Knowledge Media Institute

Efficient Parameter Learning in Bayesian Networks from Incomplete Databases

Marco Ramoni *Paola Sebastiani*

KMI-TR-41

January 1997

Efficient Parameter Learning in Bayesian Networks from Incomplete Data

Marco Ramoni
Knowledge Media Institute
The Open University

Paola Sebastiani
Department of Actuarial Science and Statistics
City University

Abstract

Current methods to learn conditional probabilities from incomplete databases use a common strategy: they complete the database by inferring somehow the missing data from the available information and then learn from the completed database. This paper introduces a new method — called *bound* and *collapse* (BC) — which does not follow this strategy. BC starts by *bounding* the set of estimates consistent with the available information and then *collapses* the resulting set to a point estimate via a convex combination of the extreme points, with weights depending on the assumed pattern of missing data. Experiments comparing BC to the Gibbs Samplings are also provided.

Keywords: Machine Learning, Probabilistic Reasoning.

Reference: KMi Technical Report KMi-TR-41 (January 1997).

Address: Marco Ramoni. Knowledge Media Institute. The Open University. Walton Hall, Milton Keynes, United Kingdom MK7 6AA. PHONE: +44 (1908) 655721, FAX: +44 (1908) 653169, EMAIL: M.Ramoni@open.ac.uk, URL: http://kmi.open.ac.uk/~marco.

1. Introduction

Bayesian Belief Networks (BBNs) provide a powerful formalism to reason under uncertainty. Although in their original concept BBNs were mainly designed to encode the knowledge of human experts, their statistical roots soon prompted for the use of learning methods to extract them directly from databases of cases rather than from the insight of human domain experts [2, 1, 6].

A common assumption made by the current learning methods is that the database at hand is complete. The reason behind this assumption is that, when facing missing data, exact Bayesian learning becomes intractable, namely its complexity is exponential in the number of missing data in the database [2]. Unfortunately, real-world databases are rarely complete: unreported, lost, and corrupted data are one of their distinguished features. Researchers in both Artificial Intelligence and Statistics developed different methods to learn conditional probabilities from databases with missing data. Best known techniques are currently *Sequential updating*, the EM *algorithm*, and the Gibbs Sampling. For an updated review of these methods, see [5]. These solutions share a common strategy: they try to complete the database by inferring the missing data from the available information and then learn from the completed database. The underlying assumption is that the unreported data are Missing at Random (MAR) so that the incomplete database is a representative sample of the complete one.

This paper introduces a new method to learn conditional probabilities from incomplete databases which does not rely on the completion of the database: the method starts by *bounding* the set of possible estimates consistent with the available information and then *collapses* the resulting interval estimate to a point estimate via a convex combination of the extreme points with weights depending on the assumed pattern of missing data. Because of this strategy, we call it *bound* and *collapse* (BC) method. The use of this method allows the encoding of prior knowledge of the pattern of missing data, such as the MAR assumption, without any need to guess the missing data. Experimental valuations show clearly that the estimates provided by BC are equivalent to the ones provided by the Gibbs Sampling, when data are MAR, and the estimates provided by BC are more robust to departure from the true pattern of missing data, thus making the computational effort of stochastic methods vane.

2. Background

A BBN is a direct acyclic graph in which nodes represent stochastic variables and arcs represent conditional dependencies among variables. We will only consider discrete variables taking a finite number of values. More formally, a BBN is defined by a set of *variables* $\mathcal{X} = \{X_1, \ldots, X_I\}$ and a structure S defining a graph of conditional dependencies among the elements of $\mathcal{X}$. A conditional dependency links a *child* variable X_i to a set of *parent* variables Π_i, and it is defined by the conditional probabilities of each state of the child variable given a configuration of the parent variables. The structure S allows us to decompose the joint probability of a particular set of values of the variables in $\mathcal{X}$, say $x_m = \{x_{1m}, \ldots x_{Im}\}$, as

$$p(\mathcal{X} = x_m) = \prod_{i=1}^{I} p(X_i = x_{im} | \Pi_i = \pi_{ij}),$$

where π_{ij} denotes the state of Π_i in x_m. Clearly, $p(X_i = x_{im}|\Pi_i = \pi_{ij}) = p(X_i = x_{im})$ for a root node X_i. In the following we will denote $X_i = x_{im}$ as x_{im}, and $\Pi_i = \pi_{ij}$ as π_{ij}. We shall consider the conditional probabilities defining the BBN as being generated by parameters $\theta = \{\theta_1, \ldots, \theta_K\}$, so that the joint probability of a case x_j is

