Submitted to *Entropy*. Pages 1 - 22.

OPEN ACCESS

Article

Efficient approximation of the conditional relative entropy with applications to discriminative learning of Bayesian network classifiers

Alexandra M. Carvalho[1,4,⋆], **Pedro Adão** [2,5] **and Paulo Mateus** [3,5]

[1] Department of Electrical Engineering, IST, Technical University of Lisbon, Portugal

[2] Department of Computer Science, IST, Technical University of Lisbon, Portugal

[3] Department of Mathematics, IST, Technical University of Lisbon, Portugal

[4] PIA, Instituto de Telecomunicações, 1049-001 Lisbon, Portugal

[5] SQIG, Instituto de Telecomunicações, 1049-001 Lisbon, Portugal

⋆ Author to whom correspondence should be addressed; E-Mail, Tel. and Fax number of the corresponding author.

Version April 2, 2013 submitted to Entropy. *Typeset by LaTeX using class file mdpi.cls*

Abstract: We propose a minimum variance unbiased approximation to the conditional relative entropy, of the distribution induced by the observed frequency estimates, for multi-classification tasks. Such approximation is an extesion of a decomposable scoring criterion, named *approximate conditional log-likelihood* (aCLL), primarily used for discriminative learning of augmented Bayesian network classifiers. Our contribution is twofold: (i) it addresses multi-classification tasks, and not only binary-classification ones; and (ii) it covers broader stochastic assumptions than uniform distribution over the parameters. Specifically, we considered a Dirichlet distribution over the parameters, which experimentally showed to be a very good approximation to CLL. In addition, for Bayesian network classifiers, a closed-form equation is found for the parameters that maximize the scoring criterion.

Keywords: Conditional relative entropy, approximation, discriminative learning, Bayesian network classifiers.

1. Introduction

Bayesian networks [24] are probabilistic graphical models that represent the joint probability distribution of a set of random variables. They encode specific conditional independence properties pertaining to the joint distribution via a *directed acyclic graph* (DAG). To achieve this, each vertex (aka *node*) in the DAG contains a random variable, and edges between them represent the dependencies between the variables. Specifically, given a DAG, a node is conditional independent of its non-descendants given its parents. Besides serving as a representation of a set of independences, the DAG also aids as a skeleton for factorizing a distribution via the chain rule of probability. The chief advantage of Bayesian networks is that they can specify dependencies only when necessary, providing compact representations of complex domains that leads to a significant reduction in the cost of learning and inference.

Bayesian networks have been widely used for classification [16,18,27], being known in this context as *Bayesian network classifiers* (BNC). The use of *generative learning* methods in choosing the Bayesian network structure as been pointed out as the likely cause for their poor performance when compared to much simpler methods [14,16]. In contrast to generative learning, where the goal is to be able to describe (or generate) the entire data, *discriminative learning* focuses on the capacity of a model to discriminate between different classes. To achieve this end generative methods usually maximize the *log-likelihood* (LL), or a score thereof, whereas discriminative methods focus in maximizing the *conditional log-likelihood* (CLL). Unfortunately, maximizing the CLL of a BNC turns out to be computationally much more challenging than maximizing LL. For this reason, the community has resorted to decomposing the learning procedure into generative-discriminative subtasks [17,18,28]. More recently, Carvalho *et al.* proposed a new scoring criterion, called *approximate conditional log-likelihood* (aCLL), for fully-discriminative learning of BNCs, exhibiting good performance both in terms of accuracy and computational cost [5]. The proposed scoring criterion showed to be the *minimum variance unbiased* (MVU) approximation to CLL.

Despite the aCLL good performance, the initial proposal has three significant restrictions. First, it was only devised for binary-classification tasks. Second, it was derived under the assumption of uniform distribution over the parameters. Third, the parameters of the network structure that maximize the score resorted unknown. In this paper we address all these shortcomings, which makes possible to apply aCLL

in a broader setup while maintaining the desired property of a MVU approximation to CLL. In order to solve the first two restrictions, we start by deriving the approximation for multi-classification tasks under much more relaxed stochastic assumptions. In this context, we considered multivariate symmetric distributions over the parameters, and detailed the pertinent cases of uniform and Dirichlet distributions. The constants required by the approximation are computed analytically for binary and ternary uniform distributions. In addition, a Monte Carlo method is proposed to compute these constants numerically for other distributions, including the Dirichlet. In addressing the third shortcoming, the parameters of the BNC that maximize the proposed approximation to the *conditional log-likelihood* (CLL) are derived. Finally, maximizing CLL is shown to be equivalent to minimizing the conditional relative entropy of the (conditional) distribution (of the class given the other variables) induced by the observed frequency estimates and the one induced by the learned BNC model.

To gauge the performance of the proposed approximation, we conducted a set of experiments over 89 biologically relevant datasets already used in previous works [1,4]. The results show that the models that maximized aCLL under a symmetric Dirichlet assumption attain a higher CLL, with great statistical significance, in comparison with the generative models that maximized LL and the discriminative models that maximized aCLL under a symmetric uniform distribution. In addition, aCLL under a symmetric uniform distribution also significantly outperforms LL in obtaining models with higher CLL.

The paper is organized as follows. In Section 2 we review the essentials of BNCs and revise the aCLL approximation. In Section 3 we present the main contribution of this paper, namely, we extend aCLL to multi-classification tasks under general stochastic assumptions, and derive the parameters that maximize aCLL for BNCs. Additionally, in Section 4 we relate the proposed scoring criterion with the conditional relative entropy, and in Section 5 we provide experimental results. Finally, we draw some conclusions and future work in Section 6.

2. Background

In this section we review basic concepts of BNCs required to understand the proposed methods. We then discuss the difference between generative and discriminative learning of BNCs and present the aCLL scoring criterion [5].

2.1. Bayesian network classifers

Let X be a *discrete random variable* taking values in a countable set $\mathcal{X} \subset \mathbb{R}$. In all that follows, the domain $\mathcal{X}$ is finite. We denote an $(n+1)$-dimensional *random vector* by $\mathbf{X} = (X_1, \ldots, X_n, C)$

where each component X_i is a random variable over $\mathcal{X}_i$ and C is a random variable over $\mathcal{C} = \{1, \ldots, s\}$. The variables $X_1, \ldots, X_n$ are called *attributes*, or *features*, and C is called the *class variable*. For each variable X_i, we denote the elements of $\mathcal{X}_i$ by $x_{i1}, \ldots, x_{ir_i}$ where r_i is the number of values that X_i can take. We say that x_{ik} is the k-th value of X_i, with $k \in \{1, \ldots, r_i\}$. The probability that $\mathbf{X}$ takes value $\mathbf{x}$ is denoted by $P(\mathbf{x})$, conditional probabilities $P(\mathbf{x} \mid \mathbf{z})$ being defined correspondingly.

