An interactive approach for cleaning noisy observations in Bayesian networks with the help of an expert

Andrés R. Masegosa and Serafín Moral
{andrew,smc}@decsai.ugr.es
Department of Computer Science and Artificial Intelligence
University of Granada

Abstract

When using Bayesian networks in real applications it is often the case that the empirical evidence or observations we employ for making inferences are corrupt and contain noise: Failure in a sensor, outliers, human errors, etc. Although many methods have been proposed in the literature for data cleaning (i.e. detect and correct noisy data values), all of these methods perform this task automatically. In this paper we argue that, if available, expert knowledge should be used for this task and we propose two methods which explicitly interact with an expert for detecting and correcting noisy observations.

1 Introduction

A Bayesian network (BN) is a statistical model (Pearl, 1988) that graphically encodes, via a directed acyclic graph, the conditional independencies among the domain variables. The applications of BNs in real problems are very well known and diverse (Pourret et al., 2008): Medicine, recognition, gambling, monitoring, forensic reasoning, genomics, etc.

One of the key properties of these models is the possibility of introducing empirical evidence about any of the variables (e.g. the result of a medical test) and, by means of any of the available evidence propagation algorithms (Jensen and Nielsen, 2007), updating the beliefs about the distribution over the rest of the variables (e.g. the causes of a given disease). However, in real applications, it may happen that the empirical observations or evidence introduced in the model are corrupt and, in consequence, the results of the reasoning provided by these systems may be corrupt too. For example, in a medical diagnosis system relying in a BN model, a doctor may wrongly introduce the result of a test by pressing an incorrect keyboard key and this could cause a wrong disease diagnosis. But the corruption of empirical evidence may be due to many other causes: Failure in a sensor; noise in a transmission channel, outliers, etc. There are many problem domains where the empirical evidence is subject to some kind of corruption process (Zhu et al., 2007).

One of the most studied corruption processes is the presence of noisy values in the empirical observations or data used to make inferences (Zhu and Wu, 2006). For data mining problems, there is a large amount of literature on automatic data cleaning methods (Maletic and Marcus, 2010), and some of these methods were specifically designed for BNs (Doshi et al., 2003).

In this work, we explore an alternative approach to the problem of detecting and correcting noisy observations in multinomial data. We propose new methods which have two main properties. First, we explicitly model the noisy process which determines the value of the noisy observable variables O'_i as a function of the true value of these observations O_i. Secondly, we assume that an expert will be available to clean up noisy observations. Thus, in contrast with the previous approaches, based on automatic methods in the sense that they do not allow human intervention, we propose a method which explicitly interacts with an expert. So, given a BN and an observation vector of some of these vari-

ables (i.e. the variables that can be observed, such as a medical test or a symptom), and given that some of the values can be noisy, our method will try to detect these noisy values and correct them to their actual values. For this purpose, our method will interact with an expert by requesting information about some particular values of evidence (i.e. the expert will have to confirm if an observed value is noisy or not and, if noisy, provide the actual value). Our method will try to ease and minimize this interaction. In the field of natural language processing, similar approaches are quite popular to correct misspelled words when typing on a mobile phone. Instead of correcting automatically a misspelled word, these systems display a set of alternative correct words and let the user decide which is the correct one.

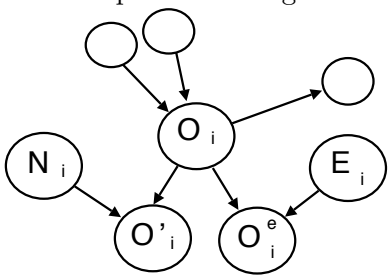
The remainder of this paper is organized as follows. Section 2 details the noisy and the expert model considered in this approach. Section 3 introduces our two proposed methods for cleaning noisy observations with the help of an expert. In Section 4 we experimentally evaluate our approach and, finally, Section 5 outlines the main conclusions and plans for future work.

