Bounding the Complexity of Structural Expectation-Maximization

Marco Benjumeda [*1] **Sergio Luengo-Sanchez** [*1] **Pedro Larrañaga** [1] **Concha Bielza** [1]

Abstract

Structural expectation-maximization is the most common approach to address the problem of learning Bayesian networks from incomplete datasets. Its main limitation is that its computational cost is usually extremely demanding when the number of variables or the number of instances is not small. The bottleneck of this algorithm is the inference complexity of the model candidates. Thus, bounding the inference complexity of each Bayesian network during the learning process is key to make structural expectation-maximization efficient. In this paper, we propose a tractable adaptation of structural expectation-maximization and perform experiments to analyze its performance.

1. Introduction

Bayesian networks (BNs) (Pearl, 1988; Koller & Friedman, 2009) provide a compact and self-explanatory representation of multidimensional probability distributions. A BN $\mathcal{B} = (\mathcal{G}, \boldsymbol{\theta})$ is composed of a structure $\mathcal{G}$, a directed acyclic graph that encodes conditional independences among triplets of variables in the network, and a set of parameters $\boldsymbol{\theta}$, i.e., the conditional probability distributions of each variable given its parents in the graph.

In the presence of missing values or hidden variables, BNs can be learned using Friedman's structural expectation-maximization algorithm (SEM) (Friedman, 1997), which extends the well-known expectation-maximization algorithm (Dempster et al., 1977; McLachlan & Krishnan, 2008) to simultaneously learn the structure and parameters of a BN. Because of its iterative nature, SEM is known to be a very computationally demanding algorithm. Moreover, as inference in BNs is NP-hard (Cooper, 1990), its computational cost may be prohibitive when the inference complexity of the network candidates is high.

Very recently, Scanagatta et al. (2018) have proposed the SEM-kMAX algorithm, a method for learning Bayesian networks with bounded treewidth from partially observed data. Unlike Friedman's SEM, they use hard assignments to complete the data in each iteration because keeping soft completions of the data in memory is infeasible for their proposal.

The main difference between using soft and hard assignments in SEM is that they involve optimizing over different objective functions. Soft assignments guarantee that the model is optimized with respect to the observed data, while hard assignments involve optimizing over both the model and the learned assignment to the missing values. In the problem of learning BNs from incomplete data, the objective function to be optimized is the former, given that the model that best explains the observed data is sought.

In this paper we propose a tractable adaptation of Friedman's SEM that uses soft assignments to guarantee that models are optimized with respect to the observed data. Additionally, hard assignments allow us to efficiently search for promising structure candidates at each iteration.

2. Tractable SEM

The most common approach to limit the inference complexity of the models is to bound its treewidth. Nevertheless, treewidth does not consider the cardinality of each variable, which can greatly influence the inference complexity of the networks. The below scoring function directly penalizes log-likelihood of the model for dataset $\mathcal{D}$ with the cost of inference:

$$\mathrm{sc}(\mathcal{D}, (\mathcal{G}, \boldsymbol{\theta})) = \ell(\boldsymbol{\theta}|\mathcal{D}) - k \cdot \mathrm{size}(\mathcal{G}), \quad (1)$$

where $k > 0$ represents the weight of the inference complexity penalization given by $\mathrm{size}(\mathcal{G})$, which is the number of arithmetic operations (sums and products) required to perform inference with variable elimination (Shachter, 1990) for the BN $\mathcal{B}$. Note that $\mathrm{size}(\mathcal{G})$ depends on the chosen elimination order. An optimal elimination order for $\mathcal{G}$ should minimize $\mathrm{size}(\mathcal{G})$, but finding it is an NP-hard problem (Arnborg et al., 1987). In the rest of the paper we assume the use of any heuristic with polynomial complexity (e.g.,

---

[*]Equal contribution [1]Computational Intelligence Group, Departamento de Inteligencia Artificial, Universidad Politécnica de Madrid, Madrid, Spain. Correspondence to: Marco Benjumeda <marco.benjumeda.barquita@upm.es>, Sergio Luengo-Sanchez <sluengo@fi.upm.es>.

Working Notes of a ICML/FAIM 2018 Workshop, Stockholm, Sweden, 2018.

