Efficient Model Determination for Discrete Graphical Models

Paolo Giudici, Peter Green, Claudia Tarantola *

Abstract

We present a novel methodology for bayesian model determination in discrete decomposable graphical models. We assign, for each given graph, a Hyper Dirichlet distribution on the matrix of cell probabilities. To ensure compatibility across models such prior distributions are obtained by marginalisation from the prior conditional on the complete graph. This leads to a prior distribution automatically satisfying the hyperconsistency criterion. Our contribution is twofold. On one hand we improve an existing methodology, the MC^3 algorithm by Madigan and York (1995). On the other hand we introduce an original methodology based on the use of the Reversible jump sampler by Green (1995) and Giudici and Green (1999). Legal movement, that is leading to a decomposable graph, are identified making use of the junction tree representation of the considered graph.

Keywords: Bayesian model selection; Contingency table; Dirichlet distribution; Hyper Markov distribution; Junction tree; Markov Chain Monte Carlo.

*Paolo Giudici is Assistant Professor, Dipartimento di Economia Politica e Metodi Quantitativi, Università di Pavia Via San Felice 5,I-27100, Italy (E-mail:giudici@unipv.it). Peter Green is Professor,Department of Mathematics, University of Bristol, Bristol BS8 1TW, UK (E-mail: P.J.Green@bristol.ac.uk). Claudia Tarantola is a postdoc student, Department of Statistics, Athens University of Economics and Business ,76,Patission str GR-10434 Athens, Greece (E-mail: claudia@verdea.stats.aueb.gr).

1 Introduction

A graphical model (see, for instance, Lauritzen, 1996), is a family of probability distributions incorporating the conditional independence assumptions represented by a graph. It is constructed by specifying *local dependencies* of each node of the graph in terms of its immediate neighbours. It is then possible to work locally, obtaining better results in terms of statistical inference and computational efficiency.

Our motivation here is to develop efficient procedures for Bayesian model determination in discrete graphical models, employed for the analysis of contingency tables. For high-dimensional contingency tables the set of plausible models is large, and a full comparison of all the posterior probabilities associated to the competing models becomes infeasible. In fact the number of graphical structures to examine increases more than exponentially with the number of vertices.

Various solutions to this problem have been proposed, the one we suggest is based on the application of MCMC techniques. This possibility has already been exploited. Madigan and York (1995), for example, introduce an MCMC sampler, called *Markov chain Monte Carlo composition* (MC^3 hereafter), for the analysis of decomposable models. They construct a Metropolis Hastings sampler that permits to explore the space of all decomposable models. Alternatively, Dellaportas and Foster (1999) have developed a MCMC sampler for model choice in loglinear models which include discrete graphical models.

In this paper we shall present two different MCMC samplers for the analysis of decomposable discrete graphical models, which are fully based on local computations and, therefore, efficient. The first one is a revised version of the MC^3 algorithm by Madigan and York (1995). It differs from the original version mainly because it incorporates a local condition for checking decomposability. Furthermore, we shall propose an extension which allows for a hierarchical prior on the cell counts. This methodology is suitable only for quantitative learning.

The second sampler is based on the Reversible jump RJMCMC by Green (1995) and

can be used both for quantitative and qualitative learning. Our methodology parallels that presented in Giudici and Green (1999) for the analysis of decomposable gaussian models. As in the gaussian case, at each step of the algorithm we update not only the graphical structure (as in MC^3), but also the associated parameter vector. Essentially, in the gaussian case, pairwise conditional independence is dictated by the absence of a single parameter, whereas in the discrete case this corresponds in general to non linear constraints on the cell probabilities. Furthermore, in the continuous case the parameter space is polynomial in the number of variables whereas in the discrete case is exponential. This leads to substantial differences in the data structure.

Section 2 contains some preliminary background on the Bayesian analysis of discrete graphical models. Section 3 contains our proposed MCMC model determination methods. Finally, Section 4 contains a comparison of the performances, with reference to two data-sets: the well-known "Woman and mathematics" dataset and a dataset presented in Fahrmeir and Hamerle (1994) for the analysis of the credit scoring problem.

2 Bayesian analysis of discrete graphical models

2.1 Discrete graphical models

In this section we briefly review the literature on discrete graphical models relevant for our work, making use extensively of the terminology and notation from Dawid & Lauritzen (1993) (DL hereafter), and Lauritzen (1996).

A graph is a mathematical object consisting of two sets, a finite set of *vertices*, V, and a set of *edges*, E, of pairs of elements taken from V, $g = (V, E)$. We will consider only undirected graphs, such that if $(a, b) \in E$ then $(b, a) \in E$. A graph is *complete* if all vertices are joined by an edge. A subset of vertices is complete if it induces a complete subgraph. A complete subset that is *not* contained within another complete subset is called a *clique*. An ordering of the cliques of an undirected graph, $\mathcal{C} = (C_1, \ldots, C_n)$, is said to be *perfect* if the vertices of each clique C_i also contained in previous cliques

$C_1, \ldots, C_{i-1}$ are all members of *one* previous clique. The set $\mathcal{S} = \{S_2, \ldots, S_n\}$ identifies the *separators*. If an undirected graph admits a perfect ordering is said to be *decomposable*. In the following we refer to subsets of the form $A \in \mathcal{A}(g) = \mathcal{C} \cup \mathcal{S}$ where $\mathcal{C}$ is the collection of cliques of g and $\mathcal{S}$ a system of separators in a perfect ordering of such cliques.

For computational aspects it is useful to organize the cliques of the considered graph through a *junction tree*. A junction tree is a graph with vertex set corresponding to the set $\mathcal{C}$ of cliques of the graph g examined. It must satisfy the *running intersection properties*, that is for any two cliques $C_i, C_j \in \mathcal{C}$, and any C' on the unique *path* between them, $C_i \cap C_j \subset C'$. A collection of disjoint junction trees forms a *junction forest*. An example of a graph, and its associated junction tree, is given in Figure 1.

Figure 1 about here

A *graphical model* is a family of probability distributions *Markov* with respect to a graph g; for brevity $P \in M(g)$. That is, the probability distributions considered must satisfy the conditional independence restrictions inherent in g, but are otherwise arbitrary.

Discrete graphical models describe the relation between a set of $k = |V|$ *discrete* random variables $X_V = (X_v)_{v \in V}$, each of which takes values in $\mathcal{I}_v$, with $\mathcal{I} = \times_{v \in V} \mathcal{I}_v$ the complete table of counts.

A graphical model is then characterised by the constraints imposed on the cell probabilities $\theta \in \Theta$ by the conditional independences embodied in a graph g. In order to underline this dependence the parameter space will be indicated as Θ_g, where $\Theta_g = \{\theta : \theta \in M(g)\}$.

When the graph is decomposable an arbitrary distribution $\theta_g \in \Theta_g$ is determined by the marginal probability tables $\theta_A = (\theta_A(i_A))_{i_A \in \mathcal{I}_A}$, with elements $\theta_A(i_A) = pr\{X_A = i_A\}$ as in the following:

$$\theta_g(i) = \frac{\prod_{A \in \mathcal{C}} \theta_A(i_A)}{\prod_{A \in \mathcal{S}} \theta_A(i_A)}. \tag{1}$$

We remark that the symbol i_A indicates a cell of the marginal contingency table corresponding to the variables in A, a subset of V corresponding to a clique or to a separator.

In this case, if we indicate with $x_V^{(n)}$ an observed realisation of a random sample of n observations $X_V^{(n)} = (X_V^1, \ldots, X_V^n)$ from the distribution $\theta_g \in \Theta_g$, the likelihood $L(\theta_g) = pr\{X_V^{(n} = x_V^{(n)} | \theta, g\}$ can be written as follows:

$$L(\theta_g) = \frac{\prod_{A \in \mathcal{C}(\mathcal{G})} pr\{X_A^{(n)} = x_A^{(n)} | \theta_A\}}{\prod_{A \in \mathcal{S}(\mathcal{G})} pr\{X_A^{(n)} = x_A^{(n)} | \theta_A\}} = \frac{\prod_{A \in \mathcal{C}(\mathcal{G})} \prod_{i_A \in \mathcal{I}_A} (\theta_A(i_A))^{n_A(i_A)}}{\prod_{A \in \mathcal{S}(\mathcal{G})} \prod_{i_A \in \mathcal{I}_A} (\theta_A(i_A))^{n_A(i_A)}}, \tag{2}$$

were $n_A(i_A) = \sum_{j:j_A = i_A} n(j)$ is the observed count in the cell i_A of the marginal table of X_A.