2 Modelling noisy observations and expert knowledge

Let us assume that we have a problem domain which is defined by two vectors of multinomial random variables: A vector of explanatory variables $\mathbf{X} = \{X_1, ..., X_n\}$ (variables for which we want to make inferences, e.g., the cause of given disease) and a vector of observable variables $\mathbf{O} = \{O_1, ..., O_p\}$ (e.g. the result of a test, the measure of a given sensor, etc.) Let us also assume that the joint probability distribution of these variables in our problem domain, $P(\mathbf{X}, \mathbf{O})$, is modelled using a known Bayesian network (BN), denoted by $\mathcal{B}$.

As mentioned in the introduction, we consider that when gathering observations from the observable variables $\mathbf{O}$ in our problem domain, there are mechanisms which add noise and corrupt some of the observed values. To account for these corruption processes, we introduce new

Figure 1: A expanded BN modelling noisy observations and expert knowledge



variables in the BN. $\mathbf{O'} = \{O'_1, ..., O'_p\}$ denotes a vector of noisy observable variables where O'_i represents a noisy observable variable which can be equal to O_i or not, depending on whether a noisy event (which corrupts the observed value) takes place or not when observing this variable. We assume that the set of possible values of O'_i, $\Omega(O'_i)$, is identical to the set of possible values $\Omega(O_i)$ of O_i. The noisy events are also explicitly modelled by a vector of variables denoted by $\mathbf{N} = \{N_1, ..., N_p\}$ where N_i models the occurrence (or not) of a noisy event when observing the value associated to variable O_i. Thus, each N_i has two values: $\Omega(N_i) = \{Noise, NoNoise\}$. These new sets of variables are related to each other as shown in Figure 1, i.e. we assume that the noisy event variables are independent and that each observed variable O'_i only depends on the true value O_i and the noisy event N_i.

We use the notation $P(N_i = Noise) = \tau_i$, where $\tau_i \in [0, 1]$ but usually is a small probability value. The conditional probability $P(O'_i|O_i, N_i)$ defines the noise model and it will vary depending on the particular problem we are dealing with. In this work we assume a simple noise model defined as follows: $P(O'_i = o_i|O_i = o_i, N_i = noNoise) = 1$, i.e. the identity function; $P(O'_i = o_i|O_i \neq o_i, N_i = Noise) = \frac{1}{|\Omega(O_i)|-1}$ where $|\cdot|$ is the cardinality operator, i.e. when noise is present, the conditional is a uniform distribution over the rest of values which differ from the current value of O_i.

Similarly, we also model the possibility of introducing expert knowledge, as discussed in the introduction, by adding two new sets of variables $\mathbf{O^e} = \{O^e_1, ..., O^e_p\}$ and $\mathbf{E} = \{E_1, ..., E_p\}$.

Variable O_i^e models the knowledge provided by the expert about the real value of variable O_i and variable E_i models the correctness of the expert's knowledge (i.e. the expert may be wrong and provide incorrect knowledge): $\Omega(E_i) = \{Right, Wrong\}$. Figure 1 shows how these variables are connected in the BN used to model the interaction procedure. $P(E_i = Wrong) = \eta_i$, where $\eta_i \in [0, 1]$ and usually is a small probability value. The conditional probability $P(O_i^e|O_i, E_i)$ is defined as follows: $P(O_i^e = o_i|O_i = o_i, E_i = Correct) = 1$, i.e. the identity function; $P(O_i^e = o_i|O_i \neq o_i, E_i = Wrong) = \frac{1}{|\Omega(O_i)|-1}$, i.e. the conditional when the expert is wrong is a uniform distribution over the rest of values which differ from the current value of O_i.

3 Cleaning noisy observations with the help of an expert

Our problem starts with a vector of noisy observations, denoted by $\mathbf{o}' = \{o'_1, ..., o'_p\} \in \Omega(\mathbf{O}')$ and our goal is to recover the vector of actual observations, denoted by $\mathbf{o} \in \Omega(\mathbf{O})$.

In this work we explore two different strategies to approach the problem of deciding which knowledge should be requested to an expert.

3.1 An entropy-based approach

Here we follow a pure inference approach where we do not consider the existence of any specific cost or utility when cleaning noisy observations. With this approach, the output is the most probable explanation (Gamez, 2003), denoted by $\mathbf{o}^{MPE}$, of the observable variables given the noisy observation, $\mathbf{o}'$, and the information provided by the expert:

$$\mathbf{o}^{MPE} = \arg\max_{\mathbf{O}=\mathbf{o}} P(\mathbf{O} = \mathbf{o}|\mathbf{O}' = \mathbf{o}', \mathbf{o}^e)$$

where $\mathbf{o}^e$ denotes the knowledge provided by the expert which is introduced as evidence for some of the variables in $\mathbf{O}^e$ (see Figure 1).

