Structural Learning of Bayesian Networks with Mixtures of Truncated Exponentials

Vanessa Romero, Rafael Rumí and Antonio Salmerón
Departament of Statistics and Applied Mathematics
University of Almería, Spain
{avrofe,rrumi,Antonio.Salmeron}@ual.es

Abstract

In this paper we introduce a hill-climbing algorithm for structural learning of Bayesian networks from databases with discrete and continuous variables. The process is based on the optimisation of a metric that measures the accuracy of a network penalised by its complexity. The result of the algorithm is a network where the conditional distribution for each variable is a mixture of truncated exponentials (MTE), so that no restrictions on the network topology are imposed. The performance of the proposed method is tested using artificial and real world data.

1 Introduction

Mixtures of truncated exponentials, abbreviated as MTE, were introduced as a model for dealing with discrete and continuous variables simultaneously in Bayesian networks without imposing any restriction on the network topology and avoiding the rough approximations of methods based on the discretisation of the continuous variables (Moral et al., 2001). The ability of MTEs for fitting several common probability models has been widely studied in the last two years (Cobb and Shenoy, 2003; Cobb et al., 2004).

The problem of learning Bayesian networks with MTEs can be structured into three tasks: learning the structure of the network, estimating the marginal distributions for the root nodes (univariate MTEs) and obtaining the conditional distributions for non-root nodes (conditional MTEs). There are methods for learning univariate (Moral et al., 2002) and conditional MTEs (Moral et al., 2003), but the structural learning has not been solved so far.

The paper is organised as follows. The necessary concepts relative to the MTE distribution are reviewed in section 2. Section 3 is devoted to introduce the proposed algorithm for structural learning. The performance of the method is experimentally tested as reported in section 4 and the paper ends with conclusions in section 5.

2 The MTE model

Throughout this paper, random variables will be denoted by capital letters, and their values by lowercase letters. Boldfaced characters will be used for random vectors. The domain of the vector $\mathbf{X}$ is denoted by $\Omega_{\mathbf{X}}$. The MTE model is defined by its corresponding potential and density as follows (Moral et al., 2001):

Definition 1 (MTE potential) *Let* $\mathbf{X}$ *be a mixed* n*-dimensional random vector. We say that a function* $f : \Omega_{\mathbf{X}} \mapsto \mathbb{R}_0^+$ *is a* mixture of truncated exponentials potential (MTE potential) *if one of the next two conditions holds:*

i. f *can be written as*

$$f(\mathbf{x}) = a_0 + \sum_{i=1}^{m} a_i \exp\left\{\sum_{j=1}^{n} b_i^{(j)} x_j\right\} \quad (1)$$

for all $\mathbf{x} \in \Omega_{\mathbf{X}}$*, where* a_i*,* $i = 0, \ldots, m$ *and* $b_i^{(j)}$*,* $i = 1, \ldots, m$*,* $j = 1, \ldots, n$ *are real numbers.*

ii. *There is a partition* $\Omega_1, \ldots, \Omega_k$ *of* $\Omega_{\mathbf{X}}$ *verifying that the domain of the continuous*

variables in $\mathbf{X}$ *is divided into hypercubes and such that* f *is defined as*

$$f(\mathbf{x}) = f_i(\mathbf{x}) \quad \text{if} \quad \mathbf{x} \in \Omega_i \ ,$$

where each f_i, $i = 1, \ldots, k$ *can be written in the form of equation (1).*

Definition 2 (MTE density) *An MTE potential* f *is an* MTE density *if*

$$\sum_{\mathbf{y} \in \Omega_{\mathbf{Y}}} \int_{\Omega_{\mathbf{Z}}} f(\mathbf{y}, \mathbf{z}) d\mathbf{z} = 1 \ , \qquad (2)$$

where $\mathbf{Y}$ *and* $\mathbf{Z}$ *are the discrete and continuous coordinates of* $\mathbf{X}$ *respectively.*

In a Bayesian network, we find two types of densities:

1. For each variable X which is a root of the network, a density $f(x)$ is given.
2. For each variable X with parents $\mathbf{Y}$, a conditional density $f(x|\mathbf{y})$ is given.