When g is *not* decomposable, the factorisation in (1) is no longer valid and, consequently, the likelihood cannot be factorised into local pieces.

In this paper we consider the case in which the variables examined are dichotomous. Each element of the vector $X_V = (X_v)_{v \in V}$ is a random variable taking values in the set $\{0, 1\}$ and the vector X_V takes values in the Cartesian product $\{0, 1\}^{|V|}$ of the set $\{0, 1\}$ with itself.

2.2 Hyper dirichlet prior distributions for Bayesian learning

In the literature on graphical models we can distinguish two main aspects of inference, *quantitative* and *qualitative* learning. Quantitative learning means that the information available is used to estimate the unknown parameters θ_g. On the other hand, structural learning has the objective of establishing which graphs, and thus which graphical models are best supported by the data and the prior information available. In this paper we consider both problems.

As a prior distribution on Θ_g we consider the *Hyper-Markov laws* introduced by DL for the analysis of decomposable models. In particular, we consider the *hyper Dirichlet* laws, that can be used for the Bayesian analysis of discrete graphical models, see for instance Madigan and York (1995) and Giudici and Tarantola (1996).

Before proceeding, we recall some important properties of the Dirichlet distribution. Let $\lambda = (\lambda(i), i \in \mathcal{I})$ be a vector of positive constants, and let $A \subseteq V$ and $B = V \setminus A$ be a partition of V. If $\mathcal{L}(\theta) = \mathcal{D}(\lambda)$ then:

(i) $\mathcal{L}(\theta_A) = \mathcal{D}(\lambda_A)$;

(ii) $\theta_{B|A}(\cdot|i_A)$ are all independent and distributed as $\mathcal{D}(\lambda_{B|A}(\cdot|i_A))$;

(iii) $\theta_A \perp\!\!\!\perp \theta_{B|A}$,

where $\lambda_A(i_A) = \sum_{j:j_A=i_A} \lambda(j)$ and $\lambda_{B|A}(i_B|i_A) = \lambda(i)$.

In order to construct a hyper Dirichlet distribution, we must satisfy the condition of consistency of the matrices of cell probabilities, that is, for any two cliques C and D:

$$\theta_{C\cap D}(i_{C\cap D}) = \sum_{j_C:j_{C\cap D}=i_{C\cap D}} \theta_C(j_C) = \sum_{j_D:j_{C\cap D}=i_{C\cap D}} \theta_D(j_D). \tag{3}$$

A hyper Dirichlet prior distribution on $\theta \in M(g)$ is then constructed by assigning to the probabilities θ_C of each clique a Dirichlet distribution $D(\lambda_C)$ with density:

$$\pi(\theta_C|\lambda_C) \propto \prod_{i_C\in\mathcal{I}_C} \theta_C(i_C)^{\lambda_C(i_C)-1}$$

on the set where $\sum_{i_C} \theta_C(i_C) = 1$ and $\theta_C(i_C) > 0$.

Furthermore, the hyperparameters λ_C are constrained in order to satisfy the hyper-consistency condition of the corresponding distribution $D(\lambda_C)$, that is:

$$\lambda_{C\cap D}(i_{C\cap D}) = \sum_{j_C:j_{C\cap D}=i_{C\cap D}} \lambda_C(j_C) = \sum_{j_D:j_{C\cap D}=i_{C\cap D}} \lambda_D(j_D). \tag{4}$$

The constraints in (4) are automatically satisfied by assigning a Dirichlet distribution on the parameter θ corresponding to the complete graph, and obtaining the laws on the cliques by *marginalisation.* Alternatively, as suggested by DL one can take:

$$\lambda(i) = \frac{\prod_{C\in\mathcal{C}(\mathcal{G})} \lambda_C(i_C)}{\prod_{S\in\mathcal{S}(\mathcal{G})} \lambda_S(i_S)}.$$

For a comparison between the two approaches, with reference to smoothing effects on the cell counts, see Giudici (1998).

Dl show that, given any such hyperconsistent collection $\lambda_C = (\lambda_C)_{C\in\mathcal{C}}$ there exists a unique law for θ called *hyper Dirichlet*, denoted by $\mathcal{HD}_g(\lambda)$ which is hyper Markov over $M(g)$ and has $\mathcal{L}(\theta_C) = D(\lambda_C)$ has its clique marginals. Furthermore, this distribution is

strong hyper Markov. This implies that, if we confine our attention to θ_C with prior law $\mathcal{D}(\lambda_C)$, and the data n_C from the marginal table corresponding to clique C, the posterior law for θ_C given n_C will be $\mathcal{D}(\lambda_C + n_C)$. If the prior law is $\mathcal{HD}_g(\lambda)$ the posterior law will be $\mathcal{HD}_g(\lambda + n)$.

Regarding model comparison, DL give a closed form expression for the marginal likelihood of g, $p(x_V^{(n)}|g)$:

$$p(x_V^{(n)}|g) = \frac{\prod_{C \in \mathcal{C}(\mathcal{G})} p(x_C^{(n)})}{\prod_{S \in \mathcal{S}(\mathcal{G})} p(x_S^{(n)})}, \tag{5}$$

where, given a complete subset of vertices $A \in \mathcal{A}(g)$, it turns out that:

$$p_A(x_A^{(n)}) = \frac{\Gamma(\lambda)}{\Gamma(\lambda + n)} \prod_{i_A \in \mathcal{I}_A} \left(\frac{\Gamma(\lambda_A(i_A) + n_A(i_A))}{\Gamma(\lambda_A(i_A))} \right).$$

Note that we have not yet specified the *density* of the Hyper Dirichlet distribution. In fact, the previous result show that it is not needed for structural learning, at least when the hyperparameters are not random. However, in more general problems, such a density may be necessary. We shall thus present its expression, as derived by Madigan and York (1997).

Let $\mathcal{C} = (C_1, \ldots, C_k)$ be a perfect ordering of the cliques of the examined graph. Consider a clique C_1 and assign on θ_{C_1} a Dirichlet distribution with parameter λ_{C_1}, whose density is (with respect to the Lebesgue measure):

$$f(\theta_{C_1}) = \frac{\Gamma(\sum_{i_{C_1} \in \mathcal{I}_{C_1}} \lambda_{C_1}(i_{C_1}))}{\prod_{i_{C_1} \in \mathcal{I}_{C_1}} \Gamma(\lambda_{C_1}(i_{C_1}))} \prod_{i_{C_1} \in \mathcal{I}_{C_1}} \theta_{C_1}(i_{C_1})^{\lambda_{C_1}(i_{C_1})-1}$$

The distribution of a generic clique C_j is obtained conditioning on $\theta_{C_1}, \ldots, \theta_{C_{j-1}}$ for $j > 1$. We note that conditions (4) and (3) need to be satisfied. Since θ is *strong hyper Markov* the distribution θ_{C_j}, given all previous cliques depends only upon θ_{S_j}. It results that the density of such conditional distribution (with respect to the Lebesgue measure) is:

$$f(\theta_{C_j}|\theta_{S_j}) = \frac{\prod_{i_{S_j}\in\mathcal{I}_{S_j}}\Gamma(\lambda_{S_j}(i_{S_j}))\prod_{i_{C_j}\in\mathcal{I}_{C_j}}\theta_{C_j}(i_{C_j})^{\lambda_{C_j}(i_{C_j})-1}}{\prod_{i_{C_j}\in\mathcal{I}_{C_j}}\Gamma(\lambda_{C_j}(i_{C_j}))\prod_{i_{S_j}\in\mathcal{I}_{S_j}}\theta_{S_j}(i_{S_j})^{\lambda_{S_j}(i_{S_j})-1}},$$

for θ_{C_i} satisfying (3).

