Towards Effective Structure Learning for Large Bayesian Networks

Prashant Doshi
pdoshi@cs.uic.edu
Dept. of Computer Science
Univ of Illinois, Chicago, IL 60607

Lloyd Greenwald
lgreenwa@mcs.drexel.edu
Math and Computer Science Dept
Drexel Univ, Philadelphia, PA 19104

John Clarke
jclarke@gradient.cis.upenn.edu
Dept of Surgery, MCP-
Hahnemann, Philadelphia, PA 19104

Abstract

The effectiveness of Bayesian network construction is a function of the predictive ability and speed of inference of the resulting network, the effort and expertise required to construct the network, and the effort and expertise required to understand the network. We empirically evaluate three alternative methods for constructing Bayesian networks from data, considering both objective performance measures and subjective construction cost measures. Empirical results are obtained for a large real-world medical database. We provide results comparing network structure laboriously elicited from a domain expert to structure automatically induced by two alternative structure learning algorithms. The parameters of the Bayesian network produced by each method are induced using the Bayesian MAP approach. Additionally, we introduce the use of classification paths as an aggregation technique to reduce the size and structural complexity of Bayesian networks. Empirical results are obtained both with and without this complexity reduction technique.

Introduction

Structure-learning algorithms induce statistically predictive relationships from data. These algorithms have evolved in response to the tedious and error-prone process of manual knowledge elicitation from experts employed in early Bayesian network construction efforts. Growth of such algorithms necessitates comparisons of hand-built and machine-built hypotheses to aid the effective practice of building Bayesian networks from large real-world databases. An objective approach for evaluating belief elicitation techniques in probabilistic models is given in (Wang, Dash, & Druzdel 2001). In this paper we empirically evaluate alternative structure elicitation methods to provide additional insight into the effective use of structure learning.

We construct and test three different types of Bayesian network: (1) the **Naive-Bayes** network in which *n-1* of the *n* random variables are conditionally independent of each other and causally related to the remaining variable, (2) the **Learned-Bayes** network whose structure is learned using a structure-learning algorithm, and (3) the **Expert-Bayes** network whose structure is elicited from a domain expert. We evaluate the performance of these three networks in terms of their predictive ability and how well they model the target distribution. Through this empirical analysis, we examine the performance of each network-construction approach with respect to the costs of building the network. For each of the networks, we also model the correlation between the size of training set and the performance of the network.

While predictive ability is an important performance metric, the effectiveness of Bayesian network construction is additionally a function of the speed of inference of the resulting network, the effort and expertise required to construct the network, and the effort and expertise required to understand the network. Models elicited from large data sets tend to have a high number of nodes and causal relationships, resulting in increased structural complexity. Moreover, in such networks, only a small subset of variables may be targets for probabilistic inference. Such huge networks prove to be unwieldy for performing fast and efficient inference. Thus, it is highly desirable to provide techniques that reduce the size of a network without affecting its scope or performance. Node *aggregation* reduces the size and structural complexity of a network. In this paper, we introduce a node aggregation technique based on *classification paths*. In this method, a subset of nodes chosen for aggregation is replaced with a single node. This new node contains as its states (domain) the classification paths induced from the data contained in the former node. We empirically evaluate this tradeoff between structural complexity and state expansion.

The empirical studies in this paper are based on a large data set procured from the Pennsylvania Trauma Systems Foundation Registry. This data set contains information on 412 features of 148,501 patients admitted to the various trauma centers of the state of Pennsylvania. Each record contains clinical information, diagnoses, operative/non-operative procedures and timestamps, and final patient outcome. Noise in this large real-world data set was filtered using novel data cleansing techniques discussed elsewhere (Doshi 2001). The cleansed data set, discretized using an entropy-based discretization technique (Fayyad & Irani 1993) and filtered, contains 152 features on each patient. Of the approximately 22.5 million data points, 70% are missing motivating a prudent selection of techniques that can handle a large percentage of missing values. The different probabilistic models in this evaluation are outlined in the next section.

Techniques for Constructing Bayesian Networks

Recent successful commercial applications of Bayesian networks (Sahami *et al.* 1998) have illuminated the importance of effective network design methods. The use of graphical representation with well-defined local semantics is an important component of effective design. However, designing Bayesian networks in complex domains such as medicine requires extensive domain expertise. The process of knowledge acquisition has proved to be tedious. These limitations motivate the current study.