A *conditional MTE density* $f(x|\mathbf{y})$ is an MTE potential $f(x, \mathbf{y})$ such that fixing $\mathbf{y}$ to each of its possible values, the resulting function is a density for X.

In (Moral et al., 2001) a data structure was proposed to represent MTE potentials, called *mixed probability trees* or mixed trees for short. Mixed trees can represent MTE potentials defined by parts. Each entire branch in the tree determines one sub-region of the space where the potential is defined, and the function stored in the leaf of a branch is the definition of the potential in the corresponding sub-region. An example of an MTE potential represented as a mixed tree can be seen in figure 1.

The operations required for probability propagation in Bayesian networks (restriction, marginalisation and combination) can be carried out by means of algorithms very similar to those described for discrete probability trees in (Kozlov and Koller, 1997; Salmerón et al., 2000).

3 Structural learning algorithm

Given a mixed random vector $\mathbf{X} = \{X_1, \ldots, X_n\}$, and a sample of $\mathbf{X}$,

$$D = \{\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(m)}\} \ ,$$

our aim is to design a method for obtaining a Bayesian network with variables $\mathbf{X}$, that agrees with the data D.

Basically, the problem of learning Bayesian networks from data can be approached as repeating the next three steps until an optimal network is obtained:

1. Determining a candidate structure G.
2. Estimating the conditional distributions, $\hat{\theta}$, for G.
3. Measuring the quality of $(G, \hat{\theta})$.

Our proposal consists of performing a hill-climbing algorithm with greedy search in order to explore the space of possible networks. The starting point will be a network without arcs. With respect to the movement operators, we have considered arc insertion, deletion and reversal. After each movement, the conditional distributions corresponding to the families involved in the change are estimated. The search process is guided by selecting the operator that best increases the quality of the current network.

3.1 Estimating the conditional distributions for a candidate network

The problem of estimating the parameters of truncated distributions has been previously studied (Smith, 1957; Tukey, 1949), as in the case of the truncated Gamma (Hegde and Dahiya, 1989; Nath, 1975), but the number of parameters is usually one, and the maximum likelihood estimator (that not always exists) or the UMVUE (Uniformly Minimum Variance Unbiased Estimator) is obtained by means of numerical methods (El-Taha and Evans, 1992; Sathe and Varde, 1969). In the case of the MTE models, no similar techniques have been applied

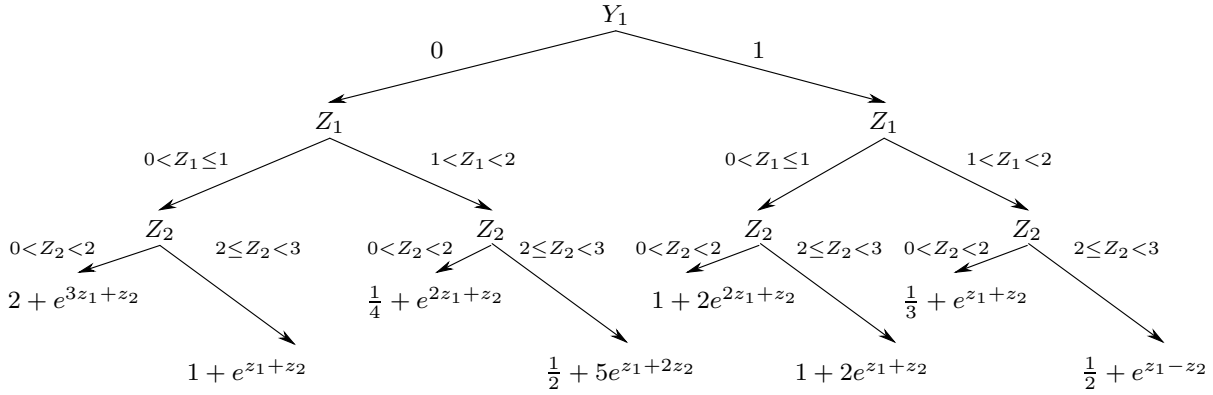

Figure 1: An example of mixed probability tree.

so far, due to the high number of parameters involved in the MTE densities.

