Combining Multiple Perspectives

Bikramjit Banerjee BIKRAM@EULER.MCS.UTULSA.EDU
Sandip Debnath DEBNATH@EULER.MCS.UTULSA.EDU
Sandip Sen SANDIP@KOLKATA.MCS.UTULSA.EDU
Department of Mathematical & Computer Sciences, University of Tulsa, Tulsa, OK 74104 USA

Abstract

We consider a group of Bayesian learners whose interactions with the environment and other agents allow them to improve their model of the dependency among various factors that have influence on their interactions with the environment. Effective collaboration can improve the performance of isolated individual learners. We present a mechanism to pool together the knowledge of many modelers in the domain, each of whom may have only partial access to the environment. The application domain used in this study is a multiagent negotiation problem. We present results to compare the performance of such knowledge-composition against isolated learners, as also against a learner who has complete access to the environment.

1. Introduction

Multiagent systems may consist of self-interested agents who have individual goals. Such agents may constantly learn and update their beliefs regarding the environment, but they are essentially solitary and may have only partial access to their environment. The individual refinements of knowledge may prove inadequate when exploration of the environment is costly or time-consuming. However, the chief limitation of individual explorations in partially accessible environments is that each agent may end up with incomplete models. In such cases, agents may benefit from pooling together their individual knowledge to develop a more complete model of the environment.

In this paper, we consider a group of learners who represent their belief-structures regarding their domains by Bayesian networks. Such a learner will incrementally update its *a priori* domain model, given their observations of events in the environment. Our goal is to collect individual belief-estimates and models of environment from these agents and produce a more consistent and comprehensive model of the environment.

We study this problem in the context of negotiations among self-interested agents in a multiagent environment. In most domains, agent-behaviors are strongly coupled in the sense that an agent can influence actions of other agents, i.e. an agent A can create a context that prompts another agent B to act in a way that preferred by A. To be an effective negotiator, however, agent A must be aware of the belief-structure of B. This means that agent A must model the various factors and their relative influences on B's actions. Such network of influences can be derived by observing B's actions under various environmental conditions. Any one agent, however, may be privy to only a subset of factors influencing B's decision or may observe only some of B's activities. Multiple observers then stand to gain by pooling together their observations and learned models. The combined mode may allow the collaborators to better negotiate with B.

In this paper we present a method for combining such multiple models and experimentally compare the performance-improvement over individual learners. We also found that the combined model falls short of the model that can be built if a single agent had full observation, i.e., could see all the factors influencing B and observe all its actions.

2. Bayesian Networks

A Bayesian network (Jensen, 1996; Charniak, 1991) is a graphical method of representing dependencies and independencies among different variables that together define a model of a real-world situation. Technically, it is a directed acyclic graph with nodes being the variables and each directed edge representing a dependence between two of them. In addition to its structure, a Bayesian network is also specified by a set of parameters θ, that quantify the network.

Consider a vector $\mathbf{X}$ of variables and an instantiation-vector $\mathbf{x}$ (that assigns a value x_i to each variable X_i in $\mathbf{X}$ from its domain D_i). If the immediate parents of a variable X_i is the vector Π_{X_i}, with its instantiation π_{x_i}, then

$$\mathbf{Pr}[\mathbf{X} = \mathbf{x}|\theta] = \prod_i \mathbf{Pr}[X_i = x_i|\mathbf{\Pi}_{X_i} = \pi_{x_i}, \theta]$$

Whenever the instantiation is clear from the context, the above is also written as,

$$\mathbf{Pr}[\mathbf{X}|\theta] = \prod_i \mathbf{Pr}[X_i|\mathbf{\Pi}_{X_i}, \theta]$$

This defines the joint distribution of the variables in $\mathbf{X}$, where each variable X_i is conditionally independent of its non-descendents given its parents, or conditioning variables. For our purpose, we actually use an extension of Bayesian networks, called *influence diagrams* (Shachter, 1994), that incorporate action/decision nodes in the network. We represent belief structures as Bayesian networks for the following major reasons:

- Bayesian networks can readily handle incomplete data sets. This is because Bayesian networks offer a way to encode the correlations among the input variables.
- Bayesian networks allow one to learn about causal relationships. This is useful to gain understanding about a problem domain. In addition it allows to make predictions in the presence of interventions.
- Bayesian networks in conjunction with Bayesian statistical techniques facilitate the combination of domain knowledge and data.
- Bayesian networks in conjunction with Bayesian methods offers an efficient and principled approach to avoiding over fitting of data.
- Bayesian networks offer a method of updating the belief or the probability of occurrence of the particular event for the given causes.