In the sections below we outline three different methods for capturing the structural relationships among variables in our medical data set.

Minimal Structure Configuration

Naive Bayesian networks lie at one extreme in the range of structural complexities. In a Naive Bayesian network containing *n* random variables, a single variable is selected as a *class* variable and the remaining *n-1* variables are presumed to be conditionally independent of each other given the class variable.

Such a minimal structure entails little domain knowledge and is thus very simple to construct. Yet networks constructed this way often perform well and have been empirically shown to be robust to violations of the conditional independence assumption exhibited in the training data.

Automated Structure Learning

Structure-learning algorithms (Tong & Koller 2001; Heckerman, Geiger, & Chickering 1994) induce a well-performing structure by searching a potentially exponential space of possible network structures. A structure is found that closely models the target distribution. A variety of structure-learning algorithms have been proposed. In this evaluation we employ a deterministic structure-learning method called *Bound and Collapse* (Ramoni & Sebastiani 1997). This method has the advantages of low computational cost and good performance on data sets with a large percentage of missing values such as ours.

Several training sets of 700 cases[1] each were randomly isolated to learn the structure. Multiple training sets were necessary to test uniformity in underlying predictive relationships. The learned structures were found to be identical and a representative was randomly selected for evaluation. Despite the relatively small size of the training set, an extensive structure was generated and most learned relationships were validated by the domain expert on our team. The resulting Bayesian network which we call **Learned-Bayes** is displayed in Fig 1.

Expert Structure Elicitation

A prevalent technique for constructing Bayesian networks is to elicit network structure evaluations from a domain expert or similar source of domain knowledge (such as existing literature). The flow of knowledge from the domain expert to the network designer can be a time-consuming process. An additional source of concern for this technique is that the resulting network is frequently assumed authoritative without objective evaluation. An expert-elicited structure is expensive to construct, compile and perform inference on. These added costs are generally believed to come with increased predictive ability and a closer fit to the target distribution.

Assisted by a Trauma Surgeon and Professor of Surgery [2] we constructed a third structural representation for the Bayesian network. The structural relationships were based on familiarity with the domain of the database and pertinent medical literature. All tasks related to eliciting the structure such as selection of variables consistent with the data set and establishing the causal relationships were carried out using domain knowledge.

Classification as an Aggregation Technique

Bayesian networks constructed from large data sets tend to have a high number of nodes and causal relationships, resulting in increased structural complexity. Structural complexity in a Bayesian network is a function of the number of random variables, average number of states in the variables and the causal relationships present in the network. Structural complexity impacts the speed of inference as well as the effort required to understand the resulting network. An effective way of simplifying any graph-based representation is node aggregation, replacing a subset of nodes with a single node. In this section we discuss reducing structural complexity through node aggregation, in a way that retains the scope and performance measures of the original Bayesian network.

We performed node aggregation on the same 24 nodes in all three networks discussed in the previous section. These 24 nodes correspond to the set of *patient prehospital information*. Using C4.5, a popular decision tree generator (Quinlan 1992) and *Patient Outcome* as the decision variable, we induced classifications from data contained in this isolated set of nodes. Each classification path in the resulting decision tree, from the root to a leaf node forms a state in the aggregated *Classifications_node* that replaces the corresponding set of nodes and their links in the Bayesian network. Additionally, a *leak classification* state is also added to the other mutually exclusive states to account for previously unseen cases present in test sets. The classifications tree and the resulting *Classifications_node* are shown in Fig. 2.

The underlying framework condensed in *Classifications_node* follows the *information gain* heuristic employed by decision tree generators. Thus, the conditional independence information encoded in the original structure is replaced by an entirely different framework. This substitution however, retains the scope of the model: findings accepted by this new node are identical to those entered in the replaced set of nodes, since the states of the new node are induced from the training data set itself. The new findings,

[1] Software limitations restricted the training set size to a maximum of 700 cases.

[2] Co-author John Clarke, M.D.

