An Empirical Evaluation of Costs and Benefits of Simplifying Bayesian Networks by Removing Weak Arcs

Parot Ratnapinda[1] and **Marek J. Druzdzel**[1,2]
[1] Decision Systems Laboratory, School of Information Sciences and Intelligent Systems Program,
University of Pittsburgh, Pittsburgh, PA 15260, USA
[2] Faculty of Computer Science, Białystok University of Technology, Wiejska 45A, 15-351 Białystok, Poland

Abstract

We report the results of an empirical evaluation of structural simplification of Bayesian networks by removing weak arcs. We conduct a series of experiments on six networks built from real data sets selected from the UC Irvine Machine Learning Repository. We systematically remove arcs from the weakest to the strongest, relying on four measures of arc strength, and measure the classification accuracy of the resulting simplified models. Our results show that removing up to roughly 20 percent of the weakest arcs in a network has minimal effect on its classification accuracy. At the same time, structural simplification of networks leads to significant reduction of both the amount of memory taken by the clique tree and the amount of computation needed to perform inference.

Introduction

Practical models based on Bayesian networks (BNs) (Pearl 1988) reach often the size of hundreds or even thousands of variables. When these are densely connected, both the amount of memory to store a compiled clique tree and the amount of computation necessary to perform belief updating may become prohibitive (Cooper 1990). In some applications, where models are built dynamically and increase in size with time, (such as Query-Based Diagnosis (Agosta, Gardos, and Druzdzel 2008)), it is a necessity to control their growth. Otherwise, at a certain point, an uncontrolled model is bound to become intractable.

One way of controlling the growth of a model is to systematically simplify its structure by removing its weakest arcs. There have been two approaches to arc removal in Bayesian networks. The first approach focuses on minimizing the KL-divergence between the joint probability distributions represented by the original and the approximated networks (e.g., (Kjaerulff 1994; van Engelen 1997; Choi, Chan, and Darwiche 2005; Choi and Darwiche 2006a; 2006b; Renooij 2010)). The second approach introduces a measure of arc strength and then approximates the model by removing its weakest arcs. Boerlage (1992) defines link strength for arcs connecting binary nodes as the maximum influence that the parent node can exert on the child node. Nicholson and Jitnah (1998) use mutual information as a measure of link strength. They show how inference can be simplified by averaging the conditional probabilities of all parents. Koiter (2006) proposed another measure of strength of influence, resting on the analysis of differences among the posterior marginal probability distributions of a child node for different states of the parent node. He calculates the difference between distributions using four distance measures: Euclidean, Hellinger, J-divergence and CDF. This strength of influence has been applied in practice for the purpose of model visualization (e.g., (Hsu et al. 2012; Theijssen et al. 2013)), and has been a standard element of the GeNIe software for the last seven years.

The question that we pose in this paper is how much the accuracy of classification models will suffer as we simplify them by removing their weakest arcs. We pose two questions: (1) how many arcs can we remove with minimal impact on the model's accuracy?, and (2) what are the benefits of removing weakest arcs in terms of the reduction of memory requirements and computation time? We describe an experiment, in which we use several real data sets from the UC Irvine Machine Learning Repository (Frank and Asuncion 2010) to create Bayesian network models. We subsequently use these models as gold standards to test how removing their weakest arcs impacts their accuracy, memory demands, and inference time. We perform this for each of the four measures proposed by Koiter (2006).

Empirical Evaluation

Our goal was evaluating the practical impact of model simplification by removing weak arcs on performance measures such as classification accuracy, memory requirements, and computational demands. We selected six data sets from the UC Irvine Machine Learning Repository in order to create gold standard Bayesian network models for our experiments. We decided to use real rather than synthetic data sets because we wanted our experiments to be as close as possible to real world applications.