2.1 Learning Bayesian Networks

The problem of learning Bayesian network from data has been addressed by several researchers (Buntine, 1994; Chow & Liu, 1968; Heckerman et al., 1995; Provan & Singh, 1996; Sprites et al., 1993). There are broadly a couple of trends in learning Bayesian networks from data. One is based on non-Bayesian approach (e.g. Sprites et al. (1993)) and other is using the Bayesian framework itself (Cooper & Herskovits, 1992; Buntine, 1994; Heckerman et al., 1995). More recently learning Bayesian networks using feature selection is addressed by Provan and Singh (1996) who select a subset of features that maximize the predictive accuracy of the network.

The problem of learning Bayesian networks involves two tasks. The first is to use the data set to best fit the graphical model and the second is to extract the conditional probabilities defining the dependencies. Herskovitz and Cooper (1990) show that learning complex structures is possible for sample sizes over $10,000$.

In general, data are provided as instantiation vectors, and are used to incrementally improve the network. The main approaches to learning a Bayesian network can be broadly classified into following categories:

- Hypotheses testing approaches using greedy search techniques (Edwards & Havranek, 1987).
- Bayesian Scoring Metrics (Cooper & Herskovits, 1992; Heckerman et al., 1995) that optimize a log-likelihood or an entropy score, over various possible structures.
- The minimum description length principle (Bouckaert, 1995; Lam & Bacchus, 1994). These techniques are asymptotically optimal, given complete data.
- Extended likelihood approaches, which use a different scoring criterion to be maximized rather than the maximum likelihood (Sclove, 1994; Poland & Shachter, 1994).
- Minimum information, such as minimum description length (Rissanen, 1987), minimum message length (Wallace & Freeman, 1987) or minimum complexity (Barron & Cover, 1991) techniques.

The training process, however, is often not reliable if one or more of the variables is missing in some instantiations. Learning Bayesian networks from incomplete or missing data has been addressed by Friedman (1997; 1998), Ghahramani and Jordan (1994), and Singh (1997) among others. All of these deal with online rather than batch learning from incomplete data. Though maintaining the instance vectors in memory for batch learning is a possibility, it is not a viable solution. The concept of *sufficient statistics* (Heckerman, 1995) enables one to compute $\mathbf{Pr}[X_i|\mathbf{\Pi}_{X_i}]$, by keeping counts of each distinct instantiation of X_i and its parents, or in general, the counts of distinct instantiations of the vector $\mathbf{X}$, given by $N(\mathbf{x})$. Though this is a significant tool in batch learning Bayesian networks from complete data, its sufficiency in case of incomplete data has not been addressed.

Though dealing with incomplete data for a single agent has received significant attention, we have found very little concern for cases where a group of self-interested agents having partial knowledge (incomplete data) and learning in parallel, combine their knowledge to learn better models of their environment.

3. Summarization From Multiple Bayesian Networks

We assume a collection of agents, M, each of which has summarized its findings as a network structure, S^i, and a set of sufficient statistical records of the form $N^i(\mathbf{x})$, where the superscript i indicates the ith agent. Suppose the ith agent has seen n^i training examples in all, and the amount of overlap in the training examples is n_0. The first step is to find the combined network structure that is intuitively given by $\bigcup_{i=1}^{|M|} S^i$. This union is performed starting at the union of the sink nodes, and gradually proceeding upwards, for the union of their parents, and so on. For the $|M|$ agents in the system, this union is performed for each agent with the $|M|-1$ other agents, sequentially. If the ordering of the nodes imposed by the individual networks are inconsistent, then the one having maximum posterior probability is chosen and the structures of the rest are made compatible with it by performing necessary arc-reorientations. See Matzkevich and Abrasamson (Matzkevich & Abramson, 1993) for an efficient approach for the same.