Finally, putting together the previous two expressions we obtain:

$$\begin{aligned} f(\theta) &= f(\theta_{C_1})\prod_{i=2}^{k} f(\theta_{C_i}|\theta_{C_1},\ldots,\theta_{C_{i-1}}) \\ &= \frac{\prod_{j=1}^{k}\prod_{i_{C_j}\in\mathcal{I}_{C_j}}\theta_{C_j}(i_{C_j})^{\lambda_{C_j}(i_{C_j})-1}}{\Psi(\lambda)\prod_{j=2}^{k}\prod_{i_{S_j}\in\mathcal{I}_{S_j}}\theta_{S_j}(i_{S_j})^{\lambda_{S_j}(i_{S_j})-1}}, \end{aligned} \tag{6}$$

where:

$$\Psi(\lambda) = \frac{\prod_{j=1}^{k}\prod_{i_{C_j}\in\mathcal{I}_{C_j}}\Gamma(\lambda_{C_j}(i))}{\Gamma(\sum_{i_{C_1}\in\mathcal{I}_{C_1}}\lambda_{C_1}(i_{C_1}))\prod_{j=2}^{k}\prod_{i_{S_j}\in\mathcal{I}_{S_j}}\Gamma(\lambda_{S_j}(i))}.$$

Remark. From the previous construction, it follows that, even for a fixed graph, a very large number of hyperparameters is to be specified. Recall that $\lambda_C(i)$ indicates a collection of positive constants related to the *a priori* expected counts in each cell of the marginal contingency table of the variables in C. Furthermore, it is necessary for them to be hyperconsistent. Finally, as argued by DL among others, it is highly desirable that the priors be compatible across models, thus further complicating prior specification.

In the following we shall assume that the Dirichlet distributions on each single clique are obtained by marginalisation from a *unique* distribution on the complete graph. Since the hyperparameters can be interpreted as hypothetical marginal data counts, this notion of compatibility, which is the same as that in DL, is equivalent to requiring that each model has the same amount of hypothetical data.

More precisely in the following we shall indicate with $\lambda_0 = \sum_{i\in\mathcal{I}}\lambda(i)$ the prior precision of the complete graph. Notice that, since the prior distributions on each single cliques are obtained by marginalisation from the prior distribution on the complete graph, λ_0 can be equivalently obtained as $\sum_{i_C\in\mathcal{I}_C}\lambda_C(i_C)$, for any generic clique C.

Regarding the value of the hyperparameters, one possibility is to assign $\lambda(i) = 1/2$, following the Jeffreys prior for multinomial sampling, or $\lambda(i) = 1$, following a uniform

distribution. For a discussion on the choice of the hyperparameters see, for instance, Dellaportas and Foster (1999).

Another possible formulation is to consider a more flexible, and easier to specify, hierarchical prior, for instance letting λ_0 to be a random variable, to be assigned a prior distribution.

3 MCMC discrete graphical model determination

In this section we shall present two different methodologies for Bayesian model determination for decomposable discrete graphical models. Both methodologies are based on the application of MCMC methods.

The first methodology presented extends the MC^3 algorithm by Madigan and York (1995) by allowing for a *hierarchical* hyper Dirichlet on the cell probabilities. We also improve computational efficiency of MC^3, replacing the decomposability test of Madigan and York (1995) with the recent proposal by Giudici and Green (1998) (GG for brevity).

The second methodology is based on the RJMCMC algorithm by Green (1995): at each step of the procedure we update the model and the corresponding vector of parameters. This methodology is quite general, and can deal with any type of priors on the parameters, such as hierarchical. Furthermore, it allows to draw posterior inferences on any quantity of interest, whereas this is not generally possible for MC^3.

For comparison purposes, we shall consider the same two classes of prior distributions for both cases, namely, a *hyper Dirichlet* prior on the vector of cell parameters. Concerning the model space, for simplicity, and without loss of generality, we consider a discrete uniform prior distribution over the set of all decomposable graphical models.

3.1 Identification of legal moves

As stated by Frydenberg and Lauritzen (1989), (FL hereafter), the space of all decomposable graphs can be traversed by adding and deleting single edges at a time. Such

changes are convenient for MCMC implementation (in terms of algebraic tractability and statistical efficiency) and will be used as basic steps for our sampling algorithms.

Given a graph g we propose to consider a new graphical structure $g^{'}$, obtained by adding/deleting one single edge. Naturally, we can decide not to change the current graph.

At each step, we can then choose between three different move types:

1) remain with the current model;

2) create a new model $g^{'}$ via the addition of one more edge;

3) create a new model $g^{'}$ via the removal of an existing edge.

Note that not all moves will be available at each step, for example we cannot add an edge to the complete graph (graph with all edges present), nor remove one from the null graph (graph with no edges present).

We shall only consider moves, called *legal*, that lead to a decomposable graph. The problem is how to characterise such moves.

For *legal deletion* we can use a result in FL that states that an edge can be removed iff it is contained in *only one* clique. On the other hand, for *legal additions*, we now introduce an efficient condition, recently proposed by GG, that permits identifying legal movements in *advance*, that is before doing the move.

We first remark that by adding/deleting an edge we modify only a local part of the junction tree, as in Figure (2).

Figure 2 about here

For this reason, GG propose to identify legal addings by a condition that permits checking decomposability considering only the *section* of the junction forest represented in Figure (2).

Theorem (GG). Let $g = (V, E)$ be an undirected decomposable graph in which vertices a and b are not adjacent, and let g' denote the graph modified by the addition of edge (a, b). The new graph g' is decomposable if and only if either:

(i) $[a] \neq [b]$, or

(ii) $[a] = [b]$ and there exist $R, T \subset V$ such that $a \cup R$ and $b \cup T$ are cliques, and $S = R \cap T$ is a separator on the path between $a \cup R$ and $b \cup T$ in a junction forest representation of the graph g

Where with $[v]$ we indicate the set of all vertices that are connected to v.

The above Theorem provides a simple and local condition for rejecting illegal addition in advance. Often, a and b are adjacent so that the search will be very fast. Furthermore, the procedure proposed by GG constructs the new junction forest so that the cliques are ready for use in probability calculations.

An alternative possibility is to reject illegal moves *a posteriori* by running an appropriate algorithm (such as the Maximum Cardinality Search, MCS) after each graphical update, to check if the proposed graph g' is decomposable. This is the solution implemented in the MC^3 algorithm by Madigan and York (1995). However rejecting moves *a posteriori* can be inefficient when the graphical structure is complex. GG provide empirical evidence to support this claim.

We now present the GG procedure in algorithmic form.

Adding

1) Starting from a clique containing a, search through the current junction tree containing a for the first clique containing b (say $b \cup T$). If none exists, the graph is disconnected: go to 4).

2) Starting from $b \cup T$, go backwards through the junction tree along the path found in 1), until the first clique containing a is found (call it $a \cup R$). Check if $R \cap T \neq \emptyset$ and $R \cap T$ is a separator on the path. If not, reject the move (the proposed g' is not decomposable): return.

3) If $a \cup R$, $b \cup T$ are not adjacent, permute the junction tree until they are.

4) Decide whether to accept the proposed move.

5) If the move is accepted, update the graph and the junction forest.

6) Return.

Deleting

1) Starting from a clique containing a search through the current junction tree while the cliques contain a until all cliques containing b are found.

2) If none is found, there is an error, a and b are not adjacent. If only one is found go to 3). Otherwise, if more than one are found, reject the move (the proposed $g^{'}$ is not decomposable): return.

3) Decide whether to accept the proposed move.

4) If the move is accepted, update the graph and the junction forest.

5) Return.

We finally remark that, according to Figure 2, in our algorithm we treat separately four particular cases: a) $R = T = R \cap T$; b) $R = R \cap T \neq T$; c) $R \neq R \cap T = T$; d) $R \cap T = \emptyset$.

3.2 A new version of MC^3

We shall first recall the original version of the MC^3 algorithm. It permits constructing a Markov Chain having $p(g|x)$ as its target distribution.

Given a graph g, indicate with $nbd(g)$ its neighbourhood consisting of g itself and the set of graphs with either one more or one fewer edge than g. Suppose that from g the only possible move is to a graph $g^{'}$ belonging to its neighbourhood. Each $g^{'}$ can be chosen with the same probability.

The transition probability $q(g, g^{'})$ is then equal to 0 for all $g^{'} \notin nbd(g)$ and constant for all $g^{'} \in nbd(g)$.

Suppose that from a graph g we propose to move to graph $g^{'}$ obtained by adding one more edge between vertices a and b.