Algorithm 1 Pseudocode of Tractable SEM (TSEM)

1: **Input:** Incomplete dataset $\mathcal{D}$
2: choose $\boldsymbol{\theta}_0$
3: **for** $j = 0, 1, \dots$ **until** convergence **do**
4: let $\mathcal{D}^{\mathrm{s}}_{j+1}$ and $\mathcal{D}^{\mathrm{h}}_{j+1}$ be the soft and the hard completion of $\mathcal{D}$ according to $\boldsymbol{\theta}_j$
5: let $\mathcal{G}'$ be the result of applying the local change that maximizes $\mathrm{sc}(\mathcal{D}^{\mathrm{h}}_{j+1}, (\mathcal{G}', \boldsymbol{\theta}'))$ where $\boldsymbol{\theta}'$ are the maximum-likelihood parameters of $\mathcal{G}'$ for $\mathcal{D}^{\mathrm{h}}_{j+1}$
6: let $\boldsymbol{\theta}_{j+1}$ and $\boldsymbol{\theta}'_{j+1}$ be the maximum-likelihood parameters of $\mathcal{G}$ and $\mathcal{G}'$ for $\mathcal{D}^{\mathrm{s}}_{j+1}$, respectively
7: **if** $\mathrm{sc}(\mathcal{D}^{\mathrm{s}}_{j+1}, (\mathcal{G}', \boldsymbol{\theta}'_{j+1})) > \mathrm{sc}(\mathcal{D}^{\mathrm{s}}_{j+1}, (\mathcal{G}, \boldsymbol{\theta}_{j+1}))$ **then**
8: $(\mathcal{G}_{j+1}, \boldsymbol{\theta}_{j+1}) \leftarrow (\mathcal{G}', \boldsymbol{\theta}'_{j+1})$
9: **else**
10: **return** $(\mathcal{G}, \boldsymbol{\theta}_{j+1})$
11: **end if**
12: **end for**

Markowitz (1957) or Kjærulff (1990)) for this purpose.

Algorithm 1 describes our proposal. In order to guide the structure search towards models with low inference complexity, Algorithm 1 scores each model according to Equation (1). The bottleneck of SEM is the computation of the expected sufficient statistics (ESS) for each network candidate. This can be very computationally demanding even when inference can be performed efficiently. To address this problem, our approach heuristically selects the most promising candidate structure at each iteration (line 5), using hard assignments to complete the data. Given a completed dataset the scoring function is decomposable, and the search of the optimal local change can be done efficiently. Subsequently, a soft completion of the data is used to compare the candidate structure with the previous one (lines 6–7). This ensures that the score at Equation (1) is improved with respect to the observed data at each iteration, guaranteeing its convergence.

2.1. Complexity of Algorithm 1

Completing dataset $\mathcal{D}^{\mathrm{h}}_{j+1}$ (line 4) requires performing M inference queries, where M is the number of instances of $\mathcal{D}$. This can be done efficiently when the complexity of inference is bounded. Completing dataset $\mathcal{D}^{\mathrm{s}}_{j+1}$ (line 4) requires exponential time and space in the number of missing values. Nevertheless, computing the ESS of $\mathcal{D}^{\mathrm{s}}_{j+1}$ for a structure is clearly less computationally demanding. Efficient inference methods as junction trees would require M inference queries to compute the ESS for a given structure candidate. Algorithm 1 computes the ESS of only two candidates at each iteration (line 6), which can be done efficiently. It is evident that lines 5–11 can be computed in tractable time given a completed dataset. Finally, the number of iterations of the loop at line e depends on the stopping criterion. If the stopping criterion is $\mathcal{G}_{j+1} = \mathcal{G}_j$ and the local changes considered at line 5 are only arc additions the maximum number of iterations that this algorithm could perform is bounded by n^2, where n is the number of variables in $\mathcal{D}$.

Table 1: Comparison of the mean $\pm$ standard deviation obtained with TSEM and SEM-kMAX in the 10 datasets. L_time is the learning time (in seconds), L_acc is the imputation accuracy and tw is the treewidth of the output model. The best results are denoted in boldface.

	Method	L_time	L_acc	tw
WI	TSEM	**199±13**	**0.956±0.001**	4.9±0.3
	SEM-kMAX	1036±295	0.943±0.003	3.2±0.4
PA	TSEM	**667±96**	**0.903±0.002**	2.3±0.5
	SEM-kMAX	1239±176	0.864±0.005	3.1±0.3
MU	TSEM	3821±399	**0.910±0.001**	2.9±0.3
	SEM-kMAX	**1976±284**	0.885±0.002	3.1±0.3

3. Experimental Results

In this section we compare our approach with SEM-kMAX to highlight the advantages and drawbacks of the proposed strategy. We generated 10 datasets of 2000 instances and 50% of missing values from the following real-world BNs: WIN95PTS (Horvitz et al., 1998), PATHFINDER (Heckerman et al., 1992), and MUNIN1 (Andreassen et al., 1989). We refer to these networks as WI, PA and MU, respectively.

