An Empirical Study of Encoding Schemes and Search Strategies in Discovering Causal Networks

Honghua Dai, Gang Li, and Yiqing Tu

School of Computing and Mathematics, Deakin University
221 Burwood Highway, Vic 3125, Australia
{hdai,gangli}@deakin.edu.au
yiqingtu@hotmail.com

Abstract. Efficiently inducing precise causal models accurately reflecting given data sets is the ultimate goal of causal discovery. The algorithm proposed by Wallace et al. [10] has demonstrated its ability in discovering *Linear Causal Models* from data. To explore the ways to improve efficiency, this research examines three different encoding schemes and four searching strategies. The experimental results reveal that (1) specifying parents encoding method is the best among three encoding methods we examined; (2) In the discovery of linear causal models, local Hill climbing works very well compared to other more sophisticated methods, like Markov Chain Monte Carto (MCMC), Genetic Algorithm (GA) and Parallel MCMC searching.

1 Introduction

Graphical Model is a powerful knowledge representation and reasoning tool under uncertainty [8]. However, the manually construction of *Graphical Model* is usually time-consuming and subject to mistakes. Therefore, algorithms for automatic construction, that occasionally use the information provided by an expert, can be of great help [5]. As *Graphical Model* can often be plausibly understood as describing causal relations, the automatic construction of *Graphical Model* is usually referred as *Causal Discovery*.

In social sciences, there is a class of limited *Graphical Model*, usually referred as *Linear Causal Models*, including *Path Diagram* [12], and *Structural Equations Model* [1]. In *Linear Causal Models*, effect variables are strictly linear functions of exogenous variables. Although this is a significant limitation, its adoption allows for a comparatively easy environment in which to develop causal discovery algorithms. In 1996, Wallace et al. successfully introduced an information theoretic approach to the discovery of *Linear Causal Models*. This algorithm uses Wallace's *Minimum Message Length* (MML) criterion [9] to evaluate and guide the search of *Linear Causal Model*, and their experiments indicated that MML criterion is capable of recovering *Linear Causal Model* which is quite accurate reflection of the original model [10]. In 1997, Dai et al. further studied the reliability and robustness issues in causal discovery [2], they closely examined the

T. Elomaa et al. (Eds.): ECML, LNAI 2430, pp. 48–59, 2002.

relationships among the complexity of the causal model to be discovered, the strength of the causal links, the sample size of given data set and the discovery ability of individual causal discovery algorithms.

Two main issues involved in the process of *Causal Discovery* using MML are *Encoding* and *Searching*. In order to improve the efficiency of discovery algorithm, an optimal encoding scheme and an efficient search strategy are highly demanded. In this paper, we examine three different encoding schemes for describing the structure of *Linear Causal Models*, and compare four different search strategies to explore the possibilities to improve discovery efficiency while preserving discovery accuracy.

The paper is organized into 5 sections. Section 2 describes two structure encoding schemes proposed in [10], and gives a new encoding scheme. Section 3 describes four different search strategies. In Sec. 4 three encoding schemes and four search strategies are compared. Finally, we conclude this paper in Sec. 5.

2 Causal Structure Encoding Schemes

As reported in [10], the basic idea of causal discovery via MML is that an encoding scheme based on the minimum message length principle needs to be provided to describe

1. the causal structure, which is a *Directed Acyclic Graph* (DAG) for *Linear Causal Model*
2. the strength (model parameters) of the causality in the *Linear Causal Model*
3. the data assuming the *Linear Causal Model* is true

For each candidate model from the model space, we calculate the total message length based on the given data, and the one with the shortest total message length will be chosen as the best model. According to information theory, the total message length $L(M, D)$ is given by,

$$\begin{aligned} L(M, D) &= -logP(M) - logP(D|M) \\ &= L(M) + L(D|M) \end{aligned} \tag{1}$$

where $L(M)$ is the cost of encoding the causal model M, and $L(D|M)$ is the cost of encoding the given data D assuming the model M is true. As a model is represented with a DAG and the path coefficients, so $L(M)$ is composed of two main parts: the cost of encoding the causal structure, $L^{(s)}$, and the cost of encoding the model parameters, $L^{(p)}$, i.e.,

$$L(M) = L^{(s)} + L^{(p)} \tag{2}$$

In general, the encoding scheme for describing model parameters and the given data is relatively stable and mature. Here we mainly examine the encoding schemes for describing the casual structure, a *Directed Acyclic Graph* (DAG).