The proposed move is accepted with probability equal to:

$$\alpha = \min\{1, R_a\} \tag{7}$$

where:

$$R_a = \frac{\#(nbd(g))p(g^{'}|x)}{\#(nbd(g^{'}))p(g|x)} \tag{8}$$

Since $p(g|x) \propto p(x|g)p(g)$, (7) involves the data only through the Bayes factor $p(x|g^{'})/p(x|g)$. We can then apply the results presented in section 2 and calculate the previous ratio by local computations, that is:

$$R_a = \frac{p(x|g^{'})}{p(x|g)} = \frac{p_S(x_S^{(n)})p_{abS}(x_{abS}^{(n)})}{p_{aS}(x_{ab}^{(n)})p_{bS}(x_{bS}^{(n)})}. \tag{9}$$

Notice that calculations involve only the local part of the junction tree represented in Figure (2).

We propose to modify the algorithm described above in two directions leading to a nonhierarchical and to a hierarchical version.

A nonhierarchical model

The main difference with respect to the original formulation is that, in order to check legal addings, we use the condition proposed by GG instead of MCS.

Furthermore, in the new version the proposal ratio is determined differently, that is contrasting the probability of adding and deleting an edge from the considered graph. This leads to a different probability of choosing between adding or deleting an edge. However, conditionally on this decision, any candidate edge has the same probability of being changed in both cases.

More precisely, given a graph with n vertices, the probability of adding an edge, A_g, can be obtained as the product of the probability of adding times the probability of choosing a particular edge between the ones eligible for addition, that is:

$$A_g = \left(\frac{\binom{n}{2} - E_g}{\binom{n}{2}}\right) \times \frac{1}{\binom{n}{2} - E_g} = \binom{n}{2}^{-1}, \tag{10}$$

where E_g is the number of edges present in the current graph g.

In a similar way we obtain the probability, D_g, of deleting an edge from the current graph. Since $A_g = D_g$ the proposal ratio is equal to 1.

The move is then accepted with probability equal to:

$$\alpha = \min\{1, R_a\}$$

where:

$$R_a = \frac{p(g^{'}|x)}{p(g|x)},$$

which can be calculated as in equation (9).

A hierarchical model

The essential difference with the previous case is that we now allow for a further level of hierarchy. For instance, we can let λ_0 (the total prior precision) become a random quantity, to be assigned a suitable prior distribution. As previously discussed, a hierarchical prior is, even when not strictly necessary, easier to specify *a priori*.

It seems reasonable to assume that λ_0 and g are independent. As a prior for λ_0 we assign a *Gamma* distribution, with mean f and variance fs, namely:

$$\pi(\lambda_0) \propto \lambda_0^{(f/s)-1} e^{-\lambda_0/s}$$

where $f > 0$ and $s > 0$ are positive constants appropriately chosen.

Another important advantage of this new setting occurs when we have *incomplete* data. An important example of this occurrence is given in Madigan and York (1997): one or more cell counts may not be available. Let $n' = (n(i), i \in \mathcal{I}'_z \subset \mathcal{I})$ indicate the missing cell counts. In this case we can let n' be a random vector, to be assigned an appropriate prior distribution, possibly according to the sampling scheme of the data. Structural learning can then proceed, conditionally on the sampled values of n'.

In general, let τ denote the extra random component considered. We propose an algorithm consisting of two stages:

1) We change the graphical structure adding/deleting only one edge at a time.

2) We update the random parameter τ

Concerning the first step, the proposed move is accepted with probability equal to:

$$\alpha = \min\{1, R_a\} \tag{11}$$

where

$$R_a = \frac{\pi(g^{'}, \tau|x)}{\pi(g, \tau|x)}.$$

As in the nonhierarchical model the proposal ratio is equal to 1. Since $\pi(g, \tau|x) \propto \pi(x|g, \tau)\pi(g, \tau)$, R_a simplifies to:

$$R_a = \frac{p(x|g^{'}, \tau)}{p(x|g, \tau)}.$$

Concerning the second step, the new value $\tau^{'}$ is sampled from a normal distribution, centered around the current value. More precisely the proposal is: $q(\tau^{'}|\tau) = N(\tau, \sigma_\tau^2)$, where σ_τ^2 is a spread parameter, to be appropriately chosen.

The move is then accepted with probability:

$$\alpha = \min\{1, R_a\}.$$

where:

$$R_a = \frac{p(x|g, \tau^{'})p(\tau^{'})}{p(x|g, \tau)p(\tau)}$$

as the proposal ratio is equal to 1, being the proposal symmetric.

One complete pass over these two moves will be called a sweep and is the basic step of our algorithm.

Clearly, when an edge is proposed for deletion the move is accepted with probability $\alpha = \min\{1, R_d\}$, where $R_d = 1/R_a$.

In the examples below we consider the case in which we update the total prior precision λ_0 following the procedure described above. The single $\lambda(i)^{'}$ is then obtained as $\lambda(i)^{'} = \lambda_0/|\mathcal{I}|$.

A more complex procedure could be adopted. For example one possibility is to update each single $\lambda(i)$ separately. However, the advantages of this do not seem to compensate

for the increased complexity of the sampler and the predicted extra computational effort, which discourage their implementation.

3.3 Reversible jump MCMC for discrete graphical models

A problem with the previous approaches is that they are designed for structural, but not for quantitative learning. For instance one could be interested in deriving posterior estimates, typically not available exactly (such as posterior odds ratios) .

Hence the necessity to develop a different methodology. The solution we propose is based on the application of the Reversible jump MCMC sampler (Green, 1995). At each step we move inside the space of models and of the corresponding parameters, that is we propose to move from (g, θ_g) to $(g^{'}, \theta_{g^{'}})$; in the following we shall indicate with y the pair (g, θ_g).

More precisely following GG we propose an algorithm consisting of two steps. At the first step we propose to modify the graphical structure adding or deleting only one edge. This move will be done in order to keep invariant the distribution on the cliques that are not involved in the change, and to assign a distribution on the new clique consistent with them. At the second step we update the matrix of cell probabilities given a graphical structure. Our procedure does not change the set of cliques and separators, but modifies the set of probabilities associated with them in a suitable way. We create a different distribution belonging however to the same Markov family; the new distribution and the old one incorporate the same set of conditional independences. If the model is hierarchical we had a further step in which we update the total prior precision.

In the following we shall describe in more details the procedure used.

Update g move. Let $g^{'}$ be a graph obtained from g adding one more edge between vertices a and b. By adding the edge (a, b) we create a new clique abS, in addition or substitution to the preexisting cliques as shown in Figure (2).

In order to obtain a Markov distribution with respect to the newly created graph $g^{'}$, the distribution corresponding to clique abS, θ_{abS}, must be consistent with the distribu-

tions on all the other cliques of the graph. This can be obtained imposing the consistency of the new matrix on the subsets aS and bS.

In order to obtain a Markov distribution with respect to the newly created graph $\mathcal{G}'$, the distribution corresponding to clique abS, θ_{abS}, must be consistent with the distributions on all the other cliques of the graph. This can be obtained imposing the consistency of the new matrix on the subsets aS and bS. In fact all the other cliques of the graph are by construction consistent on these subsets. Since we are working with dichotomous variables the number of free parameters is equal to $2^{|S|}$.The results can be easily extended to the case in which the considered variables are no more dichotomous.

We shall fix a configuration of the separator and work in terms of the conditional distribution of $ab|S = s$, $\theta_{ab|S=s}$ with assigned marginals $\theta_{a|S}$ and $\theta_{b|S}$. For each table we can fix only one value, say ϵ, that must be sampled from a suitable distribution. The corresponding value of θ_{abS} is then obtained multiplying the sampled valued by the marginal distribution of the separator.

We propose to sample the new value from a normal distribution centred on $(p_i + q_i)/2$ and to reject the new value if it does not belong to the interval $(\max(0, p_i + q_i - 1), \min(p_i, q_i))$.

The proposed move, adding and edge (a, b), is accepted with probability equal to $\alpha = \min\{1, R_a\}$, where:

$$\begin{aligned} R_a &= \frac{\pi(y')}{\pi(y)} \times \frac{r_m(y')}{r_m(y)q(z)} \times |J| \\ &= R_{post} \times R_{prop} \times |J|, \end{aligned} \tag{12}$$

where $|J|$ indicates the Jacobian of the transformation.