Another usual way to compute maximum likelihood estimates in mixture models is the *EM algorithm* (Dempster et al., 1977; Redner and Walker, 1984). The difficulty to apply this method to the MTE models lies in the fact that we may have negative coefficients for some of the densities we are combining and also in the computation of the conditional expectations in each iteration of the algorithm.

Due to the difficulties described above, the seminal paper on estimating MTEs from data (Moral et al., 2002), followed an approach based on regression techniques for the case of univariate densities. Besides the estimation of the parameters, the construction of an MTE density involves the determination of the number of terms and the splits into which its domain is partitioned. Heuristics to approach these issues are proposed in (Moral et al., 2002).

This method for constructing estimators for the parameters of the univariate MTE density is not valid for the conditional case, since more restrictions should be imposed over the parameters in order to force the MTE potential to integrate up to 1 for each combination of values of the conditioning variables, i.e. to force the MTE potential to actually be a conditional density. This problem was approached in (Moral et al., 2003) by partitioning the domain of the conditioning variables and then fitting a univariate density in each one of the splits using the method described above. More precisely, the algorithm learns a mixed tree in which the leaves contain MTE densities that depend only on the child variable, and that represents the density of the child variable given aby of the values contained in the region determined by the corresponding branch of the mixed tree. The tree is learnt in such a way that the leaves discriminate as much as possible, following a schema similar to the construction of classification trees (Quinlan, 1986).

Here we will follow the procedures introduced in (Moral et al., 2002; Moral et al., 2003) to learn the parameters of the candidate networks.

3.2 Measuring the quality of a candidate network

In order to measure the quality of a Bayesian network, we propose to use a metric based on the asymptotic approximation to the classical Bayesian metric proposed in (Castillo et al., 1997) for networks with continuous variables.

The idea is to construct a score that takes into account the likelihood of the data given the candidate network but penalising those ones with complex structure.

We define the following metric:

$$Q(G|D,\hat{\theta}) = \log L(D;G,\hat{\theta}) - \frac{\log m}{2}\mathrm{Dim}(G), \quad (3)$$

where $L(D;G,\hat{\theta})$ is the likelihood of the data given the current network and $\mathrm{Dim}(G)$ is the number of parameters needed to specify the network G. The number of parameters of the conditional density of any variable given its parents is equal to the sum of the number of parameters in the leaves of the corresponding mixed tree. Along this paper, for the sake of simplicity, we will assume that all the MTE potentials that will appear in the learnt network will have a constant number of parameters, say k, which means that the potentials will have the form

$$f(x) = a_0 + a_1 e^{b_1 x} + \cdots + a_t e^{b_t x} \ ,$$

with $k = 2t + 1$.

If we denote by $|X|$ the number of values of X if it is discrete, or the number of splits into which its domain is divided, if X is continuous, the dimension of G can be expressed as

$$\begin{aligned}\mathrm{Dim}(G) &= \sum_{X\in\mathcal{C}}\left(k\prod_{Y\in fa(X)}|Y|\right) + \\ &\quad \sum_{X\in\mathcal{D}}\left(|X|\prod_{Y\in fa(X)}|Y|\right) \ , \quad (4)\end{aligned}$$

where $fa(X_i)$ is the family of X_i, i.e. X_i and its parents, $pa(X_i)$, and $\mathcal{C}$ and $\mathcal{D}$ are the sets of continuous and discrete variables in G respectively.