Once we have the combined network structure, the next step is to compute the parameters of the network, in a way so as to use collective knowledge to better represent the model than the individual agents. The key idea is simple : if the network structures are identical, then the combined parameters are average of those of the individual agents. Otherwise for every variable missing in a structure, we assume it has been summed through all of its possible values. In the latter case we use frequencies from agents who have observed those variables, to fill in for agents who have not. We assume that the networks learned by all the agents have the same set of sink nodes. Starting from these nodes, and proceeding successively for higher nodes with non-zero indegree, we perform the following steps for each pair of agents i, j:

- Let P_i and P_j be the set of parents of the sink node G, in i and j respectively. Let the set of parents common to them be $I = P_i \bigcap P_j$. Also let the sets $P_i \backslash P_j = Y_i$ and $P_j \backslash P_i = Y_j$.

- **Case:** $I = \Phi, \quad Y_i \neq \Phi, \quad Y_j \neq \Phi$
Here we expand the distribution table of i to include Y_j and that of j to include Y_i. The new values are [1]

$$N^j(Y_i Y_j G) = \frac{N^i(Y_i G)}{N^i(G)} N^j(Y_j G),$$

$$N^i(Y_i Y_j G) = \frac{N^j(Y_j G)}{N^j(G)} N^i(Y_i G).$$

- **Case:** $I \neq \Phi, \quad Y_i \neq \Phi, \quad Y_j = \Phi$
Here the probability tables for i remain intact, and that of j is enriched, according to the rule

$$N^j(I Y_i G) = \frac{N^i(I Y_i G)}{N^i(I G)} N^j(I G).$$

- **Case:** $I \neq \Phi, \quad Y_i \neq \Phi, \quad Y_j \neq \Phi$
This case can be treated similarly as in the first case. Figure 2 shows an example, based on the network in figure 1, for this case. The table modification rules are

$$N^j(I Y_i Y_j G) = \frac{N^i(I Y_i G)}{N^i(I G)} N^j(I Y_j G),$$

$$N^i(I Y_i Y_j G) = \frac{N^j(I Y_j G)}{N^j(I G)} N^i(I Y_i G).$$

In all the above cases, the final tables of i and j contain the same set of variables, and we bypass cases where $I \neq \Phi, \quad Y_i = \Phi, \quad Y_j = \Phi$ since these preprocessing steps are meant to render the different networks of the agents, identical in structure (we also ignore the possibility of any of P_i and P_j being null). Having pointed all the distinct possibilities, we can combine the updation rules in the 3 cases, into a single one given by the 3rd case (which is the most general)

$$N^i(I Y_i Y_j G) = \frac{N^j(I Y_j G)}{N^j(I G)} N^i(I Y_i G)$$

for the ith agent, when paired with the jth agent, again ignoring the case $I \neq \Phi, \ Y_i = \Phi, \ Y_j = \Phi$, which we deal later. Finally, when the network structures of the agents are rendered identical (after $\begin{pmatrix} |M| \\ 2 \end{pmatrix}$ steps), we compute the combined parameters from the

[1]For the sake of brevity, we write $N^i(X)$ to mean the number of observations of agent i for a certain instantiation of the node-set X.

total amount of overlap in observations of the agents for all the $|M|$ agents.

The combined estimate of $N(I,G)$ from all agents is computed using a weighted average for estimates of the overlaps (the weights being the total number of observations of each agent) as

$$N(I,G) = \sum_{i=1}^{|M|} N^i(IG)(1 - \frac{n_0}{n^i} + \frac{n_0}{\sum_i n^i}).$$

The first three cases, as noted earlier, are preprocessing steps, that are repeatedly run for each internal node for each pair of agents. At the end of the procedure, we compute $\mathbf{Pr}[X_i|\mathbf{\Pi_{X_i}}]$ as

$$\mathbf{Pr}[X_i|\mathbf{\Pi_{X_i}}] = \frac{N(X_i, \mathbf{\Pi}_{X_i})}{N(\mathbf{\Pi}_{X_i})}.$$