$$p(x_j|\theta) = \prod_{i=1}^{I} \theta_{ijm},$$

where θ_{ijm} parameterizes the probability of x_{im} given the parent configuration π_{ij}. Suppose we are given a database of J cases $\mathcal{D} = \{x_1, \ldots, x_J\}$. Given a network structure S, the task here is to learn the conditional probabilities defining the dependencies in the BBN from $\mathcal{D}$. In the Bayesian approach the parameters θ_k are regarded as random variables, whose prior distribution represents the observer's belief about the parameters before observing any data. Given the information in the database, the prior density $p(\theta)$ is updated in the posterior density using Bayes' theorem, and hence

$$p(\theta|\mathcal{D}) = \frac{p(\theta)p(\mathcal{D}|\theta)}{p(\mathcal{D})}$$

where $p(\mathcal{D}) = \int_{\mathcal{R}^K} p(\theta)p(\mathcal{D}|\theta)d\theta$ is the marginal probability of the database, and $p(\mathcal{D}|\theta) = \prod_j p(x_j|\theta)$, if different cases are independent given θ. A Bayesian estimate of θ is then the posterior expectation $E(\theta|\mathcal{D})$ of θ. Common assumptions of the Bayesian approach to learn BBNs are that the parameters are (i) mutually independent, and (ii) have a Dirichlet distribution. Under (i) the joint prior density of θ can be decomposed as $p(\theta) = \prod_{k=1}^{K} p(\theta_k)$, thus allowing local computation, while (ii) facilitates the computation of the posterior density by taking advantages of conjugate analysis. Consider for instance the variable X_i, taking values $\{x_{i1}, \ldots, x_{iM}\}$, and the parameters $\theta_i = \{\theta_{i1}, \ldots, \theta_{iM}\}$ associated to the conditional probabilities $p(x_{im}|\pi_{ij})$, $m = 1, \ldots, M$, so that $\sum_m \theta_{im} = 1$. A Dirichlet prior for θ_i, denoted as $D(\alpha_{i1}, \ldots, \alpha_{iM})$, $\alpha_{im} > 0$, is a continuous multivariate distribution with density function proportional to $\prod_m \theta_{im}^{\alpha_{im}-1}$. The quantity $\alpha_{i+} = \sum_m \alpha_{im}$ can be regarded as an imaginary sample size needed to formulate this prior information about θ_i, and the mean of θ_{im} is α_{im}/α_{i+}, $m = 1, \ldots, M$. Note that the prior mean is the marginal probability of $x_{im}|\pi_{ij}$.

With complete data, the posterior distribution of the parameters can be computed exactly and efficiently using standard conjugate analysis as $p(\theta_i|\mathcal{D}) = D(\alpha_{i1}+n(x_{i1}|\pi_{ij})), \ldots, \alpha_{iM}+n(x_{iM}|\pi_{ij})))$, where $n(x_{im}|\pi_{ij})$ is the frequency of cases in the database with $x_{im}|\pi_{ij}$. The Bayes estimate of the conditional probability of $x_{im}|\pi_{ij}$, given the information in the database, i.e. the posterior mean of θ_{im}, is then

$$\frac{\alpha_{im} + n(x_{im}|\pi_{ij})}{\alpha_{i+} + n(\pi_{ij})},$$

where $n(\pi_{ij}) = \sum_m n(x_{im}|\pi_{ij})$. Unfortunately, the situation is quite different when some entries in the database are missing. When a datum is missing, there is a set of possible

complete databases, one for each possible completion of the entry in the database for which the observation is missing. Exact analysis would require the computation of the joint posterior distribution of the parameters given each possible completion of the database, and then mix these over all possible completions. This is apparently infeasible and approximate methods therefore are required. Deterministic methods as sequential updating [7, 3] or the EM algorithm [4] provide a possible solution. However the need to process data one by one of the former, and the slow rate of convergence of the latter can impair their applicability to large databases.