Our approach requires to fix the weight of the complexity penalization k for the score (Equation (1)). We empirically set k to 0.05. Other small values of k produced similar results. We set the parameters of SEM-kMAX to the values suggested by Scanagatta et al. (2018). Concretely, they set an execution time of n seconds (i.e., a second for each variable) to compute the cache of best parent sets and $n/10$ seconds for the structure search.

Table 1 shows the experimental results that compare the above approaches. TSEM outperformed SEM-kMAX in terms of imputation accuracy in all the evaluated datasets. Apparently, this is caused by the differences between using soft and hard completions of the data. Analyzing the learning times, TSEM is faster in datasets generated from medium-sized networks and slower in those generated from the largest network. This can be explained by the bound in execution time set for SEM-kMAX which forces its learning time to scale linearly.

4. Conclusions

In this paper we proposed an efficient adaptation of SEM, providing guarantees on its convergence. TSEM showed promising experimental results, outperforming the state-of-the-art.

Acknowledgements

This work has been partially supported by the Spanish Ministry of Economy and Competitiveness through the Cajal Blue Brain (C080020-09; the Spanish partner of the Blue Brain initiative from EPFL) and TIN2016-79684-P projects, by the Regional Government of Madrid through the S2013/ICE-2845-CASI-CAM-CM project, and by Fundacin BBVA grants to Scientific Research Teams in Big Data 2016. This project has received funding from the European Union's Horizon 2020 Framework Programme for Research and Innovation under Specific Grant Agreement No. 785907 (HBP SGA2). M. Benjumeda is supported by a predoctoral contract for the formation of doctors from the Spanish Ministry of Economy, Industry and Competitiveness (BES-2014-068637).

References

Andreassen, S., Jensen, F., Andersen, S., Falck, B., Kjærulff, U., Woldbye, M., Sørensen, A., Rosenfalck, A., and Jensen, F. MUNIN–An expert EMG assistant. *Computer-Aided Electromyography and Expert Systems*, 66:S4, 1989.

Arnborg, S., Corneil, D., and Proskurowski, A. Complexity of finding embeddings in a k-tree. *SIAM Journal on Algebraic Discrete Methods*, 8(2):277–284, 1987.

Cooper, G. The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence*, 42(2):393–405, 1990.

Dempster, A., Laird, N., and Rubin, D. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B*, 1:1–38, 1977.

Friedman, N. Learning belief networks in the presence of missing values and hidden variables. In *Proceedings of the International Conference on Machine Learning*, volume 97, pp. 125–133, 1997.

Heckerman, D., Horvitz, E., and Nathwani, B. Toward normative expert systems: Part I. The Pathfinder project. *Methods of Information in Medicine*, 31(2):90–105, 1992.

Horvitz, E., Breese, J., Heckerman, D., Hovel, D., and Rommelse, K. The Lumière project: Bayesian user modeling for inferring the goals and needs of software users. In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, pp. 256–265. Morgan Kaufmann Publishers Inc., 1998.

Kjærulff, U. Triangulation of graphs-algorithms giving small total state space. Technical report, R-90-09, Department of Mathematics and Computer Science, Aalborg University, Denmark, 1990.

Koller, D. and Friedman, N. *Probabilistic Graphical Models: Principles and Techniques*. The MIT Press, 2009.

Markowitz, H. The elimination form of the inverse and its application to linear programming. *Management Science*, 3(3):255–269, 1957.

McLachlan, G. and Krishnan, T. *The EM Algorithm and Extensions*. John Wiley & Sons, 2008.

Pearl, J. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

Scanagatta, M., Corani, G., Zaffalon, M., Yoo, J., and Kang, U. Efficient learning of bounded-treewidth Bayesian networks from complete and incomplete data sets. *International Journal of Approximate Reasoning*, 95:152–166, 2018.

Shachter, R. Evidence absorption and propagation through evidence reversals. In *Proceedings of the 5th Annual Conference on Uncertainty in Artificial Intelligence*, pp. 173–190. North-Holland Publishing Co., 1990.