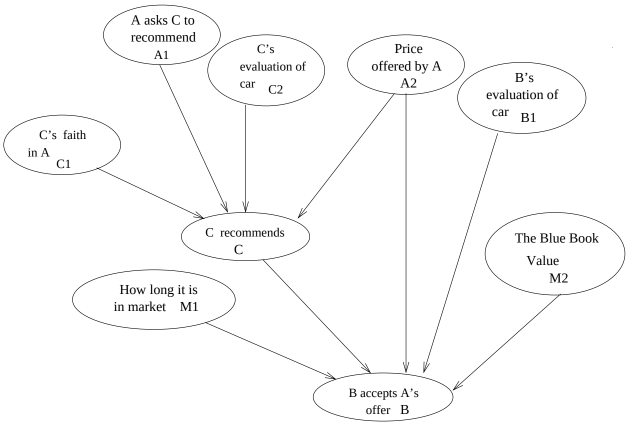

Figure 1. A Bayesian network for negotiation.

4. Combining Perspectives For Negotiation

In a previous work, we had applied Bayesian networks in multi-agent negotiation, where we had considered a buyer-seller scenario and allowed the buyer to model the seller's strategies to maximize its payoff (Banerjee et al., 1999). We recall the example briefly here.

4.1 A Multi-agent Negotiation Scenario

In this example, an agent A needs to sell some goods to another agent B (e.g a second hand car). Now A and B will negotiate the price of the car. The financial capacity of agent B influences its *reservation price* i.e the upper limit of payment that B is unwilling to exceed. The agent A may find it useful to ask agent

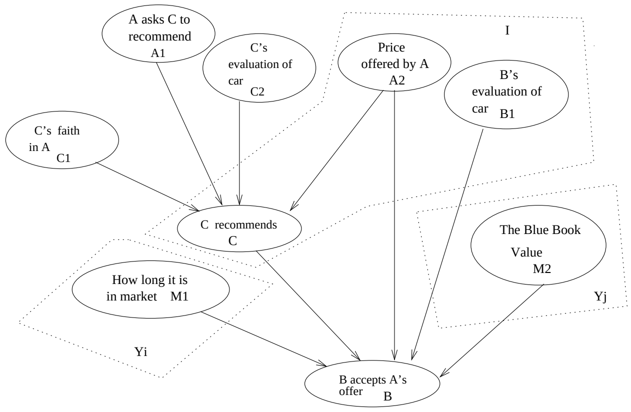

Figure 2. When the sink node is B, one agent does not know M_1 and another does not know M_2. The intersection node set is I.

C to recommend its car to agent B. This recommendation depends upon several other beliefs on part of C. Whether C will recommend A's car depends on C's faith in A, the fairness of A's offer and C's own perception of the condition of the car. In Zeng and Sycara (1997) bargainers had beliefs only regarding each other's reservation prices, which they tried to model, in order to settle to a point within the zone of agreement. This assumes the existence of such an overlapping zone. However an intermediate agent (such as C here) may persuade one of the bargainers i.e. agent B (note that its reservation price is not the upper limit of its payment-capacity, but that of what it is willing to pay in the deal) to be flexible in its dispositions thus *raising the utility of the other*, even when an initial zone of agreement does not exist. We consider the viewpoint of only the seller,A, who models both B and C, and updates its beliefs regarding both, in order to decide whether mediation by C is conducive to its end, as also whether the price-offer to B should be modified or not. Though the joint utility of the system is likely to suffer, A however, benefits from such modeling.

The goal of the seller is to set the *negotiation context* by instantiating the nodes that influence the buyer's decision in such a way as to increase the likelihood of the buyer accepting its offer. We can imagine a situation where multiple sellers perform mutual consultation to arrive at a comprehensive model of the buyer's behavior. This process of consultation, leading to a *conformal belief structure*, can be effectively captured by the procedure suggested in this paper. We note that this process can be symmetric, in the sense that if the buyers were modeling the marketplace, they could also perform such mutual consultations. We do not consider the latter possibility here.

Table 1. A portion of the conditional probability table used to generate data for the agents.