The introduction of expert knowledge is used to discard alternative explanations or assignments which have a non-negligible probability, $P(\mathbf{O} = \mathbf{o}|\mathbf{O}' = \mathbf{o}') > 0$ with $\mathbf{o} \neq \mathbf{o}^{MPE}$, and improve the confidence in the retrieved explanation $\mathbf{o}^{MPE}$. To quantify this confidence we employ the conditional entropy of the observable variables: $H(\mathbf{O}|\mathbf{O}' = \mathbf{o}') = \sum_{\mathbf{o}\in\Omega(\mathbf{O})} P(\mathbf{o}|\mathbf{o}') \ln P(\mathbf{o}|\mathbf{o}')$. A null conditional entropy means that $\mathbf{o}^{MPE}$ accumulates all the mass probability, while a maximum conditional entropy means that all explanations are equally likely. So, our approach is based on requesting knowledge about the real value of the variable O_i which most reduces this entropy measure. That is to say, we iteratively ask the expert for her/his belief about the value of the O^e_{max} variable with the highest information gain or mutual information about the observable variables $\mathbf{O}$:

$$O^e_{max} = \arg\max_{O_i^e} IG(\mathbf{O}; O_i^e|\mathbf{o}', \mathbf{o}^e)$$

where $\mathbf{o}^e$ refers to the set of answers (i.e. evidence for some of the O_i^e variables) given by the expert so far (at the beginning, this set is empty, i.e. there is no expert knowledge).

The computation of $IG(\mathbf{O}; O_i^e|\mathbf{o}', \mathbf{o}^e)$ seems to be complex, as $\mathbf{O}$ is a multidimensional variable, but it can be easily computed if we take into account that O_i^e is independent of $\mathbf{O} - \{O_i\}$ given O_i. So each variable O_i^e only gives information about its associated variable, O_i:

$$IG(\mathbf{O}; O_i^e|\mathbf{o}', \mathbf{o}^e) = H(O_i^e|\mathbf{o}', \mathbf{o}^e) - \sum_{\mathbf{o}\in\Omega(\mathbf{O})} P(\mathbf{o}|\mathbf{o}', \mathbf{o}^e) H(O_i^e|\mathbf{o}', \mathbf{o}^e, \mathbf{o})$$

and $\sum_{\mathbf{o}\in\Omega(\mathbf{O})} P(\mathbf{o}|\mathbf{o}', \mathbf{o}^e) H(O_i^e|\mathbf{o}', \mathbf{o}^e, \mathbf{o}) = \sum_{o_i\in\Omega(O_i)} P(o_i|\mathbf{o}', \mathbf{o}^e) H(O_i^e|o_i)$. Taking into account the probabilities relating O_i with O_i^e, we have that $H(O_i^e|o_i) = -\tau_i \ln \tau_i - (1 - \tau_i) \ln \frac{1-\tau_i}{|\Omega(O_i)|-1} = H(E_i) + (1-\tau_i) \ln(|\Omega(O_i)|-1)$. Putting everything together, we get that $IG(\mathbf{O}; O_i^e|\mathbf{o}', \mathbf{o}^e)$ is equal to:

$$H(O_i^e|\mathbf{o}', \mathbf{o}^e) - H(E_i) + (1 - \tau_i) \ln(|\Omega(O_i)| - 1)$$

which can be computed by standard propagation algorithms.

We stop requesting knowledge when the information given by O^e_{max} is below a given threshold, denoted by λ. This parameter defines the information gain we are willing to accept in exchange for a question to the expert. A pseudocode description is given in Algorithm 1.