2.1 Scheme 1: Specifying a Total Ordering and Arc Connections

A *Directed Acyclic Graph* (DAG) with K nodes can be encoded by specifying a total ordering (requiring $\log K!$ bits) and specifying which pairs of nodes are connected [1]; this requires $\frac{K(K-1)}{2}$ bits on the assumption that the probability that a link is present is 1/2. It corresponds to maximal ignorance about the degree of connectedness of the graph. We avoid the use of explicit prior information about the causal models we are looking for. It is enough to specify the presence or absence of arcs, since directionality is implied by the ordering already provided. Since more than one ordering is consistent with the DAG, actually specifying a *particular* ordering is inefficient, so we reduce the message length by the number of bits needed to select among the ϕ total orderings consistent with the DAG. Hence,

$$L_1^{(s)} = log\ K! + \frac{K(K-1)}{2} - log\ \phi. \tag{3}$$

2.2 Scheme 2: Specifying Total Acyclic Orientations

The second method for calculating the cost of describing a DAG begins by specifying the undirected graph, which costs $\frac{K(K-1)}{2}$ bits, and then specifies the particular direction which each arc is to assume. This results in an acyclic graph. That is, we count the number of possible acyclic orientations; the logarithm of that number is the number of additional bits required. In order to do this count, we can subtract the number of cyclic orientations ρ from the number of total orientations, which is 2^ν, where ν is the number of undirected arcs. Hence,

$$L_2^{(s)} = \frac{K(K-1)}{2} + log(2^\nu - \rho). \tag{4}$$

Previous experimental results show that these two methods result in MML costs that are very close for a wide variety of simple graph structures[10] we tested, so we can expect that the choice of encoding method will make little difference to experimental results. In practice, until the introduction of $L_3^{(s)}$, our implementation of $L_1^{(s)}$ is faster. To further improve the efficiency of the discovery algorithm, we introduced the following new encoding scheme.

2.3 Scheme 3: Specifying Parents Set

The structure of a *Directed Acyclic Graph* can be described by specifying its parents set *Parents(x)* for each node of the DAG. This description consists of the number of parents, followed by the index of the set *Parent(x)* in some enumeration of all sets of its cardinality. So the cost for encoding the causal structure can be calculated using:

$$L_3^{(s)} = \sum_i \left(\log K + \log \binom{K}{r_i} \right) \tag{5}$$

[1] Schemes 1 and 2 were introduced by Wallace, Korb and Dai in 1996 [10].

Where $r_i = |Parents(X_i)|$. To avoid intensive computational time cost in calculating $\log\binom{K}{r_i}$, we use *Stirling's approximation* formula $x! = x^x e^{-x}\sqrt{2\pi x}$, so we get,

$$\log\binom{K}{r_i} \approx (K - r_i)\log\left(\frac{K}{K - r_i}\right) + r_i \log\left(\frac{K}{r_i}\right) \tag{6}$$

Thus, we have,

$$L_3^{(s)} = \sum_i \left(\log K + (K - r_i)\log\left(\frac{K}{K - r_i}\right) + r_i \log\left(\frac{K}{r_i}\right)\right) \tag{7}$$

This encoding scheme works much faster than using the formula $L_1^{(s)}$ and $L_2^{(s)}$. This can be seen from the experimental results reported in section 4.