A2	C2	A1	C1	C	$\overline{C}$
0	0	0	0	0.67	0.33
0	0	0	1	0.565	0.435
0	0	0	2	0.095	0.905
0	0	1	0	0.55	0.45
0	0	1	1	0.47	0.53
0	0	1	2	0.066	0.934
0	1	0	0	0.42	0.48
0	1	0	1	0.34	0.66
0	1	0	2	0.055	0.945
0	1	1	0	0.32	0.68
2	2	1	0	0.29	0.71
2	2	1	1	0.19	0.81
2	2	1	2	0.043	0.957

5. Experimental Results

We have tested our procedure based on the conditional probability table, part of which is given in Figure 1. Multiple instance-vectors are generated according to this table. Each of the agents has been given a different training data set by sampling with replacement from this master instance set. Also, they observe only a subset of the variables in each instantiation; e.g. one of the agents does not observe M_1 in Figure 2, and the other does not observe M_2. We also allow a portion of the instance-set observed by the agents to overlap. Each agent first learns the structure of the network from its instance-set, using structure-learning from Bayesian Knowledge Discoverer (Ramoni & Sebastiani,), and then computes the frequencies of the distinct instantiations to form the *sufficient statistics*. Our combination-procedure is then used to arrive at the composite network and the parameters are computed for this network. Finally, these are compared with the values in Figure 1. The error between the composite-network parameters and the original values is measured by a metric (similar to cross-entropy), which is defined as:

Definition 1 *If $Q(x)$ is the composite-network distribution, while $P(x)$ is that for the original network (figure 1)) over x, then the error $e(P,Q)$, between these distributions is given by*

$$e(P,Q) = \sum_x (P(x) - Q(x)) * \log[\frac{P(x)}{Q(x)}].$$

The overlap in the data-sets has been varied from 0% to 100% in step of 5. The average error, e, of the agents A, B, and their combined knowledge have been plotted in figure 3 against these overlaps.

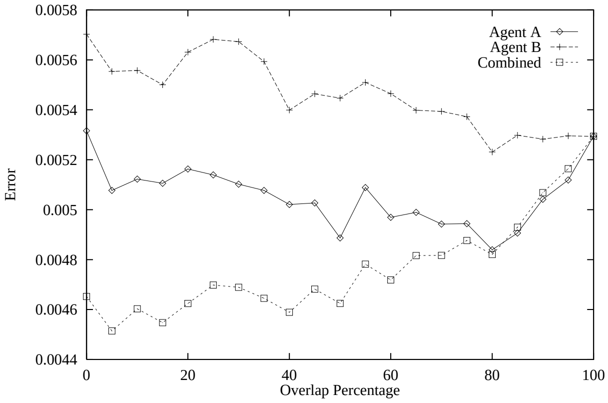

Figure 3. Relative performances of the two agents A,B, against our procedure, represented by the Combined result.

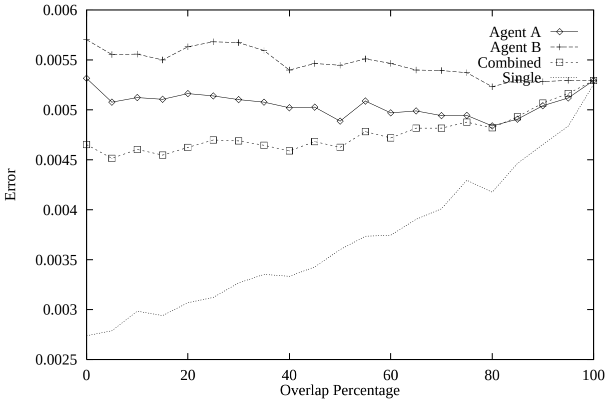

Figure 4. Relative performances of the two agents A,B, and our procedure, against the Single agent.

The error-data in the plots are averaged over 250 training data sets of $10,000$ instances each. When there is no overlap of observations, each of the agents A and B has seen a different set of $10,000$ data vectors for each run, and the combined knowledge has the least average error. On the other hand, for a 100% overlap between these two training sets, the errors of both the agents converge with that of the combined knowledge. This is because all three have seen essentially the same data set. To summarize, with more of distinct data (i.e. in case of less overlap of observations between the individual learners), the combined model provides greater improvement.

In our experiments, the error is computed as the average of the error metric in definition 1, in predicting the probability values over all the children nodes over all the 250 data sets. If the size of a data set is increased, each of the agents has a better view of the network and consequently the error is reduced.