Algorithm 1. The entropy based method

1: $\mathbf{o}^e = \emptyset$.
2: end=false;
3: **repeat**
4: Compute the O^e_i variable with the highest information gain:

$$O^e_{max} = arg\max_{O^e_i} IG(\mathbf{O}; O^e_i|\mathbf{o}', \mathbf{o}^e)$$

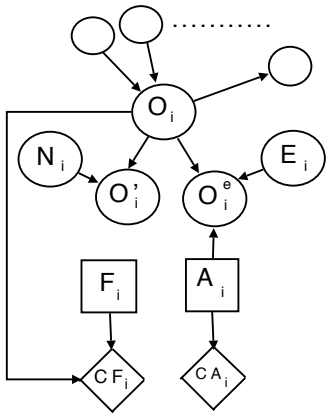
5: **if** $IG(\mathbf{O}; O^e_i|\mathbf{o}', \mathbf{o}^e) > \lambda$ **then**
6: Ask to expert about O^e_{max} and introduce expert's answer: $\mathbf{o}^e = \mathbf{o}^e \cup o^e_{max}$.
7: **else**
8: end=true;
9: **end if**
10: **until** end
11: $\mathbf{o}^{MPE} = \arg\max_{\mathbf{O}=\mathbf{o}} P(\mathbf{O} = \mathbf{o}|\mathbf{O}' = \mathbf{o}', \mathbf{o}^e)$;
12: **return** $\mathbf{o}^{MPE}$;

This algorithm converges and stops the querying process for a positive threshold because the information gain is by definition lower than the entropy of the observable variables $\mathbf{O}$ and this entropy is always reduced with each new answer from the expert. The algorithm also involves the computation of the most probable explanation (line 11). Although this task can be intractable for some networks, many efficient approximate methods have been proposed in the literature (Jensen and Nielsen, 2007).

3.2 A cost-based approach

In this new approach we explicitly consider that there are costs or utilities associated with this problem. We assume that there is a specific and known cost, denoted by CF, which depends on the differences between the actual values of the observed variables $\mathbf{o}$ and the values assigned for the system, and that there is also a specific and known cost, denoted by CA_i, when we ask the expert about the noisy observable variable O_i. So, this problem can be approached as a decision making problem where we are going to have observations in variables $\mathbf{O'}$ and, based on these observations, we have to make the following decisions: If we request expert knowledge about a noisy observable variable O'_i, this decision is denoted by A_i and has two states, $\{Ask, NotAsk\}$; and the decision regarding the necessity of correcting this noisy variable, O'_i, and the way of doing, is denoted by F_i and has the same set of states of O_i (i.e when $F_i = O'_i$, we are deciding that there is no noise in this observation, otherwise F_i takes the corrected value). When the F_i decisions are taken, all the noisy observable variables are fixed.

Figure 2: The Influence Diagram employed in the cost-based approach of Section 3.2



Although the above decision making problem can be represented by a decision graph or an influence diagram, solving this problem (i.e. finding an optimal policy) is not computationally feasible if the number of observable variables p is relatively large, for two reasons: There is no sequential order in the decisions (except that decisions about fixing are made after decisions about asking), i.e. we have an unconstrained influence diagram (Jensen and Nielsen, 2007) with exponential complexity in time; and each decision depends on the information of $2 \cdot p$ variables (i.e. $\mathbf{O'} \cup \mathbf{O}^e$). Thus, the size of the decision node tables is exponential in p.

In order to make feasible the solution of this decision problem for a relatively large number of noisy observable variables, we propose the following simplifications: (i) the decision making problem is defined and solved only for a partic-

ular noisy observation vector $\mathbf{o}'$ (i.e. we assume that there are no alternative noisy observations besides the current one); (ii) the overall cost of incorrectly fixing a noisy observation vector, $\mathbf{o}'$, can be computed as a sum of independent costs for each variable, $CF = \sum_{i=1}^{p} CF_i$; (iii) we assume that we have p different decision problems. Each of them, $\mathcal{D}_i$, is the problem obtained by considering only decision A_i and all the fixing-related decision variables $\{F_1, ..., F_p\}$. When solving $\mathcal{D}_i$ we assume that we do not ask for the rest of the variables: $A_j = NotAsk$ $(j \neq i)$. To implement this last simplification we initially set the evidence in the variables O_j^e with $j \neq i$ to value "NA", which means that we do not have yet any answer from the expert regarding variable O_j^e. Thus, we extend the set values of O_i^e, $\Omega(O_i^e) = \Omega(O_i) \cup \{NA\}$, and redefine the conditional probability of O_i^e, which now depends on the decision A_i. This conditional probability is equally defined as detailed in Section 2 when $A_i = Ask$, $P(O_i^e = NA|O_i, E_i, A_i = Ask) = 0$, but when $A_i = NotAsk$ we define that $P(O_i^e = NA|O_i, E_i, A_i = NotAsk) = 1$.