3 Model Space Search Strategies

For a given data set, the number of possible causal structures is exponential in the number of variables. To find out the best structure from this huge space, a efficient search strategy is highly demanded [5] [2]. Hill-climbing search was used in our previous work [3, 10] [3].

In the past decade, there is an increasing amount of work on the application of *Markov Chain Monte Carlo* and *Evolutionary Algorithm* to complex learning, search and optimization problems. In 1993, Madigan proposed the MC^3 algorithm[4] which uses Metropolis sampling to search over structures for graphical models [7]. In 1996, Larrañaga et al. tackle the problem of searching for a Bayesian Network structure that maximizes the BDE metric with hybrid genetic algorithm, given a total order of all variables [6]. This algorithm was later extended to the general case that no ordering between the variables is assumed, and they make use of a repair operator to convert offspring structures that are cyclic into acyclic. In 1999, Wong et al. used MDL metric and evolutionary programming for the optimization in the search process [11]. In 1998, Holmes used genetic operators to inform the proposal distribution for a Metropolis sampling algorithm [4], proposed the *Parallel Markov Chain Monte Carlo* algorithm and found that their sampler converged quicker than standard Metropolis sampling.

In this section, we describe four different search strategies for discovering causal models: *Hill-climbing*, *Genetic Algorithm*, *Markov Chain Monte Carlo* (MCMC) and parallel MCMC. Their performance will be compared in Sec. 4.

3.1 Hill-Climbing (HC)

This search method could start with a seed DAG provided by user, or a null graph without any edge, then attempt to add an edge if there is none or to

[2] The search is guided by message length, and we have an assumption that the model with minimum message length is the best model.

[3] Random Restarting can be integrated to overcome the problem of local optima.

[4] MC^3 means Markov Chain Monte Carlo Model Composition.

delete or to reverse it if there already is one. Such adding, deleting or reversing is done only if such changes result in a decrease of the total message length of the new structure. If the new structure is better, it is kept and then try another change. This process continues until no better structure is found within a given number of Hill-climbing steps, or the search from the whole structure space is completed.

3.2 Markov Chain Monte Carlo (MCMC)

Given a data set D, the posterior probability $p(M|D)$ of each model M can be directly calculated from Bayes theorem,

$$p(M|D) = \frac{p(D|M)p(M)}{p(D)} \tag{8}$$

Where $p(M)$ is the prior probability for the model M, $p(D|M)$ is the likelihood of the model for the data set D, and $p(D)$ can be thought as a normalizing constant.

If we are interested in finding the model that maximizes the posterior probability, causal discovery can be formulated as an optimization problem that finds the maximum a posterior probability:

$$M^* = arg \max_M p(M|D) \tag{9}$$

The MCMC method for solving this problem is to generate samples M from the distribution $p(M|D)$ and select the best. If we can generate independent samples from the target distribution, the law of *Large Number* ensures that the approximation can be made as accurate as desired by increasing the sample size. MCMC method draws samples independently from the target distribution through a Markov chain having $p(M|D)$ as its stationary distribution.

To perform this sampling, we use a version of the Metropolis algorithm: The current model structure is represented by a Connectivity Matrix, in which a cell $C[i][j]$ having a non-zero value indicates there exists an edge $i \rightarrow j$ in the model structure. Sampling from the posterior over model structures proceeds by a Markov process which steps from one DAG to another in such a way that the chance of visits to a DAG is proportional to its posterior probability. The proposal distribution is determined by the following four variation operators:

- Select two distinct variables i and j uniformly from the domain. If there exists an edge between them, attempt to remove it. Otherwise, attempt to add an edge in either direction.
- Select three distinct variables i, j and k uniformly from the domain, if there exists an edge $i \rightarrow j$, then remove it and add another edge $i \rightarrow k$.
- Select three distinct variables i, j and k uniformly from the domain, if there exists an edge $i \rightarrow j$, then remove it and add another edge $k \rightarrow j$.
- Check the resulting structure, if it contains a cycle, then randomly select and remove an edge from the cycle, so that the resulting structure is a DAG.