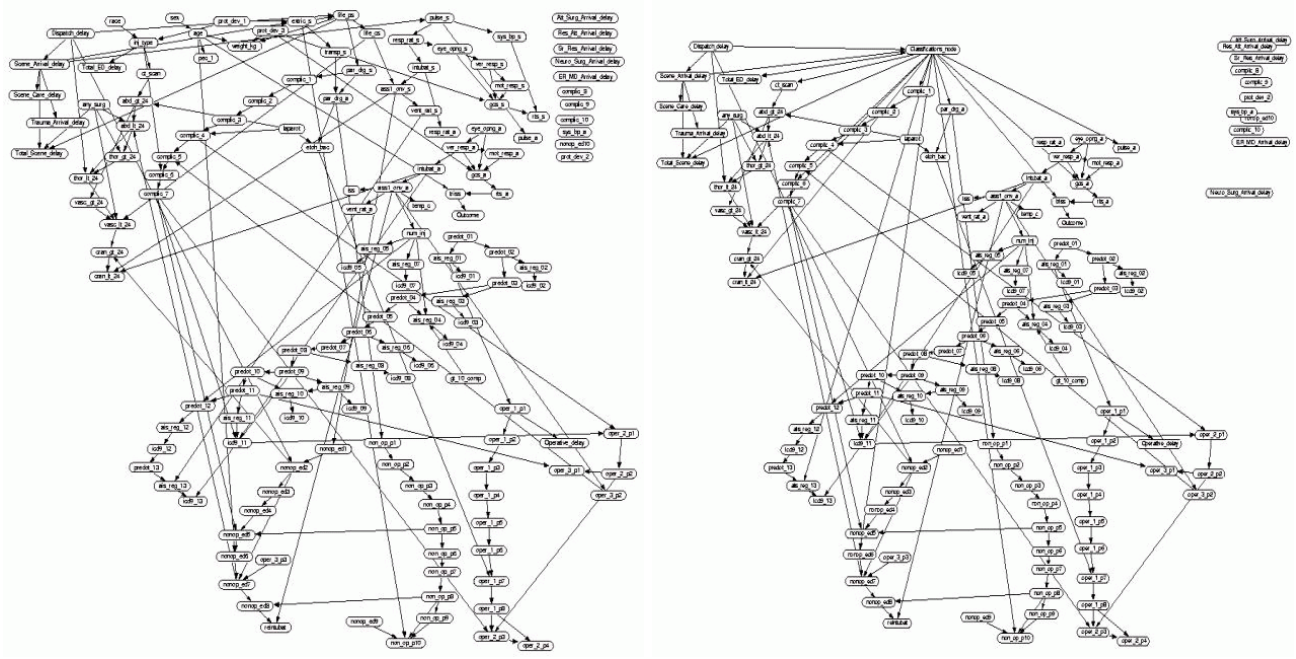

Figure 1: The structure of the Bayesian network on the left was learned using the Bound and Collapse technique. On the right, we can observe the reduction in structural complexity in the top portion of the same network due to the aggregation.

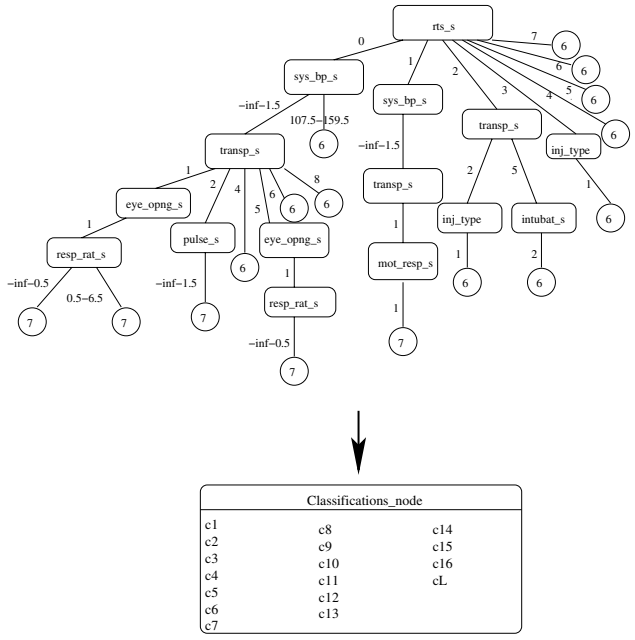

Figure 2: Graphical representation of the decision tree consisting of the important classification paths. The nodes of the decision tree represent the variables aggregated. Traversing the tree from left to right, each path from root to the leaf is labelled as c1,c2,....,c16. These labels alongwith the leak classification (cL) appear as states of Classifications_node.

composed of a conjunction of the former findings, are classified into the appropriate path ie. state of the aggregation node. The nodes to be aggregated should (1) not be targets for probabilistic inference since the original nodes can no longer be queried individually, (2) be linked to atleast one other node selected for aggregation, and (3) be of peripheral interest in understanding and evaluating the network. Furthermore, information exhibited by nodes selected for aggregation should preferably belong to the same subdomain. For the aggregation in our example, we selected all variables pertinent to *patient prehospital information*.