Since $\pi(g, \theta|x) \propto p(x|g, \theta)\pi(\theta|g)p(g)$ the posterior ratio $\pi(y')/\pi(y)$ simplifies as:

$$R_{post} = \frac{p(x|g', \theta)\pi(\theta|g')}{p(x|g, \theta)\pi(\theta|g)}.$$

Applying equation (6) we notice that calculations can be made locally considering only the four complete subsets represented in Figure (2). That is:

$$\frac{\pi(\theta|g^{'})}{\pi(\theta|g)} = \frac{f(\theta_{abS}|\theta_{aS})f(\theta_{bT}|\theta_{bS})}{f(\theta_{bT}|\theta_{S})}.$$

Consider now the proposal ratio:

$$R_{post} = \frac{r_m(y')}{r_m(y)q(z)}.$$

It can be decomposed in two different terms: $R_r = r_m(y')/r_m(y)$, obtained contrasting the probability of adding and deleting one edge from the considered graph, and $q(z)^{-1}$, the probability distribution of the *auxiliary* variable considered in order to satisfy the dimension matching problem.

From (10) it follows that $R_r = 1$. Concerning $q(z)$ it is obtained by the product of $2^{|S|}$ independent normal distributions. Finally the Jacobian of the considered transformation is equal to $\prod_{i_s \in \mathcal{I}_s} \theta_S(i_S)$.

When (a, b) is proposed for deletion we leave θ_{abS} unspecified and the acceptance ratio of the proposed move is obtained as $R_d = 1/R_a$.

Update θ.

This move does not involve a change in dimensionality, and the acceptance ratio will be calculated applying the standard Metropolis Hastings algorithm. At each step we update only one single clique; we choose it randomly and we perturb each elements of its vector of cell probabilities in a suitable way.

Let $\theta_C = (\theta_1, \ldots, \theta_k)$ be the vector of cell probabilities corresponding to clique C. Each element of the previous vector will be perturbed with a realisation from a uniform random variable. More precisely, the θ_i' are obtained as $\theta_i' = \theta_i + y_i$, with $y_i \sim \mathit{Uniform}(-\epsilon_i, \epsilon_i)$. Subsequently, we correct the newly created matrix in order to maintain invariant the marginal distributions on the current separators.

The procedure used will now be described here in algorithmic form.

1. Select randomly one clique C with matrix of cell probabilities $\theta_C = (\theta_1, \ldots, \theta_k)$ and indicate with $S' = \{S_1, \ldots, S_J\}$ the set of separators having a non empty intersection with C.

2. Sample y_i from a $Uniform(-\epsilon, +\epsilon)$ and set $\theta'_i = \theta_i + y_i$, $i = 1, \ldots, k$.

3. Correct the vector of cell probabilities θ' in order to leave invariant the marginal distributions of each separator, S_i say, in S'. To do this, first calculate, for each separator in S', its marginal probability table. θ' is then corrected by a factor equal to the difference between the new marginal of separator S_i and the old marginals. Finally, divide the obtained results by $2^{|C \setminus S_i|}$.

4. Choose a different separator and go back to step 3.

It can be shown that the final result is *invariant* to the order in which the separators are considered for the corrections.

We remark that the proposed change in θ is accepted with probability $\alpha = \min\{1, R_a\}$, where:

$$\begin{aligned} R_a &= \frac{\pi(y')}{\pi(y)} \times \frac{q(y')}{q(y)} \\ &= R_{post} \times R_{prop} \end{aligned} \tag{13}$$

Since we are modifying only the distribution corresponding to clique C, the posterior ratio results equal to:

$$R_{post} = \frac{p(x|g, \theta'_C)\pi(\theta'_C|g)}{p(x|g, \theta_C)\pi(\theta_C|g)}$$

The proposal ratio is equal to 1 since the proposal distribution is symmetric.

We finally remark that the algorithm presented can easily become hierarchical, introducing an extra random variable τ, as done in the previous subsection.

In particular, a change in τ, will be accepted with probability equal to:

$$\alpha = \min\{1, R_a\},$$

where

$$R_a = \frac{p(\theta|g, \tau')p(\tau')}{p(\theta|g, \tau)p(\tau)}.$$

4 Performance of the proposed methods

In order to evaluate the performance of the proposed methods, we shall first consider a complete data-set, already analysed in the Literature: the *Women and Mathematics* data set, which will be used to compare standard Occam's razor methods (as in Madigan and Raftery, 1994) to our proposed MC^3 methods (nonhierarchical and hierarchical), and show the advantages of a hierarchical prior, in terms of higher robustness of the posterior inference. We also compare the results with those obtained with our reversible jump MCMC methodology.

We then consider a more challenging sparse table, concerning a credit scoring data-set, where we compare, using a hierarchical prior, our extended MC^3 and the reversible jump MCMC algorithm.

We remark that all of our algorithms have been previously tested with simulations from the prior distribution. In this way it has been possible to test the correct implementation of the algorithm. Furthermore, after having obtained the final results from the posterior, we have run the usual convergence diagnostics, and found satisfactory performance of the algorithm.

4.1 Woman and mathematics data-set

This set of data concerns the attitude of New Jersey high-school students towards mathematics, the source is Fowlkes et al (1988). For a description of the problem and the data-set we also refer to Madigan and Raftery (1994) who analysed this data-set using Bayesian discrete graphical models.

The random variables of interest are:

(X_1) WAM Lecture Attendance: attended or did not attend;

(X_2) Sex: female, male;

(X_3) School type: suburban or urban;

(X_4) "I'll need mathematics in my future": agree or disagree;

(X_5) Subject Preference: math/science or liberal arts;

(X_6) Future Plans: college or job.

The aim of the research was to investigate whether the attendance of scientific lectures with female teachers had some influence on the interest of females towards mathematics.

There are 32768 possible models, of which about the 20% are non decomposable. Note that there is no close expression that permits calculating the number of decomposable graphs; we can empirically obtain this percentage running our algorithm to sample from a uniform prior on the model space and considering the graphs never visited.

We have analysed this data set both with the nonhierarchical and the hierarchical model, using our extended MC^3 algorithm. With the nonhierarchical model the posterior distribution is highly sensitive to the value of the hyperparameter λ_0; it is more concentrated for low values of λ_0 than for higher values.

With $\lambda_0 = 1$ the best two models take into account more than 80% of the posterior probability. On the other hand, with a more precise prior, with $\lambda_0 = 64$ we must consider 10 models to take into account 60% of the posterior probability. Furthermore, model ranking in terms of probability depends highly on the value of the hyperparameters.

We shall now present in detail results of the MCMC simulation, in the hierarchical case, taking $f = 1$ and $s = 0.1$. We have used different values of the hyperparameters, but the results do not differ markedly from those presented below, thus showing robustness of the hierarchical prior.

We first check performance issues of the algorithm. Mixing over g has been monitored looking at an appropriate measure of g, the number of edges present in the graph, which describes the graph complexity. In Figure (3) we have some diagnostic graphs on the number of edges. More precisely we represent, for a run of 100000 iterations, thinned every 100, the number of edges present at each iteration and the corresponding cumulative mean, autocorrelation and cumulative occupancy fractions.

Figure 3 about here

Note that the chain explores more frequently graphs with 6 or 7 edges; the cumulative mean of edges is quite stable, apart from few initial values due to the effect of the burn-in; for all lags greater than 1 the values of the autocorrelation function are not significantly different from zero. We can thus conclude there is indication of good stability of the MCMC output.

Figure 4 reproduces the most plausible graphs, according to the posterior distribution of g. We notice that the posterior distribution is less concentrated than in the nonhierarchical case: the best graph receives about 23% of the posterior probability, and this is substantially confirmed changing the values of the hyperparameters. In order to obtain 70% of the posterior probability we must consider at least 9 graphical structures. Note also that there is strong evidence for the marginal independence of variable X_1.

Figure 4 about here

The results can be compared with those obtained by Madigan and Raftery (1994), who select, with a nonhierarchical model and using Occam's razor on this same data-set, two different graphical structures. We remark that their best model is the same as ours, the second one correspond to the third one in our selection. However, their results are more sensible to the prior.