A *Bayesian network classifier* (BNC) is a triple $B = (\mathbf{X}, G, \Theta)$ where $\mathbf{X} = (X_1, \ldots, X_n, C)$ is a random vector. The network structure $G = (\mathbf{X}, E)$ is a *directed acyclic graph* (DAG) with nodes in $\mathbf{X}$ and edges E representing direct dependencies between the variables. We denote by Π_{X_i} the (possibly empty) set of *parents* of X_i in G. For efficiency purposes it is common to restrict the dependencies between the attributes and the class variable, imposing all attributes to have the class variable as parent; rigorously, these are called *augmented naive Bayes classifiers* but it is common to refer to them abusively as BNCs. In addition, the parents of X_i without the class variable are denoted by $\Pi^*_{X_i} = \Pi_{X_i} \setminus \{C\}$. We denote the number of possible configurations of the parent set $\Pi^*_{X_i}$ by q^*_i. The actual parent configurations are ordered (arbitrarily) and denoted by $w^*_{i1}, \ldots, w^*_{iq^*_i}$ and we say that w^*_{ij} is the j-th configuration of $\Pi^*_{X_i}$, with $j \in \{1, \ldots, q^*_i\}$. Taking into account this notation, the third element of the BNC triple denotes the *parameters* Θ given by the families $\{\theta_{ijck}\}_{i\in\{1\ldots n\},\, j\in\{1,\ldots,q^*_i\},\, c\in\{1,\ldots,s\},k\in\{1,\ldots,r_i\}}$ and $\{\theta_c\}_{c\in\{1,\ldots,s\}}$ that encode the *local distributions* of the network via

$$P_B(C = c) = \theta_c \text{ and} \tag{1}$$

$$P_B(X_i = x_{ik} \mid \Pi^*_{X_i} = w^*_{ij}, C = c) = \theta_{ijck}. \tag{2}$$

A BNC B defines a unique joint probability distribution over $\mathbf{X}$ given by

$$P_B(X_1, \ldots, X_n, C) = P_B(C) \prod_{i=1}^{n} P_B(X_i \mid \Pi_{X_i}). \tag{3}$$

The conditional independence properties pertaining to the joint distribution are essentially determined by the network structure. Specifically, X_i is conditionally independent of its non-descendants given its parents Π_{X_i} in G [24]; and so, C depends on all attributes (as desired).

The problem of learning a BNC given data D consists in finding the BNC that best fits D. This can be achieved by a score-based learning algorithm, where a *scoring criterion* is considered in order to quantify the fitting of a BNC. Contributions in this area of research are typically divided in two different problems: *scoring* and *searching*. The scoring problem focus on devising new scoring criteria to measure the goodness of a certain network structure given the data. On the other hand, the searching problem concentrates on identifying one or more network structures that yield a high value for the scoring

criterion in mind. If the search is conducted with respect to a neighborhood structure defined on the space of possible solutions then we are in the presence of *local score-based learning*.

Local score-based learning algorithms can be extremely efficient if the scoring criterion employed is *decomposable*, that is, if the scoring criterion can be expressed as a sum of local scores associated to each network node and its parents. In this case, any change over the network structure carried out during the search procedure is evaluated by considering only the difference to the score of the previous assessed network. The most common scoring criteria employed in BNC learning are reviewed in [3,19,31]. We refer the interested reader in newly developed scoring criteria to the works of Carvalho *et al.* [5], de Campos [12] and Silander *et al.* [26].

Unfortunately, even performing local score-based learning, searching for unrestricted BNC structures from data is NP-hard [8]. Worse than that, finding for unrestricted approximate solutions is also NP-hard [11]. These results led the community to search for the largest subclass of BNCs for which there is an optimal and efficient learning algorithm. First attempts confined the network to *tree augmented naive* (TAN) structures [16] and used Edmonds [15] and Chow-Liu [9] optimal branching algorithms to learn the network. More general classes of BNCs have eluded efforts to develop optimal and efficient learning algorithms. Indeed, Chickering [6] showed that learning networks constrained to have in-degree at most 2 is already NP-hard.

2.2. Generative versus discriminative learning of Bayesian network classifiers

For convenience, we introduce a few additional notation. Let data D be given by

$$D = \{\mathbf{y}_1, \ldots, \mathbf{y}_N\}, \text{ where } \mathbf{y}_t = (y_t^1, \ldots, y_t^n, c_t).$$

Generative learning reduces to maximizing the likelihood of the data, by using the log-likelihood scoring criterion or a score thereof (for instance, [12,26]). The *log-likelihood* scoring criterion can be written as:

$$\mathrm{LL}(B \mid D) = \sum_{t=1}^{N} \log P_B(y_t^1, \ldots, y_t^n, c_t). \tag{4}$$

On the other hand, *discriminative learning* concerns maximizing the conditional likelihood of the data. The reason why this is a form of discriminative learning is that it focus on correctly discriminating between classes by maximizing the probability of obtaining the correct classification. The *conditional log-likelihood* (CLL) scoring criterion can be written as:

$$\mathrm{CLL}(B \mid D) = \sum_{t=1}^{N} \log P_B(c_t | y_t^1, \ldots, y_t^n). \tag{5}$$

Unlike LL, CLL does not decompose over the network structure, and therefore, there is no closed-form equation for optimal parameter estimates for the CLL scoring criterion. This issue was first approached by spliting the problem into two distinct tasks: find optimal-CLL parameters [17,28] and find optimal-CLL structures [2,18]. Although showing promising results, these approaches still present a problem of computational nature. Indeed, optimal-CLL parameters have been achieved by resorting to gradient descent methods, and optimal-CLL structures have been found only with global search methods, which turn both approaches very inefficient. Recently, a least-squares approximation to CLL, called *approximate conditional log-likelihood* (aCLL), was proposed which enables full discriminative learning of BNCs in a very efficient way [5]. The aCLL scoring criterion is presented in detail in the next section.

2.3. A first approximation to the conditional log-likelihood

In this section we present the approximation to the conditional log-likelihood proposed in [5]. Therein it was assumed that the class variable was binary, that is, $\mathcal{C} = \{0, 1\}$. For the binary case the conditional probability of the class variable can then be written as

$$P_B(c_t \mid y_t^1, \ldots, y_t^n) = \frac{P_B(y_t^1, \ldots, y_t^n, c_t)}{P_B(y_t^1, \ldots, y_t^n, c_t) + P_B(y_t^1, \ldots, y_t^n, 1 - c_t)}. \tag{6}$$

For convenience, the two terms in the denominator were denoted by

$$\begin{aligned} U_t &= P_B(y_t^1, \ldots, y_t^n, c_t) \qquad \text{and} \\ V_t &= P_B(y_t^1, \ldots, y_t^n, 1 - c_t), \end{aligned} \tag{7}$$

so that Eq. (6) becomes simply

$$P_B(c_t \mid y_t^1, \ldots, y_t^n) = \frac{U_t}{U_t + V_t}. \tag{8}$$

The BNC B was omitted from the notation of both U_t and V_t for the sake of readability.

The log-likelihood (LL), and the conditional log-likelihood (CLL) now take the form

$$\mathrm{LL}(B \mid D) = \sum_{t=1}^{N} \log U_t, \qquad \text{and} \tag{9}$$

$$\mathrm{CLL}(B \mid D) = \sum_{t=1}^{N} \log U_t - \log(U_t + V_t). \tag{10}$$

As mentioned in Section 2.1, an efficient scoring criterion must be decomposable. The LL score decomposes over the network structure. To better see why this is the case, substitute the expression for the joint probability distribution, in Eqs. (1)–(3), into the LL criterion, in Eq. (9), to obtain

$$\mathrm{LL}(B \mid D) = \sum_{i=1}^{n} \sum_{j=1}^{q_i} \sum_{k=1}^{r_i} \sum_{c=1}^{s} N_{ijck} \log \theta_{ijck} + \sum_{c=1}^{s} N_c \log \theta_c, \tag{11}$$

where N_{ijck} is the number of instances in D where X_i takes its k-th value, its parents (excluding the class variable) take their j-th value, and C takes its c-th value, and N_c is the number of instances where C takes its c-th value.