The Data

We selected six data sets: Chess (King-Rook vs. King-Pawn), Letter, Molecular Biology (Splice-junction Gene Sequences), Mushroom, Nomao, and Optical Recognition of Handwritten Digits (ORHD) using the following selection criteria:

- The data include a known class variable so that we could test the accuracy of the models on a real problem.
- The data set contains a reasonably large number of records (more than 1,000). The main reason for this is that we used the EM algorithm (Dempster, Laird, and Rubin 1977), which learns parameters more accurately from large data sets. In addition, because we check the accuracy of the models on the original data, the larger the data set, the more reliable our results.
- The selected data sets should not be too small in terms of the number of attributes (16–72), so that we obtain models with reasonably large number of arcs and a challenging total clique tree size.
- The majority of the attribute types should be discrete in order to reduce the need for discretization, which would be a possible confounding factor in our experiments.
- The data set should not contain too many missing values (not more than 1/3 of the data set). Missing values require special treatment in structure learning algorithms, which would be an additional confounding factor in our experiments.
- The probability over the class variable distribution is not too strong biased toward one class. Nomao has the probability of the most likely class equal to 73% while the other data sets varies from 11% to 52%.

Table 1 lists the key properties of the data sets selected for our experiments.

Table 1: Data sets used in our experiments. #I denotes the number of records, #A denotes the number of attributes, #C denotes the number of classes, #R denotes the number of arcs, #CB denotes total clique tree size of the original BN model, #CT denotes total clique tree size of the original TAN model, and M indicates presence of missing values.

Data set	**#I**	**#A**	**#C**	**#R**	**#CB**	**#CT**	**M**
Chess	3196	36	2	100	311KB	284B	N
Letter	20000	16	26	39	664KB	9KB	N
M.Biology	3190	60	3	101	10MB	4KB	N
Mushroom	8124	22	2	47	21KB	894B	Y
Nomao	28575	72	2	279	168MB	2KB	N
ORHD	5620	64	10	82	317KB	150KB	N

Experiments

To learn the gold standard Bayesian networks, we applied the standard Bayesian search-based learning algorithm (BS) proposed by Cooper and Herskovits (1992) and Tree Augmented Naive Bayes algorithm (TAN) proposed by Friedman, Geiger, and Goldszmidt (1997). Both algorithms do not handle missing values and continuous variables. We first discretized continuous attributes using equal frequency discretization with 5 intervals, removed all records with missing values, and used each algorithm to learn the model structure. Subsequently, we used the entire data sets (i.e., including the records with missing values) to learn the models' numerical parameters. We present some characteristics of the resulting models in Table 1. No BS models resemble Naive Bayes structure. The BS models have much larger total clique size than TAN models.

We used the models constructed in this way as our gold standard models, which we subsequently simplified by removing weak arcs using four distance measures: Euclidean, Hellinger, J-divergence and CDF. We calculated the strength of influence for each arc in the gold standard network. Subsequently, we removed all arcs that had the strength of influence below a threshold setting (0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, and 1.0). For example, when we set the threshold at 0.1, we removed all arcs that had the strength of influence less than 0.1.

We tested the classification accuracy of the simplified Bayesian network models by means of 10-fold cross validation on the original data sets from Table 1. We measured the total clique tree size and the CPU time consumed by performing inference on each of the simplified networks. We performed our tests on a Windows Vista computer with 4 GB of memory, and an Intel Core 2 Quad Q6600 processor running at 2.4 GHz. We implemented all our code in C++.

Results

Because the qualitative behavior of the accuracy of the networks is more important in our experiment than the precise numerical results, we present the results of our experiment graphically. Due to space constraints, we only present some results for the Bayesian Search networks. Results for the TAN networks are qualitatively similar. For the full set of results please refer to Ratnapinda (2014).

Classification accuracy In testing the classification accuracy of the simplified models on the original UCI Machine Learning Repository data sets, we used the simplest possible criterion, which is that the model guesses the most likely class to be the correct class for each record.

We show four plots of models' classification accuracy as a function of the four strength of influence measures in Figure 1. Figure 2 shows four plots of model's classification accuracy as a function of the percentage of the arcs removed for each of the networks.