We have also compared the combined result with a single learner who is privy to the entire master training set. Such an agent learns the graphical structure from which the data was generated, if sufficient number of training instances were given. It learns better than the individual agents with partial information as also against our procedure. We expected the combined-model to be more accurate than that learned by the agents A, B but not as good as that of the single agent. This is corroborated by experimental results in Figure 4. The difference of the average error between the single learner and combined learner is a result of the information-loss that occurs due to the vertical-partitioning of data (partial observation by individual learners) and redundant observations (some data are observed by both agents which skews the combined distribution).

6. Related Work

There has been some study in multiagent belief revision in context of revising an agent's model of other agents' models (Gaspar, 1991; van der Meyden, 1994). More recently Kfir-Dahav and Tennenholtz (1996) consider multi-agent belief revision in heterogeneous systems where each agent has some private and some shared knowledge about the environment. In that work, the authors have assumed a two-phase iterative belief-revision process using each incoming instance. In the first phase either only the observing agent updates its belief in private domain or all the agents update their beliefs in shared domain depending on which domain the current observation belongs to (in the latter case the observation is communicated by the observing agent to the other agents) and in the second phase the agents coordinate their beliefs. The authors have identified some requirements for such a process and have shown that all of them are not satisfiable simultaneously. In this paper, we have considered Bayesian learners with identical portions of networks representing shared domain and unique portions for private belief, but have not considered online iterative belief revision. Instead we have focused on how the possibly different Bayesian network models learned by several agents performing joint exploration of their environment can be combined to better summarize their knowledge. As such we have not dealt with learning per se, but assumed that the individual agents have already learned their networks from partially overlapping set of training examples. [2] In Freund et al. (1998), several "weak" rankings or preferences are combined by boosting to yield a better preference order, but the problem of combining the qualitative and quantitative beliefs (as in Bayesian networks) from several learners in heterogeneous partially-observable environments has not been addressed. We have proposed a simple but novel method to perform such combination and have evaluated its performance for various amounts of experience-overlap.

7. Conclusion And Future Work

We have studied the problem of individual learners being limited by the incompleteness of their observation of domain-variables that influence other agents' decision-making. We presented a model-composition method by which multiple learners can compose their incomplete models learned from partially overlapping (both in terms of actual instances and in terms of the domain-variables observed) data-sets. Combined models are found to be more consistent with the actual domain model though not as accurate as that learned by a single agent with full observability.

We plan to do further experiments by varying the overlap of the set of variables observed. Another dimension to be investigated is the number of agents involved in the composition process.

Acknowledgement

The authors gratefully acknowledge the use of BKD (Bayesian Knowledge Discoverer), provided by a joint effort of the Knowledge Media Institute and the Department of Statistics of The Open University. This work has been supported, in part, by an NSF CAREER Award: IIS-9702672.

[2] Hence we have allowed concurrent exploration instead of just one agent perceiving a data-tuple at a time as in Kfir-Dahav and Tennenholtz (1996).

References

Banerjee, B., Debnath, S., & Sen, S. (1999). Using Bayesian networks to aid negotiations among agents. In *Working notes of theAAAI-99 workshop on negotiation: Settling conflicts and identifying opportunities*. AAAI Press.

Barron, A. R., & Cover, T. M. (1991). Minimum complexity density estimation. *IEEE Transactions on Information Theory*, *37*, 1034–1054.

Bouckaert, R. (1995). *Bayesian belief networks: From construction to inference*. Doctoral dissertation, University of Utrecht, The Netherlands.

Buntine, W. L. (1994). Operations for learning with graphical models. *Journal of Artificial Intelligence Research*, *2*, 159–225.

Charniak, E. (1991). Bayesian networks without tears. *AI Magazine*, *12*, 50–63.

Chow, C. K., & Liu, C. N. (1968). Approximating discrete probability distributions with dependence trees. *IEEE Transaction on Information theory*, *14*, 462–467.

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

Edwards, D., & Havranek, T. (1987). A fast model selection procedure for large families of models. *Journal of American Statistical Association*, *82*, 205–211.

Freund, Y., Iyer, R., Schapire, R. E., & Singer, Y. (1998). An efficient boosting algorithm for combining preferences. *Proceedings of the Fifteenth International Conference on Machine Learning* (pp. 170–178). San Francisco, CA: Morgan Kaufman.