Under these simplifications, our decision making problem can be represented by means of the unconstrained influence diagram depicted in Figure 2, which extends our previous BN model detailed in Section 2. This problem is then solved by greedily selecting the sequence of variables O_i for which we ask the expert (i.e. the order in which the decision problems $\mathcal{D}_i$ are solved) and, after that, by choosing the optimal set of fixing decisions F_i. Each time we decide to ask the expert about a variable O_i, the obtained value $O_i^e = o_i^e$ is added to the current set of observations $\mathbf{O}^e = \mathbf{o}^e$. The following decision about asking is made after conditioning each one of the original problems $\mathcal{D}_i$ to $\mathbf{O}^e = \mathbf{o}^e$, $\mathbf{O}' = \mathbf{o}'$ and only for these concrete values. If we have already asked the expert about variable O_i, then in subsequent problems it is not necessary to solve $\mathcal{D}_i$, and in all problems $\mathcal{D}_j$ $(j \neq i)$ we assume that $A_i = Ask$.

When solving a problem $\mathcal{D}_i$ we evaluate the influence diagram for $A_i = Ask$ (problem $\mathcal{D}_i^a$) and for $A_i = NotAsk$ (problem $\mathcal{D}_i^n$). To evaluate these two problems we fix the decision variable to the corresponding value, propagate the information, and optimize the decision variables about fixing $(F_1, \ldots, F_q)$. In that step we have to take into account that the optimal values of these variables do not depend on the order in which we decide to fix them, as each time we fix a variable $F_j = f_j^\star$, this value does not change the probabilities of any other chance variable in the problem. Also, the selection can be independently computed for any of the variables F_j. In problem $\mathcal{D}_i^n$, we must compute for each variable F_i the value $f_j^\star$ such that

$$f_j^\star = arg\min_{f_j} \sum_{o_j} CF_j(f_j, o_j) P(o_j|\mathbf{o}', \mathbf{o}^e)$$

where the above minimization problem can be solved in constant time, $O(|\Omega(O_i)|)$ after the observations have been propagated. For example, if we have a 0/1 cost error (i.e. $CF(f_i, o_i) = 1$ if $f_i \neq o_i$ and $CF(f_i, o_i) = 0$ if $f_i = o_i$), the above problem reduces to selecting the o_i with the highest probability.

Let us denote by $CF_j(\mathbf{o}', \mathbf{o}^e)$ the value $\sum_{o_j} CF_j(f_j^\star, o_j) P(o_j|\mathbf{o}', \mathbf{o}^e)$. The cost of problem $\mathcal{D}_i^n$, $c(\mathcal{D}_i^n)$, will be equal to $\sum_j CF_j(\mathbf{o}', \mathbf{o}^e)$.

The cost of problem $\mathcal{D}_i^a$ can be computed in a similar way. The only differences are that now we have to add the cost of asking for a variable CA_i and that each time we fix the value of variable O_j (we compute $f_j^\star$), we will have an additional observation $O_i^e = o_i^e$. If we denote by $CF_j(\mathbf{o}', \mathbf{o}^e \cup o_i^e)$ the optimal cost of fixing variable O_j with these observations, the cost of the problem will be $c(\mathcal{D}_i^a) = \sum_{o_i^e} P(o_i^e|\mathbf{o}', \mathbf{o}^e)(\sum_j CF_j(\mathbf{o}', \mathbf{o}^e \cup o_i^e)) + CA_i$.