Our results show that for all link strength measures, except J-divergence, the classification accuracy does not decrease much from the gold standard when the threshold is below 0.2 (this corresponds to removal of around 20 percent of all original arcs). Then the accuracy drops sharply and reaches a plateau after roughly 0.6 (when roughly 60–80% of the arcs have been removed; see Figure 1).

We explored further the reason for the sudden drop in the curves and found that this is related to removal of arcs between the class node and the nodes belonging to its Markov blanket. In our experiment, five of the six data sets contained no missing data. When there are no missing data, any node that is not in the Markov blanket of the class node will not affect the accuracy. In TAN models, all feature nodes belong to the Markov blanket of the class node. We show in Figure 3 that the accuracy reaches the plateau point when all the arcs in the Markov blanket are removed.

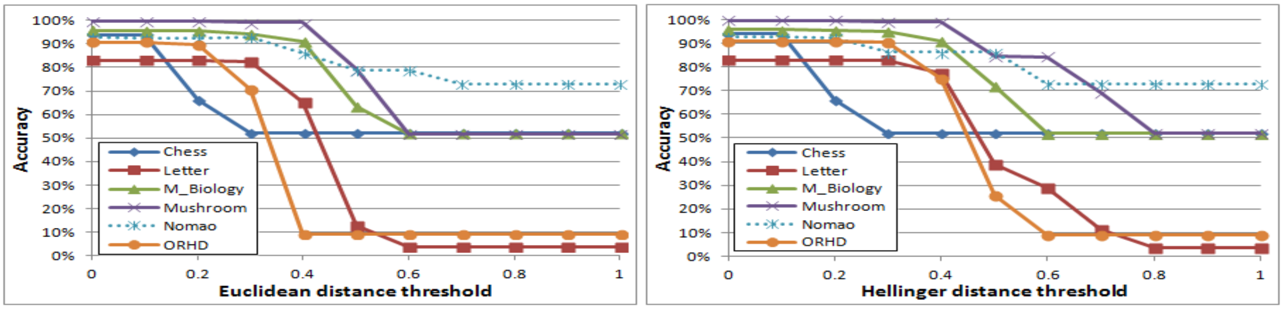

Figure 1: Classification accuracy as a function of the distance threshold for two of the four measures of the link strength