We shall now present briefly the results obtained by the analysis of the same set using our reversible jump MCMC methodology, using the same values for the hyperparameters. At first note that with the nonhierarchical model the convergence is slower than with the corresponding version of the MC^3 algorithm. In fact in order to reach a reasonable diagnosis of convergence, similar to that in Figure 3, we must consider at least 200000 iterations with a burn-in of 20000.

This results confirm what expected, remember that now we generate a new realisation both for the graphical structure and for the matrix of cell probability. Furthermore, some proposed values of the considered probabilities may be rejected because out of the boundaries.

With the hierarchical model the distribution is more concentrated. In fact after 200000

iterations and a burn in of 20000 the best model takes into more than 50% of the posterior probability. See figure 5 for the posterior probability of the graphs.

Figure 5 about here

All calculations have been made on a pc with a Pentium II microprocessor with 266 Mhz and 128.0 MB of Ram and do take at most 30 minutes of elaboration (for the reversible Jump hierarchical models).

4.2 Credit scoring data-set

Credit scoring is a class of statistical methods employed to classify creditors in two risk categories: "good" and "bad" payers. By credit risk we mean the probability of a delay in the repayment of the credit granted.

Statistical credit scoring is a procedure to determine the probability that an applicant for credit will repay on time the amount of credit he is granted. We shall say that the applicant is credit reliable. Such a procedure is built on a database of information concerning the credit behaviour of individuals; for instance, in a bank such information may be taken from the operations registered on the individual's account.

For a review on credit scoring, see e.g. Hand and Henley (1997). Here we follow the graphical modelling approach suggested in Hand et al. (1998), who select, from a frequentist viewpoint, the graphical model that best describes the relationships between credit reliability and other variables, describing the "banking status" of each individual. They then draw inferences conditionally on the selected model. Here we follow our proposed Bayesian approach, so that, when drawing inferences, we take more correctly into account inference due to model uncertainty (see e.g. Madigan and Raftery, 1994).

The dataset we consider consists of 1000 observations on clients of a southern German bank, who were given credit, for which 21 variables are available. The data can be downloaded from the web page of the University of Munich: http://www.stat.uni-muenchen.de/data-sets/credit.

Given the extremely high sparseness of the data, we have performed a preliminary screening of the variables, following Fahrmeir and Hamerle (1994). Therefore, the binary random variables we shall consider are:

(X_1) Gender

(X_2) Marital status: single, non single

(X_3) Banking account ?

(X_4) Good history of banking account ?

(X_5) Good repayment of past credits ?

(X_6) Large amount of the given credit ?

(X_7) Use of the credit: private, professional

(X_8) Credit deadline: short or long term

(X_9) Credit reliability

An important point is that the sample is stratified: in the sample, 700 individuals are credit reliable and 300 are not credit reliable.

A classical backward procedure, with a significance level of 5% leads to the following results:

a) Credit reliability is conditionally independent on gender.

b) Credit reliability is conditionally independent on the amount of the given credit.

c) Credit reliability is conditionally independent on having an account, but not on having a good account.

d) Credit deadline seems to be the variable which is mostly related to the others.

Consider now the application of our proposed Bayesian methodology, with the MC^3 algorithm. We have taken $f = 1.0$ and $s = 0.1$.

Figure 6 describes diagnostic output on the simulation, for a run of $n = 200000$ iterations plus $n = 10000$ of burn-in.

Figure 6 about here

Note that the Markov chain explores most frequently graphs with 9, 10 or 11 edges (out of the possible 36); the cumulative average number of edges seems to indicate a good stability of the results.

Table 1 reports the overall estimated probability of an edge being present.

Table 1 about here

Differently from what done in the WAM case, the high number of possible graphs makes difficult to discriminate between them on the basis of their posterior probabilities. Instead we suggest to build a representative graph, which contains all edges with a probability of being present, as evaluated in Table 1, greater than a certain threshold, such as 90%.

Comparing the result with the classical results note that the Bayesian model is more parsimonious. Credit reliability is conditionally independent on the variables which were also previously such, however there is one further independence, with marital status.

However, it is important, especially for edges whose presence is uncertain, to look at the posterior marginal odds ratios, averaged across all models. For instance consider edge (5,7). However, recall that, to do quantitative learning, we need to consider the reversible jump MCMC methodology.

We remark that, as in the WAM case, we need a longer run in order to achieve stability, comparable to that obtained for the MCS.

Figure 7 presents convergence diagnostics for the reversible jump MCMC approach, with a burn-in of 50000 and $n = 500000$ subsequent iterations.

Figure 7 about here

The chain reaches stability in correspondence to a simpler structure. In this case the mean number of edges corresponds to 9. Furthermore, we remark that edge $(5, 7)$ is now almost always present in the graph and the expected posterior odds ratio is estimated to be equal to 2.15.

5 Concluding remarks

In this paper we have concentrated on the problem of Bayesian model determination for discrete graphical models, showing that Markov Chain Monte Carlo techniques can be a useful tool in this field.

Our main contribution is the development of new MCMC techniques to model determination in discrete graphical models. On one hand we have improved an existing methodology, the MC^3 algorithm by Madigan and York (1995). On the other hand we have introduced an original methodology based on the use of the RJMCMC sampler by Green (1995).

Our results suggest employing hierarchical prior distributions, as they have two main advantages with respect to nonhierarchical priors: on one hand, they are easier to specify, and can thus constitute an "automatic" default choice, especially for highly complex problems; on the other hand, they seem to lead to inferences less sensitive to the prior, as they allow "borrowing strength" of sample information between different clique domains.

Although more difficult to implement and test than the MC^3, the Reversible Jump algorithm allows the extraction of posterior inference on *any* quantity of interest, in *both* the hierarchical and the nonhierarchical model. For instance, posterior estimates of the odds ratios, giving the strengths of the associations, can be easily obtained.

Both our algorithms are fully based on local computations, leading to efficient computations. On the other hand, a possible disadvantage is that we are restricted to decomposable graphical models. However, as shown in GG, quantitative learning in non-decomposable models can be reasonably well approximated by learning from mixtures of decomposable models. Alternatively, one can use the approach suggested in Dellaportas

and Foster (1999), which does RJMCMC model determination for both decomposable and non-decomposable graphs. However, their approach is not based on local calculations on cliques and separators, and is less suitable for the use of Hyper Markov priors.

Another important weakness of the methodology is that it becomes slow for very large domains, as the dimension of the model space increases more than exponentially with the number of vertices. Research is needed in the design of proposal moves which can improve the speed of convergence as well as on the related issue of monitoring the convergence of the algorithm.

We finally remark that our proposed methodology is quite general, and can be extended to other families of graphical models. In particular, some aspects for future research which we have not considered are :

(i) Merging the results obtained in the gaussian and in the discrete case in order to construct a sampler for the analysis of mixed models.

(ii) Extension of the number of factor levels for each variables allowed, from two to arbitrary finite values. This can be done quite easily with few modifications in the code.

(iii) Application of the methodology to directed graphs. In this case the graph can be updated both by adding/deleting one single edge, or by changing the direction of an arrow.

Acknowledgement

This work has been supported by EU TMR network ERB-FMRX-CT96-0095 on "Computational and Statistical methods for the analysis of spatial data".The authors acknowledge Phil Dawid for helpful comments provided during the HSSS conference on "Graphical Models" held in Tirano in September 1998; David Madigan for sending the MC^3 code; Ludwig Fahrmeir for providing the credit scoring data, and Stefano Farro for the results of the classical analysis of the credit scoring data. The third author acknowledges support

from University of Trento for Ph.D. grant, University of Bristol and Athens University for computing facilities

References

Agresti, A. (1990). *Categorical Data Analysis*, Wiley, New York.

Dawid, A.P. and Lauritzen, S.L.(1993). Hyper Markov Laws in the statistical analysis of decomposable graphical models. *Ann. Statist.*, **21**, 1272-317.

Dellaportas, P. and Forster, J.J. (1999). Markov Chain Monte Carlo model determination for hierarchical and graphical log-linear models. *To appear in Biometrika*

Fahrmeir and Hamerle (1994) *Multivariate statistical modelling based on generalized linear models.* Springer, New York, 1994

Fowlkes, E.B., Freeny, A.E. and Landwehr, J.M. (1988). Evaluating logistic models for large contingency tables. *J. Americ. Statist. Assoc.*, 83, 611-622.

