A BAYESIAN NETWORKS STRUCTURAL LEARNING ALGORITHM BASED ON A MULTIEXPERT APPROACH

Francesco Colace, Massimo De Santo, Mario Vento
DIIIE, Università degli Studi di Salerno, Via Ponte Don Melillo 1, 84084, Fisciano (Salerno), Italy

Pasquale Foggia
DIS, Università di Napoli "Federico II", Via Claudio, 21, 80125 Napoli, Italy

Keywords: Bayesian Networks, MultiExpert System

Abstract: The determination of Bayesian network structure, especially in the case of large domains, can be complex, time consuming and imprecise. Therefore, in the last years, the interest of the scientific community in learning Bayesian network structure from data is increasing. This interest is motivated by the fact that many techniques or disciplines, as data mining, text categorization, ontology building, can take advantage from structural learning. In literature we can find many structural learning algorithms but none of them provides good results in every case or dataset. In this paper we introduce a method for structural learning of Bayesian networks based on a multiexpert approach. Our method combines the outputs of five structural learning algorithms according to a majority vote combining rule. The combined approach shows a performance that is better than any single algorithm. We present an experimental validation of our algorithm on a set of "de facto" standard networks, measuring performance both in terms of the network topological reconstruction and of the correct orientation of the obtained arcs.

1 INTRODUCTION

Bayesian belief networks (or shortly Bayesian networks) are powerful knowledge representation and reasoning tool for managing conditions of uncertainty. A Bayesian belief network is a directed acyclic graph (DAG) with a conditional probability distribution for each node. The DAG structure of such networks contains nodes representing domain variables, and arcs between nodes representing probabilistic dependencies. In the last period this model is becoming a popular representation for encoding uncertain knowledge. The main advantages of Bayesian Networks are discussed in detail in various papers (Heckerman,1997)(Cheng,1997) and can be summarized in the following points:

- Bayesian Networks can handle incomplete data sets
- Bayesian Networks allow learning about causal relationships
- Bayesian Networks facilitate the combination of background knowledge and experimental data avoiding the over fitting problem typical of methods based exclusively on experimental data

An interesting problem is the learning of Bayesian Networks structure from a finite set of data samples. This task is not easy to solve and in literature we can find many different approaches for "structural learning". The main aim of structural learning algorithms is to infer the relationships among the entities of the domain and to specify the causality dependencies from the observations of domain variables values. Generally, these algorithms can be grouped into two categories (Singh,1995)(Bell,1997): the first category uses heuristic search methods to construct a model and evaluates it using a scoring measure. This process continues until the score of the model obtained at the current iteration is not significantly better than the previous one. Different scoring criteria have been proposed in these algorithms, such as, Bayesian scoring, entropy based scoring, and minimum description length (Glymour,1987)(Cooper,1992)(Lauritzen,1989). The second category builds the dependency relationships by analysing pairs of nodes. The dependency relationships are measured by using some kind of

Colace F., De Santo M., Vento M. and Foggia P. (2005).
A BAYESIAN NETWORKS STRUCTURAL LEARNING ALGORITHM BASED ON A MULTIEXPERT APPROACH.
In *Proceedings of the Seventh International Conference on Enterprise Information Systems*, pages 194-200
DOI: 10.5220/0002521401940200

conditional independence (CI) test. The algorithms described in (Fung,1990)(Cooper,1992) belong to this category. Both of these two categories have their advantages and disadvantages: generally, algorithms in the first category have less time complexity in the worst case (when the underlying DAG is densely connected), but it may not find the best solution due to its heuristic nature. The second category of algorithms is usually asymptotically correct when the probability distribution of data is DAG-Isomorphic, but CI tests with large condition-sets may be unreliable unless the volume of data is enormous (Cooper,1992). In this paper we propose a structural learning algorithm based on a multiexpert approach. The proposed Multi-Expert System combines five algorithms (Bayesian algorithm (Heckermann,1995), K2 (Cooper,1992), K3 (Bouckaert,1993), PC (Spirtes,2001) and TPDA (Cheng,1997)) selected among those presented in the literature that show the better results. To evaluate this algorithm, we present the experimental results on eight networks datasets selected among those regarded standard in the literature. The reported experimental results show not only that proposed system performances are better than the ones of original experts but also the ability of the Multi-Expert System of exploting the strengths of each expert overcoming at same time its weakness. The paper is organized as follows: in section 2 we describe the general structure and the various approaches of Structural Learning Algorithms, the selected algorithms and our MultiExpert system. In section 3 we describe the reference datasets and the obtained results.

2 ALGORITHMS OF STRUCTURAL LEARNING

As previously said the main aim of a structural learning algorithm is to point out relationships between the entities of a domain and to specify the causality bonds starting from the observations of domain variables values. In general a structural learning algorithm includes the following steps:

- Collection of experimental data
- Determination of the network nodes from the acquired data
- Construction of an initial graph
- Choice of the search method
- Initialitation of the Structural Learning process
- Costruction of the network

The earliest result in structure learning was the Chow and Liu algorithm (Chow, 1968). This algorithm learns a Bayesian Network whose shape is a tree. Problems like structural learning become very difficult when datasets are smaller because of overfitting in the structure space. The main limitation of the method by Chow and Liu was that it did not take any countermeasure to reduce overfitting. Most subsequents works on structural learning apply standard statistical methodologies for fitting models and avoiding overfitting. It is important to note that the role of a statistical methodology is to convert a learning problem into an optimization problem in order to apply techniques aimed at avoiding local minima. First family of methods is based on the maximum likelihood or the minimum cross entropy. The maximum likelihood approach tries to find the network structure S_m for which the maximum likelihood over parameter θ_m (characterizing associated to the given structure) is the largest:

$$S = \arg\max_{S_m} \max_{\theta_m} p(sample|\theta_m, S_m)$$

The minimum cross entropy approach tries to find the structure whose minimum cross entropy with the data is the smallest. It has been demonstrated that these two approaches are equivalent. For Bayesian networks the maximum likelihood approach has been applied by (Geiger, 1992). A number of extensions to the maximum likelihood approach have been proposed. They replace the sample likelihood by a modified score that is to be maximized. Examples of modified score can be the penalized likelihood, Akaike information criteria (AIC), the Bayesian information criteria (BIC). Some algorithms minimize some information complexity measure, for instance minimum description lenght, minimum message length and minimum complexity (Rissanen,1978). One advantage of this approach is that it requires no “a priori” knowledge and is hence objective. Da For Bayesian networks, MDL has been applied by (Suzuki,1999)(Lam,1994). Another class of algorithms is based on the hypotesis testing approach that is the standard model selection strategy from classical statistics. As mentioned before, the problem is that this approach is a viable only if there is a small number of hypotheses that need to be tested. Sub-Optimal search techniques (e.g.) greedy search tecniques can help here by reducing the number of hypothesis tests required. Finally one of the most important families of algorithms is based on the Bayesian approach. Actually we can say that there is a rich variety of Bayesian methods and most of the previous methodologies can be reduced to some form of

Bayesian approximation. In its complete form the Bayesian approach requires specification of a prior probabilities. The Bayesian approach has many different approximations: the simplest is the MAP approach. In general the full bayesian approach is predictive: rather than returning the single best network with respect to observer data, the aim is to maximize the expected performance also for new cases. The key distinction between Bayesian and non Bayesian methods is the use of priors. Unfortunately priors computation can be complex mathematically, so poorly chosen priors can make a Bayesian method perform worse than other methods. Some approaches use a two phases algorithm: in the first phase a statistical method is used to obtain a reasonable estimate of prior probabilities which is then exploited in the second phase bayesian method. None of described approach obtains good results in every case because, as previously described, they have diffent strategies useful in well defined cases (for example sparse networks, huge datasets). In order to obtain a structural learning system able to perform its task under the most diverse condition we propose a new algorithm based on a MultiExpert approach. We have selected five different algorithms that represent all the categories previously described and we have combined their results according to a combining rule to obtain the final output of our system. In the next subsections we will show the selected algorithms and the architecture of our MultiExpert System.

2.1 The Bayesian Algorithm

The bayesian algorithm resolves the problem of the Structural Learning from data determining the structure **m** that maximizes the probability p(**M=m**|D), where **M**ε{**m**$_1$, ...,**m**$_n$} that is a set of models that contains the *true model* of a domain **X and** D is the set of the observed samples. According to this approach if we have two models **m**$_i$ and **m**$_j$ representing the domain **X**, we will choose **m**$_i$ if p(**m**$_i$|D) > p(**m**$_j$|D). We can choose as our scoring function the logarithm of p(**D**|m). In fact with simple passages we can show that:

$$\log(p(\mathbf{m}|D)) = \log(p(\mathbf{m}))+\log(p(D|\mathbf{m}))-\log(p(D))$$
$$=\log(p(D|\mathbf{m}))+\text{Constant}$$

hence the model maximizing log(p(D|m)) will also maximize p(m|D), under the condition that log(p(D)) and log(p(**m**)) are constant values (complete “a priori” ignorance of the domain structure). This formulation is based on the statistical criterion of Maximum Likelihood; in cases where the models have not the same prior probability (p(m)) the algorithm can use instead the Maximum a Posteriori principle (MAP). As regard the searching methodology we can choose between two different approaches:

model selection: the search is aimed at obtaining a single model with in a family of considered models chosen according to a scoring function. In case of ties the algorithm performs a not deterministic choice.

selective model averaging: the search is aimed at obtaining a set of “good models”, i.e. models with a good scoring value; then a single model obtained by means of some averaging criterion over this set.

Many papers have experimentally shown that the selection of a single model, using a greedy search algorithm, supplies accurate models (Chickering,1996)(Heckerman,1997). The selective model averaging, instead, must be applied in conjunction with sampling methods such as Montecarlo method in order to obtain good results (Heckerman,1997). In this paper we will refer to a representative algorithm of this approach based on a “model selection". In order to select the best model the algorithm performs a “hill climbing search” with respect to a fixed scoring function: given an initial structure *S* (either a graph without arcs that represents complete ignorance on the relationship between the network variables or an acyclic graph constructed inserting arcs in random way or a net that represents the “a priori” expert knowledge) the algorithm iteratively modifies the edges choosing at each step the modification wich involves the maximum gain in the scoring function. The procedure ends when it find a local maximum of scoring function or when it reaches the maximum number of iterations.

2.2 The K2 Algorithm

This algorithm is representative of the approach based on a bayesian framework with different definition of the scoring function (Cooper,1992). The K2 procedure differs from a typical bayesian algorithm also for the initialization phase: while in the pure bayesian approach the initial graph incorporates the “a priori” knowledge of an expert, in the K2 approach the user must provide the initial topological ordering (from parents to children) of the nodes. In fact this information gradely reduces the cardinality of the searching space of the models. However also with the sorting procedure, the number of possible models remains high because the distribution of combined probability P(X$_1$,X$_2$,...,X$_n$) can be rewritten in many different ways even after fixing one of the *n*! possible configurations. In this approach the scoring function is defined as:

$$p(B_s, D) = P(B_s)\prod_{i=1}^{n} g(X_i, \pi_i)$$

where D is a data set of the **k** complete cases and M_s is the structure of a bayesian network. The function $g(X_i,\pi_i)$ represents the variation obtained in the scoring function after the introduction of a new dependence relation and possibly of a new parent node for X_i. The core of this approach is a greedy search algorithm starting from an initial structure where nodes have no parents. The search for parents nodes ends when all candidate nodes have been examined or when the maximum number of parents for a node has been achieved. The main disadvantage of this approach is the impossibility of deleting an arc after its introduction in the network.

2.3 The K3 Algorithm

This type of algorithm, introduced in the paper (Bouckaert,1993), is based on a bayesian approach, but as in K2 algorithm gives a new definition for the scoring function. In this case the scoring function is based on the Minimum Description Length (MDL) metric. According to MDL approach (Rissanen,1978) the optimal model minimizes the total length of description. In other words it aims at establishing the best statistical compromise between the “a priori” complexity of the model and the quality of the “a posteriori” estimates. In the MDL approach the learned network must minimize the *total description length* defined as the sum of the description length of the samples (the source) and the description length of a pre-existent network structure supplied by an expert or generated in a previous learning process. In this approach, samples and the pre-existent network structure are considered independent in order to process them separately. The scoring function is defined as follows:

$$L(B,D) = \log(P(B)) + N * H(B,D) - \frac{1}{2} k \log(N)$$

$$H(B,D) = \sum_{i=1}^{n}\sum_{j=1}^{q_i}\sum_{k=1}^{r_i} - \frac{N_{ijk}}{N}\log(\frac{N_{ijk}}{N_{jk}})$$

$$k = \sum_{i=1}^{n} q_i (r_i - 1)$$

where B represents a possible structure, D is the n samples of data set, the value r_i represents the number of states associated to node $X_{i,}$ q_i is the number of possible configurations of parents nodes for each node X_i and N_{ijk} are the occurrences in D of X_i with state k and fathers configuration j and H(B,D) is the entropy. Also this approach needs an initial ordering of the nodes.

2.4 The PC Algorithm

This algorithm is based on a constraint satisfaction approach (Spirtes,2001). In fact it derives the Bayesian network structure through suitable statistical independence tests on the samples. The PC algorithm needs, together with the observation of the random discrete variables associated to nodes, also a matrix whose element ij represents the confidence about the indipendece of the nodes i and j according to a fixed independence test. The PC procedure consists of an initialization phase where a fully connected DAG, associated to a domain X and with iteration t equal to zero, is set up (so assuming that all variables are mutually dependent) then iteratively tha algorithm removes edges of this DAG according to the D-Separation property derived from statistical indipendence tests of the same orders as the iterations number. The algorithm stops when it can not find further nodes to each the D-Separation can be applied. After this process we obtain a not oriented graph: in order to determinate the arcs orientation the algorithm use consideration based on conditional independence. The reliability of the test results is related to the number of samples number: increasing the number of nodes we usually have an increase of the dependencies and so the samples number must be greater in order to obtain reliable results. Concerning the significance level, its high value means many dependencies to extract from the database of samples. This is obvious because increasing the threshold the probability that the independence test can supply a incorrect result increases. A high value of the confidence level (>0.6) is used with small databases, on the contrary a low value is appropriate in presence of a considerable number of observations.

2.5 The TPDA Algorithm

Also the TPDA (Three-Phase Dependence Analysis) algorithm (Cheng,1997) is a dependence-based algorithm and learns the Bayesian Network structure starting from the independence relationships among data. The input of the algorithm, like in the PC algorithm, is the dataset and a threshold ε used in the independence tests. The TPDA divides the process of learning in three phases: Drafting, Thickening and Thinning. The "Drafting" phase produces an initial relations set through test on cross entropy value between the variables of the domain. After this phase we obtain is a single connected dag (i.e. there is only one path connecting any to nodes). The second phase, "thickening", adds no arcs if it is not possible to d-separate two nodes. The resulting graph contains all arcs of the true model and some extra-arcs. The third phase, "thinning", consists in the examination of all arcs and their exclusion if the two linked nodes are conditionally independent. At the end of this phase the algorithm estabilishes the arc’s orientation with an approach similar to PC algorithm.

2.6 The proposed approach

The main idea of this paper is to use a multi expert approach in the Bayesian networks structural learning problem. The idea of combining various experts with the aim of compensanting the wekness of each single expert while preserving its own strenght has been considered appealing by many researchers in the last few years (Ho,1994)(Kittler,1998). The rational of this approach is that the performance obtained combining the results of a set of expert can result better than that of any single experts. The successful implementation of a multiexpert system depends both on the definition of suitable combining rule and on the choice of experts that are as much as possible complementary. One of the simplest combining rules, the majority vote, assigns the input samples to the class for which a relative or absolute majority of experts agrees. In our approach we have adopted a relative majority voting rule. In particular we used this rule to decide both if an arc should be placed between two nodes and which orientation should be assigned to the arc. This rule has proved to be quite effective and ha the advantage of not requiring the training of a parameters set.

3 EXPERIMENTAL RESULTS

We have selected eight networks in order to test the algorithms previously described. These networks are mentioned in several papers and represent the reference networks in literature (Table 1).

Table 1: Analysed Networks and Datasets

Network Name	Nodes Number	Arcs Number	Data Set Samples
Alarm (Pearl,1991)	37	46	10.000
Angina (Cooper,1992)(Lauritzen,1989)	5	5	10.000
Asia (Glymour,1987)	8	8	5.000
College (Singh,1995)	5	6	10.000
Hailfinder (Cheng,1997)	56	66	20.000
Led (Fung,1990)	8	8	5.000
Pregnancy (Lauritzen,1989)	4	3	10.000
Sprinkler (Suzuki,1999)	5	5	400

3.1 Obtained Results

For evaluating the performance of our method we have designed and implemented a Java based software tool based on the previous scheme. We have implemented all algorithms previously described according the authors instructions and a majority voting combiner. In order to evaluate the performances of algorithm we have used two indexes (Colace,2004):

$$\text{Topological Learning} = \frac{\sum \text{Correct Arcs}}{\sum \text{Correct Arcs} + \sum \text{Missing Arcs} + \sum \text{Added Arcs}}$$

$$\text{Global Learning} = \frac{\sum \text{Well Oriented Arcs}}{\sum \text{Well Oriented Arcs} + \sum \text{Wrong Oriented Arcs} + \sum \text{Added Arcs} + \sum \text{Missing Arcs}}$$

The first index measures the ability of the algorithm in the learning of correct topology of the net. The second index measures the ability of the algorithm in the learning of correct networks. In figure 1 and 2 we show the results obtained by the proposed MultiExpert System vs the best single expert.

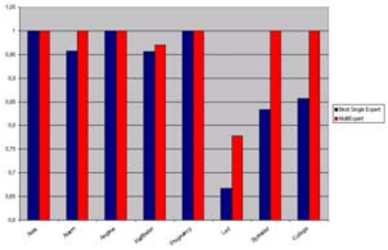

Figure 1: Obtained results fot the Topological Index (in red the multiexpert results)

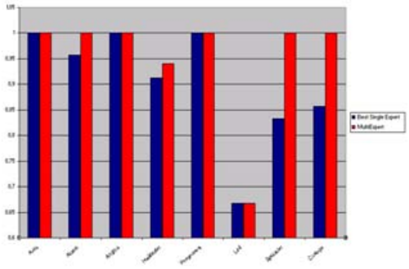

Figure 2: Obtained results fot the Global Index (in red the multiexpert results)

The obtained results show that the multiexpert approach has higher performances than the best expert both from the topological point of view and the global point of view. In particular the MultiExpert approach is able to obtain the correct network in the 75% of considered networks versus the 37,5% obtained by the single best expert. Furthermore there is no network for which the multiexpert approach has performance lower than that of any single expert. In general we have a performance increase of multiexpert system versus the best single expert. In particular in the case of sprinkler network (a dataset with a very low number of samples) the performance increase is very impressive: 16.7%.

4 CONCLUSION

In this paper we introduced a MultiExpert system for structural learning of Bayesian Networks. We showed the most important approaches in literature. None of these approaches allows a correct building in every case. So we selected five algorithms in order to build a MultiExpert system based on majority vote approach. Aiming to evaluate the results of our approach we selected eight networks and their samples datasets. The obtained results show that the multiexpert approach provide better results than any single experts. In order to improve the performance of MultiExpert system we are working to the introduction of new experts and new, more sophisticated, combining rules.

REFERENCES

Singh, M., Valtorta, M., Construction of Bayesian Network Structures from Data: a Brief Survey and an Efficient Algorithm. International Journal of Approximate Reasoning, 1995, 12:111-131

Glymour, C., Scheines, R., Spirtes P. and Kelly, K., Discovering Casual Structure, Academic Press, 1987

Fung R. M., Crawford S. L., Constructor: a System For The Induction of Probabilistic Models, Proceedings of AAAI-90, 1990, 762-769

Pearl J., Verma T., A Theory of Inferred Causation, Principles of Knowledge Representation and Reasoning, 1991, 441-452, Morgan Kaufmann

Cooper G. F., E. Herskovits, A Bayesian Method For The Induction of Probabilistic Networks From Data, Machine Learning. 1992, 9, 309-347

Lauritzen S., Thiesson B., Spiegelhalter D., Diagnostic Systems Created by Model Selection Methods: A Case Study., AI and Statistics IV, Volume Lecture Notes in Statistics, 143-152. Springer Verlag, New York, 1989

Suzuki J., Learning Bayesian Belief Networks Based on the MDL Principle: an Efficient Algorithm Using the Branch and Bound Technique, IEICE Trans. Inf. & Syst., Vol. E82, No. 2 February, 1999

Cheng J., Greiner R., Learning Bayesian Belief Network Classifiers: Algorithms and System, Lecture Notes in Computer Science 2056, 141-160, 2001

D. M. Chickering, Learning Bayesian NP-Complete, Learning from Data: AI and Statistics, Springer and Verlag, 1996

D. Heckermann, Bayesian Networks for Data Mining, Journal of Knowledge Discovery and Data Mining 1(1), pag. 79-119, Kluwer Academic Publishers, 1997

Bouckaert R., Probabilistic Network Construction Using the Minimum Description Length Principle, Lecture Notes in Computer Science, Vol. 747, 1993

Rissanen J., Modeling by shortest data description, Automatica, Vol. 14, pp. 465-471, 1978

Spirtes, P., Glymour, C., Scheines, R, Causation, Prediction and Search, MIT press, 2001

Cheng , J., Bell, D., Liu, W., Learning belief networks from data: an information theory based approach, Proceedings of the Sixth ACM International Conference on Information and Knowledge Management, 1997

Heckermann, D., Geiger, D., and Chickering, D.. Learning Bayesian Networks. The Combination of Knowledge and Statistical Data. Machine Learning, 1995 20(3):197-243

Cheng , J., Bell, D., Liu, W., Learning Bayesian networks from data: an efficient approach based on information theory, Conference on Information and Knowledge Management, 1997

Bell, D., Cheng , J., Liu, W., An Algorithm for Bayesian Belief Network Construction from Data, Proceedings of AI&STAT'97, Ft. Lauderdale, Florida, 1997

Chow, C.K., Liu, C.N., Approximating Discrete Probability Distribution with Dependence Trees, IEEE Trans. Information Theory, vol.14, 1968

Geiger, D., An Entropy Based Learning Algorithm of Bayesian Conditional Trees, Dubois et al., pp. 92-97

Lam, W., Bacchus, F., Learning Bayesian Belief Networks: an Approach Based on the MDL principle, Computational Intelligence, Vol. 10-4, 1994

Colace, F., De Santo, M., Foggia, P., Vento, M., Bayesian Network Structural Learning from Data: an Algorithms Comparison, Proceedings of International Conference on Enterprise Information Systems, Porto, 2004

Ho TK, Hull JJ, Srihari SN, Decision Combination in Multiple Classifiers, IEEE Trans. On PAMI, vol. 16, 1994

Kittler J., Hatef D., Matas J., On Combining Classifiers, IEEE Trans. On PAMI, vol. 20 n. 3, 1998