If we assume symmetricity in the proposal distribution [5], then the candidate model M' will replace current model M with the following probability:

$$\alpha = \min\{1, \frac{p(M'|D)}{p(M|D)}\} \tag{10}$$

This acceptance rule says that the candidate model is always accepted when its posterior probability is higher than that of current model. Otherwise, it is accepted according to the ratio of two probabilities.

From the formula 8, it can be seen that the posterior probability can be calculated as the ratio of joint probabilities. On the other hand, the theory of MML inference shows that the total message length $L(M, D)$ closely approximates the negative logarithm of the joint probability of model M and data set D, like this

$$\begin{aligned} L(M, D) &= -\log p(M) - \log p(D|M) \\ &= -\log p(D, M) \end{aligned} \tag{11}$$

So we have

$$\begin{aligned} \frac{p(M'|D)}{p(M|D)} &= \frac{p(D, M')}{p(D, M)} \\ &= 2^{L(M,D)-L(M',D)} \end{aligned} \tag{12}$$

Finally, the acceptance probability can be written as

$$\alpha = \min\{1, 2^{L(M,D)-L(M',D)}\} \tag{13}$$

The sampling proceeds until the chain is thought to be converged.

3.3 Genetic Algorithm (GA)

The third method we considered is a Genetic Algorithm. Chromosome representation for the model structure is an $K \times K$ connectivity matrix, in which a cell $C[i][j]$ having a non-zero value indicates there exists an edge $i \rightarrow j$ in the model structure.

The fitness of a chromosome is defined according to the MML cost of the corresponding model structure: The less MML cost, the higher its fitness.

With matrix representation in mind, one crossover operator, three mutate operators, and one repair operator are defined as follows:

Crossover Binary tournament selection is used to select pairs of structures for crossover. One structure with higher fitness from two randomly selected structures, and another one with higher fitness from another two randomly selected structures are selected to crossover. The crossover operator uniformly exchanges parent sets for each variable.

[5] The algorithm is also referred as Metropolis-Hastings algorithm

Simple Mutate Randomly select two distinct variables i and j from the domain. If there exists an edge between them, then attempt to remove it. Otherwise, attempt to add an edge in either direction.

Parent Shift Mutate Randomly select an edge, randomly set the end of the edge to another variable.

Child Shift Mutate Randomly select an edge, randomly set the start of the edge to another variable.

Repair Illegal structures may be generated from above operators. Repair operator try to locate and break cycles in the structure, until a directed acyclic graph is got.

During the process of evolution, *Roulette wheel* selection strategy is adopted to ensure that better structures have a higher probability to be selected. Evolution proceeds until a given termination criterion is satisfied (In the experiments of this paper the evolution process stops when the number of MML calculations reaches a pre-set limit).

3.4 Parallel MCMC (PMCMC)

In 1998, Holmes and Mallick suggested to exchange information between different samplers as a way to improve mixing in MCMC samplers [4]. They demonstrated this method on a parameter training problem for a neural network, and a knot selection problem. They found that their algorithm can propose large changes without sacrificing acceptance probabilities.

The parallel MCMC method uses a population of samplers to estimate features of the target distribution $p(M|D)$ in an attempt to select a proposal distribution as close as possible to the target distribution. In this algorithm, candidate structures are not only generated by those operators as defined in 3.2, but also generated by a crossover operator as defined in 3.3. However, instead of using the candidate directly as in a standard Genetic Algorithm, PMCMC uses the formula 13 to either accept the candidate or remain unchanged.

4 Empirical Results and Analysis

In this section, we compare three encoding schemes and evaluate four different search strategies. Eight *Linear Causal Models* are used in our experiments: *Fiji, Evans, Blau, Rodgers&Maranto, case9, case10, case12* and *case15*, which have 4, 5, 6, 7, 9, 10, 12 and 15 variables respectively.

4.1 Comparison on Encoding Schemes

In order to compare the computational cost of these encoding schemes, we incorporate them with a Hill-Climbing search strategy. The CPU time cost of search process using different encoding schemes are compared in Table 1. From this table, we can see that scheme 1 is faster than scheme 2, this coincides with previous experimental reported in [10], and the encoding scheme 3 is the fastest

Table 1. Comparison of Time Cost of Discovery using different Encoding Schemes

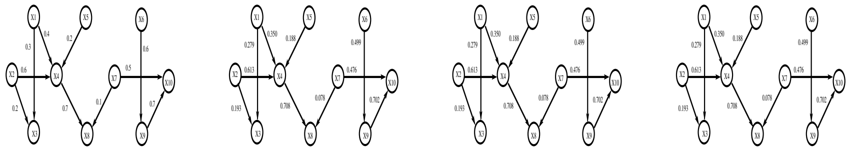
Data Set	Scheme 1	Scheme 2	Scheme 3
Fiji	0.84 seconds	0.92 seconds	0.96 seconds
Evans	2.29 seconds	3.01 seconds	2.25 seconds
Blau	5.18 seconds	6.35 seconds	3.42 seconds
Case9	19.23 seconds	23.19 seconds	16.32 seconds
Case10	59.97 seconds	71.19 seconds	20.10 seconds
Case12	126.20 seconds	181.61 seconds	36.20 seconds
Case15			265.50 seconds

one among three different encoding schemes. As the problem of computational cost, the search process using encoding scheme 1 or 2 can not give a result for data set (*case15*) which has more than 12 variables within reasonable time, however, the search process using encoding scheme 3 is capable of discovering more complicated models with larger number of variables.

Although the encoding costs using different schemes are different, the final discovery results are very close for data sets we tested. As an example, we give results on data set *Case10* and *Case12* by Hill-climbing based on different encoding schemes in Fig. 1 and 2. From which, we can see that all these search process returns the same results. This indicates that encoding scheme 3 can improve the efficiency of the discovery process while preserving the discovery accuracy.

4.2 Test Results on Search Strategies

To evaluate the performance of different search strategies, a common representation of the causal structure is used with four different search strategies. All the message lengths of causal structures here are calculated using scheme 3 as described in Sec. 2.



(a) Original (b) Scheme 1 (c) Scheme 2 (d) Scheme 3

Fig. 1. Comparison of Discovery Result on *Case10* using different Schemes

Table 2. Minimum Message Length Comparison

	Fiji	*Evans*	*Blau*	*Rodgers*	*Case9*	*Case10*	*Case12*	*Case15*
Original	5495.3	5470.3	7348.2	9346.2		3302.4	4462.8	17355.7
HC	5489.0	5466.7	7352.3	9352.6	10208.1	3302.4	4462.8	17371.0
MCMC	5488.6	5462.3	7348.2	9344.3	10208.2	3305.0	4470.2	17355.7
GA	5489.0	5466.7	7348.1	9345.2	10217.1	3302.4	4468.4	17482.4
PMCMC	5489.6	5466.7	7351.5	9344.3	10217.1	3302.4	4462.8	17355.7

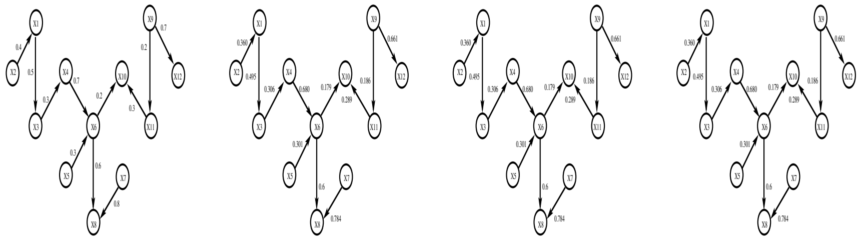
Considering the fact that the most computationally intensive part in search process is the calculation of MML cost, we use the times of calculating the MML cost as the basis of the termination rule for all the algorithms. For those models with less than 8 nodes, the search proceeds until the times of calculating MML reaches 2000, for those network with 9 or more variables, the search proceeds until the calling times reaches 5000.

For the genetic algorithm and the parallel MCMC algorithm, we run a set of exploratory experiments to find a proper set of parameters for them. In the following experiments, population size is set to 12, crossover probability is set to 0.5, and mutation probability is 0.3.

Table 2 illustrates the message length of eight original models and the MML length of the corresponding models discovered by the HC, MCMC, GA and PMCMC search strategies.

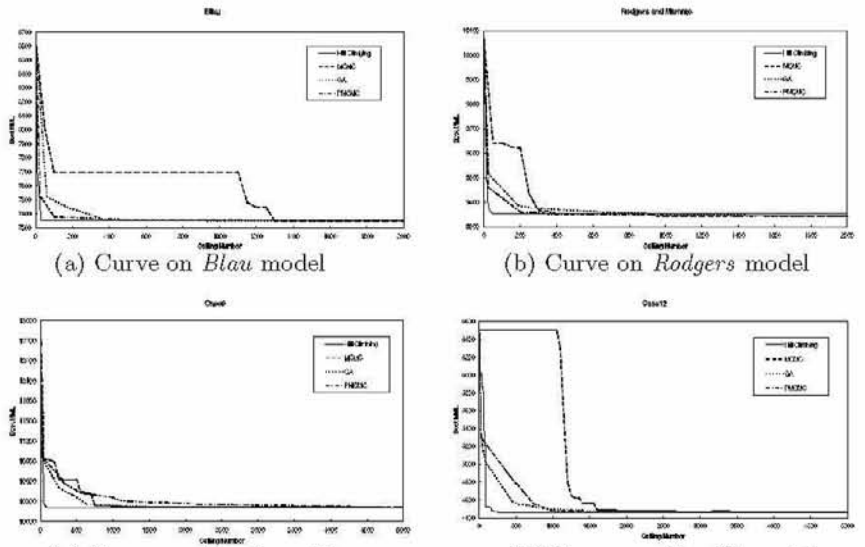
It should be noted that MML-based procedure is derived from asymptotic approximations, thus for models with few variables (like *Fiji* and *Evans*), minimizing MML doesn't coincide with the original model. For models with 9 or more variables, MML-based procedure works well.

From Table 2, we can see that Hill-climbing search can discover 3 original models from 8, MCMC and PMCMC search strategies can discover 4 original models from 8, and GA can find 2 original models out of 8. Although in some



(a) Original (b) Scheme 1 (c) Scheme 2 (d) Scheme 3

Fig. 2. Comparison of Discovery Result on *Case12* using different Schemes



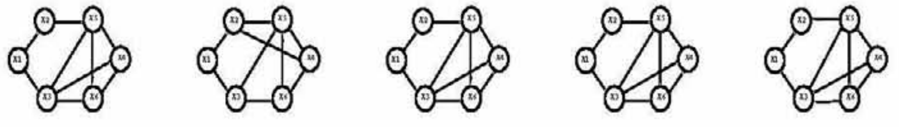
(a) Curve on *Blau* model

(b) Curve on *Rodgers* model

(a) Curve on *Case9* model

(b) Curve on *Case12* model

Fig. 3. Search Curve on Test Models



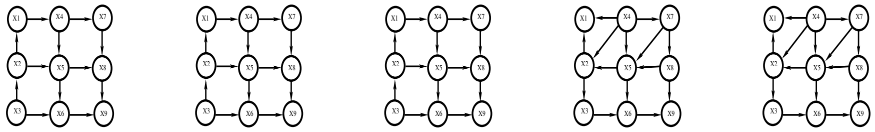
(a) Original (b) HC (c) MCMC (d) GA (e) PMCMC

Fig. 4. Original Model and Search results on *Blau* model

cases the algorithms do not find original models, all these algorithms can approximate the original one accurately.

Figure 3 illustrate the convergence speed of the four search strategies in discovering causal models from four different data sets. In these figures, X-axis is the number of MML calculation so far, and Y-axis is the best MML cost found. Although all different search strategies converge towards the original model, we can see that the Hill-climbing converge faster than the other three.

In theory, the main drawback of Hill-climbing method is that its greedy search nature determines that this method can be very easily stuck in local optima. More theoretic-robust methods like MCMC, and GA were proposed as means to overcome this problem. However, from our experiments, we can see that within reasonable number of calculations, MCMC and GA seem to have no apparent advantages over Hill-climbing. As for Parallel MCMC, although Holmes' work



(a) Original (b) HC (c) MCMC (d) GA (e) PMCMC

Fig. 5. Original Model and Search results on *Case9* model

showed that it can converge faster than standard MCMC, this method can be viewed as a Genetic Algorithm using a different selection strategy.

So in order to overcome problem of *the curse of dimension*, standard GA or MCMC don't seem to be a potential direction, at least for causal discovery from complete data set.

5 Conclusions

To improve the efficiency of discovery algorithm, this paper examined three different encoding schemes for describing the structure of a *Linear Causal Model*, and compared four different search strategies. Our empirical results indicated that the encoding scheme 3 is an improvement over our previous work in terms of learning efficiency while preserving the discovery accuracy.

This paper also reported the comparison results of four search strategies for the discovery of causal models from the model space. The experimental results revealed that more sophisticated search strategies seem to have no apparent advantages over Hill-climbing, which works very well for the task of Causal Discovery from complete data sets. This result keeps consistent with what we have found previously.

References

[1] K. A. Bollen. *Structural Equations with Latent Variables.* Wiley, New York, 1989. 48

[2] Honghua Dai, Kevin Korb, Chris Wallace, and Xindong Wu. A study of causal discovery with small samples and weak links. In *Proceedings of the 15th International Joint Conference On Artificial Intelligence* **IJCAI'97**, pages 1304–1309. Morgan Kaufmann Publishers, Inc., 1997. 48

[3] Honghua Dai and Gang Li. An improved approach for the discovery of causal models via MML. In *Proceedings of The 6th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD-2002)*, pages 304–315, Taiwan, 2002. 51

[4] C. Holmes and B. Mallick. Parallel markov chain monte carlo sampling: an evolutionary based approach. Technical report, Imperial College, London, 1998. 51, 54

[5] Michael I. Jordan. *Learning in Graphical Models*. MIT Press, Cambridge, MA, 1 edition, 1998. 48, 51

[6] P. Larrañaga, M. Poza, Y. Yurramendi, R. H. Murga, and C. M. H. Kuijpers. Structure learning of bayesian networks by genetic algorithms: A performance analysis of control parameters. *IEEE Transactions on Pattern analysis and Machine Intelligence*, 18(9):912–926, 1996. 51

[7] D. Madigan and J. York. Bayesian graphical models for discrete data. Technical Report TR-259, University of Washington Department of Statistics, Seattle, WA, 1993. 51

[8] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann, San Mateo, California, 1988. 48

[9] Chris Wallace and D. M. Boulton. An information measure for classification. *Computer Journal*, 11(2):185–194, 1968. 48

[10] Chris Wallace, Kevin B. Korb, and Honghua Dai. Causal discovery via MML. In *Proceedings of the 13th International Conference on Machine learning (ICML'96)*, pages 516–524, San Francisco, 1996. Morgan Kauffmann Publishers. 48, 49, 50, 51, 54

[11] W. L. Wong, W. Lam, and K. S. Leung. Using evolutionary computation and minimum description length principle for data mining of probabilistic knowledge. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(2):174–178, 1999. 51

[12] Sewall Wright. The method of path coefficients. *Annals of Mathematical Statistics*, 5:161–215, 1934. 48