Frydenberg, M. and Lauritzen, S.L.(1993). Hyper Markov Laws in the statistical analysis of decomposable graphical models. *Ann. Statist.*

Giudici, P. (1998). Smoothing sparse contingency tables: a graphical Bayesian approach. *Metron* , vol. LVI, pp. 171-188.

Giudici, P. e Green, P.J. (1999). Decomposable graphical gaussian model determination. *To appear in Biometrika.*

Giudici, P. and Tarantola, C. Global prior distribution for discrete graphical models. *JISS* , **5**, 129-147.

Green, P. J. (1998). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, **82**, 711-32.

Hand, D.J. and Henley, W.E. (1997). Statistical classification methods in consumer credit scoring: a review. *J.R.Stat. Sci. A*, **160**, 523-541.

Hand, D.J., McConway, K.J. and Stanghellini, E. (1997). Graphical Models of Applicants for Credit. *IMA Journal of Mathematics Applied in Business and Industry*, **8**, 143-155.

Lauritzen, S.L.(1996). *Graphical Models.* Oxford, Oxford University Press.

Madigan, D. and Raftery, A.E. (1994). Model Selection and accounting for model uncertainty in graphical models using Occam's window. *J. Americ. Statist. Assoc.*, **89**, 1535-46.

Madigan, D. and York, J. (1995). Bayesian graphical models for discrete data. *International statistical Review*, **63**,215-232.

Madigan, D. and York, J. (1997). Bayesian methods for the estimation of the size of a closed population. *Biometrika*, **84**,19-31.

Spiegelhalter, D. J., Dawid, A. P., Lauritzen, S. L., and Cowell, R. G. (1993). Bayesian analysis in expert systems (with discussion). *Statistical Science*, **8**, 219-247 and 204-283.

Table 1: **Estimated probability of an edge being present**

Edge	**1/100**	**Edge**	**1/100**	**Edge**	**1/100**
[2,1]	1.000000	[6,3]	0.000270	[8,4]	0.004520
[3,1]	0.000910	[6,4]	0.000155	[8,5]	0.651470
[3,2]	0.000000	[6,5]	0.014205	**[8,6]**	1.000000
[4,1]	0.000305	[7,1]	0.022965	[8,7]	0.005060
[4,2]	0.000380	[7,2]	0.004690	[9,1]	0.093475
[4,3]	1.000000	[7,3]	0.009405	[9,2]	0.037830
[5,1]	0.018675	[7,4]	0.034695	[9,3]	0.892105
[5,2]	0.002710	[7,5]	0.288675	**[9,4]**	0.999835
[5,3]	0.010265	[7,6]	0.005105	**[9,5]**	1.000000
[5,4]	0.229645	**[8,1]**	0.917005	[9,6]	0.010960
[6,1]	0.121920	[8,2]	0.049140	[9,7]	0.766750
[6,2]	0.003020	[8,3]	0.005115	**[9,8]**	0.999805

Figure 1: **Example of a junction tree**

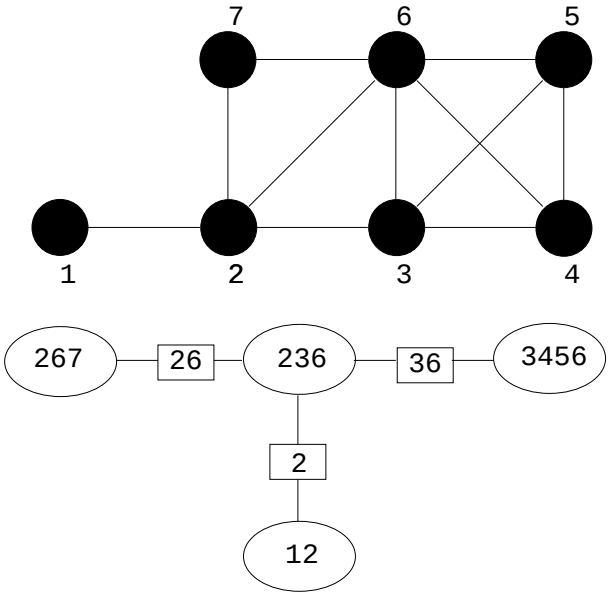

Figure 2: **Change in the Junction tree after adding an edge between vertices a and b**

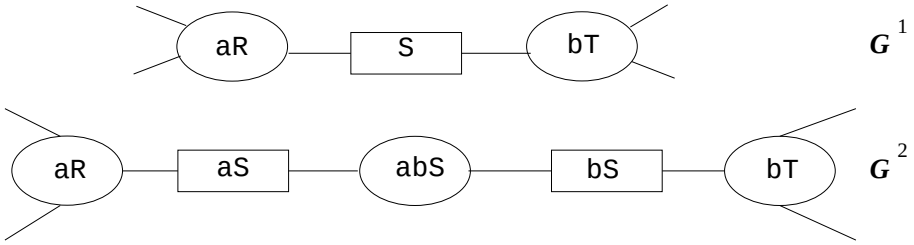

Figure 3: **Woman and mathematics: diagnostic on the number of edges present in the graph, with the hierarchical MC^3 method**

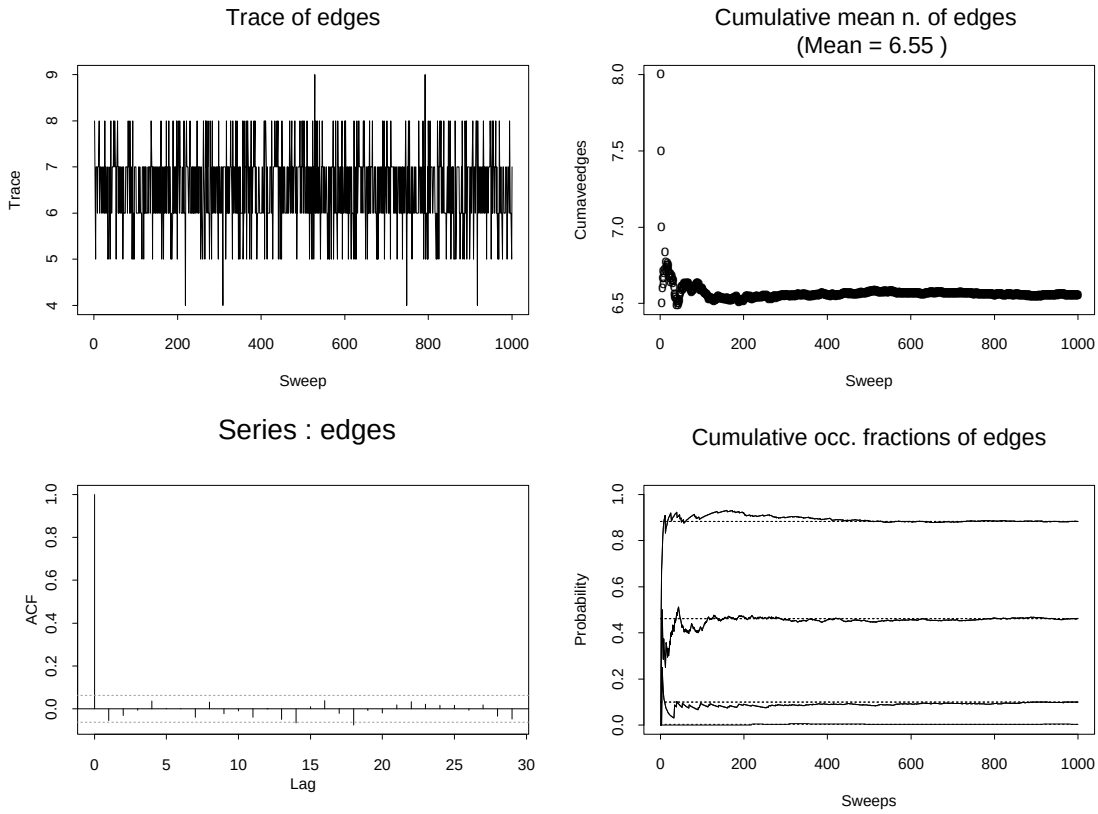

Figure 4: **Woman and mathematics: most probable graphs, with the hierarchical MC^3 method**