To select the variable O_i for which we are going to ask the expert, we compute the differences $c(\mathcal{D}_i^a) - c(\mathcal{D}_i^a)$. This value is denoted by $CG(A_i|\mathbf{o}', \mathbf{o}^e)$ and called the expected cost gain if we ask for variable O_i. We decide to ask for the variable O_i with a greatest cost gain $CG(A_i|\mathbf{o}', \mathbf{o}^e)$. The observation provided by the expert $O_i^e = o_i^e$ is added to $\mathbf{o}^e$ and the process is repeated. The stop condition is that this greater cost gain value is lower or equal than 0 (we experiment a loss when asking for any of the variables). So, this method resembles the

one detailed in Algorithm 1 but using the cost gain instead of the information gain and, finally, retrieving $\mathbf{f}^\star$ instead of the most probable explanation.

Note that value $c(\mathcal{D}_i^a)$ is the same for all the decision problems $\mathcal{D}_i$ and should be computed only once.

3.3 Using the EM algorithm to estimate the noise rate

In the two previously described methods we assumed that the Bayesian network $\mathcal{B}$ relating explanatory and observable variables, $(\mathbf{X}, \mathbf{O})$ and the conditional probabilities of variables $\mathbf{O}'$ are known. For this procedure we are going to assume that the prior probabilities for noisy observations $P(N_i = Noise) = \tau_i$ are not known; instead, we are given a pool of noisy observations, $D = \{\mathbf{o}'_{(1)}, ..., \mathbf{o}'_{(M)}\}$ where we can estimate them. In this section we describe how these unknown parameters $\tau = (\tau_1, ..., \tau_p)$ can be estimated by means of the EM algorithm (Dempster et al., 1977) (once estimated, the previously detailed cleaning approaches can be used for data cleaning as usual). Using this algorithm, we get the maximum a posteriori (MAP) estimate of the parameters: $\hat{\tau} = \max_\tau L(D;\tau)$, where $L(D;\tau)$ is a function on τ proportional to the posterior probability which is computed as follows:

$$L(D;\tau) = P(\tau) \prod_m \sum_{\mathbf{N},\mathbf{O}} P(\mathbf{o}'_{(m)}, \mathbf{N}, \mathbf{O}|\tau)$$

where $P(\tau)$ is the prior over the parameters. In this work, it is defined as follows: $P(\tau) = \prod_i P(\tau_i) = \prod_i B(\tau_i; 0.1, 1.9)$, where $B(\cdot;\cdot)$ is the Beta distribution. In this Beta distribution we assume that the prior belief about the parameter τ (the error rate) is $1/20 = 0.05$.

The EM algorithm is an iterative method which converges to the MAP estimate, $\hat{\tau}$, by alternating the two following steps:

An **Expectation (E) step**, which computes the expectation of the log-likelihood evaluated using the current estimate for the parameters in step k , $\tau^{(k)}$. This is implemented here by computing $P(N_i = noise|\mathbf{o}'_{(m)}, \tau^{<k>})$ for each of the noisy observation vectors in D.

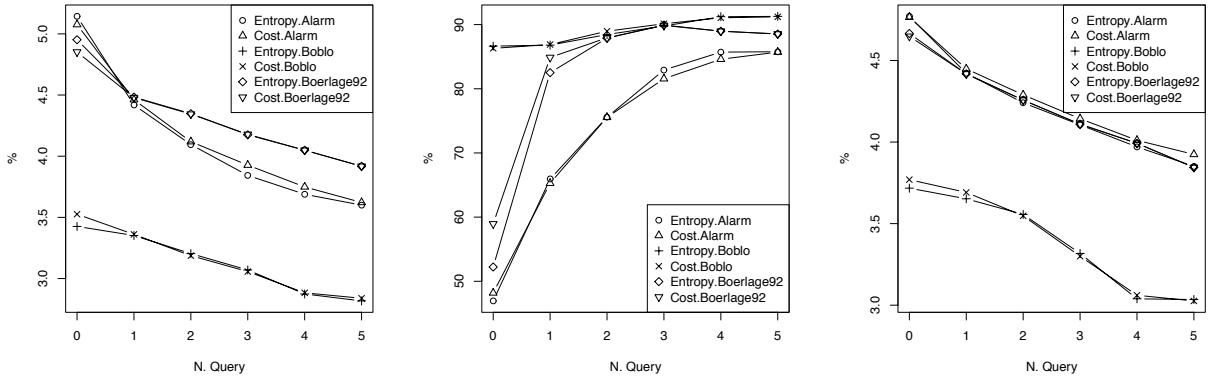
A **Maximization (M) step**, which updates the parameters maximizing the expected log-likelihood found on step E. This is implemented in this problem as follows: $\tau_i^{<k+1>} = \frac{1}{M}\sum_{m=1}^{M} P(N_i = noise|\mathbf{o}'_{(m)}; \tau^{<k>})$.