By employing an effective decision tree induction method, the increase in complexity due to the enlarged state size is mitigated by the reduction in complexity due to aggregation. We are currently investigating a theoretical formulation for this tradeoff. In the next section, we empirically evaluate this tradeoff between structural complexity and state expansion.

Performance Comparison

In previous sections, we outlined three basic network structure construction alternatives: (1) **Naive-Bayes**, which has minimal structure and is easy to construct; (2) **Learned-Bayes**, whose structure is automatically learned from data using a structure-learning algorihtm; and (3) **Expert-Bayes**, whose structure is elicited from domain knowledge. In the following empirical studies we additionally, in all three networks, replace a selected subset of nodes with a single node whose states represent the classification paths induced from data contained in the replaced variables. This step helps in further reducing the structural complexity.

We objectively evaluate the performance of these networks both in terms of predictive ability and how well they model the target distribution. In addition to this objective

empirical analysis, we also discuss both a subjective and an objective measure of the costs of constructing each of the three models.

Evaluating the Cost of Construction

As mentioned earlier, the effectiveness of a Bayesian network construction method includes the effort and expertise required to determine network structure. To model this cost we use the following two mutually independent metrics:

1. Amount and expertise of the domain knowledge required – Acquiring and representing domain knowledge is a tedious and expensive process requiring both domain expertise and Bayesian network construction skills. The three structure-construction methods differ widely in the degree of required domain knowledge.
2. Structural complexity – Structural complexity is a function of the number of random variables, the average number of states, in the variables, and the causal relationships present in the network. We are currently exploring a more precise definition of structural complexity based on well-known graph-theoretic and Bayesian network-specific properties.

   Networks exhibiting complex structures require greater compilation and inference time. Hence complicated structures generally lead to a greater cost of construction and evaluation.

A useful classification of the alternative network structure-construction methods is possible using these metrics. Constructing a Naive-Bayes network requires no domain knowledge beyond choosing the class variable. Also, the structure of Naive-Bayes is the simplest since it involves the minimum number of links.[3] Constructing a Learned-Bayes network may require partial domain knowledge depending on the structure-learning algorithm. *Scoring-based* algorithms (Cooper & Herskovits 1991; Herskovits & Cooper 1990) generally require a partial ordering of the variables whereas algorithms using *conditional independence* tests (Chow & Liu 1968) do not. Additionally, causal relationships are limited to those exhibited by the given data set. The structure of an Expert-Bayes network is completely elicited from domain knowledge. The extent of expertise required is application-specific, but is generally far more than that required by the previous two methods. Furthermore, during the process of construction, variables are linked using domain knowledge even though such relationships may not be exhibited by the available data. Hence typically, such networks suffer from far more structural complexity than the previous two networks.

Table 1 summarizes the construction costs involved for each of the three networks.

Empirical Results

In this section we analyze the performance of the various Bayesian networks. For each method, we trained the parameters of the networks with varying sizes of training samples and tested with varying sizes of holdout samples (test sets). Both the training and test sets were drawn from the same distribution. For learning the parameters of each of the three models, we used the Bayesian *MAP* approach using Dirichlet functions as prior probability distributions. We used Beta functions in cases of nodes with only two states. Assuming the prior distributions to be Dirichlet generally does not result in a significant loss of accuracy, since precise priors aren't usually available, and Dirichlet functions can fit a wide variety of simple functions. The parameter learning algorithm also makes the assumption that the conditional probabilities being learned are mutually exclusive. This assumption does not penalize predictive performance when there are a large number of cases (data). In addition only nodes for which the current case in consideration supplies a value (finding) and supplies values for all its parents have their probability tables updated. The complete algorithm appears in (Spiegelhalter *et al.* 1993). We used *predictive ability, average log loss* and the *Brier Score* to compare performance. Our training sets ranged in size from 41,769 cases to 104,420 cases, as follows:

Training Sets (TR 1-4): 41769, 62654, 83359, 104420 cases

Test Sets: 22041, 44081 cases

Using the network-construction techniques introduced previously, five Bayesian networks were constructed from the same set of random variables. The test procedure consisted of entering findings for 52 nodes that comprise *patient prehospital data*, constructing a junction tree for inference (Spiegelhalter, Dawid, & Cowell 1993) and using *Patient Outcome* as the query node.