Thus, the metric in equation (3) can be expressed as

$$\begin{aligned}Q(G|D,\hat{\theta}) &= \log L(D;G,\hat{\theta}) \\ &\quad -\frac{\log m}{2}\mathrm{Dim}(G) \\ &= \sum_{i=1}^{m}\sum_{j=1}^{n}\log p_j(x_j^{(i)}|pa(x_j^{(i)})) \\ &\quad -\frac{\log m}{2}\sum_{X\in\mathcal{D}}\left(k\prod_{Y\in fa(X)}|Y|\right) \\ &\quad -\frac{\log m}{2}\sum_{X\in\mathcal{C}}\left(|X|\prod_{Y\in fa(X)}|Y|\right) \ .\end{aligned}$$

This metric can be decomposed as

$$Q(G|D,\hat{\theta}) = \sum_{j=1}^{n} Q(X_j|D,\hat{\theta}) \ ,$$

where

$$\begin{aligned}Q(X_j|D,\hat{\theta}) &= \sum_{i=1}^{m}\log p_j(x_j^{(i)}|pa(x_j^{(i)})) \\ &\quad -\frac{\log m}{2}\left(k\prod_{Y\in fa(X_j)}|Y|\right)\end{aligned}$$

(or replacing k by $|X|$ if X is discrete), which means that after carrying out a modification over a network, only the part of the metric corresponding to the two variables affected by the operation has to be re-computed.

4 Experimental evaluation

In order to evaluate the performance of the proposed algorithm we have implemented it in Java, and is available in the Elvira system (Elvira Consortium, 2002). We have carried out an experimental evaluation using two artificial base networks. One of them, denoted as net15 has 21 links, 13 continuous and 2 discrete variables. The other one, denoted as net10, contains 12 links, 7 continuous and 3 discrete variables. The structure of both networks has been generated as follows:

Sample size	100	1000	2000	5000
Mean LL	1.48	4.69	6.42	9
Std. Dev.	0.88	1.6	2.02	2.3
Mean CL	0.62	1.16	1.63	2.51
Std. Dev.	0.71	0.96	1.24	1.75
Mean IL	0.65	2.45	3.1	4.07
Std. Dev.	0.73	1.23	1.3	1.6
Mean NL	0.21	1.07	1.69	2.42
Std. Dev.	0.41	0.92	1.3	1.39

Table 1: Results of the experiment for net10.

Sample size	100	1000	2000	5000
Mean LL	0.98	5.08	8.2	11.78
Std. Dev.	0.65	1.73	1.76	2.33
Mean CL	0.38	2.86	4.78	6.87
Std. Dev.	0.63	1.59	2.01	2.18
Mean IL	0.58	2.1	3.14	4.44
Std. Dev.	0.49	1.27	1.66	1.52
Mean NL	0.02	0.12	0.28	0.47
Std. Dev.	0.14	0.35	0.45	0.63

Table 2: Results of the experiment for net15.

- For each variable, the number of parents is selected according to a Poisson distribution with mean 0.8.
- The parents are selected at random, among those that do not violate the DAG condition.

The experiment consisted of 100 iterations of the next procedure:

1. For each iteration, the parameters of net10 and net15 are generated as follows:
   - The number of values of each discrete variable is selected uniformly at random from the set $\{2, 3, 4\}$.
   - The values in the probability tables of the discrete variables are generated from a negative exponential distribution with mean 0.5, and they are normalised afterwards.
   - The number of splits of the domain of each continuous variable is set to 1, 2 or 3 with probability $0.2, 0.4$ and 0.4 respectively.
   - The number of exponential terms for each MTE potential is equal to 0, 1 or 2 with probability $0.05, 0.75$ and 0.2 respectively.
   - The independent term of each MTE potential is generated from a negative exponential distribution with mean 0.01.
   - The coefficients of the exponential terms in the potentials are generated from a negative exponential distribution with mean 1.
   - The coefficients of the exponents in the exponential terms are generated from a standard normal distribution.
2. A sample is generated from the obtained MTE network.
3. Using that sample, a network is learnt using the proposed algorithm.

	CL	IL	NL
liver	0	1	3
abalone	4	4	7
diabetes	1	2	4

Table 3: Comparison of the learnt networks vs. K2 for real world data.

4. For the learnt network, we record the number of links (L), number of coincident links (CL), number of inverted links (IL) and number of new links (NL), i.e. those not coincident nor inverted.