An alternative approach, very popular nowadays, is the use of a stochastic approximation of the posterior distribution using Markov Chain Monte Carlo (MCMC) methods, such as the Gibbs Sampling. The basic idea of the Gibbs Sampling algorithm is (i) for each parameter θ_i sample a value from the conditional distribution of θ_i given all the other parameters and the data in the database, (ii) repeat this for all the parameters, and (iii) iterates these steps several times. In practical applications, the algorithm iterates a number of times and then, when stability seems to be reached, a final sample from the joint posterior distribution of the parameters is taken. When some of the entries in the database are missing, the Gibbs Sampling treats the missing data as unknown parameters, so that, for each missing entry, a value is sampled from the conditional distribution of the corresponding variables, given all the parameters and the available data. The algorithm is iterated to reach stability, and then a sample from the joint posterior distribution is taken which can be used to provide empirical estimates of the posterior means [8]. Convergence rate and convergence detection are open issues of this stochastic method.

These methods share a common strategy: they complete the database by inferring somehow the missing data from the available information and then learn from the completed database. The underlying assumption is that the unreported data are MAR so that the incomplete database is a representative sample of the complete one. In the next section, we will introduce a deterministic method to estimate parameters from an incomplete database which does not guess the missing data but estimates the parameters bounding their possible values and then collapses these values in a single probability measure using a convex combination of the extreme values of this set.

3. Method

This section introduces a deterministic method, called BC method, to estimate parameters from an incomplete database which does not guess the missing data but estimates the parameters by *bounding* their possible values and then *collapsing* these values in a single probability measure using a convex combination of the extreme values of this set.

3.1 Bound

The *bound* step of the BC method computes, for each parameter, a probability interval whose extreme points are the minimum and the maximum Bayes estimate that would have been inferred from all possible completions of the database.

Let X_i be a variable in the BBN and $n^\bullet(x_{im}|\pi_{ij})$ be the frequency of cases with $X_i = x_{im}$, given the parent configuration π_{ij}, which have been obtained by completing incomplete

cases. Note that these completions can be due either to an incomplete observation of the parent configuration, or to an incomplete observation of the variable X_i itself. Suppose further that we start from total ignorance, so that the parameter θ_{im} which is associated to $p(x_{im}|\pi_{ij})$ is assigned a uniform prior. Then the Bayes estimate $E(\theta_{im}|\mathcal{D})$, that would have been computed from the complete database is bounded below by

$$p_\bullet(x_{im}|\pi_{ij}, \mathcal{D}_{inc}) = \frac{1 + n(x_{im}|\pi_{ij})}{M + \sum_m n(x_{im}|\pi_{ij}) + \max_{l \neq m} n^\bullet(x_{il}|\pi_{ij})} \tag{1}$$

and above by

$$p^\bullet(x_{im}|\pi_{ij}, \mathcal{D}_{inc}) = \frac{1 + n(x_{im}|\pi_{ij}) + n^\bullet(x_{im}|\pi_{ij})}{M + \sum_m n(x_{im}|\pi_{ij}) + n^\bullet(x_{im}|\pi_{ij})} \tag{2}$$

Each maximum probability $p^\bullet(x_{im}|\pi_{ij}, \mathcal{D}_{inc})$ is obtained from a Dirichlet distribution $D_m(1+ n(x_{i1}|\pi_{ij}), \ldots, 1 + n(x_{im}|\pi_{ij}) + n^\bullet(x_{im}|\pi_{ij}), \ldots, 1 + n(x_{iM}|\pi_{ij}))$ which identifies a unique probability for the other states of the variable X_i given π_{ij}, say $p_{m\bullet}(x_{il}|\pi_{ij}, \mathcal{D}_{inc})$, and $p_\bullet(x_{im}|\pi_{ij}, \mathcal{D}_{inc}) = \min_l\{p_{l\bullet}(x_{im}|\pi_{ij}, \mathcal{D}_{inc})\}$. It is worth noting that these bounds depend only on the frequencies of complete entries in the database and the "artificial" frequencies of the completed entries, so that the whole data can be processed at once. This method is implemented using an algorithm based on discrimination trees to efficiently store information about the parameters.

The main feature of this method is that it does not rely on any assumption on the distribution of the missing data because it does not try to infer them: the available information can only give rise to constraints on the possible estimates that could be learned from the database. Furthermore, the use of this robust method provide BC with the a new measure of information: the width of the interval accounts for the amount of information available in the database about the parameter to be estimated and represents a measure of the quality of the probabilistic information conveyed by the database about a parameter. In this way, intervals provide an explicit representation of the reliability of the estimates which can be taken into account when the extracted BBN is used to perform a particular task.