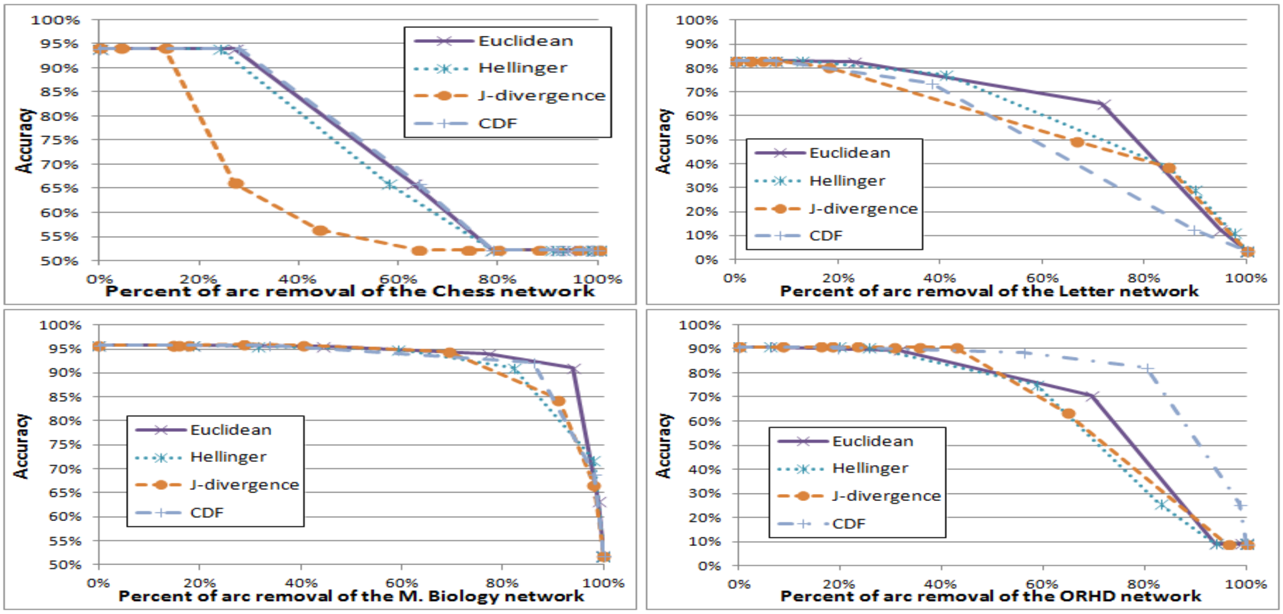

Figure 2: Classification accuracy as a function of the percentage of arcs removed for four of the six data sets

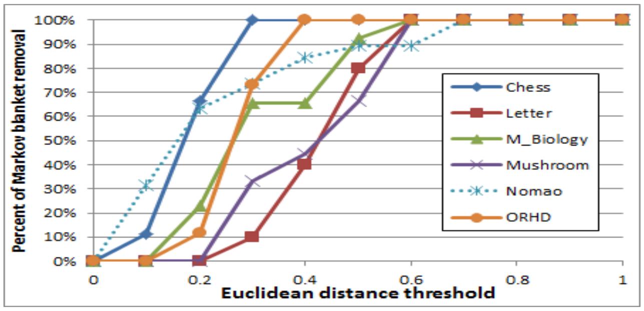

Figure 3: Percentage of arcs within the class node's Markov blanket removed as a function of the Euclidean distance threshold

Memory usage and computation time We measured the memory usage and the time taken to perform inference on simplified networks relative to the memory usage and inference time on the original (gold standard) networks. Removal of weak arcs can lead to significant savings in memory. Figure 4 shows the total clique tree size as a function of the percentage of arcs removed. We can see that even with as few as 20 percent of the weakest arcs removed the savings in memory approach an order of magnitude, which can mean a difference between an intractable and a tractable network.

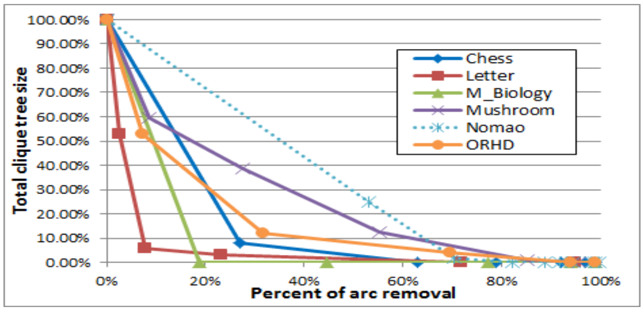

Figure 4: Total clique tree size as a function of the percentage of arc removed

Figure 5 shows the computation time as a function of the percentage of arcs removed. Here also, we can see that even with as few as 20 percent of the weakest arcs removed the computational savings can be significant. Our results here are somewhat confounded, as the computation time includes the time taken to create the clique tree, an integral part of the inference procedure as implemented SMILE.

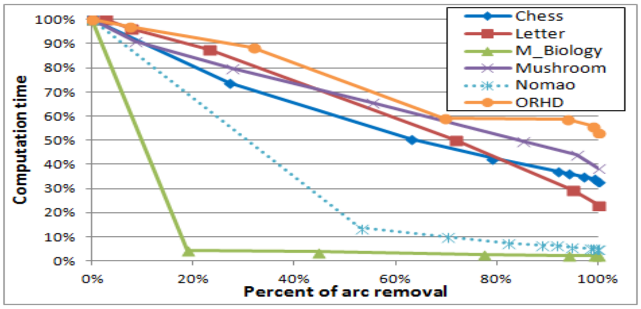

Figure 5: Computation time as a function of the percentage of arc removed

Discussion

Our results show that removing up to roughly 20 percent of the weakest arcs in a network has minimal effect of its classification accuracy. At the same time, both the amount of memory taken by the clique tree and the amount of computation needed to perform inference decreases significantly.

Acknowledgments

We acknowledge the support from XDATA program of Defense Advanced Research Projects Agency (DARPA), administered through Air Force Research Laboratory contract FA8750-12-C-0332, the National Institute of Health under grant number U01HL101066-01, and the Government of the Kingdom of Thailand. Implementation of this work is based on SMILE, a Bayesian inference engine developed at the Decision Systems Laboratory and available at `http://genie.sis.pitt.edu/`. We would like to thank Martijn de Jongh for his valuable input.

References

Agosta, J. M.; Gardos, T. R.; and Druzdzel, M. J. 2008. Query-based diagnostics. In Jaeger, M., and Nielsen, T. D., eds., *Proceedings of the Fourth European Workshop on Probabilistic Graphical Models (PGM–08)*, 1–8.

Boerlage, B. 1992. Link strength in Bayesian networks. Master's thesis, University of British Columbia.

Choi, A., and Darwiche, A. 2006a. An edge deletion semantics for belief propagation and its practical impact on approximation quality. In *Proceedings of the 21st National Conference on Artificial Intelligence - Volume 2*, AAAI'06, 1107–1114. AAAI Press.

Choi, A., and Darwiche, A. 2006b. A variational approach for approximating Bayesian networks by edge deletion. In *Proceedings of the Twenty-Second Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-06)*, 80–89. Arlington, Virginia: AUAI Press.

Choi, A.; Chan, H.; and Darwiche, A. 2005. On Bayesian network approximation by edge deletion. In *Proceedings of the Twenty-First Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-05)*, 128–135. Arlington, Virginia: AUAI Press.

Cooper, G. F., and Herskovits, E. 1992. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning* 9(4):309–347.

Cooper, G. F. 1990. The computational complexity of probabilistic inference using Bayesian belief networks. *Artificial Intelligence* 42(2–3):393–405.

Dempster, A.; Laird, N.; and Rubin, D. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B* 39(1):1–38.

Frank, A., and Asuncion, A. 2010. UCI Machine Learning Repository. http://archive.ics.uci.edu/ml.

Friedman, N.; Geiger, D.; and Goldszmidt, M. 1997. Bayesian network classifiers. *Machine Learning* 29(2-3):131–163.

Hsu, W.; Taira, R. K.; El-Saden, S.; Kangarloo, H.; and Bui, A. A. 2012. Context-based electronic health record: toward patient specific healthcare. *IEEE Transactions on Information Technology in Biomedicine* 16(2):228–234.

Kjaerulff, U. 1994. Reduction of computational complexity in Bayesian networks through removal of weak dependencies. In *Proceedings of the Tenth Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-94)*, 374–382. San Francisco, CA: Morgan Kaufmann.

Koiter, J. R. 2006. Visualizing inference in Bayesian networks. Master's thesis, Delft University of Technology.

Nicholson, A. E., and Jitnah, N. 1998. Using mutual information to determine relevance in Bayesian networks. In *Proceedings of the 5th Pacific Rim International Conference on Artificial Intelligence*. Springer.

Pearl, J. 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA: Morgan Kaufmann Publishers, Inc.

Ratnapinda, P. 2014. *Theoretical And Practical Aspects Of Decision Support Systems Based On The Principles Of Query-Based Diagnostics*. Ph.D. Dissertation, School of Information Sciences, University of Pittsburgh.

Renooij, S. 2010. Bayesian network sensitivity to arc-removal. In *Proceedings of the Fifth European Workshop on Probabilistic Graphical Models (PGM–2010)*, 233–240.

Theijssen, D.; ten Bosch, L.; Boves, L.; Cranen, B.; and van Halteren, H. 2013. Choosing alternatives: using Bayesian networks and memory-based learning to study the dative alternation. *Corpus Linguistics and Linguistic Theory* 9(2):227–262.

van Engelen, R. A. 1997. Approximating Bayesian Belief networks by arc removal. *IEEE Transactions on Pattern Analysis and Machinde Intelligence* 19(8):916–920.