Standard statistical metrics formed the basis for evaluating the performance of each of the networks. We give the metrics and resulting plots below.

Error rate For each record in a test set, from the calculated distribution on the query node, the state with the highest probability was compared with the true state provided by the test record. The error rate thus measures the predictive ability of the network by calculating the percentage of the cases in a test set for which the network predicted a wrong value on the query node. The line plots for the error rate calculated for each combination of training and test sets for each of the five networks are displayed in Fig 3.

The Naive-Bayes as well as the Naive-Bayes-Classfn exhibit a poorer predictive ability (higher error rate) compared to the Learned-Bayes and the Expert-Bayes networks. The error rates of all networks for both the test sets increase with larger sizes of the training sets. This behavior can be explained due to the predictive bias present in our data set that induces overfitting. Approximately 92% of the patients in our training sets are alive with only 7% dead. Hence initial training sets have few dead patients. As the size of the training sets increase, the network gets trained to predict more patients *alive* resulting in an increase in the number of alive mispredictions (false positives).

Rather than simply comparing the most likely predicted state to the true state, the next two metrics consider the actual belief levels of the states in determining how well they agree with the value in the test set. These measures are there-

[3] The number of links required for a Naive-Bayes network is one less than the number of variables.

Table 1: Summary of the construction costs for the networks.

Type of Network	Metric-based Evaluation	Cost
Naive-Bayes	No domain knowledge required Minimal structural complexity	Least expensive
Learned-Bayes	Minimal domain knowledge required Some structural complexity	Expensive
Expert-Bayes	Most domain knowledge required Most structural complexity	Most expensive

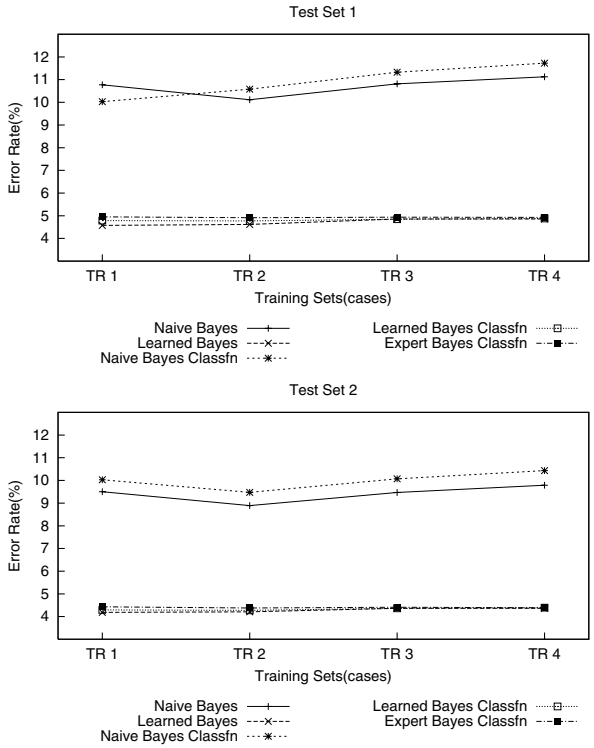

Figure 3: Line plots displaying error rate on the two test sets for the various configurations. Suffix *Classfn* indicates network configurations with the aggregation node.

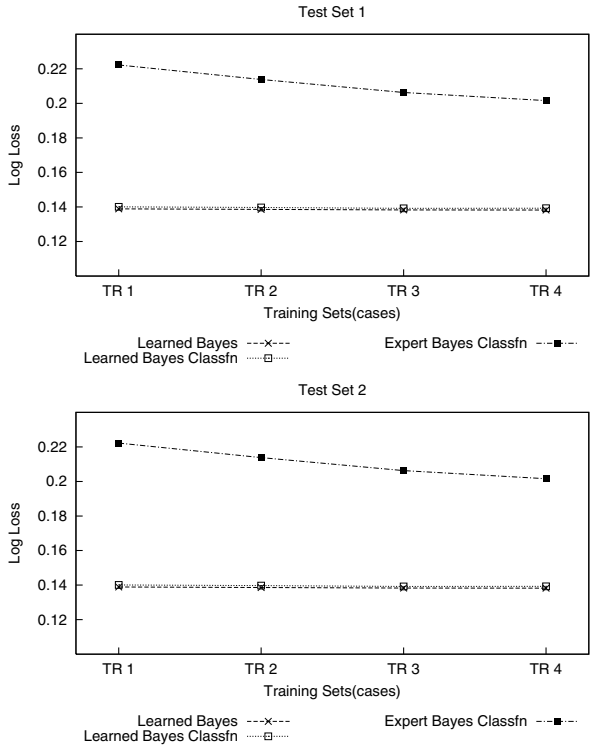

Figure 4: Line plots displaying log loss for the various configurations.

fore not very sensitive to the predictive bias present in our data set, and model more accurately the predictive ability of our networks. For detailed reference on these measures see (Morgan & Henrion 1990; Brier 1950). The error plots conclusively show the poorer predictive power of the Naive-Bayes configurations. Thus and for lucidity, we exclude the Naive-Bayes networks in further analysis.

Logarithmic Loss The following formula is used to calculate the log loss at the query node.

$Log\ Loss = \frac{\Sigma_N - log(P_c)}{N}$

where,

P_c: Probability predicted for the correct state c.

N: Number of cases in the test set.

For a correct prediction $P_c = 1$ and hence the Log Loss evaluates to zero. For non-zero Log Loss, lower values imply better performance. In Fig 4, we give log losses calculated for each combination of training and test sets.

Brier Score This measure also referred to as the Quadratic Score is calculated using the following expression.

$Brier\ Score = \frac{\Sigma_N(1-2*P_c)+\Sigma_{j=1..n}(P_j)^2}{N}$

where,

P_j: Probability predicted for state j.

n: Number of states of the query node.

Other notations are similar to those employed in the calculation of Log Loss. Similar to Log Loss, lower values of Brier Score also imply better performance by the network. The line plots for the performance of the configurations using Brier Score as the metric are given in Fig 5.

For each of the three configurations, both Log Loss and Brier Score values decrease with increasing sizes of the training set indicating an improving fit of the target distribution. The lower values for Learned-Bayes suggest a better performance compared to Expert-Bayes. We are currently testing the difference in performances for statistical significance. Furthermore, networks without the *classifications* node did not perform significantly better than networks with

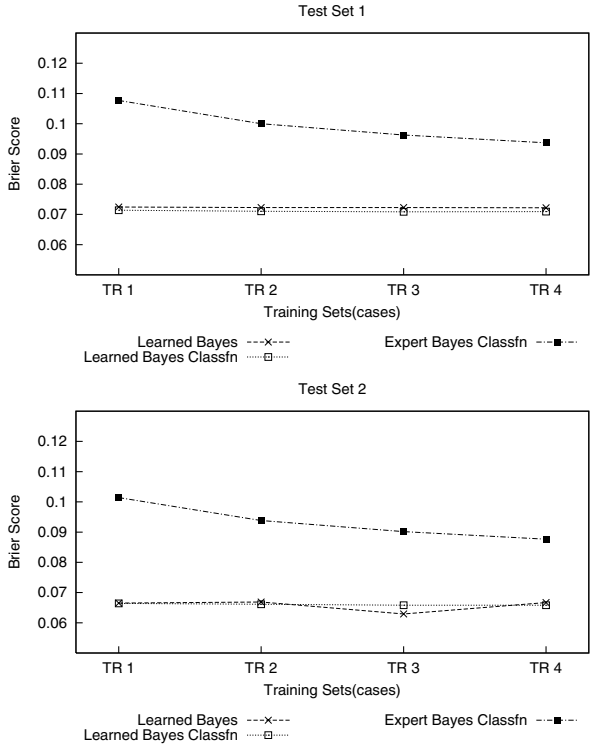

Figure 5: Line plots displaying Brier score values for the various configurations.

the node present, indicating that this form of aggregation does not alter the performance of the network. Duration of the testing phase for Bayesian networks with aggregation averaged 25% less than the testing phase for Bayesian networks with no such aggregation.

Conclusion and Future Work

A prevalent technique for constructing Bayesian networks is to elicit network structure evaluations from a domain expert. For example, in a domain such as medicine, a physician-user may be unwilling to apply the results of a system to patient care if the structure of its knowledge representation can not be investigated and understood. This expensive practice may introduce a misleading confidence in the optimality of the network. An expert-elicited network is often assumed authoritative without objective evaluation. Furthermore, hand-built models are believed to have a better predictive ability and a closer fit to the target distribution.

The advent of powerful structure-learning algorithms provides a viable alternative to the process of eliciting the structure from a domain expert. In this paper we provide a method that combines the use of automated structure-learning algorithms with a method to simplify the resulting network so that it is easier to evaluate and use.

We constructed and tested five network configurations that are the product of three different structure-construction schemes. The *inexpensive* Naive-Bayes has no structure at all, the absence of which significantly lowers its predictive ability. The Expert-Bayes falls slightly short in comparison to the Learned-Bayes. In addition, we introduce a node aggregation technique that reduces structural complexity while retaining the scope and performance of the original networks. The reduction in structural complexity translated into a substantial gain in inferencing time. Duration of the testing phase for Bayesian networks with aggregation averaged 25% less than the testing phase for Bayesian networks with no such aggregation. In addition, the performance of the aggregated networks remained unaffected.

Through a subjective analysis we also gain insight into the comparatively higher costs involved in constructing models using the traditional method of eliciting knowledge from experts.

One weakness of our evaluation is the reliance on a single medical expert to aid in the construction of the Expert-Bayes network. We plan further studies involving multiple medical experts. We also intend to consider multiple nodes for inference to avoid the misleading notion of our task being one of classification. Similarly, expanding the range of structure-learning algorithms in future experiments may provide further insight into the choice of a structure-construction scheme. Furthermore, we are investigating a theoretical formulation for use of classification as an aggregation technique. Finally, we understand that the domain of trauma is inherently complex and may not facilitate efficient construction of hand-built models. Thus our results may only tenuously extend to other domains.

Acknowledgements

This research is sponsored in part by a Drexel/MCP-Hahnemann Research Synergies award and, in part by a National Science Foundation (NSF) Instrumentation Award under grant CISE-9986105.

References

Brier, G. W. 1950. Verification of forecasts expressed in terms of probabilities. *Monthly Weather Review* 78:1–3.

Chow, C., and Liu, C. 1968. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory* 114:462–467.

Cooper, G. F., and Herskovits, E. H. 1991. A Bayesian method for the induction of probabilistic networks from data. In D'Ambrosio, B. D.; Smets, P.; and Bonissone, P. P., eds., *Uncertainty in Artificial Intelligence: Proceedings of the Seventh Conference*, 86–94.

Doshi, P. 2001. Effective methods for building probabilistic models from large noisy data sets. Master's thesis, Drexel University.

Fayyad, U. M., and Irani, K. B. 1993. Multi-interval discretization of continuous-valued attributes for classification learning. In *IJCAI-93*.

Heckerman, D.; Geiger, D.; and Chickering, D. M. 1994. Learning bayesian networks: The combination of knowledge and statistical data. In *KDD Workshop*, 85–96.

Herskovits, E., and Cooper, G. 1990. Kutato: an entropy-driven system for construction of probabilistic expert sys-

tems from database. In *Proceedings of the Sixth Conference on Uncertainty in Artificial Intelligence*, 54–62.

Morgan, M. G., and Henrion, M. 1990. *Uncertainty, A guide to dealing with uncertainty in qualitative risk and policy analysis*. Cambridge: Cambridge University Press.

Quinlan, J. R. 1992. *C4.5: Programs for Machine Learning.* Morgan Kaufmann.

Ramoni, M., and Sebastiani, P. 1997. Learning Bayesian networks from incomplete databases. In Geiger, D., and Shenoy, P. P., eds., *Proceedings of the 13th Conference on Uncertainty in Artificial Intelligence (UAI-97)*, 401–408. San Francisco: Morgan Kaufmann Publishers.

Sahami, M.; Dumais, S.; Heckerman, D.; and Horvitz, E. 1998. A bayesian approach to filtering junk e-mail. In *AAAI-98 Workshop on Learning for Text Categorization, 1998.*

Spiegelhalter, D.; Dawid, A.; Lauritzen, S.; and Cowell, R. 1993. Bayesian analysis in expert systems. *Journal of Statistical Science* 8(3):219–283.

Spiegelhalter, D.; Dawid, A.; and Cowell, S. 1993. Queries and updates in probabilistic networks. *Journal of Statistical Science.* 8(3):219–283.

Tong, S., and Koller, D. 2001. Active learning for structure in Bayesian networks. In *Proceedings of the International Joint Conference on Artificial Intelligence*.

Wang, H.; Dash, D.; and Druzdel, M. J. 2001. A method for evaluating elicitation schemes for probabilities. In *Conference of Florida Artificial Intelligence Research Society*.