3.2 Collapse

The second step of BC method collapses the interval estimated in the *bound* step into a point-valued estimate using a convex combination of the lower and upper bounds of this interval. This convex combination can be achieved either by the use of external information about the pattern of missing data or by a dynamic estimation of this pattern from the available information.

3.2.1 Using Prior Information

Suppose that some information on the pattern of missing data is available, which allow the user to formulate a probability distribution describing, for each incomplete case in the database, the probability of a completion as:

$$p(x_{im}|\pi_{ij}, X_i =?, \theta) = \phi_{ijm} \tag{3}$$

where $m = 1, ..., M$ and $\sum_{m=1}^{M} \phi_{ijm} = 1$. Such information can be used to summarize the interval estimate in a point estimate via a convex combination of the extreme probabilities i.e.:

$$\begin{aligned} &\hat{p}(x_{im}|\pi_{ij}, \mathcal{D}_{inc}, \phi_{ijm}) = \\ &\sum_{l \neq m} \phi_{ijl} p_{l\bullet}(x_{im}|\pi_{ij}, \mathcal{D}_{inc}) \quad + \quad \phi_{ijm} p^{\bullet}(x_{im}|\pi_{ij}, \mathcal{D}_{inc}). \end{aligned} \tag{4}$$

This point estimate is the expected Bayes estimate that would be obtained from the complete database when the mechanism generating the missing data in the database is described by (3). Note that the estimates so found define a probability distribution since

$$\begin{aligned} &\sum_{m=1}^{M} \hat{p}(x_{im}|\pi_{ij}, \mathcal{D}, \phi_{ijm}) = \\ &\sum_{m=1}^{M} \phi_{ijm} p^{\bullet}(x_{im}|\pi_{ij}, \mathcal{D}) + \sum_{m=1}^{M} \sum_{l \neq m=1}^{M} \phi_{ijl} p_{l\bullet}(x_{im}|\pi_{ij}, \mathcal{D}_{inc}) \\ &= \sum_{m=1}^{M} \phi_{ijm} \{p^{\bullet}(x_{im}|\pi_{ij}, \mathcal{D}) + \sum_{l \neq m=1}^{M} p_{m\bullet}(x_{il}|\pi_{ij}, \mathcal{D})\} = \\ &\sum_{m=1}^{M} \phi_{ijm} = 1. \end{aligned}$$

The intuition behind (4) is that the lower bound of the estimate $p(x_{im}|\pi_{ij}, \mathcal{D})$ is achieved when all the incomplete cases are assigned to one different value assumed by the variable $X_i|\Pi_{ij}$, and the upper bound is achieved when all the incomplete cases are completed as $x_{im}|\pi_{ij}$. Thus if the user specifies that $p(x_{im}|\pi_{ij}, X_i =?, \theta) = 1$ for a particular m, then (4) will return the upper bound of the interval probability as estimate of $p(x_{im}|\pi_{ij})$, and the lower bounds of the other interval probabilities as estimates of $p(x_{ih}|\pi_{ij})$, $h \neq m$. This case corresponds to the assumption that data are *systematically* missing in the database. On the other hand, when no information on the mechanism generating missing data is available and therefore any pattern of missing data is equally likely, then $\phi_{ijm} = 1/M$.

3.2.2 Using the Available Information

Suppose now that the user has sufficient information to believe that data are MAR. Let $\mathcal{D}_{obs}$ and $\mathcal{D}_{mis}$ denote respectively the part of the database with complete entries and that with incomplete cases. The MAR assumption would correspond to assuming that $\mathcal{D}_{obs}$ is a *representative sample* of $\mathcal{D}$, and hence we can use $\mathcal{D}_{obs}$ to estimate the probability of a completion as

$$\hat{\phi}_{ijm} = p(x_{im}|\pi_{ij}, \mathcal{D}_{obs}) = \frac{1 + n(x_{im}|\pi_{ij})}{M + \sum_m n(x_{im}|\pi_{ij})},$$

and then $\hat{\phi}$ can be used to compute (4). If $n^\bullet(x_{im}|\pi_{ij}) = n_{ij}$ for all m, (4) becomes