In the experiments, the value for k in equation (4) has been set to 5. It means that in each leaf of the mixed tree corresponding to a conditional distribution, the fitted MTE potential has 5 parameters:

$$f(x) = a_0 + a_1 e^{b_1 x} + a_2 e^{b_2 x} \ .$$

Furthermore, the number of splits into which the domain of the variables is split is set to 3. The result of the experiment for net10 and net15 can be found in tables 1 and 2 respectively, where the mean and standard deviation (Std. Dev.) of the values of L, CL,IL and NL for the 100 iterations of the experiment are displayed.

The results of the experiments suggest that the algorithm increases its accuracy in terms of similarity to the original structure, as the sample size grows. However, the increase is rather slow. For instance, the number of arcs in both networks for samples of size 5000 is far away to the number of links in the original networks. It means that the risk of including false independencies in the learnt model is high. With respect to the inverted links, some of them may introduce new independencies, but also, those ones that do not change the independencies in the original model can change the distribution of the learnt model, due to the heuristic nature of the parametric estimation employed in this work.

We have also tested the algorithm using three real-world databases taken from the UCI Machine Learning Repository (Blake and Merz, 1998). The description of these databases is as follows:

- liver: Liver-disorders Database with 7 continuous variables and 345 instances.
- abalone: Database for predicting the age of abalone from physical measurements. It contains 8 variables (7 continuous and 1 discrete), and 4177 instances.
- diabetes: Pima Indians Diabetes Database, with 8 continuous variables and a binary class variable. It contains 768 instances.

In this case, there are no original networks to compare with, so we have tested them versus the K2 algorithm (Cooper and Herskovits, 1992) with the continuous variables previously discretised using equal frequency intervals, limiting the number of parents in the learnt network to a maximum of 5. We have found that the learnt structure in the case of database diabetes is a naive Bayes (excluding the disconnected parts), in which the class variable has no parents and the rest of the variables are children of it, which seems to be sensible. The results of the comparison of the leant MTE networks vs. K2 can be found in table 3. The results suggest that for large samples the structures provided by both methods become more similar. However, there are big differences in the case of databases liver and diabetes.

5 Conclusions

We have introduced an algorithm for learning the structure of Bayesian networks with discrete and continuous variables simultaneously, in which the MTE model is used. So far, algorithms for estimating marginal and conditional MTE densities existed (Moral et al., 2002; Moral et al., 2003), but the obtainment of the network structure remained unsolved.

The method proposed here is rather preliminary, but it is nevertheless useful since the

user can obtain a Bayesian network from any kind of mixed data, without worrying about the structural restriction imposed by the Conditional Gaussian model (Lauritzen, 1992), which requires discrete nodes not to have continuous parents.

However, still much effort must be invested in order to reach a satisfactory solution for the structural MTE learning problem. For instance, the metric used in this paper is known to have good properties when the conditional distributions in the network belong to the curved exponential family (Haughton, 1988). The MTE distribution does not belong to this family, and thus the asymptotic properties of the metric should be studied. Furthermore, we think that the logarithm of the likelihood of the data given the candidate network is not the best way to measure the accuracy in the case of density functions. The Kullback-Leibler divergence should be a better choice. At this point, we are studying a method to compute that divergence for the MTE model.

Another aspect that influences the performance of the structural learning algorithm is the estimation of the parameters. The method we have used here is based on regression techniques, which are used to fit a curve expressed as an MTE potential to the empirical histogram obtained from the sample. However, the empirical histogram is usually a bad model for the density of the data. More precisely, it may have many peaks, specially when the sample size is small. Besides, the sample size is actually reduced, since all the points under the same rectangle of the histogram are assigned the same density value. We have found that the estimation of the MTEs can be improved using smoother empirical densities, as the ones provided by Kernel methods (see e.g. Simonoff (1996)). Currently we are implementing the improved estimation procedure using kernels.

With respect to the search scheme, we are planning to used methods that try to avoid reaching local optima, such as the *stochastic variable neighbourhood search* algorithm (de Campos and Puerta, 2001).

Acknowledgments

This paper has been supported by the Spanish Ministry of Science and Technology, through project TIC2001-2973-C05-02, which is partly funded with European Funds for Regional Development from the EU. We want to acknowledge the work of the anonymous reviewers, for their valuable and accurate comments.

References

C.L. Blake and C.J. Merz. 1998. UCI repository of machine learning databases. www.ics.uci.edu/∼mlearn/MLRepository.html. University of California, Irvine, Dept. of Information and Computer Sciences.

E. Castillo, J.M. Gutiérrez, and A.S. Hadi. 1997. *Expert systems and probabilistic network models*. Springer-Verlag, New York.

B. Cobb and P.P. Shenoy. 2003. Inference in hybrid Bayesian networks with mixtures of truncated exponentials. In *Proceedings of 6th Workshop on Uncertainty Processing*, pages 47–63, Hejnice, Czech Republic.

B. Cobb, P.P. Shenoy, and R. Rumí. 2004. Approximating probability density functions with mixtures of truncated exponentials. In *Proceedings of Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU-2004)*, Perugia, Italy. In press.

G.F. Cooper and E. Herskovits. 1992. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9:309–347.

L.M. de Campos and J.M. Puerta. 2001. Stochastic local and distributed search algorithms for learning belief networks. In *Proceedings of the III International Symposium on Adaptive Systems (ISAS): Evolutionary Computation and Probabilistic Graphical Models*, pages 109–115.

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

M. El-Taha and W. Evans. 1992. A new estimation procedure for the right-truncated exponential distribution. In *Proceedings of the 23th Pittsburgh Conference on Modelling and Simulation*, pages 427–434.

Elvira Consortium. 2002. Elvira: An environment for creating and using probabilistic graphical models. In J.A. Gmez and A. Salmern, editors, *Proceedings of the First European Workshop on Probabilistic Graphical Models*, pages 222–230.

D.M.A Haughton. 1988. On the choice of a model to fit data from an exponential family. *Annals of Statistics*, 16:342–355.

L.M. Hegde and R.C. Dahiya. 1989. Estimation of the parameters of a truncated Gamma distribution. *Communications in Statistics. Theory and Methods*, 18:561–577.

D. Kozlov and D. Koller. 1997. Nonuniform dynamic discretization in hybrid networks. In D. Geiger and P.P. Shenoy, editors, *Proceedings of the 13th Conference on Uncertainty in Artificial Intelligence*, pages 302–313. Morgan & Kaufmann.

S.L. Lauritzen. 1992. Propagation of probabilities, means and variances in mixed graphical association models. *Journal of the American Statistical Association*, 87:1098–1108.

S. Moral, R. Rumí, and A. Salmerón. 2001. Mixtures of truncated exponentials in hybrid Bayesian networks. In *Lecture Notes in Artificial Intelligence*, volume 2143, pages 135–143.

S. Moral, R. Rumí, and A. Salmerón. 2002. Estimating mixtures of truncated exponentials from data. In J.A. Gámez and A. Salmerón, editors, *Proceedings of the First European Workshop on Probabilistic Graphical Models*, pages 156–167.

S. Moral, R. Rumí, and A. Salmerón. 2003. Approximating conditional MTE distributions by means of mixed trees. In *Lecture Notes in Artificial Intelligence*, volume 2711, pages 173–183.

G.B. Nath. 1975. Unbiased estimates of reliability for the truncated Gamma distribution. *Scandinavian Actuarial Journal*, (3):181–186.

J.R. Quinlan. 1986. Induction of decision trees. *Machine Learning*, 1:81–106.

R. Redner and H. Walker. 1984. Mixture densities, maximum likelihood and the EM algorithm. *SIAM Review*, 26:195–239.

A. Salmerón, A. Cano, and S. Moral. 2000. Importance sampling in Bayesian networks using probability trees. *Computational Statistics and Data Analysis*, 34:387–413.

Y. Sathe and S. Varde. 1969. Minimum variance unbiased estimates of reliability for the truncated exponential distribution. *Technometrics*, 11:609–612.

J.S. Simonoff. 1996. *Smoothing methods in Statistics*. Springer.

W. Smith. 1957. A note on truncation and sufficient statistics. *Annals of Mathematical Statistics*, 28:247–252.

J. Tukey. 1949. Sufficiency, truncation and selection. *Annals of Mathematical Statistics*, 20:309–311.