Friedman, N. (1997). Learning Bayesian networks in the presence of missing values and hidden variables. *Proceedings of the Fourteenth International Conference on Machine Learning* (pp. 125–133). San Francisco, CA: Morgan Kaufmann.

Friedman, N. (1998). The Bayesian structural EM algorithm. *Proceedings of the Fifteenth Conference on Uncertainty in Artificial Intelligence* (pp. 129–138). San Francisco, CA: Morgan Kaufmann.

Gaspar, G. (1991). Communication and belief changes in a society of agents: Towards a formal model of an autonomous agent. In Y. Demazeau and J. P. Muller (Eds.), *Decentralized AI 2*, 245–255. North-Holland/Elsevier.

Ghahramani, Z., & Jordan, M. (1994). *Learning from incomplete data* (Technical Report). Artificial Intelligence Lab, Massachesets Institute of Technology, Cambridge, MA.

Heckerman, D. (1995). *A tutorial on learning Bayesian Networks, msr-tr-95-06* (Technical Report). Microsoft Research, Redmond, WA.

Heckerman, D., Geiger, D., & Chikering, D. M. (1995). Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, *20*, 197–243.

Herskovitz, E. H., & Cooper, G. F. (1990). Kutato: An entropy driven system for construction of probabilistic expert system from databases. *Proceedings of the Sixth Conference on Uncertainty in Artificial Intelligence* (pp. 54–62). San Mateo, CA: Morgan Kaufmann.

Jensen, F. V. (1996). *An Introducton to Bayesian Networks*. New York: Springer Verlag.

Kfir-Dahav, N. E., & Tennenholtz, M. (1996). Multi-agent belief revision. *Proceedings of Sixth Conference on Theoretical Aspects of Rationality and Knowledge* (pp. 175–194).

Lam, W., & Bacchus, F. (1994). Learning Bayesian belief networks: An approach based on the mdl principle. *Computational Intelligence*, *10*, 269–293.

Matzkevich, I., & Abramson, B. (1993). Deriving a minimal I-map of a belief network relative to a target ordering of its nodes. *Proceedings of the Ninth Conference on Uncertainty in Artificial Intelligence* (pp. 159–165). San Mateo, CA: Morgan Kaufmann.

Poland, R. B., & Shachter, R. D. (1994). Three approaches to probability model selection. *Proceedings of the Tenth Conference of Uncertainty in Artificial Intelligence* (pp. 478–483). San Mateo, CA: Morgan Kaufmann.

Provan, G. M., & Singh, M. (1996). Learning Bayesian networks using feature selection. In D. Fisher and H. Lenz (Eds.), *Learning from data*, vol. 112 of *Lecture Notes in Statistics*, 291–300. Berlin: Springer-Verlag.

Ramoni, M., & Sebastiani, P. Bayesian knowledge discoverer, http://kmi.open.ac.uk/projects/bkd/.

Rissanen, J. (1987). Stochastic complexity. *Royal Statistical Society*, *49*, 223–239.

Sclove, S. L. (1994). Small-sample and large sample statistical model selection criteria. In P. Cheeseman and R. Oldford (Eds.), *Selecting models from data: Artificial intelligence and statistics IV*, 31–39. Berlin: Springer-Verlag.

Shachter, R. (1994). Evaluating influence diagrams. *Operations Research*, *34*, 871–882.

Singh, M. (1997). Learning Bayesian networks from incomplete data. *Proceedings of the Fourteenth National conference on Artificial Intelligence* (pp. 534–539). Menlo Park, CA: AAAI Press/The MIT Press.

Sprites, P., Gleymour, C., & Scheines, R. (1993). *Causation, prediction, and search.* New York: Springer Verlag.

van der Meyden, R. (1994). Mutual belief revision. *Proceedings of the Fourth International Conference on Principles of Knowledge Representation and Reasoning* (pp. 595–606).

Wallace, C. S., & Freeman, P. R. (1987). Estimation and inference by compact encoding. *Royal Statistical Society*, *49*, 240–265.

Zeng, D., & Sycara, K. (1997). Benefits of learning in negotiation. *Proceedings of the Fourteenth National Conference on Artificial Intelligence* (pp. 36 – 41). Menlo Park, CA: AAAI Press/MIT Press.