$$\hat{p}(x_{im}|\pi_{ij}, \mathcal{D}, \phi_{ijm}) = \frac{1 + n(x_{im}|\pi_{ij}) + n^\bullet_{ij}\hat{\phi}_{ijm}}{M + \sum_m n(x_{im}|\pi_{ij}) + n^\bullet_{ij}} \tag{5}$$

which is an approximation of the expected Bayesian estimate given $\mathcal{D}_{obs}$ and the assumption that data are MAR:

$$\frac{1 + n(x_{im}|\pi_{ij}) + n^\bullet_{ij}p(x_{im}|\pi_{ij}, \theta)}{M + \sum_m n(x_{im}|\pi_{ij}) + n^\bullet_{ij}}.$$

The Gibbs Sampling would complete the database by guessing the missing data from the current estimate of $p(x_{im}|\pi_{ij}, \theta)$, use the completed database to learn the parameters and iterate the procedure until convergence is reached. Clearly the estimates of the conditional probabilities returned in (5) are the expected estimates, and for large samples they will be the same. On the other hand, the computational cost of BC is reduced to the cost of two exact updating — one for each extreme distribution — plus the cost of a convex combination for each parameter in the BBN.

4. Experimental Evaluation

The Gibbs Sampling is currently considered one of the most feasible solutions to the problem of learning BBNs from databases with missing data, although its limitations are well-known: the convergence rate is slow and resource consuming. The aim of these experiments were to compare the accuracy of the parameter estimates provided by the Gibbs Sampling and our method as the available information in the database decreases.

4.1 Materials and Methods

We compared an implementation of our method to the implementation of the Gibbs Sampling provided by the program BUGS [8]. In the following experiments, we used the implementation of BUGS version 0.5 running on a Sun Sparc 5 under SunOS 5.5 and an implementation of our method written in Common Lisp running on the same platform under CLISP version 1996/10/10. Figure 1 shows the graphical structure of the simple BBN — defined by three binary variables A, B, and C — used for this comparison. We generated a database of 1000 random cases from the following probability distribution:

$$\begin{aligned}
p(A = 1) &= 0.5 \\
p(B = 1) &= 0.4 \\
p(C = 1|A = 1, B = 1) &= 0.95 \\
p(C = 1|A = 1, B = 2) &= 0.05 \\
p(C = 1|A = 2, B = 1) &= 0.1 \\
p(C = 1|A = 2, B = 2) &= 0.8
\end{aligned}$$

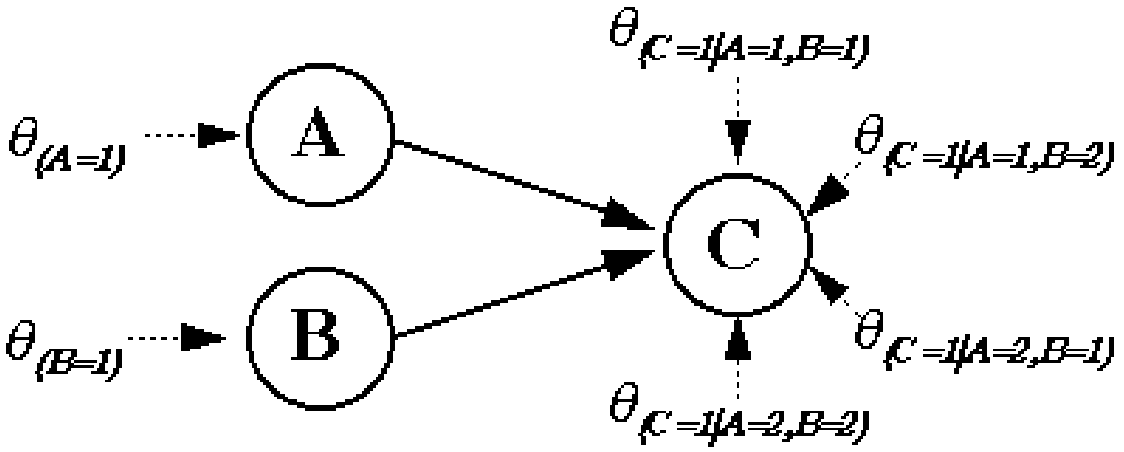

Figure 1: The simple network structure of the BBN used for the experimental evaluation.