Unfortunately, CLL does not decompose over the network structure because $\log(U_t + V_t)$ cannot be expressed as a sum of local contributions. To achieve decomposability of CLL an approximation

$$\hat{f}(U_t, V_t) = \alpha \log U_t + \beta \log V_t + \gamma \tag{12}$$

of the original function

$$f(U_t, V_t) = \log U_t - \log(U_t + V_t)$$

was proposed in [5], where α, β, and γ are real numbers to be chosen so as to minimize the approximation error. To determine suitable values of α, β and γ, uniformity assumptions about U_t and V_t were made. To this end, let $\Delta^2 = \{(x, y) : x + y \leq 1 \text{ and } x, y \geq 0\}$ be the 2-simplex set. As U_t and V_t are expected to become exponentially small as the number of attributes grows, it was assumed that

$$U_t, V_t \leq p < \frac{1}{2}$$

for some $0 < p < \frac{1}{2}$. Combining this constraint with

$$(U_t, V_t) \sim \text{Uniform}(\Delta^2)$$

yielded the following assumption.

Assumption 2.1 *There exists a small positive $p < \frac{1}{2}$ such that*

$$(U_t, V_t) \sim \textit{Uniform}(\Delta^2)|_{U_t, V_t \leq p} = \textit{Uniform}([0, p] \times [0, p]).$$

In [5] it was shown that under Assumption 2.1, the values of α, β and γ that minimize the *mean square error* (MSE) of $\hat{f}$ w.r.t. f are given by:

$$\alpha = \frac{\pi^2 + 6}{24}, \quad \beta = \frac{\pi^2 - 18}{24} \quad \text{and} \quad \gamma = \frac{\pi^2}{12 \ln(2)} - \left(2 + \frac{(\pi^2 - 6) \log p}{12}\right). \tag{13}$$

This resulted in a decomposable approximation for the CLL, called *approximate CLL*, defined as

$$\text{aCLL}(B \mid D) = \sum_{c=0}^{1} (\alpha N_c + \beta N_{1-c}) \log \theta_c + \sum_{i=1}^{n} \sum_{j=1}^{q_i} \sum_{k=1}^{r_i} \sum_{c=0}^{1} (\alpha N_{ijck} + \beta N_{ij(1-c)k}) \log \theta_{ijck} + N\gamma. \tag{14}$$

This decomposable approximation has some desirable properties. It is *unbiased*, that is, the mean difference between $\hat{f}$ and f is zero for all values of p. In addition, $\hat{f}$ is the approximation with the lowest variance amongst unbiased ones, leading to a *minimum variance unbiased* (MVU) approximation

of f. Moreover, since the goal is to maximize $\mathrm{CLL}(B \mid D)$, the constant γ from Eq. (14) can be dropped, yielding an approximation that disregards the value p (used in Assumption 2.1). For this reason, p needs not to be known to maximize aCLL. Despite these advantageous properties, the parameters that maximize aCLL resorted unknown. In addition, the aCLL score is not directly applied to multi-classification tasks and no other stochastic assumptions rather that uniformity of U_t and V_t were studied.

3. Extending the approximation to CLL

The shortcomings of the initial aCLL proposal make it natural to explore a variety of extensions. Specifically, in this section, we derive a general closed-form expression for aCLL grounding in regression theory. This yields a new scoring criterion for multi-classification tasks, with broader stochastic assumptions than the uniform one, while maintaining the desirable property of a MVU approximation to CLL. In addition, we also provide the parameters that maximize this new general form in the context of BNCs.

3.1. Generalizing aCLL to multi-classification tasks

We now set out to consider multi-classification tasks under a generalization of the Assumption 2.1. In this case, let

$$U_{t,c_t} = P_B(y_t^1, \ldots, y_t^n, c_t)$$

so that the conditional probability $P_B(c_t|y_t^1, \ldots, y_t^n)$ in Eq. (8) becomes now

$$P_B(c_t \mid y_t^1, \ldots, y_t^n) = \frac{U_{t,c_t}}{\sum_{c=1}^{s} U_{t,c}},$$

where $U_{t,c} = P_B(y_t^1, \ldots, y_t^n, c)$, for all $1 \leq c \leq s$. Observe that the vectors $(y_t^1, \ldots, y_t^n, c)$, for all $1 \leq c \leq s$ with $c \neq c_t$, called the *complement samples*, may or may not occur in D. Hence, for multi-classification tasks, the conditional log-likelihood in Eq. (10) can be rewritten as

$$\mathrm{CLL}(B \mid D) = \sum_{t=1}^{N} \log U_{t,c_t} - \log \left(\sum_{c=1}^{s} U_{t,c} \right).$$

In this case, the approximation $\hat{f}$ in Eq. (12) consists now in approximating

$$f(U_{t,1}, \ldots, U_{t,s}) = \log U_{t,c_t} - \log \left(\sum_{c=1}^{s} U_{t,c} \right)$$

by a function of the form

$$\hat{f}(U_{t,1}, \ldots, U_{t,s}) = \alpha \log U_{t,c_t} + \sum_{c \neq c_t} \beta_c \log U_{t,c} + \gamma. \tag{15}$$

By taking $\beta_{c_t} = \alpha - 1$, the approximation in Eq. (15) is equivalent to the following linear approximation

$$\begin{aligned} -\log\left(\sum_{c=1}^{s} U_{t,c}\right) &= f(U_{t,1},\ldots,U_{t,s}) - \log U_{t,c_t} \\ &\approx \hat{f}(U_{t,1},\ldots,U_{t,s}) - \log U_{t,c_t} \\ &= (\alpha-1)\log U_{t,c_t} + \sum_{c\neq c_t} \beta_c \log U_{t,c} + \gamma \\ &= \sum_{c=1}^{s} \beta_c \log U_{t,c} + \gamma. \end{aligned} \tag{16}$$

Therefore, we aim at minimizing the expected squared error for the approximation in Eq. (16). We are able to achieve this if we assume that the joint distribution $(U_{t,1},\ldots,U_{t,s})$ is *symmetric*.

Assumption 3.1 *For all permutations* $(\pi_1,\ldots,\pi_s)$ *we have that* $(U_{t,1},\ldots,U_{t,s}) \sim (U_{t,\pi_1},\ldots,U_{t,\pi_s})$.

Assumption 3.1 imposes that $U_{t,c}$ and $U_{t,c'}$ are identically distributed, for all $1 \leq c, c' \leq s$, that is, $U_{t,c} \sim U_{t,c'}$. This is clearly much more general than the symmetric uniformity imposed in Assumption 2.1 for binary classification.

Under Assumption 3.1 the approximation in Eq. (16) is such that $\beta_1 = \cdots = \beta_s$. Let β denote the common value and consider that

$$A = -\log\left(\sum_{c=1}^{s} U_{t,c}\right) \quad \text{and} \quad B = \sum_{c=1}^{s} \log U_{t,c}. \tag{17}$$

Our goal is to find β and γ that minimize the following expected value

$$E[(A - (\beta B + \gamma))^2]. \tag{18}$$

Let σ_A^2 and σ_B^2 denote the variance of A and B, respectively, and let σ_{AB} denote the covariance between A and B. Standard regression allow us to derive the next result [21].

Theorem 3.2 *Assume that* $\sigma_B^2 > 0$. *The unique values of* β *and* γ *that minimize Eq.* (18) *are given by*

$$\beta = \frac{\sigma_{AB}}{\sigma_B^2} \quad \textit{and} \quad \gamma = E[A - \beta B]. \tag{19}$$

Moreover, it follows that

$$E[A - (\beta B + \gamma)] = 0, \quad \textit{and} \tag{20}$$

$$E[(A - (\beta B + \gamma))^2] = \sigma_A^2 - \frac{\sigma_{AB}^2}{\sigma_B^2}. \tag{21}$$

Proof: Let

$$\begin{aligned} h(\beta,\gamma) &= E[(A-(\beta B+\gamma))^2] \\ &= E[A^2]-2\beta E[AB]+\beta^2 E[B^2]-2\gamma E[A]+2\beta\gamma E[B]+\gamma^2. \end{aligned}$$

Then, the solution for $\frac{\partial h}{\partial \gamma}=0$ is $\gamma = E[A]-\beta E[B]$. By replacing γ in h with the solution and solving $\frac{\partial h}{\partial \beta}=0$ on β we get

$$\beta = \frac{E[A]E[B]-E[AB]}{E[B]^2-E[B^2]} = \frac{\sigma_{AB}}{\sigma_B^2}.$$

Moreover, since the Hessian of h is

$$2\begin{pmatrix} E[B^2] & E[B] \\ E[B] & 1 \end{pmatrix},$$

which is positive-definite if $\sigma_B^2 > 0$, we conclude that h has a unique minimum. Finally, Eq. (20) and Eq. (21) are easily derived by plugging in the solution accordingly. □

From Eq. (20) we conclude that the approximation in Eq. (16) is unbiased, and since it minimizes Eq. (18) it is a MVU approximation. As in [5], we are adopting estimation terminology for approximations. In this case, by taking $\hat{A}=(\beta B+\gamma)$, Eq. (18) is precisely $E[(A-\hat{A})^2]=\text{MSE}(\hat{A})$ where MSE is the *mean squared error* of the approximation/estimation. The MSE coincides with the variance when the approximation/estimator is unbiased, and so the approximation in Theorem 3.2 is MVU. In addition, the standard error of the approximation in Eq. (16) is given by square root of Eq. (21).

The previous result allow us to generalize the aCLL scoring criterion for multi-classification tasks. The values of β and γ needed by the approximation in Eq.(16) are given by Eq.(19). This results in a decomposable approximation of CLL, and a generalization of aCLL for multi-classification as

$$\text{aCLL}(B \mid D) = \sum_{t=1}^{N}\left(\log U_{t,c_t} + \sum_{c=1}^{s}\beta \log U_{t,s} + \gamma\right). \tag{22}$$

3.1.1 Symmetric uniform assumption for multi-classification tasks

Herein, we analyze aCLL under the symmetric uniform assumption. We start by providing a general explanation on why aCLL approximation is robust to the choice of p, which was already noticed for the binary case. In addition, we confirm that the analysis in Section 2.3 for the binary case coincides with that given in Section 3.1 when $s=2$. Finally, we provide the constants β and γ for ternary-classification tasks, under the symmetric uniform assumption of $(U_{t,1},U_{t,2},U_{t,3})$.

Assumption 3.3 *Let* $(U_{t,1},\ldots,U_{t,s}) \sim$ *Uniform*$([0,p]^s)$.

Start by noticing that under Assumption 3.3, changes in p correspond to multiplying the random vector $(U_{t,1},\ldots,U_{t,s})$ by a scale factor. Indeed, if each random variable is multiplied by a common scale factor κ it results in the addition of constant values to A and B. To this end note that

$$-\log\left(\sum_{c=1}^{s}\kappa U_{t,c}\right) = -\log\left(\kappa\sum_{c=1}^{s}U_{t,c}\right) = -\log(\kappa) - \log\left(\sum_{c=1}^{s}U_{t,c}\right)$$

and

$$\sum_{c=1}^{s}\log(\kappa U_{t,c}) = \sum_{c=1}^{s}\log\kappa + \log U_{t,c} = s\log(\kappa) + \sum_{c=1}^{s}\log U_{t,c}.$$

Since these additive terms have no effect on variances and covariances, it follows that β is not affected with changes in p. Therefore, the choice of p is irrelevant for maximizing aCLL when a uniform distribution is chosen. Moreover, it is enough to obtain the parameter β as aCLL maximization is insensitive to the constant factor γ.

We also stress out that for the binary case, with $(U_{t,1},U_{t,2}) \sim \text{Uniform}([0,p]^2)$, the values of β and γ given by Eq. (19), with $\alpha = 1+\beta$, coincide with those given by Eq. (13).

Finally, by using *Mathematica 9.0*, we were able to obtain an analytical expression of β for ternary classification tasks.

Example 3.4 *For the ternary case where* $(U_{t,1},U_{t,2},U_{t,3}) \sim$ *Uniform*$([0,p]^3)$ *the constant* β *that minimizes Eq.* (18) *is*

$$\frac{1}{36}\left(-15\pi^2 - 2\left(-11 + 9\ln(3) - 12\ln(2) + 60\ln^2(2) + 72\mathit{Li}_2(-2) + 24\mathit{Li}_2(1/4)\right)\right),$$

where $\mathit{Li}_n(z)$ *is the polylogarithm function of order* n.

3.1.2 Symmetric Dirichlet assumption for multi-classification tasks

In this section we provide an alternative assumption which will lead us to a very good approximation of CLL. Instead of a symmetric uniform distribution, we assume that the random vector $(U_{t,1},\ldots,U_{t,s})$ follows a symmetric Dirichlet distribution. The use of the Dirichlet distribution is attributed to the fact that it is a conjugate family of the distribution for multinomial sample; this ties perfectly with the fact that data is assumed to be a multinomial sample when learning BNCs [20].

In order to take profit of Theorem 3.2, we consider the following symmetric Dirichlet assumption.

Assumption 3.5 *Let* $(U_{t,1},\ldots,U_{t,s}) \sim$ *Dir*$(a_1,\ldots,a_s,b)$ *where* $a_c = a$ *for all* $c = 1,\ldots,s$.

Assumption 3.5 implies that the tuple $(y_t^1,\ldots,y_t^n,c)$ occurs in the data exactly $a-1$ times, for all $c = 1,\ldots,s$. Moreover, there are $b-1$ instances in the data different from $(y_t^1,\ldots,y_t^n,c)$, for all

$c = 1, \ldots, s$. Given a dataset of size N, it is reasonable to assume $a = 1$ and $b = N$. Indeed, probabilities $U_{t,c}$ are expected to be very low, becoming exponentially smaller as the number of attributes n grows; therefore, it is reasonable to assume that $(y_t^1, \ldots, y_t^n, c_t)$ occurs only once in the data and that its complement instances $(y_t^1, \ldots, y_t^n, c)$ with $c \neq c_t$ do not even occur in the data. Thus, by starting with an uninformitive prior, that is, with distribution $\mathrm{Dir}(1, \ldots, 1, 1)$ over the simplex of dimension s, and then condition this distribution over the multinomial observation of the data of size N we obtain the prior

$$(U_{t,1}, \ldots, U_{t,s}) \sim \mathrm{Dir}(a_1, \ldots, a_s, N) \tag{23}$$

where $a_c = 1$ for $c \neq c_t$ and $a_{c_t} = 2$. However, such distribution is asymmetric and it is not in the conditions of Theorem 3.2. We address this problem by considering the symmetric distribution

$$(U_{t,1}, \ldots, U_{t,s}) \sim \mathrm{Dir}(1, \ldots, 1, N) \tag{24}$$

which is very close to the distribution in Eq. (23). We stress that the goal of any assumption is to find good approximations for the constants β and γ and that, the conditions upon which such approximations were performed need not to hold true exactly.

We now focus our attention in finding the values of β and γ numerically, via a Monte-Carlo method. To this end, several random vectors $(u_1, \ldots, u_s)$ are generated with the envisaged Dirichlet distribution in order to define a set $\mathcal{S}$ of pairs (B, A) given by $\mathrm{A} = -\log(\sum_{c=1}^{s} u_c)$ and $\mathrm{B} = \sum_{c=1}^{s} \log(u_c)$. In this way, we are sampling the random variables A and B as defined in Eq. (17). Given $\mathcal{S}$, it is straightforward to find the best linear fit of the form $\mathrm{A} = \hat{\beta}\mathrm{B} + \hat{\gamma}$, and moreover, by the strong law of large numbers $\hat{\beta}$ and $\hat{\gamma}$ converge in probability to β and γ, respectively, as the set $\mathcal{S}$ grows. The general method is described in Algorithm 1.

Algorithm 1 General Monte-Carlo method to estimate β and γ

Input: number of samples m

1. For $i = 1$ to m
2. For $c = 1$ to s
3. Generate a sample u_c=CDF$_X^{-1}$(Uniform([0,1])) where $P(X = x) = P(U_{t,c} = x | U_{t,1} = u_1, \ldots, U_{t,c-1} = u_{c-1})$.
4. Add the point $(\sum_{c=1}^{s} \log(u_c), -\log(\sum_{c=1}^{s} u_c))$ to $\mathcal{S}$.
5. Perform simple linear regression to $\mathcal{S}$ obtaining $\mathrm{A} = \hat{\beta}\mathrm{B} + \hat{\gamma}$.

Algorithm 1 works for any distribution (even for non-symmetric ones). The idea in Step 3 is to use the standard simulation technique of using the *cumulative distribution function* (CDF) of the univariate marginal for $U_{t,1}$ to generate a sample for the first component u_1 of the vector and, afterwards, apply

the CDF of the univariate conditional distributions, to generate samples for the remaining components $u_2, \ldots, u_s$.

We used Algorithm 1 to cross-check the values of β and γ for the case of a symmetric uniform distribution obtained analytically in Eq. (13) and Example 3.4, for binary and ternary classification tasks, respectively.

Example 3.6 *For binary classification under Assumption 3.3 and for $m = 20000$ we obtained an exact approximation of the analytical expression in Eq. (13) up to 4 decimal places. For ternary classification we required 80000 samples to achieve the same precision for the analytical expression given in Example 3.4.*

Although Algorithm 1 works well for Dirichlet distributions, we note that there are simpler and more efficient ways to generate samples for a distribution $\text{Dir}(a_1, \ldots, a_s, a_{s+1})$. Consider $s+1$ independent Gamma random variables $Y_i \sim \text{Gamma}(a_i, 1)$ for $i = 1, \ldots, s+1$ and let $Y_0 = \sum_{i=1}^{s+1} Y_i$. Then, it is well known that $(Y_1/Y_0, \ldots, Y_s/Y_0) \sim \text{Dir}(a_1, \ldots, a_s, a_{s+1})$. Clearly, it is much more simple to sample $s+1$ independent random variables than sampling marginal and conditional distributions. The modified algorithm is presented in Algorithm 2.

Algorithm 2 Monte-Carlo method to estimate β and γ for $\text{Dir}(a, \ldots, a, b)$

Input: number of samples m and hyperparameters a and b

1. For $i = 1$ to m
2. For $c = 1$ to s
3. Generate a sample y_c=$\text{CDF}_X^{-1}(\text{Uniform}([0,1]))$ where $X \sim \text{Gamma}(a, 1)$.
4. Generate a sample $y_{s+1} = \text{CDF}_X^{-1}(\text{Uniform}([0,1]))$ where $X \sim \text{Gamma}(b, 1)$.
5. Set $u_c = \frac{y_c}{\sum_{\ell=1}^{s+1} y_\ell}$ for $c = 1, \ldots, s$.
6. Add the point $(\sum_{c=1}^{s} \log(u_c), -\log(\sum_{c=1}^{s} u_c))$ to $\mathcal{S}$.
7. Perform simple linear regression to $\mathcal{S}$ obtaining $\mathsf{A} = \hat{\beta}\mathsf{B} + \hat{\gamma}$.

Next we illustrate the use of the Algorithm 2 in the conditions discussed in Eq. (24).

Example 3.7 *For binary classification under Assumption 3.5 and with $a_1 = a_2 = 1$, $b = 1000$ and $m = 100000$, the estimated values for β and γ are $\hat{\beta} = -0.39291$ and $\hat{\gamma} = 0.61698$, respectively.*

3.2. Parameter maximization for aCLL

The main goal for devising an approximation to CLL is to obtain an expression that decomposes over the BNC structure. By plugging in the probability expansion of Eq. (3) in the general aCLL scoring criterion for multi-classification given in Eq. (22) we obtain

$$\text{aCLL}(B|D) = \sum_{c=1}^{s} \left(\alpha N_c + \beta \sum_{c' \neq c} N_{c'} \right) \log(\theta_c) + \sum_{i=1}^{n} \sum_{j=1}^{q_i^*} \sum_{k=1}^{r_i} \sum_{c=1}^{s} \left(\alpha N_{ijck} + \beta \sum_{c' \neq c} N_{ijc'k} \right) \log(\theta_{ijck}) + N\gamma, \quad (25)$$

where $\alpha = 1 + \beta$. This score is decomposable as it allows to compute independently the contribution of each node (and its parents) to the global score. However, the values of the parameters θ_{ijck} and θ_c that maximize the aCLL score in Eq. (25) remain unknown. This problem was left open in [5] and a further approximation was required to obtain optimal BNC parameters.

We are able to obtain the optimal values of θ_{ijck} and θ_c by assuming that they are lower-bounded. This lower bound follows naturally by adopting pseudo-counts, commonly used in BNCs to smooth observed frequencies with Dirichlet priors and increase the quality of the classifier [16]. Pseudo-counts attend to impose the common sense assumption that there are no situations with probability zero. Indeed, it is a common mistake to assign probability zero to an event that is extremely unlikely, but not impossible [22].

Theorem 3.8 *Let $N' > 0$ be the number of pseudo-counts. The parameters θ_{ijck} that maximize the aCLL scoring criterion in Eq. (25) are given by*

$$\theta_{ijck} = \frac{N_{ij+ck}}{N_{ij+c}} \text{ and } \theta_c = \frac{N_{+c}}{N_+} \quad (26)$$

where

$$N_{ij+ck} = \begin{cases} \alpha N_{ijck} + \beta \sum_{c' \neq c} N_{ijc'k} & \text{if } \alpha N_{ijck} + \beta \sum_{c' \neq c} N_{ijc'k} \geq N' \\ N' & \text{otherwise,} \end{cases}$$

$$N_{+c} = \begin{cases} \alpha N_c + \beta \sum_{c' \neq c} N_{c'} & \text{if } \alpha N_c + \beta \sum_{c' \neq c} N_{c'} \geq N' \\ N' & \text{otherwise,} \end{cases}$$

$$N_{ij+c} = \sum_{k=1}^{r_i} N_{ij+ck} \text{ and } N_+ = \sum_{c=1}^{s} N_{+c},$$

constrained to $\theta_{ijck} \geq \frac{N'}{N_{ij+c}}$ and $\theta_c \geq \frac{N'}{N_+}$ for all i, j, c and k.

Proof: We only show the maximization for the parameters θ_{ijck}, as the maximization for θ_c is similar. Then, by taking the summand of Eq. (25) that depends only on the parameters θ_{ijck} we have

$$\begin{aligned} &\sum_{i=1}^{n} \sum_{j=1}^{q_i^*} \sum_{k=1}^{r_i} \sum_{c=1}^{s} (\alpha N_{ijck} + \beta \sum_{c' \neq c} N_{ijc'k}) \log(\theta_{ijck}) \\ &= \sum_{i=1}^{n} \sum_{j=1}^{q_i^*} \sum_{k=1}^{r_i} \sum_{c=1}^{s} \underbrace{N_{ij+ck} \log(\theta_{ijck})}_{(a)} + \underbrace{((\alpha N_{ijck} + \beta \sum_{c' \neq c} N_{ijc'k}) - N_{ij+ck}) \log(\theta_{ijck})}_{(b)}. \end{aligned} \quad (27)$$

Observe that if $N_{ij+ck} \geq N'$ then $N_{ij+ck} = \alpha N_{ijck} + \beta \sum_{c' \neq c} N_{ijc'k}$. Thus the summand (b) in Eq. (27) is only different from zero when $\alpha N_{ij+ck} + \beta \sum_{c' \neq c} N_{ijc'k} < N'$. In this case $N_{ij+ck} = N'$ which implies that $(\alpha N_{ijck} + \beta \sum_{c' \neq c} N_{ijc'k}) - N' < 0$. So, the value for θ_{ijck} that maximizes the summand (b) is the minimal value for θ_{ijck}, that is, $\frac{N'}{N_{ij+c}} = \frac{N_{ij+ck}}{N_{ij+c}}$. Finally, by Gibb's inequality, we derive that the distribution for θ_{ijck} that maximizes the summand (a) in Eq. (27) is $\theta_{ijck} = \frac{N_{ij+ck}}{N_{ij+c}}$. Since the maximality of the summands (a) and (b) is obtained with the same values for θ_{ijck}, we have that the values for θ_{ijck} that maximize Eq. (25) are given by $\theta_{ijck} = \frac{N_{ij+ck}}{N_{ij+c}}$. □

The role of the pseudo-counts is to guarantee that the values N_{ij+ck} or N_{+c} cannot be smaller than N'. By plugging in the parameters given in Theorem 3.8 in Eq. (26) into the aCLL criterion in Eq. (25), we obtain

$$\hat{\text{a}}\text{CLL}(G|D) = \sum_{c=1}^{s} \left(\alpha N_c + \beta \sum_{c' \neq c} N_{c'} \right) \log \left(\frac{N_{+c}}{N_+} \right) + \sum_{i=1}^{n} \sum_{j=1}^{q_i^*} \sum_{k=1}^{r_i} \sum_{c=1}^{s} \left(\alpha N_{ijck} + \beta \sum_{c' \neq c} N_{ijc'k} \right) \log \left(\frac{N_{ij+ck}}{N_{ij+c}} \right) + N\gamma \qquad (28)$$

The notation using G as argument instead of B emphasizes that once the parameters Θ are decided upon, the criterion is a function of the network structure G only.

Finally, we show that âCLL is not score equivalent. Two BNCs are said to be *equivalent* if they can represent precisely the same set of distributions. Verma and Pearl [29] showed that this is equivalent to check if the underlying DAG of the two BNCs have the same skeleton and the same v-structures. A *score-equivalent* scoring criterion is one that assigns the same score to equivalent BNC structures [7,12,31].

Theorem 3.9 *The âCLL scoring criterion is decomposable and non-score equivalent.*

Proof: Decomposability follows directly from the definition in Eq. (28). Concerning non-score equivalence, it suffices to provide a counter-example where two equivalent structures do not score the same. To this purpose consider a BNC with two attributes ($n = 2$) and $D = \{(0,0,1),(0,1,1),(1,1,0),(1,1,1)\}$ as the training set. The structures $G \equiv X_1 \rightarrow X_2$ and $H \equiv X_2 \rightarrow X_1$ (we omitted the node representing the class variable C pointing to X_1 and X_2) for B are equivalent, but it is straightforward to check that $\hat{\text{a}}\text{CLL}(G \mid D) \neq \hat{\text{a}}\text{CLL}(H \mid D)$. □

Interesting scoring criteria in the literature are decomposable, since it is unfeasible to learn undecomposable scores. On the other hand, both score-equivalent and non-score-equivalent decomposable scores can be learned efficiently, although the algorithms to learn them are different. In general, the score-equivalence property does not seem to be important, as non-score-equivalent scores typically perform better than score-equivalent ones [12,31].

4. Information-theoretic interpretation of the conditional log-likelihood

Herein, we provide some insights about information-theoretic concepts and how they relate with the CLL. These results are well known in the literature [16,23], nonetheless, they are presented here in detail to provide a clear understanding that approximating the CLL is equivalent to approximating the conditional relative entropy. We refer the interested reader to the textbook of Cover and Thomas [10] for further details.

The relative entropy is a measure of the distance between two distributions. The *relative entropy*, also known as *Kullback-Leibler divergence*, between two probability mass functions $p(x)$ and $q(x)$ for a random variable X is defined as

$$D_{KL}(p(x) \mid\mid q(x)) = \sum_x p(x) \log \frac{p(x)}{q(x)}.$$

This quantity is always non-negative and is zero if and only if $p = q$. Although interpreted as a distance between distributions, it is not a true distance as it is not symmetric and it does not satisfy the triangle inequality. However, it is a measure of the inefficiency of assuming that the distribution is q when the true distribution is p. In terms of encoding, this means that if instead of using the true distribution p to encode X, we used the code for a distribution q, we would need $H_p(X) + D_{KL}(p(x) \mid\mid q(x))$ bits on the average to describe X. In addition, the *conditional relative entropy* is the average of the relative entropies between the conditional probability mass functions $p(y|x)$ and $q(y|x)$ averaged over the probability mass function $p(x)$, that is,

$$D_{KL}(p(y|x) \mid\mid q(y|x)) = E_X[D_{KL}(p(y|X) \mid\mid q(y|X)] = \sum_x p(x) \sum_y p(y|x) \log \frac{p(y|x)}{q(y|x)}.$$

The relationship between log-likelihood and entropy is well established. Lewis had already shown that maximizing the log-likelihood is equivalent to minimizing the entropy [23]. In addition, Friedman *et al.* [16] came into the same conclusion and related it with the relative entropy. They concluded that minimizing the Kullback-Leibler divergence between the distribution $\hat{P}_D$ induced by the *observed frequency estimates* (OFE) and the distribution P_B given by the Bayesian network classifier B is equivalent to minimizing the entropy H_{P_B}, and thus maximizing $\mathrm{LL}(B \mid D)$. In detail we have:

$$\begin{aligned}
\mathrm{LL}(B \mid D) &= \sum_{t=1}^{N} \log P_B(\mathbf{y}_t) \\
&= N \sum_{\mathbf{y}_t} \hat{P}_D(\mathbf{y}_t) \log P_B(\mathbf{y}_t) \\
&= N \sum_{\mathbf{y}_t} \hat{P}_D(\mathbf{y}_t) \log P_B(\mathbf{y}_t) + N H_{\hat{P}_D}(\mathbf{X}) - N H_{\hat{P}_D}(\mathbf{X})
\end{aligned}$$

$$
\begin{aligned}
&= N \sum_{\mathbf{y}_t} \hat{P}_D(\mathbf{y}_t) \log P_B(\mathbf{y}_t) - N \sum_{\mathbf{y}_t} \hat{P}_D(\mathbf{y}_t) \log \hat{P}_D(\mathbf{y}_t) - N H_{\hat{P}_D}(\mathbf{X}) \\
&= -N \sum_{\mathbf{y}_t} \hat{P}_D(\mathbf{y}_t) \log \frac{\hat{P}_D(\mathbf{y}_t)}{P_B(\mathbf{y}_t)} - N H_{\hat{P}_D}(\mathbf{X}) \\
&= -N D_{KL}(\hat{P}_D(\mathbf{X}) \,||\, P_B(\mathbf{X})) - N H_{\hat{P}_D}(\mathbf{X}). \qquad (29)
\end{aligned}
$$

The second term $N H_{\hat{P}_D}(\mathbf{X})$ in Eq. (29) is not affected by the choice of B, therefore maximizing $\mathrm{LL}(B \mid D)$ is equivalent to minimizing the relative entropy $D_{KL}(\hat{P}_D(\mathbf{X}) \,||\, P_B(\mathbf{X}))$.

Friedman *et al.* [16] also hinted that maximizing the conditional log-likelihood is equivalent to minimizing the conditional relative entropy. Assuming that $\mathbf{A} = \mathbf{X} \setminus \{C\}$, and for the case of BNCs, this fact can be taken from

$$
\begin{aligned}
\mathrm{CLL}(B \mid D) &= \sum_{t=1}^{N} \log P_B(c_t \mid \mathbf{y}_t^-) \\
&= N \sum_{\mathbf{y}_t^-} \sum_{c_t} \hat{P}_D(\mathbf{y}_t^-, c_t) \log P_B(c_t \mid \mathbf{y}_t^-) \\
&= N \sum_{\mathbf{y}_t^-} \sum_{c_t} \hat{P}_D(\mathbf{y}_t^-, c_t) \log P_B(c_t \mid \mathbf{y}_t^-) + N H_{\hat{P}_D}(C \mid \mathbf{A}) - N H_{\hat{P}_D}(C \mid \mathbf{A}) \\
&= N \sum_{\mathbf{y}_t^-} \sum_{c_t} \hat{P}_D(\mathbf{y}_t^-, c_t) \log P_B(c_t \mid \mathbf{y}_t^-) - N \sum_{\mathbf{y}_t^-} \sum_{c_t} P_{\hat{P}_D}(\mathbf{y}_t^-, c_t) \log P_{\hat{P}_D}(c_t \mid \mathbf{y}_t^-) \\
&\quad -N H_{\hat{P}_D}(C \mid \mathbf{A}) \\
&= -N \sum_{\mathbf{y}_t^-} \sum_{c_t} \hat{P}_D(\mathbf{y}_t^-, c_t) \log \frac{\hat{P}_D(c_t \mid \mathbf{y}_t^-)}{P_B(c_t \mid \mathbf{y}_t^-)} - N H_{\hat{P}_D}(C \mid \mathbf{A}) \\
&= -N \sum_{\mathbf{y}_t^-} \sum_{c_t} \hat{P}_D(\mathbf{y}_t^-) \hat{P}_D(c_t \mid \mathbf{y}_t^-) \log \frac{\hat{P}_D(c_t \mid \mathbf{y}_t^-)}{P_B(c_t \mid \mathbf{y}_t^-)} - N H_{\hat{P}_D}(C \mid \mathbf{A}) \\
&= -N \sum_{\mathbf{y}_t^-} \hat{P}_D(\mathbf{y}_t^-) \sum_{c_t} \hat{P}_D(c_t \mid \mathbf{y}_t^-) \log \frac{\hat{P}_D(c_t \mid \mathbf{y}_t^-)}{P_B(c_t \mid \mathbf{y}_t^-)} - N H_{\hat{P}_D}(C \mid \mathbf{A}) \\
&= -N D_{KL}(\hat{P}_D(C \mid \mathbf{A}) \,||\, P_B(C \mid \mathbf{A})) - N H_{\hat{P}_D}(C \mid \mathbf{A}). \qquad (30)
\end{aligned}
$$

Indeed, the main goal in classification is to learn the model that best approximates the conditional probability, induced by the OFE, of C given $\mathbf{A}$. It follows from Eq. (30) that by maximizing $\mathrm{CLL}(B \mid D)$ such goal is achieved since we are minimizing $D_{KL}(\hat{P}_D(C \mid \mathbf{A}) \,||\, P_B(C \mid \mathbf{A}))$ which is the conditional relative entropy between the conditional distribution induced by the OFE and the conditional distribution given by the target model B.

In conclusion, since aCLL is a MVU approximation of CLL, by finding the model B that maximizes $\mathrm{aCLL}(B \mid D)$ we expect to minimize the conditional relative entropy between $\hat{P}_D(C \mid \mathbf{A})$ and $P_B(C \mid \mathbf{A})$.

5. Experimental results

We implemented the TAN learning algorithm (c.f. Section 2.1), endowed with the aCLL scoring criterion, in Mathematica 7.0 on top of the Combinatorica package [25]. For the experimental results, we considered optimal TAN structures learned with LL and aCLL under both (symmetric) uniform and Dirichlet assumptions. The main goal of this empirical assessment is to compare the CLL attained with these optimal structures, in order to unravel which one better approximates CLL. To this end, only the scoring criteria vary, and the searching procedure is fixed to yield an optimal TAN structure; this ensures that only the choice of the scoring criterion affects the learned model. To avoid overfitting, that arises naturally when complex structures are searched, we improved the performance of all BNCs by smoothing parameter estimates according to a Dirichlet prior [19]. The smoothing parameter N' was set to 5, as suggested in [16].

We performed our evaluation on 89 benchmark datasets used by Barash *et al.* [1]. These datasets constitute biologically validated data retrieved from the TRANSFAC database [30] for binary classification tasks. The aCLL constants considered for the uniform assumption are those given in Eq. (13). The constants considered under the Dirichlet assumption are those given in Example 3.7, since all the datasets have size around 1000.

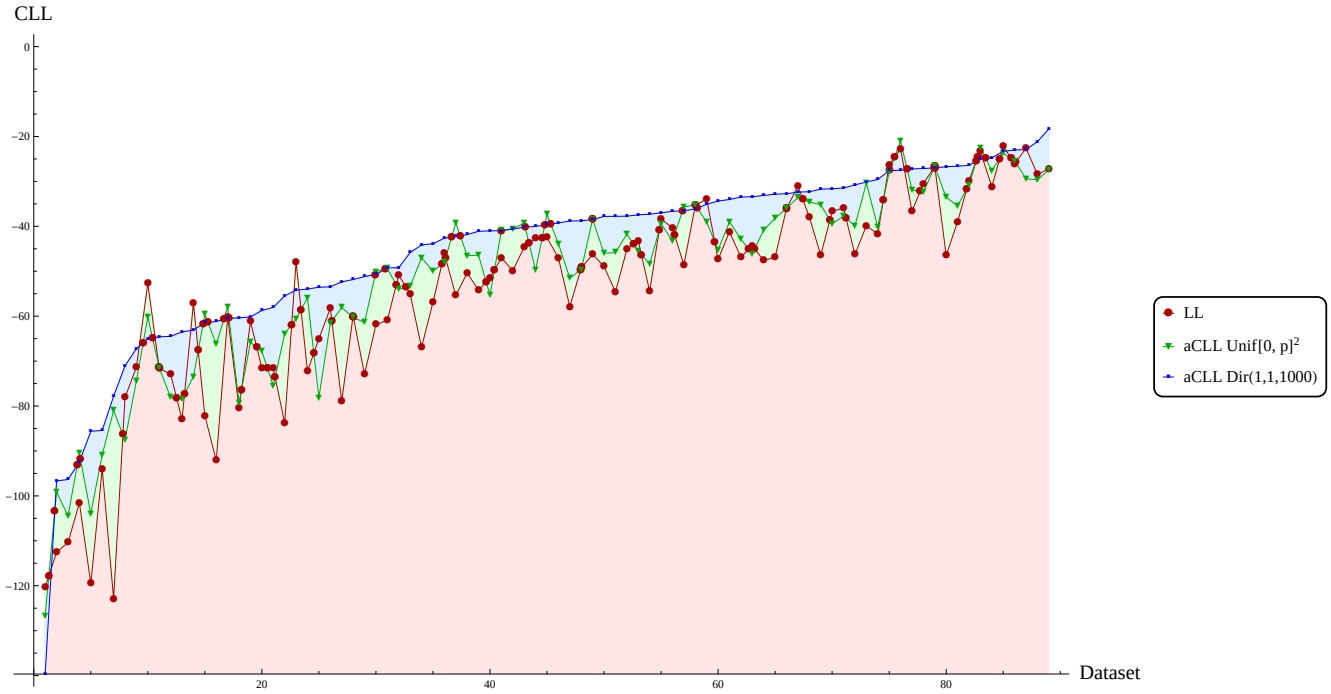
The CLL of the optimal TAN learned by each scoring criterion was computed and it is depicted in Figure 1. From the figure is clear that the scoring criterion that obtained the highest CLL was aCLL under Dir(1,1,1000), followed by aCLL under Unif$([0, p]^2)$. To evaluate the statistical significance of these results we used Wilcoxon signed-rank tests. This test is applicable when paired scoring differences, along the datasets, are independent and not necessarily normally distributed [13]. Results are presented in Table 1. Each entry of the table gives the Z-test and p-value of the significance test for the corresponding pairs of BNCs. The arrow points to the superior scoring criterion, in terms of higher CLL. From Table 1

Table 1. Wilcoxon signed-rank tests for the CLL score obtained by optimal TANs.

Searching **Score** **Assumption**	TAN LL	TAN aCLL Unif$[0, p]^2$
TAN aCLL Dir(1,1,1000)	7.08 7.21×10^{-13} $\Leftarrow$	6.84 3.96×10^{-12} $\Leftarrow$
TAN aCLL Unif$[0, p]^2$	3.52 2.15×10^{-4} $\Leftarrow$	

it is clear that aCLL is significantly better than LL for obtaining a model with higher CLL. In addition, the Dirichlet assumption is more suitable than the uniform one for learning BNCs.

Figure 1. CLL score achieved by optimal TANs with different scoring criteria.



6. Conclusions

In this work we explored three major shortcomings of the initial proposal of the aCLL scoring criterion [5]: (i) it addressed only binary-classification tasks; (ii) it assumed only a uniform distribution over the parameters; (iii) in the context of discriminative learning of BNCs, it did not provide the optimal parameters that maximize it. The effort of exploring the aforementioned limitations culminated with the proposal of a non-trivial extension of aCLL for multi-classification tasks under diverse stochastic assumptions. Whenever possible, the approximation constants were computed analytically; this included binary and ternary classification tasks under a symmetric uniform assumption. In addition, a Monte-Carlo method was proposed to compute the constants required for the approximation under a symmetric Dirichlet assumption. In the context of discriminative learning of BNCs, we showed that the extended score is decomposable over the BNC structure and provided the parameters that maximize it. This decomposition allows score-based learning procedures to be employed locally, turning full (structure and parameters) discriminative learning of BNCs very efficient. Such discriminative learning is equivalent to minimizing the conditional relative entropy between the conditional distribution of the class given the attributes induced by the OFE and the one given by the learned discriminative model.

The merits of the devised scoring criteria under two different assumptions were evaluated in real biological data. These assumptions adopted a symmetric uniform and a symmetric Dirichlet distribution over the parameters. Optimal discriminative models learned with aCLL both with symmetric uniform

and symmetric Dirichlet assumptions, showed higher CLL than those learned generatively with LL. Moreover, among the proposed criteria, the symmetric Dirichlet assumption also showed to approximate CLL better than the symmetric uniform one. This was expected as the Dirichlet is a conjugate distribution for multinomial sample, which ties perfectly with the fact that data is assumed to be a multinomial sample when learning BNCs.

Directions for future work include extending aCLL to unsupervised learning and to deal with missing data.

Acknowledgements and funding This work was partially supported by Fundação para a Ciência e Tecnologia (FCT), under grant PEst-OE/EEI/LA0008/2011, and by the projects NEUROCLINOMICS (PTDC/EIA-EIA/111239/2009) and ComFormCrypt (FCT PTDC/EIA-CCO/113033/2009), also funded by FCT.

References

1. Y. Barash, G. Elidan, N. Friedman, and T. Kaplan. Modeling dependencies in protein-DNA binding sites. In *Proc. RECOMB'03*, pages 28–37, 2003.
2. J. Bilmes. Dynamic Bayesian multinets. In *Proc. UAI'00*, pages 38–45, 2000.
3. A. M. Carvalho. Scoring functions for learning Bayesian networks. Technical report, INESC-ID Tec. Rep. 54/2009, 2009.
4. A. M. Carvalho, A. L. Oliveira, and M.-F. Sagot. Efficient learning of Bayesian network classifiers: An extension to the TAN classifier. In M. A. Orgun and J. Thornton, editors, *Proc. IA'07*, volume 4830 of *LNCS*, pages 16–25. 2007.
5. A. M. Carvalho, T. Roos, A. L. Oliveira, and P. Myllymäki. Discriminative learning of Bayesian networks via factorized conditional log-likelihood. *Journal of Machine Learning Research*, 12(Jul):2181–2210, 2011.
6. D. M. Chickering. *Learning Bayesian networks is NP-Complete*, pages 121–130. Learning from data: AI and statistics V. Springer, 1996.
7. D. M. Chickering. Learning equivalence classes of Bayesian-network structures. *Journal of Machine Learning Research*, 2:445–498, 2002.
8. D. M. Chickering, D. Heckerman, and C. Meek. Large-sample learning of Bayesian networks is NP-hard. *Journal of Machine Learning Research*, 5:1287–1330, 2004.
9. C. K. Chow and C. N. Liu. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 14(3):462–467, 1968.

10. T. Cover and J. Thomas. *Elements of information theory*. John Wiley & Sons, 2006.
11. P. Dagum and M. Luby. Approximating probabilistic inference in Bayesian belief networks is NP-hard. *Artificial Intelligence*, 60(1):141–153, 1993.
12. L. M. de Campos. A scoring function for learning Bayesian networks based on mutual information and conditional independence tests. *Journal of Machine Learning Research*, 7:2149–2187, 2006.
13. J. Demsar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.
14. P. Domingos and M. J. Pazzani. On the optimality of the simple Bayesian classifier under zero-one loss. *Machine Learning*, 29(2–3):103–130, 1997.
15. J. Edmonds. Optimum branchings. *Journal of Research of the National Bureau of Standards*, 71B:233–240, 1967.
16. N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. *Machine Learning*, 29(2-3):131–163, 1997.
17. R. Greiner and W. Zhou. Structural extension to logistic regression: Discriminative parameter learning of belief net classifiers. In *Proc. AAAI/IAAI'02*, pages 167–173, 2002.
18. D. Grossman and P. Domingos. Learning Bayesian network classifiers by maximizing conditional likelihood. In *Proc. ICML'04*, pages 46–53, 2004.
19. D. Heckerman, D. Geiger, and D. M. