BAYESIAN NETWORK STRUCTURAL LEARNING AND INCOMPLETE DATA

Philippe Leray and Olivier François

INSA Rouen - PSI Laboratory - FRE CNRS 2645
BP 08 - Av. de l'Université, 76801 St-Etienne du Rouvray Cedex, France
{Philippe.Leray , Olivier.Francois}@insa-rouen.fr

ABSTRACT

The Bayesian network formalism is becoming increasingly popular in many areas such as decision aid, diagnosis and complex systems control, in particular thanks to its inference capabilities, even when data are incomplete. Besides, estimating the parameters of a fixed-structure Bayesian network is easy. However, very few methods are capable of using incomplete cases as a base to determine the structure of a Bayesian network. In this paper, we take up the structural EM algorithm principle [9, 10] to propose an algorithm which extends the Maximum Weight Spanning Tree algorithm to deal with incomplete data. We also propose to use this extension in order to (1) speed up the structural EM algorithm or (2) in classification tasks extend the Tree Augmented Naive classifier in order to deal with incomplete data.

1. INTRODUCTION

Bayesian networks introduced by [17] are a formalism of probabilistic reasoning used increasingly in decision aid, diagnosis and complex systems control [15, 25, 24].

Let $\mathbb{X} = \{X_1, \ldots, X_n\}$ be a set of discrete random variables. A Bayesian network $\mathcal{B} =< \mathcal{G}, \Theta >$ is defined by

- a directed acyclic graph (DAG) $\mathcal{G} =< \mathbb{N}, \mathbb{U} >$ where $\mathbb{N}$ represents the set of nodes (one node for each variable) and $\mathbb{U}$ the set of edges,
- parameters $\Theta = \{\theta_{ijk}\}_{1 \leqslant i \leqslant n, 1 \leqslant j \leqslant q_i, 1 \leqslant k \leqslant r_i}$ the set of conditional probability tables of each node X_i knowing its parents' state $Pa(X_i)$ (with r_i and q_i as respective cardinalities of X_i and $Pa(X_i)$).

If $\mathcal{G}$ and Θ are known, many inference algorithms can be used to calculate the probability of any variable that has not been measured conditionally to the values of measured variables. (cf. [15, 25, 24] for a review of these methods). Bayesian networks are therefore a tool of choice for reasoning in uncertainty, based on incomplete data, which is often the case with real applications.

Determination of Θ (when $\mathcal{G}$ is given) is often based on expert knowledge, but several learning methods based on data have appeared. However, most of these methods only deal with complete data cases.

We will therefore first recall the issues relating to structural learning, and review the various ways of dealing with incomplete data, primarily for parameter estimation, and also for structure determination. We will then examine the structural EM algorithm principle, before proposing and testing a few ideas for improvement based on the extension of the Maximum Weight Spanning Tree algorithm to deal with incomplete data.

2. PRELIMINARY REMARKS

2.1. Structural learning

Robinson [28] showed that $r(n)$, the number of possible structures for Bayesian network having n nodes, is given by the recurrence formula of equation 1.

$$r(n) = \sum_{i=1}^{n} (-1)^{i+1} \binom{n}{i} 2^{i(n-i)} r(n-i) = n^{2^{\mathcal{O}(n)}} \quad (1)$$

This gives $r(1) = 1$, $r(2) = 3$, $r(3) = 25$, $r(5) = 29281$ and $r(10) \simeq 4,2 \times 10^{18}$.

Because of the super-exponential size of the search space, exhaustive search for the best structure is impossible. Many heuristic methods have been proposed to determine the structure of a Bayesian network. Some of them rely on human expert knowledge, others use real data which are -most of the time- completely observed [24, 8].

We are here more specifically interested in score-based methods, primarily GS algorithm [1] and MWST proposed by [4] and applied to Bayesian networks in [14]. GS is a greedy search carried out in DAG spaces where the interest of each structure located near the current structure is assessed by means of a BIC/MDL type measurement (equation 2)[1] or a Bayesian score like BDe (see [24] for a description of these scores).

$$BIC(\mathcal{G}, \Theta) = \log P(\mathcal{D}|\mathcal{G}, \Theta) - \frac{\log N}{2} \text{Dim}(\mathcal{G}) \quad (2)$$

where $\text{Dim}(\mathcal{G})$ is the number of parameters used for the Bayesian network representation and N is the size of the dataset $\mathcal{D}$.

[1] As [9], we consider that the BIC/MDL score is a function of the graph $\mathcal{G}$ and the parameters Θ, generalizing the classical definition of the BIC score which is defined with our notation by $BIC(\mathcal{G}, \Theta^*)$ where Θ^* is obtained by maximising the likelihood or $BIC(\mathcal{G}, \Theta)$ score for a given $\mathcal{G}$

The BIC score is decomposable. It can be written as the sum of local score computed for each node of the graph:

$$BIC(\mathcal{G}, \Theta) = \sum_i bic(X_i, P_i, \Theta_{Xi|P_i}) \quad (3)$$

where

$$bic(X_i, P_i, \Theta_{Xi|P_i}) = \quad (4)$$
$$\sum_{X_i=x_k} \sum_{P_i=pa_j} N_{ijk} \log \theta_{ijk} - \frac{\log N}{2} \text{Dim}(\Theta_{Xi|P_i})$$

with N_{ijk} the occurrence number of $\{X_i = x_k$ and $P_i = pa_j\}$ in $\mathcal{D}$.

The MWST principle is rather different. This algorithm determines the best tree that links all the variables, using a mutual information measurement [4] or the BIC score variation when two variables become linked [14]. The aim is to find an optimal solution, but in a space limited to trees. This very fast algorithm gives good results and can also be used for initializing greedy search carried out with GS [8].

2.2. Dealing with incomplete data

2.2.1. Nature of missing data.

Let $\mathcal{D} = \{X_i^l\}_{1 \leqslant i \leqslant n, 1 \leqslant l \leqslant N}$ our dataset, with $\mathcal{D}_o$ the observed part of $\mathcal{D}$, $\mathcal{D}_m$ the missing part and $\mathcal{D}_{co}$ the set of completely observed cases in $\mathcal{D}_o$. Let also $\mathcal{M} = \{M_{il}\}$ with $M_{il} = 1$ if X_i^l is missing, 0 if it is not. We have then the following relations:

$$\begin{aligned} \mathcal{D}_m &= \{X_i^l \ / \ M_{il} = 1\}_{\ 1 \leqslant i \leqslant n, 1 \leqslant l \leqslant N} \\ \mathcal{D}_o &= \{X_i^l \ / \ M_{il} = 0\}_{\ 1 \leqslant i \leqslant n, 1 \leqslant l \leqslant N} \\ \mathcal{D}_{co} &= \{[X_1^l \dots X_n^l] \ / \ [M_{1l} \dots M_{nl}] = [0 \dots 0]\}_{\ 1 \leqslant l \leqslant N} \end{aligned}$$

Dealing with missing data depends on their nature. Rubin [29] identified several types of missing data:

- MCAR (Missing Completly At Random) : $P(\mathcal{M}|\mathcal{D}) = P(\mathcal{M})$, the probability for data to be missing does not depend on $\mathcal{D}$,
- MAR (Missing At Random) : $P(\mathcal{M}|\mathcal{D}) = P(\mathcal{M}|\mathcal{D}_o)$, the probability for data to be missing depends on observed data,
- NMAR (Not Missing At Random) : the probability for data to be missing depends on both observed and missing data.

MCAR and MAR cases are the easiest to solve as observed data include all necessary information to estimate missing data distribution. The case of NMAR is trickier as outside information has to be used to model missing data distribution.

2.2.2. Determining parameters Θ when data are incomplete.

With MCAR data, the first and simplest possible approach is the *complete case analysis*. This is a parameter estimation based on $\mathcal{D}_{co}$ the set of completely observed cases in $\mathcal{D}_o$. When $\mathcal{D}$ is MCAR, the estimator based on $\mathcal{D}_{co}$ is unbiased. However, with a high number of variables the probability for a case $[X_1^l \dots X_n^l]$ to be completely measured is low and $\mathcal{D}_{co}$ may be empty.

One advantage of Bayesian networks is that, if only X_i and $P_i = Pa(X_i)$ are measured, then the corresponding conditional probability table can be estimated. Another possible method with MCAR cases is the *available case analysis*, i.e. using for the estimation of each conditional probability $P(X_i|Pa(X_i))$ the cases in $\mathcal{D}_o$ where X_i and $Pa(X_i)$ are measured, not only in $\mathcal{D}_{co}$ (where all X_i's are measured) as in the previous approach.

Many methods try to rely more on all the observed data. Among them are *sequential updating* [30], *Gibbs sampling* [12], and *expectation maximisation* (EM) [7, 18] algorithms which use the missing data MAR properties. More recently, *bound and collapse* [26] and *robust bayesian estimator* [27] algorithms try to resolve this task whatever the nature of missing data.

Algorithm 1 explains in detail how EM works. EM has been proposed by [7] and adapted by [18] to the learning of the parameters of a Bayesian network whose structure is known. Let $\log P(\mathcal{D}|\Theta) = \log P(\mathcal{D}_o, \mathcal{D}_m|\Theta)$ be the data log-likelihood. $\mathcal{D}_m$ being an unmeasured random variable, this log-likelihood is also a random variable function of $\mathcal{D}_m$. By establishing a reference model Θ^*, it is possible to estimate the probability density of the missing data $P(\mathcal{D}_m|\Theta^*)$ and therefore to calculate $Q(\Theta : \Theta^*)$ expectation of the previous log-likelihood:

$$Q(\Theta : \Theta^*) = E_{\Theta^*} [\log P(\mathcal{D}_o, \mathcal{D}_m|\Theta)] \quad (5)$$

So $Q(\Theta : \Theta^*)$ is the expectation of the likelihood of any set of parameters Θ calculated using a distribution of the missing data $P(\mathcal{D}_m|\Theta^*)$. This equation can be rewritten as follows (cf. [24]):

$$Q(\Theta : \Theta^*) = \sum_{i=1}^{n} \sum_{X_i=x_k} \sum_{P_i=pa_j} N_{ijk}^* \log \theta_{ijk} \quad (6)$$

where $N_{ijk}^* = E_{\Theta^*} [N_{ijk}] = N \times P(X_i = x_k, P_i = pa_j|\Theta^*)$ is obtained by inference in the network $< \mathcal{G}, \Theta^* >$ if the $\{X_i$,$P_i\}$ are not completely measured, or else only by mere counting.

The EM principle is very simple:

Algorithm 1 : Generic EM for parameter learning

1: **Init :** $i = 0$
   Random or heuristic choice of initial parameters Θ^0
2: **repeat**
3: $\quad i = i + 1$
4: $\quad \Theta^i = \underset{\Theta}{\text{argmax}} \ \ Q(\Theta : \Theta^{i-1})$
5: **until** $|Q(\Theta^i : \Theta^{i-1}) - Q(\Theta^{i-1} : \Theta^{i-1})| \leqslant \epsilon$

- *expectation*: estimating the N^* of equation 6 from a reference model $< \mathcal{G}, \Theta^i >$,
- *maximisation*: choosing the best value of parameters Θ^{i+1} by maximising Q,
- repeating these two steps until the optimal value of Q is reached.

Dempster et al. [7] proved this algorithm convergence, as well as the fact that it was not necessary to find the global optimum Θ^{i+1} of function $Q(\Theta : \Theta^i)$ but simply a value which would increase function Q (*Generalized EM*).

2.2.3. Determining structure Θ when data are incomplete.

The main methods for structural learning with incomplete data use the EM principle : *Alternative Model Selection EM* (AMS-EM) [9] and *Bayesian Structural EM* (BS-EM) [10]. We can also cite the *Hybrid Independence Test* proposed in [6] that can use EM to estimate the essential sufficient statistics that are then used for an independence test in a constraint-based method. [23] proposes a structural learning method based on genetic algorithm and MCMC.

We will now explain the structural EM algorithm principle in details, and then we will put forward some ideas for improvement based on the extension of the Maximum Weight Spanning Tree algorithm to deal with incomplete data.

3. STRUCTURAL EM ALGORITHM

3.1. General principle

The *expectation maximisation* principle formulated by [7], which we have described above for parameter learning, applies more generally to structural learning [9, 10] with the algorithm 2.

The parameter maximisation (Step 4) of algorithm 1 has now to be replaced by a maximisation in the joint space $\{\mathcal{G}, \Theta\}$ which amounts to searching the best structure and the best parameters corresponding to this structure. In practice, these two steps are clearly distinct[2]:

$$\mathcal{G}^i = \underset{\mathcal{G}}{\text{argmax}} \;\; Q(\mathcal{G}, \bullet : \mathcal{G}^{i-1}, \Theta^{i-1}) \tag{7}$$

$$\Theta^i = \underset{\Theta}{\text{argmax}} \;\; Q(\mathcal{G}^i, \Theta : \mathcal{G}^{i-1}, \Theta^{i-1}) \tag{8}$$

[2] the notation $Q(\mathcal{G}, \bullet : \dots)$ used in equation 7 stands for $E_\Theta[Q(\mathcal{G}, \Theta : \dots)]$ for bayesian scores or $Q(\mathcal{G}, \Theta^o : \dots)$ where Θ^o is obtained by likelihood maximisation

Algorithm 2 : Generic EM for structural learning

1: **Init :** $i = 0$
   Random or heuristic choice of the initial bayesian network $(\mathcal{G}^0, \Theta^0)$
2: **repeat**
3: $\quad i = i + 1$
4: $\quad (\mathcal{G}^i, \Theta^i) = \underset{\mathcal{G},\Theta}{\text{argmax}} \;\; Q(\mathcal{G}, \Theta : \mathcal{G}^{i-1}, \Theta^{i-1})$
5: **until** $|Q(\mathcal{G}^i, \Theta^i : \mathcal{G}^{i-1}, \Theta^{i-1}) - Q(\mathcal{G}^{i-1}, \Theta^{i-1} : \mathcal{G}^{i-1}, \Theta^{i-1})| \leqslant \epsilon$

where $Q(\mathcal{G}, \Theta : \mathcal{G}^*, \Theta^*)$ is the expectation of the likelihood of any bayesian network $< \mathcal{G}, \Theta >$ calculated using a distribution of the missing data $P(\mathcal{D}_m | \mathcal{G}^*, \Theta^*)$.

Note that the first search (equation 7) in the space of possible graphs takes us back to the initial problem, i.e. the search for the best structure in a super-exponential space. However, with *Generalised EM* it is possible to look for a better solution to function Q, rather than the best possible one, without affecting the algorithm convergence properties. This search for a better solution can then be done in a limited space, like for example $\mathcal{V}_\mathcal{G}$, the set of the neigbours of graph $\mathcal{G}$ that have been generated by removal, addition or inversion of an arc.

Concerning the search in the space of the parameters (equation 8), [9] proposes repeating the operation several times, using intelligent initialisation. This step then amounts to running the parametric EM algorithm for each structure $\mathcal{G}^i$, starting with structure $\mathcal{G}^0$ (steps 4 to 7 of algorithm 3). The two structural EM algorithms proposed by Friedman can therefore be considered as greedy search algorithms like GS, with a learning of EM type parameters at each iteration.

Friedman [9] also suggests choosing structure $\mathcal{G}^0$ judiciously by using an oriented chain graph linking all the variables.

3.2. Choice of function Q

We now have to choose the function Q that will be used for structural learning. The likelihood used for parameter learning is not a good indicator to determine the best graph since it gives more importance to strongly connected structures. Moreover, it is impossible to calculate marginal likelihood when data are incomplete, so that it is necessary to rely on an efficient approximation like those reviewed by [2]. In complete data cases, the most frequently used measurements are the BIC/MDL score and the Bayesian

Algorithm 3 : Detailed EM for structural learning

1: **Init :** $finished = false, i = 0$
   Random or heuristic choice of the initial bayesian network $(\mathcal{G}^0, \Theta^{0,0})$
2: **repeat**
3: $\quad j = 0$
4: $\quad$ **repeat**
5: $\quad\quad \Theta^{i,j+1} = \underset{\Theta}{\text{argmax}} \;\; Q(\mathcal{G}^i, \Theta : \mathcal{G}^i, \Theta^{i,j})$
6: $\quad\quad j = j + 1$
7: $\quad$ **until** convergence $(\Theta^{i,j} \rightarrow \Theta^{i,j^o})$
8: $\quad$ **if** $i = 0$ or $|Q(\mathcal{G}^i, \Theta^{i,j^o} : \mathcal{G}^{i-1}, \Theta^{i-1,j^o}) - Q(\mathcal{G}^{i-1}, \Theta^{i-1,j^o} : \mathcal{G}^{i-1}, \Theta^{i-1,j^o})| > \epsilon$ **then**
9: $\quad\quad \mathcal{G}^{i+1} = \underset{\mathcal{G} \in \mathcal{V}_{\mathcal{G}^i}}{\arg\max} \;\; Q(\mathcal{G}, \bullet : \mathcal{G}^i, \Theta^{i,j^o})$
10: $\quad\quad \Theta^{i+1,0} = \underset{\Theta}{\text{argmax}} \;\; Q(\mathcal{G}^{i+1}, \Theta : \mathcal{G}^i, \Theta^{i,j^o})$
11: $\quad\quad i = i + 1$
12: $\quad$ **else**
13: $\quad\quad finished = true$
14: $\quad$ **end if**
15: **until** $finished$

BDe score (see paragraph 2.1). When proposing the MS-EM and AMS-EM algorithms, [9] shows how to use the BIC/MDL score with incomplete data, by applying the principle of equation 5 to the BIC score (equation 2) instead of likelihood. Function Q^{BIC} is defined as the BIC score expectation by using a certain probability density on the missing data $P(\mathcal{D}_m|\mathcal{G}^*,\Theta^*)$:

$$Q^{BIC}(\mathcal{G},\Theta:\mathcal{G}^*,\Theta^*) = \quad (9)$$
$$E_{\mathcal{G}^*,\Theta^*}\left[\log P(\mathcal{D}_o,\mathcal{D}_m|\mathcal{G},\Theta)\right] - \frac{1}{2}\text{Dim}(\mathcal{G})\log N$$

As the BIC score is decomposable, so is Q^{BIC}:

$$Q^{BIC}(\mathcal{G},\Theta:\mathcal{G}^*,\Theta^*) = \quad (10)$$
$$\sum_i Q^{bic}(X_i,P_i,\Theta_{Xi|P_i}:\mathcal{G}^*,\Theta^*)$$

where

$$Q^{bic}(X_i,P_i,\Theta_{Xi|P_i}:\mathcal{G}^*,\Theta^*) = \quad (11)$$
$$\sum_{X_i=x_k}\sum_{P_i=pa_j} N^*_{ijk}\log\theta_{ijk} - \frac{\log N}{2}\text{Dim}(\Theta_{Xi|P_i})$$

with $N^*_{ijk} = E_{\mathcal{G}^*,\Theta^*}\left[N_{ijk}\right] = N * P(X_i = x_k, P_i = pa_j|\mathcal{G}^*,\Theta^*)$ obtained by inference in the network $\{\mathcal{G}^*,\Theta^*\}$ if $\{X_i,P_i\}$ are not completely measured, or else only by mere counting.

With the same reasoning, [10] proposes the adaptation of the BDe score to incomplete data.

4. MWST-EM, A STRUCTURAL EM IN THE SPACE OF TREES

4.1. General principle

Following [14]'s recommendations, [8] have shown that, in complete data cases, the MWST algorithm was able to find a simple structure very rapidly (the best tree connecting all the nodes in the space $\mathbb{T}$), which could be used as judicious initialisation by the GS algorithm. [14] also suggests using the variation of any decomposable score instead of the mutual information originally used in MWST.

Moreover, we have seen that EM algorithms proposed by [9, 10] could be considered as GS algorithms performing parametric EM at each iteration, which could be called GS-EM.

It therefore seems logical now to adapt the MWST algorithm so that it can deal with incomplete data, and to determine whether this new MWST-EM algorithm has the same properties (rapidity, "correct" solution and good initialisation of GS-EM).

4.2. Algorithm

Step 1 of algorithm 4, like all the previous algorithms, deals with the choice of the initial structure. The choice of an oriented chain graph linking all the variables proposed by [9] seems even more judicious here, since this chain graph also belongs to the tree space.

Steps 4 to 7 do not change. They deal with the running of the parametric EM algorithm for each structure $\mathcal{T}^i$, starting with structure $\mathcal{T}^0$.

There is a change from the regular structural EM algorithm in step 9, i.e. the search for a better structure for the iteration that comes next. With the previous structural EM algorithms, we were looking for the best DAG among the neighbours of the current graph. With MWST-EM, we can directly get the best tree that maximises function Q.

4.3. Calculation of function Q

In paragraph 2.1, we briefly recalled that the MWST algorithm used a similarity function between two nodes which was based on the BIC score variation whether X_j is linked to X_i or not. This function can be summed up in the following (symmetrical) matrix :

$$\left[M_{ij}\right]_{1\leqslant i,j\leqslant n} = \left[bic(X_i,P_i=X_j,\Theta_{Xi|X_j}) - bic(X_i,P_i=\emptyset,\Theta_{Xi})\right] \quad (12)$$

where the local *bic* score is defined in equation 4.

Running maximum (weight) spanning algorithms like Kruskal's on matrix M enables us to obtain the best tree $\mathcal{T}$ that maximises the sum of the local scores on all the nodes, i.e. function BIC of equation 3.

By applying the principle we described in section 3.2, we can then adapt MWST to incomplete data by replacing the local bic score of equation 12 with its expectation; to do so, we use a certain probability density of the missing data $P(\mathcal{D}_m|\mathcal{T}^*,\Theta^*)$:

$$\left[M^Q_{ij}\right] = \left[Q^{bic}(X_i,P_i=X_j,\Theta_{Xi|X_j}:\mathcal{T}^*,\Theta^*) - Q^{bic}(X_i,P_i=\emptyset,\Theta_{Xi}:\mathcal{T}^*,\Theta^*)\right] \quad (13)$$

Algorithm 4 : Structural EM in the tree space (MWST-EM)

1: **Init :** $finished = false, i = 0$
   Random or heuristic choice of the initial tree $(\mathcal{T}^0,\Theta^{0,0})$
2: **repeat**
3: $j = 0$
4: **repeat**
5: $\Theta^{i,j+1} = \underset{\Theta}{\text{argmax}}\ Q(\mathcal{T}^i,\Theta:\mathcal{T}^i,\Theta^{i,j})$
6: $j = j+1$
7: **until** convergence $(\Theta^{i,j}\rightarrow\Theta^{i,j^o})$
8: **if** $i = 0$ or $|Q(\mathcal{T}^i,\Theta^{i,j^o}:\mathcal{T}^{i-1},\Theta^{i-1,j^o}) - Q(\mathcal{T}^{i-1},\Theta^{i-1,j^o}:\mathcal{T}^{i-1},\Theta^{i-1,j^o})| > \epsilon$ **then**
9: $\mathcal{T}^{i+1} = \underset{\mathcal{T}\in\mathbb{T}}{\text{argmax}}\ Q(\mathcal{T},\bullet:\mathcal{T}^i,\Theta^{i,j^o})$
10: $\Theta^{i+1,0} = \underset{\Theta}{\text{argmax}}\ Q(\mathcal{T}^{i+1},\Theta:\mathcal{T}^i,\Theta^{i,j^o})$
11: $i = i+1$
12: **else**
13: $finished = true$
14: **end if**
15: **until** $finished$

With the same reasoning, running a maximum (weight) spanning algorithm on matrix M^Q enables us to get the best tree $\mathcal{T}$ that maximises the sum of the local scores on all the nodes, i.e. function Q^{BIC} of equation 10.

4.4. MWST-EM as an initialisation of structural EM algorithms

In [8], in order to initialise the greedy search algorithms like GS that use complete data, we proposed to take, as initial graph, the optimal tree obtained with MWST. This variant, which we called GS+T, can now be used for incomplete data; the tree obtained with MWST-EM will serve as initialisation of the SEM algorithms proposed by Friedman. This variant of the structural EM algorithm will be called AMS-EM+T.

4.5. Extension to classification problems

For classification tasks (where data are incomplete), many studies like those of [16, 20] use a structure based on an augmented nave Bayesian network, where observations (i.e. all the variables except class) are linked to the very best tree (*TANB, Tree Augmented Naive Bayes*). [11] showed it was the tree obtained by running MWST on the observations.

It is therefore possible to extend this specific structure to classification problems when data are incomplete by running our MWST-EM algorithm, and this algorithm will be called TANB-EM.

4.6. Related works

Meila [21] applies MWST algorithm and EM principle, but in another framework, learning mixtures of trees. In this work, the data is complete, but a new variable is introduced in order to take into account the weight of each tree in the mixture. This variable isn't measured so EM is used to determine the corresponding parameters.

Cohen et al. [5] deal with TANB classifiers and EM principle for partially unlabeled data. In there work, only the variable corresponding to the class can be partially missing whereas any variable can be partially missing in our TANB-EM extension.

[13] propose maximising conditional likelihood for BN parameter learning. They apply their method to MCAR incomplete data by using *available case analysis* in order to find the best TANB classifier.

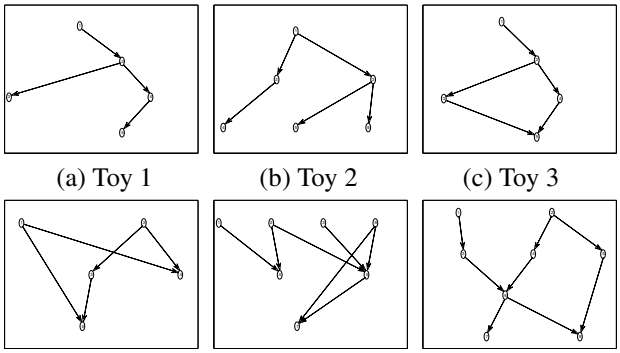
5. EXPERIMENTS

The first experiment stage aims at comparing our MWST-EM algorithm, the "classical" AMS-EM version, and our proposed initialisation for AMS-EM (AMS-EM+T).

We present the experimental protocol, the results and the first interpretations of the results below.

5.1. Protocol

We used Matlab, and more specifically the Bayes Net Toolbox [22]. We are developing and distributing a *structure learning* package (cf. [19]) based on this toolbox with the function codes implemented in the tests.



(a) Toy 1 (b) Toy 2 (c) Toy 3

(d) Toy 4 (e) Toy 5 (f) Asia

Figure 1. Reference toy structures

We tested the three following algorithms: MWST-EM – structural EM in the tree space, AMS-EM – structural EM in the DAG space, AMS-EM+T – AMS-EM using the MWST-EM result as an initialisation. The minimised score in each of these algorithms is the BIC approximation described in 4.3.

These algorithms were tested on a series of toy problems as on the classical ASIA network described in figure 1. For each problem, incomplete (MCAR) datasets were generated with various incompleteness rates ($10\%, 20\%$ and 30%). The (randomly chosen) sizes of these learning datasets are given in table 1.

To take into account the problems linked to method initialisation, and the variability of the learning datasets, each algorithm was evaluated by calculating the mean BIC score of the networks obtained when running the corresponding algorithm five times. Each BIC score was calculated on a test dataset generated by the initial network. The size of these dataset (randomly chosen) is given in figure 1.

To assess the interest of each method and to compare their respective complexity, we also measured the mean calculation time of each algorithm (seconds).

5.2. Results

The results obtained in our toy problems are summed up in table 1 from learning "quality" (BIC score of the initial network and of the obtained results) and calculation time points of view. Figure 2 gives us a graphical representation of these results by ploting the "quality" of each algorithm versus the computation time.

With initial tree structures (Toy 1 and Toy 2), MWST-EM finds a better graph than AMS-EM, and does so faster.

With more complex networks (Toy 3, Toy 4, Toy 5 and ASIA), the structure that MWST-EM finds is less good than AMS-EM, which is also logical. However, it should be note that the score of the obtained network is often closer to the best score and always with a lower calculation time.

So our algorithm logically gives better results in the tree space and will give a fast and not too bad solution when the theoretical network is not a tree. This observa-

		Toy 1	Toy 2	Toy 3	Toy 4	Toy 5	Asia
	Nlearn, Ntest	300, 500	500, 1000	1000, 1000	1000,1000	2000, 1000	2000, 1000
	DAG init	-1339.1	-2950.8	-2676.0	-2581.3	-3869.5	-2281.1
10	MWST-EM	-1374.5 (55)	-3043.7 (101)	-2850.1 (88)	-2706.1 (140)	-4119.2 (441)	-2873.2 (458)
	AMS-EM	-1384.9 (136)	-3064.2 (474)	-2829.8 (218)	-2724.1 (610)	-4039.5 (1651)	-2887.8 (8583)
	AMS-EM+T	-1402.8 (178)	-3073.8 (483)	-2800.0 (308)	-2722.2 (704)	-4068.7 (2542)	-2726.9 (4211)
20	MWST-EM	-1375.6 (64)	-3047.9 (101)	-2862.8 (89)	-2711.8 (216)	-4134.2 (534)	-2903.6 (773)
	AMS-EM	-1375.3 (137)	-3083.6 (540)	-2846.0 (282)	-2713.7 (660)	-4041.2 (1898)	-2831.9 (12393)
	AMS-EM+T	-1421.0 (187)	-3097.1 (550)	-2729.8 (356)	-2698.2 (885)	-4003.2 (2855)	-2713.0 (9633)
30	MWST-EM	-1374.4 (69)	-3034.1 (135)	-2852.7 (103)	-2708.1 (217)	-4121.5 (923)	-2885.3 (796)
	AMS-EM	-1404.5 (174)	-3098.7 (461)	-2846.9 (325)	-2694.6 (565)	-3997.6 (3123)	-2638.7 (16476)
	AMS-EM+T	-1421.1 (191)	-3112.9 (466)	-2821.2 (595)	-2721.9 (736)	-4035.2 (4482)	-2552.5 (11266)

Table 1. Mean BIC scores (on 5 runnings) calculated on test data for the initial network (figure 1) and for networks learned with the MWST-EM, AMS-EM and AMS-EM+T methods. Each algorithm is used with incomplete learning datasets, with incompleteness rates of $10\%, 20\%$ and 30%. Mean calculation times (sec.) are also given in brackets.

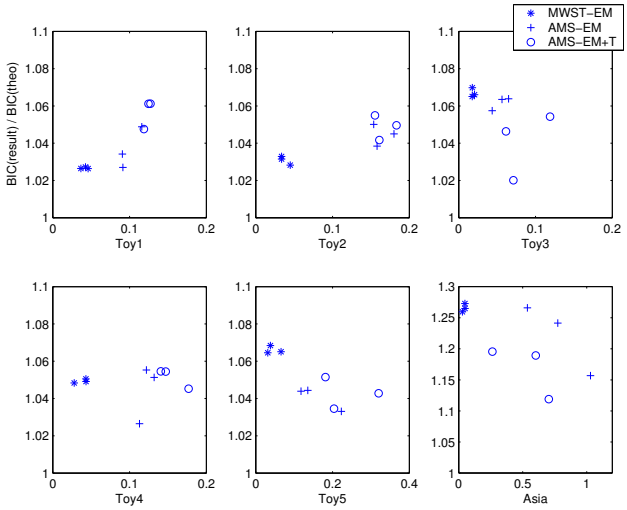


Figure 2. Mean BIC scores (on 5 runnings) calculated on test data for networks learned with the MWST-EM, AMS-EM and AMS-EM+T methods (divided by the BIC score of the initial network) versus mean calculation times (divided by Nlearn*Nvar). Each algorithm is used with incomplete learning datasets, with incompleteness rates of $10\%, 20\%$ and 30%.

tion confirms that MWST-EM should be used as AMS-EM initialisation. Indeed, table 1 also shows that the results of AMS-EM+T are often better than AMS-EM. The initialisation that we propose seems to lead to the same solution faster, or to a better one.

6. CONCLUSIONS AND PROSPECTS

Bayesian networks are a tool of choice for reasoning in uncertainty, with incomplete data. However, most of the time, bayesian network structural learning only deal complete data cases.

We have therefore proposed an original method of tree structural learning using incomplete data. In this method, the EM principle already used in DAG space is applied to a structural learning based on maximum weight spanning tree (MWST). MWST-EM has been empirically compared to AMS-EM, its equivalent in DAG space.

The first results show that this method is quite efficient and not very complex. By using it, it is possible to find structures which have a good score, and to do so more rapidly than AMS-EM. However, the drawback of our method is that it is applied in tree space, a subspace less rich than the DAG space.

We then have used our method to initialise the AMS-EM algorithm. With this initialisation, it is often possible to get better results than with the original initialisation.

This first conclusive experiment stage is not final. We are now planning to test and evaluate these algorithms on a wider range of problems, especially with various MAR datasets, and on problems specifically related to classification.

The MWST-EM and AMS-EM methods are respective adaptation of MWST and GS to incomplete data. Both structural search algorithms are applying in DAG space. Chickering and Meek [3] recently proposed an optimal search algorithm (GES) which deals with Markov equivalent space. Logically enough, the next step in our research is to adapt GES to incomplete datasets.

7. ACKNOWLEDGEMENT

This work was supported in part by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778. This publication only reflects the authors' views.

8. REFERENCES

[1] D. Chickering, D. Geiger, and D. Heckerman. Learning Bayesian networks: Search methods and experimental results. In *Proceedings of Fifth Conference on Artificial Intelligence and Statistics*, pages 112–128, 1995.

[2] D. Chickering and D. Heckerman. Efficient Approximation for the Marginal Likelihood of Incomplete

Data given a Bayesian Network. In *Proceedings of the 12th Conference on Uncertainty in Artificial Intelligence (UAI-96)*, pages 158–168. Morgan Kaufmann, 1996.

[3] D. Chickering and C. Meek. Finding optimal Bayesian networks. In A. Darwiche and N. Friedman, editors, *Proceedings of the 18th Conference on Uncertainty in Artificial Intelligence (UAI-02)*, pages 94–102, S.F., Cal., 2002. Morgan Kaufmann Publishers.

[4] C. Chow and C. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 14(3):462–467, 1968.

[5] I. Cohen, F. G. Cozman, N. Sebe, M. C. Cirelo, and T. S. Huang. Semisupervised learning of classifiers: Theory, algorithms, and their application to human-computer interaction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(12):1553–1568, 2004.

[6] D. H. Dash and M. J. Druzdzel. A robust independence test for constraint-based learning of causal structure. In *Proceedings of the 19th Conference on Uncertainty in Artificial Intelligence (UAI-03)*, pages 167–174, SF, CA, 2003. Morgan Kaufmann Publishers.

[7] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, B 39:1–38, 1977.

[8] O. Francois and P. Leray. Evaluation d'algorithmes d'apprentissage de structure pour les réseaux bayésiens. In *Proceedings of 14ème Congrès Francophone Reconnaissance des Formes et Intelligence Artificielle, RFIA 2004*, pages 1453–1460, Toulouse, France, 2004.

[9] N. Friedman. Learning belief networks in the presence of missing values and hidden variables. In *Proceedings of the 14th International Conference on Machine Learning*, pages 125–133. Morgan Kaufmann, 1997.

[10] N. Friedman. The Bayesian structural EM algorithm. In G. F. Cooper and S. Moral, editors, *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI-98)*, pages 129–138, San Francisco, July 24–26 1998. Morgan Kaufmann.

[11] D. Geiger. An entropy-based learning algorithm of Bayesian conditional trees. In *Uncertainty in Artificial Intelligence: Proceedings of the Eighth Conference (UAI-1992)*, pages 92–97, San Mateo, CA, 1992. Morgan Kaufmann Publishers.

[12] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(6):721–741, November 1984.

[13] R. Greiner and W. Zhou. Structural extension to logistic regression. In *Proceedings of the Eighteenth Annual National Conference on Artificial Intelligence (AAI02)*, pages 167–173, Edmonton, Canada, 2002.

[14] D. Heckerman, D. Geiger, and M. Chickering. Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 20:197–243, 1995.

[15] F. V. Jensen. *An introduction to Bayesian Networks*. Taylor and Francis, London, United Kingdom, 1996.

[16] E. Keogh and M. Pazzani. Learning augmented Bayesian classifiers: A comparison of distribution-based and classification-based approaches. In *Proceedings of the Seventh International Workshop on Artificial Intelligence and Statistics*, pages 225–230, 1999.

[17] J. Kim and J. Pearl. Convice; a conversational inference consolidation engine. *IEEE Trans. on Systems, Man and Cybernetics*, 17:120–132, 1987.

[18] S. Lauritzen. The EM algorithm for graphical association models with missing data. *Computational Statistics and Data Analysis*, 19:191–201, 1995.

[19] P. Leray and O. Francois. BNT structure learning package: Documentation and experiments. Technical report, Laboratoire PSI, 2004.

[20] P. Leray and O. Francois. Réseaux Bayésiens pour la classification – méthodologie et illustration dans le cadre du diagnostic médical. *Revue d'Intelligence Artificielle*, 18/2004:169–193, 2004.

[21] M. Meila-Predoviciu. *Learning with Mixtures of Trees*. PhD thesis, MIT, 1999.

[22] K. Murphy. The BayesNet Toolbox for Matlab. In *Computing Science and Statistics: Proceedings of Interface*, volume 33, 2001.

[23] J. Myers, K. Laskey, and T. Levitt. Learning Bayesian networks from incomplete data with stochastic search algorithms. In *Proceedings of the 15th Annual Conference on Uncertainty in Artificial Intelligence (UAI-99)*, pages 476–485, San Francisco, CA, 1999. Morgan Kaufmann Publishers.

[24] P. Naïm, P.-H. Wuillemin, P. Leray, O. Pourret, and A. Becker. *Réseaux bayésiens*. Eyrolles, Paris, 2004.

[25] J. Pearl. Graphical models for probabilistic and causal reasoning. In D. M. Gabbay and P. Smets, editors, *Handbook of Defeasible Reasoning and Uncertainty Management Systems, Volume 1: Quantified Representation of Uncertainty and Imprecision*, pages 367–389. Kluwer Academic Publishers, Dordrecht, 1998.

[26] M. Ramoni and P. Sebastiani. Parameter estimation in Bayesian networks from incomplete databases. *Intelligent Data Analysis*, 2(1-4):139–160, 1998.

[27] M. Ramoni and P. Sebastiani. Robust learning with missing data. *Machine Learning*, 45:147–170, 2000.

[28] R. W. Robinson. Counting unlabeled acyclic digraphs. In C. H. C. Little, editor, *Combinatorial Mathematics V*, volume 622 of *Lecture Notes in Mathematics*, pages 28–43, Berlin, 1977. Springer.

[29] D. Rubin. Inference and missing data. *Biometrika*, 63:581–592, 1976.

[30] D. J. Spiegelhalter and S. L. Lauritzen. Sequential updating of conditional probabilities on directed graphical structures. *Networks*, 20:579–605, 1990.