4.2 Results

Using the materials described above, we ran two different experiments, one considering the situation in which data are systematically missing about a particular value of a variable, and one fulfilling the MAR assumption.

4.2.1 Systematically Missing Data

In the first experiment, we started with a complete database, where all the parameters are independent and uniformly distributed, and ran both our algorithm and the Gibbs Sampling on it. Then, we proceed by iteratively deleting 2% of the database by systematically removing the entries reporting the value $B = 1$ and, each time, we ran the two learning algorithms. Each run of the Gibbs Sampling is based on 2,000 iterations, which were sufficient to reach stability, and a final sample of 2,000 cases. This procedure was iterated until no entry $B = 1$ was reported in the database. Figure 2 plots the parameter estimates given by the two systems against the percentage of entries still present in the database. Dotted lines are the bounds computed by our method. Crosses are the estimates returned by our method when any pattern of the missing data is equally likely. Circles represent the point estimates returned by our method under the MAR assumption, so that the weights used in the convex combination are computed from the complete database. Plus represent the point estimates given by the Gibbs Sampling.

As expected, the estimates returned by the Gibbs Sampling and by our method using the weight computed from the complete database are almost identical. However, an apparently surprising exception occurs in the estimation of $p(C = 1|A = 2, B = 2)$ and $p(C = 1|A = 1, B = 2)$: with the 87% of the database but all the entries $B = 1$ removed, the Gibbs Sampling returns in the first case an estimate of 0.4123 and in the second case an estimate of 0.5876. Both estimates includes an error of almost 40% with respect to the estimates computed for the same parameters with the complete databases: 0.0453 and

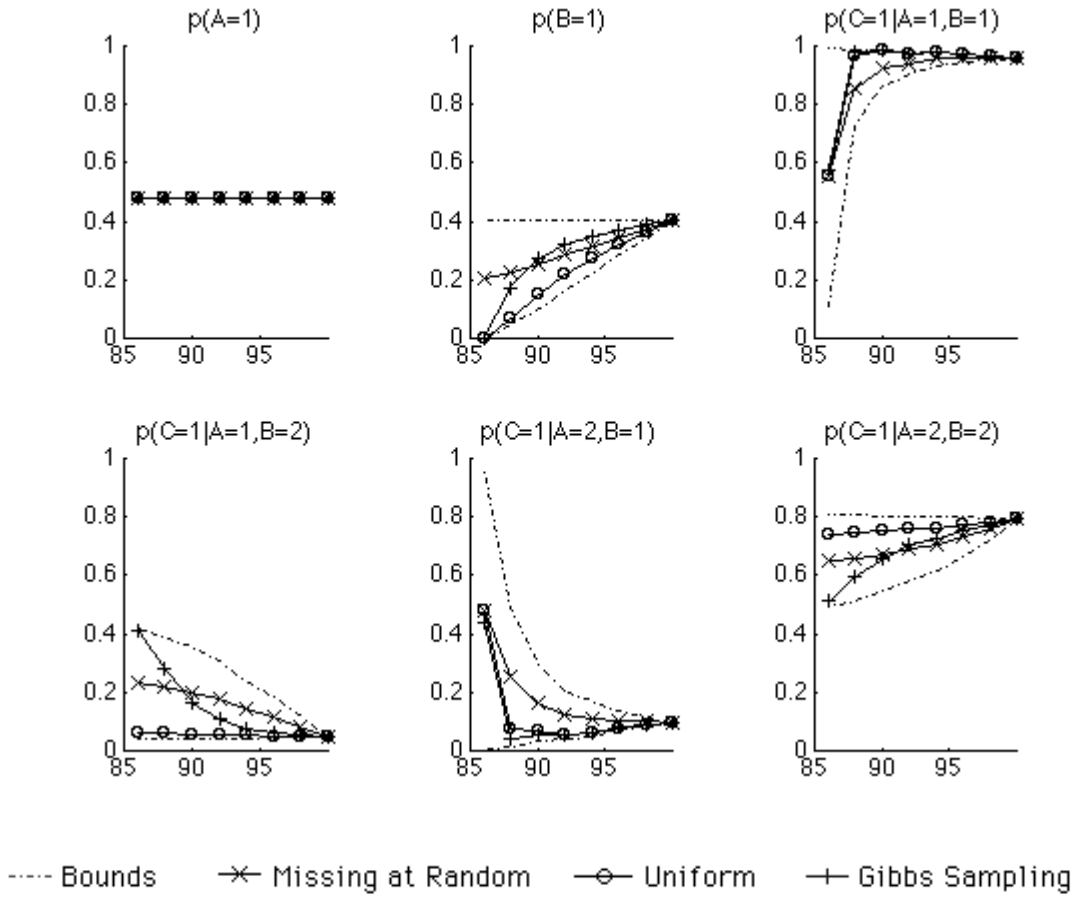

Figure 2: Parameter estimates against percentage of entries in the database. Entries B=1 are systematically removed.

0.9546 respectively. On the contrary, BC under the same MAR assumption, catches the correct value even when all the observations $B = 1$ are missing.

When the user encodes her ignorance about the pattern of missing data as a uniform distribution over the possible states of the parameters, the results are not as good as the ones obtained by BC under the assumption that data are MAR, but are still better than the ones obtained by the Gibbs Sampler. It is worth noting that, if information about the systematic pattern of missing data were available, this could have been encoded in our method by setting the weight to 1. In this case, the estimations returned by BC would have collapsed on the bounds of the reported intervals and, therefore, even more accurate than the estimates returned under the MAR assumption. A further difference between the performances of the two systems has been the execution time: in the worse case, Gibbs Sampling took over 3 minutes to run to completion, while our system ran to completion the same task in less than 20 milliseconds.

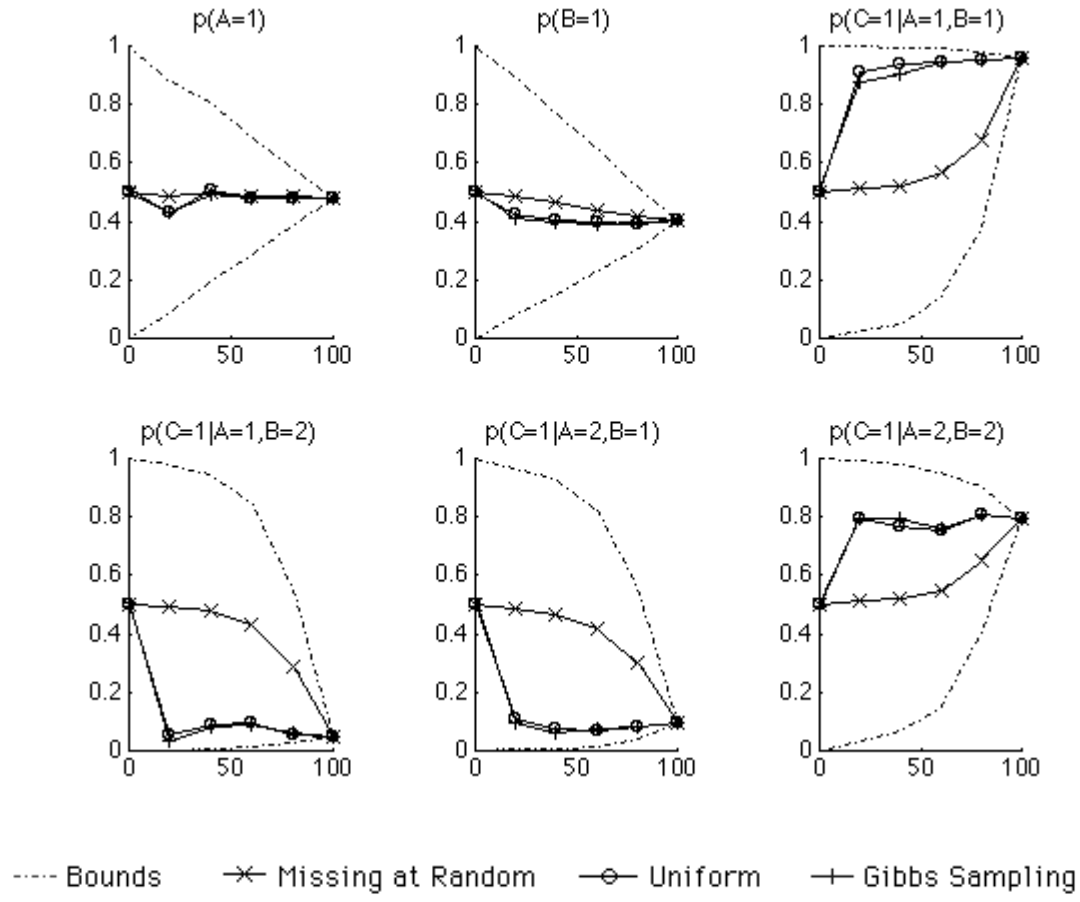

Figure 3: Parameter estimates against percentage of entries in the database. Entries are randomly missing.

4.2.2 Randomly Missing Data

The procedure followed during the second experiment is identical to the procedure used in the previous experiment, except that, at each step, the 20% of the data are randomly deleted from the database. Not surprisingly BC under the MAR assumption returns the now accurate estimates of the Gibbs Sampling, while the use of a uniform distribution over the missing data leads to a slower convergence toward the correct value. Again, the difference in computation time was significant: while our implementation of BC took always less than 20 milliseconds to complete each analysis, the Gibbs Sampler took up to 8 minutes to run to completion.

4.3 Discussion

The first important result of our experiment is the overall equivalence between the Gibbs Sampling and the BC estimates, which are almost identical when the MAR assumption is fulfilled by the database. When this assumption is violated, BC seems to be significantly

more robust than the Gibbs Sampling: the little gain in the estimates of $p(B = 1)$ is a small reward for the appalling error in the estimates of $p(C = 1|A = 1, B = 2)$ and $p(C = 1|A = 2, B = 2)$. The strategy of guessing the missing data from the available information leads the Gibbs Sampler to ascribe most of the missing entries to $B = 2$, so that the final estimates of $p(C = 1|A = 1, B = 2)$ and $p(C = 1|A = 2, B = 2)$ are based on a inflated sample. On the other hand, the BC estimates computed under the MAR assumption are based on weights which depend only on the complete sample.

The second point is the significant difference in computational cost: milliseconds against minutes. Furthermore, since BC is deterministic in nature, it provides a sound stopping rule, while, since the Gibbs Sampling is guaranteed to converge only asymptotically, heuristic methods are needed to assess whether the result can be considered satisfactory or not.

5. Conclusion

BC provides a method to learn conditional probabilities in a BBN from incomplete databases. Contrary to the current methods, BC does not try to complete the database before learning, but it computes the extreme points of the possible distributions consistent with the database at hand and then collapses them into a point-valued estimate using the available information about the pattern of missing data. This feature allows the encoding of knowledge external to the database about the pattern of missing data. BC subsumes the methods using MAR assumption as special cases and it is able to properly encode this assumption, providing performances equal or even higher in accuracy than the ones provided by the Gibbs Sampling. Furthermore, the probability intervals used by BC provide a specific measure of the *quality* of probabilistic information conveyed by the database and an explicit representation of the reliability of the estimates.

From a computational standpoint, BC has all the advantages of a deterministic method: it provides a stopping rule and its computational cost is reduced to the cost of two exact updating — one for each extreme distribution — plus the cost of a convex combination for each parameter in the BBN. Moreover, it is able to process the whole database at once.

A final remark is due: although we focused on BBNs, we must note that BC is a general method to learn conditional probabilities and may be applied to other machine learning formalisms based on probability theory, from decision trees to classification systems.

References

[1] W. L. Buntine. Operations for learning with graphical models. *Journal of Artificial Intelligence Research*, 2:159–225, 1994.

[2] G.F. Cooper and E. Herskovitz. A bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9:309–347, 1992.

[3] R G Cowell, A.P. Dawid, and P. Sebastiani. A comparison of sequential learning methods for incomplete data. In *Bayesian Statistics 5*, pages 533–542. Clarendon Press, Oxford, 1996.

[4] A. Dempster, D. Laird, and Rubin D. Maximum likelihood from incomplete data via the em algorithm. *J. Roy. Statist. Soc. B*, 39:1–38, 1977.

[5] D. Heckerman. A tutorial on learning Bayesian networks. Technical Report MSR-TR-95-06, Microsoft Research, Microsoft Corporation, 1995.

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

[7] D.J. Spiegelhalter and S.L. Lauritzen. Sequential updating of conditional probabilities on directed graphical structures. *Networks*, 20:157–224, 1990.

[8] A Thomas, D J Spiegelhalter, and W R Gilks. Bugs: A program to perform bayesian inference using gibbs sampling. In *Bayesian Statistics 4*, pages 837–42. Clarendon Press, Oxford, 1992.