, the confidence measure α_a estimates the confidence of the independence test between X and Y given the conditioning set S and is defined as:

$$\alpha_a(X,Y|S) = 1 - \frac{\log N_a}{2N_a} \times T \tag{6}$$

where $T = |X| \times |Y| \times |S|$, with $|x|$ is the cardinality of x, and where N_a is the size of the dataset D_a.

Finally, the combined independence function I_a, that computes the independence test between X and Y given S in task a with inductive transfer learning, is a linear weighted combination of the independence measures in a and in its most similar task b^* (with respect to the combined similarity Sc) :

$$\begin{aligned} Ic_a(X,Y|S) &= \alpha_a(X,Y|S) \times sgn(I_a(X,Y|S)) \\ &+ \alpha_{b^*}(X,Y|S) \times Sc_{ab^*}(X,Y|S) \times sgn(I_{b^*}(X,Y|S)) \end{aligned} \tag{7}$$

where $sgn(I)$ is $+1$ if X and Y are independent given S and -1 otherwise.

Hybrid approaches These approaches combine the features of the constraint-based and the score-based algorithms. To our knowledge, they are only proposed so far for single task learning. Generally, these algorithms start by implementing a constraint-based strategy to reduce the space of candidate DAGs, then they perform a score-based strategy to find an optimal DAG in the restricted space. In this context, we can cite the Max-Min Hill-Climbing (MMHC) algorithm proposed in [18]. MMHC is a hybrid approach for single task Bayesian structure learning that first identifies the skeleton of the graph with a constraint-based method (named MMPC for Max-Min Parent Children) then it selects and directs the interesting edges using a search-and-score procedure.

The MMPC algorithm uses an association metric $A(X, Y|S)$ such as Mutual information or χ^2 to estimate the strength of the dependency between X and Y given S and it performs a conditional independence test $I(X, Y|S)$ from this metric. The algorithm progressively identifies for each variable X a set of candidate parents and children $CPC(X)$ (without distinction between parent or child).

Throughout the edge direction assignment step, a greedy search is performed to determine the DAG that best fits the data. The important difference from standard greedy search is that the search space is constrained by the fact that candidate edges must be consistent with the $CPCs$ discovered by MMPC.

3 The MT-MMHC algorithm

The BN structure learning algorithms mentioned in section 2 mainly perform transfer learning in single task (ST) scenario with constraint based approaches [8] or multi-task consideration with search-and-score methods [10]. In our contribution, we propose a hybrid approach for multi-task problems that we call MT-MMHC. The purpose is to learn k BN structures from k similar problems simultaneously, combining benefits from constraint-based algorithms, score-and-search based algorithms, TL and MT learning techniques.

3.1 Overall process of MT-MMHC

The main idea is to extend the MMHC algorithm to the MT scenario, as shown in Fig. 1. As its ST counterpart, the procedure starts with a constraint-based phase to identify the CPC sets associated to each task. It performs a local search technique ensured by the MMPC algorithm (refer to the grey boxes in Fig. 1). For transfer learning adaptation, we propose in section 3.2 a combined association metric. In the second phase, for edge orientation, we apply the MT greedy search algorithm proposed in [10] adapted to our context by constraining it to the discovered CPCs as described in section 3.3.

3.2 The combined association measure

The first phase of the MT-MMHC algorithm that we denote MT-MMPC, consists in k parallel MMPC with a new combined association measure, to identify the upper bound CPC of the skeleton of each model (refer to the grey boxes in Fig. 1).

Inspired by the work in [8], we propose in Eq. 8 this new association metric taking into account the MT setting.

$$Ac_a(X, Y|S) = \frac{\alpha_a(X, Y|S)A_a(X, Y|S) + \alpha_{b*}(X, Y|S)Sc_{ab^*}(X, Y|S)A_{b*}(X, Y|S)}{\alpha_a(X, Y|S) + \alpha_{b*}(X, Y|S)Sc_{ab*}(X, Y|S)} \quad (8)$$

where $A_a(X, Y|S)$ is the usual association measure between two variables X and Y given a subset S from the dataset D_a and $A_{b*}(X, Y|S)$ expresses the

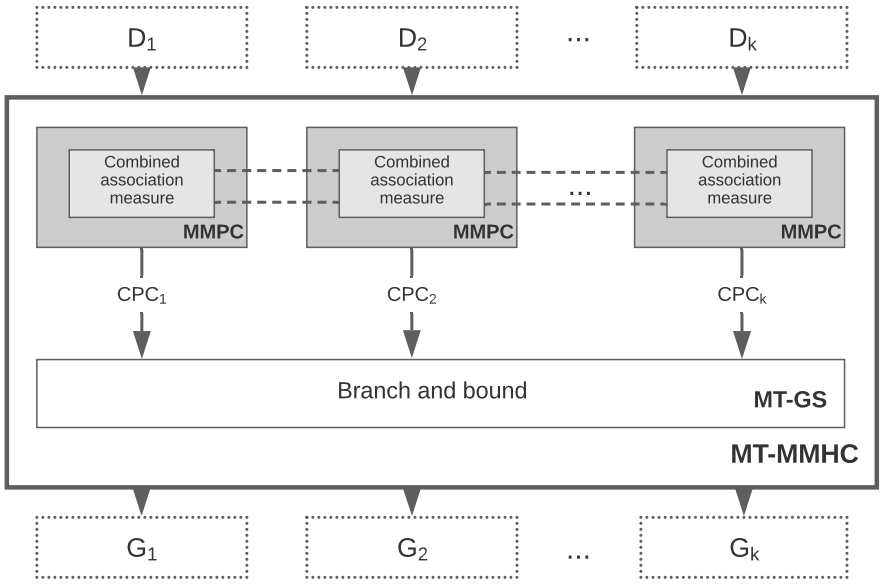

Fig. 1. Overall process of MT-MMHC.

association measure between the two same variables X and Y given the subset S from dataset D_{b*} of the closest task $b*$ determined by the combined similarity measure Sc.

This combined association measure allows us to transfer information between similar tasks while computing independence tests. We propose using Ac_a instead of the combined independence function I_a defined in Eq.7 for the following reason: as a convex combination of two association measurements, this value can also be interpreted as an association measurement. I_a is the non convex linear combination of two signs of independence tests which is only used for its sign whereas the MMPC algorithm requires also the information provided by the strength of the association between variables.

3.3 MT greedy search with CPC constraints

MT-MMPC, the constraint-based phase of MT-MMHC, outputs a CPC set $\{CPC_1, CPC_2, ..., CPC_k\}$ for each task.

In the second phase of MT-MMHC, as inspired by the single task MMHC, we propose to apply the MT-GS algorithm (cf. Fig. 1) described in section 2. The main difference lies in the input of the algorithm and how we accordingly bound the search space by using the information provided by the $CPCs$.

The neighborhood of a configuration $\{G_1, G_2, ..., G_k\}$ is generated by applying for each pair of nodes in each possible subset of graphs an edit operation (add, remove, reverse or leave unchanged an edge). We adapt the generation of

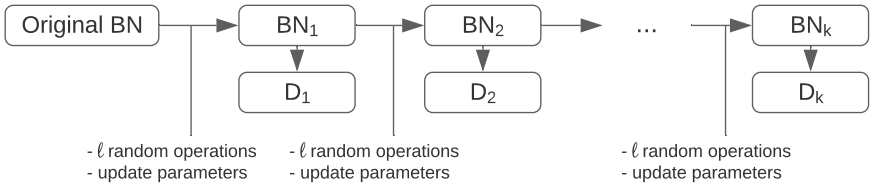

Fig. 2. A procedure to generate multi-task benchmarks from a reference BN.

Table 1. Description of the BNs used in our experimentation.

Name	Nodes	Arcs	Max in-degree	Description
Asia	8	8	2	Used for patient chest clinic diagnosis given symptoms and risk factors
Alarm	37	46	4	Medical diagnosis for monitoring intensive care patients

this neighborhood by allowing the addition of the edge $X \rightarrow Y$ in a graph G_a only if $X \in CPC_a(Y)$.

By using this definition of a neighborhood, we can perform a MT greedy search procedure while keeping all the properties of the greedy search in the context of MT with the constraints provided by the result of MT-MMPC. Hence, we can reduce the search space and decrease the computational cost of the greedy search. As the neighborhood size of a configuration $\{G_1, G_2, ..., G_k\}$ can get large for large k and n, MT-MMHC could then be more scalable than MT-GS.

4 Experiments

In this section, we present an empirical evaluation of MT-MMHC as a comparative study with state-of-the-art single task BNs structure learning algorithms GS and MMHC, and also with the MT-GS algorithm.

4.1 Experimental protocol

Benchmark and data generation When benchmarks for BN structure learning in single task context are quite popular with reference models such as ASIA [6] or ALARM [2] (see Table. 1), there are no benchmarks for evaluating MT algorithms. We propose here one simple procedure that takes as an input one of the reference models used for ST learning, and generates a set of k MT reference models by controlling the similarity between each model.

As a first version of this procedure, we generate k similar networks by applying one random walk between each model (by applying randomly ℓ usual operators: add, remove or reverse edge) and recomputing randomly the parameters (see Fig. 2).

We can then generate one dataset for each model by applying the usual forward sampling algorithm, with a potentially different number of observations that we call data size N_a for each task a.

In our work, we generated three series of experiments, with small datasets size ($N_a \in [500, 1000[$), medium size ($N_a \in [1000, 5000[$) and large size ($N_a \in [5000, 10000[$) for $k = 5$ different tasks generated with $\ell = 1$.

Algorithms We have implemented several algorithms in PILGRIM[1], our C++ library dedicated to probabilistic graphical models. We propose to compare MT-MMHC (described in section 3), MT-GS (cf. section 2) and the independent running of k single task structure learning algorithms kST-GS and kST-MMHC. In our experiments, we used mutual information as a measure of association, with $\alpha = 5\%$ for the independence tests, and BIC as an approximation of each marginal likelihood $P(D_a|G_a)$ in Eq. 1 and $\delta = 1e - 7$ as a penalty in Eq. 2.

Evaluation metrics We measure performances in terms of run time and Structural Hamming Distance (SHD) between the true structures (from which we generated sampling data) and the learned ones, and more exactly the distance between the essential graphs as proposed in [18]. For each experiment, we propose the mean and standard deviation of SHD over 10 runs x 5 tasks, and the mean and standard deviation of the execution time over the 10 runs.

4.2 Empirical results

MT-MMHC versus kST-GS and kST-MMHC Fig. 3 (top) presents performances in terms of SHD (the lower the better) for the three approaches MT-MMHC, kST-GS and kST-MMHC with respect to three categories of data size, and MT benchmarks generated from ASIA and ALARM networks.

The trends in the performances are as expected: learned networks are more accurate with larger datasets, and kST-MMHC performs better than kST-GS except for small datasets. It is worth to note that MT-MMHC is able to find better networks than the single task approaches.

Fig. 3 (bottom) shows the average execution time to learn five tasks BNs. Single task MMHC is the fastest approach in all experiments, but we can notice that MT-MMHC is much faster than ST greedy search in a medium network like Alarm thanks to the space reduction strategy. For the small network Asia, MT-MMHC is the slowest but still runs in quite reasonable time (from 2.5 to 17.5 seconds on average). For instance, in the case of the largest data slice from ASIA, MT-MMHC is 17% slower but 40% more accurate than kST-GS.

MT-MMHC versus MT-GS Fig. 4 presents performances in terms of SHD and execution time for MT-MMHC and MT-GS with respect to the sizes of datasets, for MT benchmarks generated from the ALARM network.

For small datasets, MT-GS is slightly better than MT-MMHC with a much higher execution time. For medium and larger datasets MT-MMHC performs better in terms of quality of the learned model and runs with an affordable time cost.

[1] https://pilgrim.univ-nantes.fr/

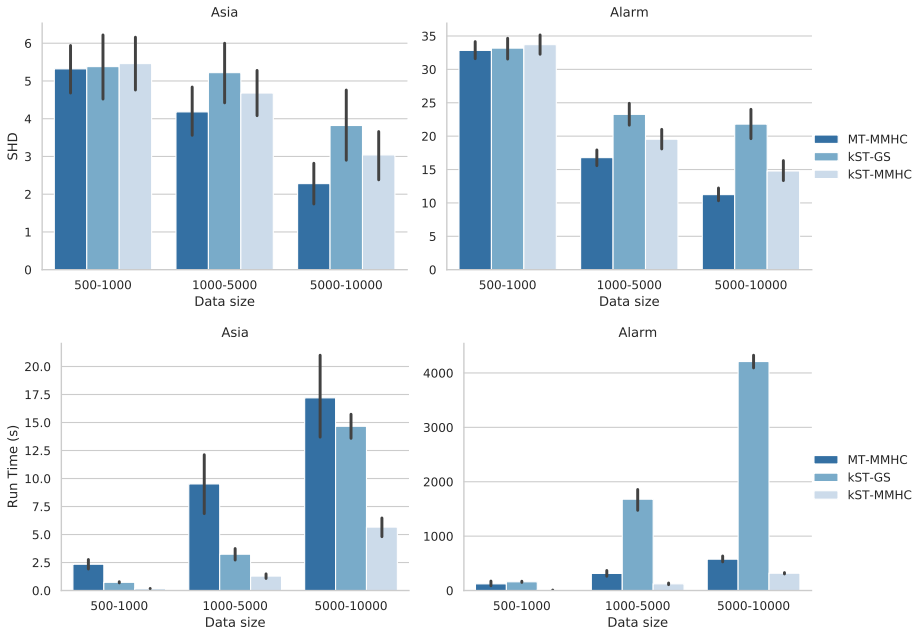

Fig. 3. SHD (top) and run time (bottom) with respect to data size for MT-MMHC, kST-GS and kST-MMHC (MT benchmarks are generated respectively from the ASIA and ALARM networks).

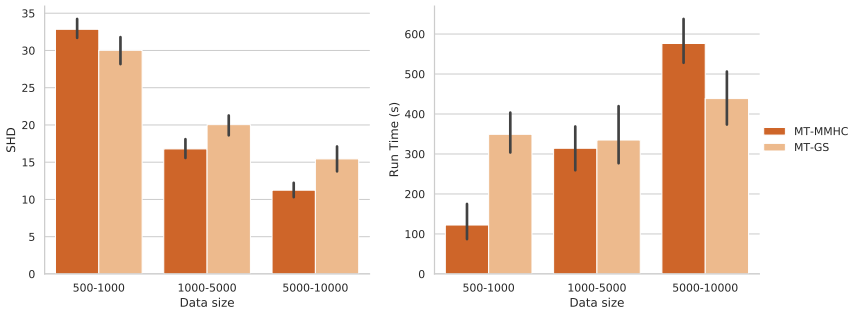

Fig. 4. SHD (left) and run time (right) woth respect to datasets size for MT-MMHC and MT-GS for MT benchmarks generated from the ALARM network

5 Conclusion

In this paper, we propose an extension of an efficient BN structure discovery algorithm MMHC to a multi-task context. Our algorithm MT-MMHC is the first hybrid transfer learning algorithm. MT-MMHC can learn multiple BN structures

simultaneously by inducing information between similar tasks with the help of a new combined association measure.

In order to validate our approach, we have also proposed one procedure to generate MT benchmarks from any reference model, by controlling the similarity between tasks.

The results of our experiments show that it is more beneficial to learn related tasks simultaneously than considering them individually and, both our combined association measure and the use of the CPC discovered by MT-MMPC help in finding accurate models with an affordable time cost. In these experiments, MT-MMHC was able to learn better models than kST-GS, kST-MMHC and MT-GS in medium to large MT benchmarks.

This work is the first step of our research, with several perspectives. In a very short term, we intend to perform larger experiments with other datasets to consolidate the interest of our proposition. We are also planning to work on additional procedures to generate MT benchmarks, for instance by using "longer" random walks, or by creating tasks that don't necessarily have all variables in common.

Finally, our objective is to combine such MT structure learning algorithms with differential privacy techniques (already used for BN ST learning in [19] for instance) in order to propose one general framework to Federated Learning of Bayesian networks, i.e. collaborative Structure and Parameter Learning with privacy considerations and no shared training data. Our interest is also to perform this BN federated learning for the development of BN-based medical assistants such as Medical Companion [9].

Acknowledgment

This work was supported by ANR AIby4 (Artificial Intelligence by Humans, for Humans) PhD program (ANR-20-THIA-0011) and Atlanstic2020 regional program.

References

1. Azzimonti, L., Corani, G., Scutari, M.: Structure Learning from Related Data Sets with a Hierarchical Bayesian Score. In: Jaeger, M., Nielsen, T.D. (eds.) Proceedings of the 10th International Conference on Probabilistic Graphical Models. Proceedings of Machine Learning Research, vol. 138, pp. 5–16. PMLR (2020)
2. Beinlich, I.A., Suermondt, H.J., Chavez, R.M., Cooper, G.F.: The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks. In: Hunter, J., Cookson, J., Wyatt, J. (eds.) AIME 89, Second European Conference on Artificial Intelligence in Medicine. Lecture Notes in Medical Informatics, vol. 38, pp. 247–256. Springer (1989)
3. Carvalho, A.M.: Scoring functions for learning Bayesian networks. Tech. Rep. 54/2009 Apr 2009, INESC-ID (2009)
4. Daly, R., Shen, Q., Aitken, S.: Learning Bayesian networks: approaches and issues. The Knowledge Engineering Review **26**(2), 99–157 (2011)

5. Jia, H., Wu, Z., Chen, J., Chen, B., Yao, S.: Causal discovery with Bayesian networks inductive transfer. In: Liu, W., Giunchiglia, F., Yang, B. (eds.) Knowledge Science, Engineering and Management - 11th International Conference, KSEM 2018, Changchun, China, August 17-19, 2018, Proceedings, Part I. Lecture Notes in Computer Science, vol. 11061, pp. 351–361. Springer (2018)
6. Lauritzen, S.L., Spiegelhalter, D.J.: Local computations with probabilities on graphical structures and their application to expert systems. Journal of the Royal Statistical Society. Series B (Methodological) **50**(2), 157–224 (1988)
7. López-Cruz, P.L., Larrañaga, P., DeFelipe, J., Bielza, C.: Bayesian network modeling of the consensus between experts: An application to neuron classification. International Journal of Approximate Reasoning **55**(1), 3–22 (2014)
8. Luis, R., Sucar, L.E., Morales, E.F.: Inductive transfer for learning Bayesian networks. Machine learning **79**(1), 227–255 (2010)
9. Mouchabac, S., Leray, P., Adrien, V., Gollier-Briant, F., Bonnot, O.: Beyond big data in behavioral psychiatry, the place of Bayesian network. example from a preclinical trial of an innovative smartphone application to prevent suicide relapse. Journal of Medical Internet Research **16/03/2021:24560**, (in press) (2021)
10. Niculescu-Mizil, A., Caruana, R.: Inductive transfer for Bayesian network structure learning. In: Meila, M., Shen, X. (eds.) Proceedings of the Eleventh International Conference on Artificial Intelligence and Statistics. Proceedings of Machine Learning Research, vol. 2, pp. 339–346. PMLR, San Juan, Puerto Rico (21–24 Mar 2007)
11. Oyen, D., Lane, T.: Leveraging domain knowledge in multitask Bayesian network structure learning. Proceedings of the AAAI Conference on AI **26**(1) (2012)
12. Oyen, D., Lane, T.: Bayesian discovery of multiple Bayesian networks via transfer learning. In: In IEEE International Conference on Data Mining (2013)
13. Salaken, S.M., Khosravi, A., Nguyen, T., Nahavandi, S.: Extreme learning machine based transfer learning algorithms. Neurocomput. **267**(C), 516–524 (2017)
14. Scutari, M., Vitolo, C., Tucker, A.: Learning Bayesian networks from big data with greedy search: computational complexity and efficient implementation. Stat. Comput. **29**(5), 1095–1108 (2019)
15. Silander, T., Leppä-Aho, J., Jääsaari, E., Roos, T.: Quotient normalized maximum likelihood criterion for learning Bayesian network structures. In: International Conference on Artificial Intelligence and Statistics. pp. 948–957. PMLR (2018)
16. Spirtes, P., Glymour, C., Scheines, R.: Causation, Prediction, and Search. MIT press, 2nd edn. (2000)
17. Thung, K.H., Wee, C.Y.: A brief review on multi-task learning. Multimedia Tools Appl. **77**(22), 29705–29725 (2018)
18. Tsamardinos, I., Brown, L.E., Aliferis, C.F.: The Max-Min Hill-Climbing Bayesian Network Structure Learning Algorithm. Machine Learning **65**(1), 31–78 (2006)
19. Zhang, J., Cormode, G., Procopiuc, C.M., Srivastava, D., Xiao, X.: PrivBayes: Private data release via Bayesian networks. ACM Trans. Database Syst. **42**(4) (2017)
20. Zhang, Y., Yang, Q.: A survey on multi-task learning. IEEE Transactions on Knowledge and Data Engineering pp. 1–20 (2021)
21. Zhou, Y., Hospedales, T.M., Fenton, N.: When and where to transfer for Bayesian network parameter learning. Expert Systems with Applications **55**, 361–373 (2016)