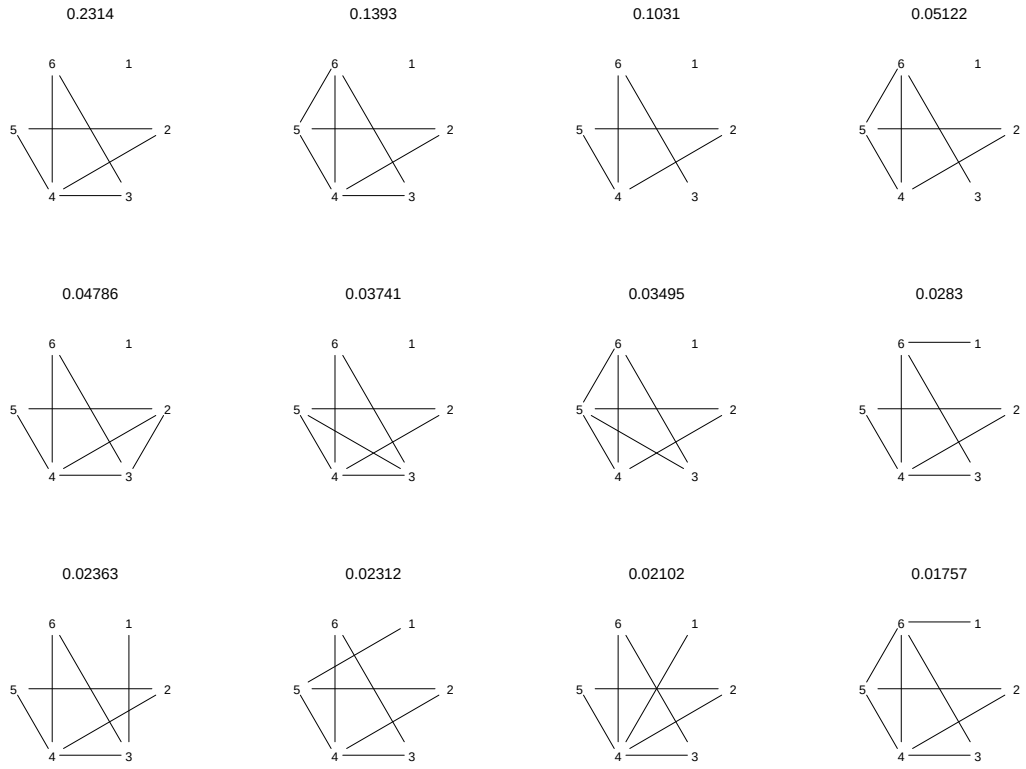

Figure 5: **Woman and mathematics: most probable graphs, with the hierarchical reversible jump MCMC method**

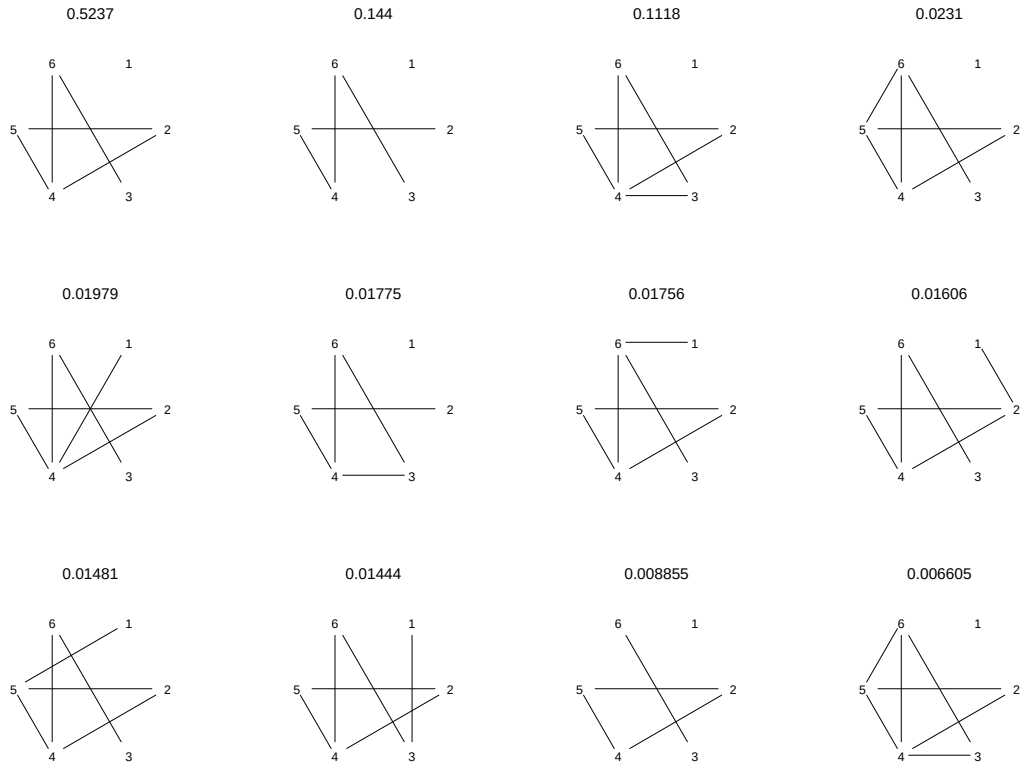

Figure 6: **Credit scoring : diagnostics on the number of edges, with the hierarchical MC^3 method**

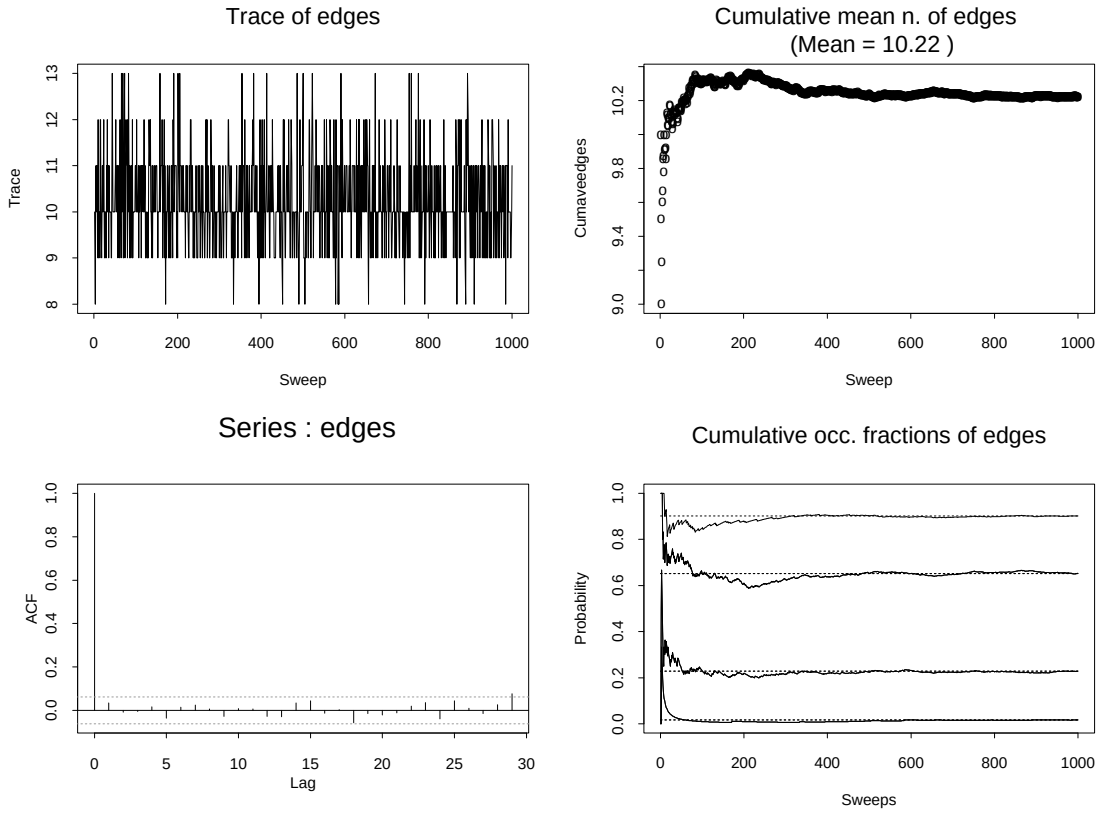

Figure 7: **Credit scoring : diagnostics on the number of edges, with the hierarchical Reversible Jump MCMC method**