The algorithm is run until convergence: $\sum_i |\tau_i^{<k+1>} - \tau_i^{<k>}| < 1e-5$.

4 Experimental evaluation

In this experimental evaluation we assess the techniques presented in Section 3. For that purpose, we consider three different BNs (the Alarm network with 37 nodes, the Boblo network with 23 nodes, and the Boerlage network, with 23 nodes too) and, by means of logic sampling, we artificially generate different random samples $(\mathbf{o}, \mathbf{o}', \mathbf{n}, \mathbf{o}^e)$ of the variables in $\mathbf{O}$, $\mathbf{O}'$, $\mathbf{N}$ and $\mathbf{O}^e$, extending previously each model according to Figure 1. For each noisy observation generated, $\mathbf{o}'$, we then apply the two methods detailed in Section 3.1 and 3.2, in order to evaluate their capacity to detect and correct the noisy observations (i.e. those o'_i such that $n_i = Noise$). The presence of expert knowledge is simulated by accessing the o_i^e values, considering also the possibility that the expert gives a wrong answer. Finally, once the cleaning method has finished we can check how well this method works by comparing the cleaned observations with the actual ones, $\mathbf{o}$. Two different error measures are considered to evaluate the performance of the methods: the *a posteriori noise rate* as the rate of values in $\mathbf{o}'$ which are still noisy (i.e. $o_i^{MPE} \neq o_i$ in the entropy-based approach or $f_i^\star \neq o_i$ in the cost-based approach) after the interaction with the expert; and *the precision of the fixing decisions* as the ratio between the number of correctly identified noisy values (those o'_i such that $n_i = Noise$ and which are correctly cleaned) over the total number of cleaned observations (i.e. those values in $\mathbf{o}'$ which are either correctly or incorrectly cleaned because they have been identified as noisy observations by our methods). With this second measure we aim to evaluate if our meth-

Figure 3: Evaluation of the entropy and the cost based methods for different fixed number of queries and for data samples where 5% of the observations are noisy (i.e. when the noise rate is below this value, it means that some noisy observations have been correctly fixed).



(a) Noise Rate (b) Precision (c) Noise Rate - EM

ods fix only those observations which are really noisy, and do not incorrectly fix those which are not noisy.

In this evaluation we consider all the variables in the three BNs mentioned above as observable variables (i.e. $\mathbf{X} = \emptyset$). We also set a noise rate equal to 5% for all the variables (i.e. $\forall i$ $\tau_i = 0.05$). We have evaluated different noise rates and the conclusions are quite similar. The reliability of the expert is set to 99% (i.e. $\forall i$ $\eta_i = 0.01$). For the cost-based approach, we have considered a standard 0/1 cost error.

The results of this evaluation are displayed in Figures 3 and the displayed error measures of the two cleaning methods (labeled as *Entropy-<BN>* and *Cost-<BN>*) are computed as an average value over 1000 observation vectors sampled for the three BNs. Although the two proposed methods consider different stop criteria which depend on the particular problem, we have only evaluated the performance of the methods using a preset number of queries which is displayed in the X-axis of the figures. As can be seen, we have considered from 1 to 5 queries, but also the case where there are no queries submitted to the expert, *N.Query=0*. In this last case, we have a fully automatic method without expert intervention.

In Figure 3 (a), we can firstly see that for Boerlage92 and Alarm networks there is hardly any improvement in the noise rate using the methods that do not require help from the expert (N.Query = 0), i.e. simply cleaning the noisy observations using the most probable explanation in the case of the entropy based method (see Section 3.1) or using the fixing decisions with the lowest expected cost in the case of the cost-based approach (see Section 3.2). Actually, in the case of the Alarm network there is a slight increment in the noise rate (5.14% for the entropy method and 5.08% for the cost method). This is due, as can be seen in Figure 3 (b), to the *precision* of the fixing decisions, which is very low at N.Query=0. This contrasts with the Boblo network, where the precision is high and the noise rate is reduced. In conclusion, we can see that, although there are models where automatic cleaning methods work, there are other models where it can be very hard to successfully clean the noisy observations without the help of an expert, even having perfect knowledge of the underlying probability distribution.

In any case, we can see that the noise rates of the observations decrease with our methods, as more queries are submitted to the expert. The strength and the pace of this decrease strongly depend on the particular model. The same trend appears for the precision of noisy values detection. The introduction of expert knowl-

edge increases the precision of noisy observations detection. This is also remarkable because there is a significant decrease in the number of false positive detections (i.e observations incorrectly considered as noisy).

On the other hand, we can see that the entropy-based and the cost-based methods perform quite similarly in these models, although we stress that the latter method can be tailored to situations where there are asymmetric costs when cleaning noisy observations. It also allows to explicitly consider the cost of requesting knowledge to an expert and, thus, it clearly states when to stop asking questions.

Finally, in Figure 3 (c), we show the results of applying our methods, but this time using the EM algorithm as a previous step (see Section 3.3) to estimate the noise rates of the noisy observable variables. These estimates are obtained using the same pool of 1000 samples used in the previous evaluation. As can be seen in this figure, the overall result is quite similar to the case where the noise rates are known. We cannot give details about the estimations of noise rates τ_i obtained by using EM due to lack of space, but they are accurate and tend to be below the true value. This happens because some noisy observations cannot be identified and the EM considers them as not noisy. We then show that a large set of noisy observations can be employed to obtain estimates of the noise rates and then apply successfully our proposed methods to integrate the expert knowledge and fix many of the noisy observations.

5 Conclusions and future works

In this work we have presented two approaches for cleaning noisy observations using expert knowledge. As opposed to fully automatic methods, our proposals interact with an expert and request knowledge about particular noisy observations. Our empirical evaluation shows that the inclusion of expert knowledge correctly fixes many noisy values. We also show that, in some cases, this can not be achieved without the help of an expert.

Future work will be focused on the extension of these methods to scenarios where the Bayesian network that models the joint probability distribution is not known and must be learned from a pool of noisy data samples.

Acknowledgements

This work has been jointly supported by the research programme Consolider Ingenio 2010, the Spanish Ministerio de Ciencia de Innovación and the Consejería de Innovación, Ciencia y Empresa de la Junta de Andalucía under projects CSD2007-00018, TIN2010-20900-C04-01, TIC-6016 and P08-TIC-03717, respectively. We also thank the reviewers for their constructive suggestions.

References

A. P. Dempster, N. M. Laird, and D. B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1):1–38.

Prashant Doshi, Lloyd Greenwald, and John R. Clarke. 2003. Using Bayesian networks for cleansing trauma data. In Ingrid Russell and Susan M. Haller, editors, *FLAIRS Conference*, pages 72–76. AAAI Press.

J.A. Gamez. 2003. Abductive inference in Bayesian networks: A review. Technical report, Department of Computer Science, University of Castilla-La Mancha, UCLM.

Finn V. Jensen and Thomas D. Nielsen. 2007. *Bayesian Networks and Decision Graphs*. Information Science and Statistics. Springer-Verlag, New York, USA.

Jonathan I. Maletic and Andrian Marcus. 2010. Data cleansing: A prelude to knowledge discovery. pages 19–32.

J. Pearl. 1988. *Probabilistic Reasoning with Intelligent Systems*. Morgan & Kaufman, San Mateo.

Olivier Pourret, Bruce Marcot, and Patrick Naim. 2008. *Bayesian networks: a practical guide to applications*. Statistics in Practice. Wiley, Chichester.

Xingquan Zhu and Xindong Wu. 2006. Error awareness data mining. In *Granular Computing, 2006 IEEE International Conference on*, pages 269 – 274, may.

Xingquan Zhu, Taghi M. Khoshgoftaar, Ian Davidson, and Shichao Zhang. 2007. Editorial: Special issue on mining low-quality data. *Knowl. Inf. Syst.*, 11(2):131–136, February.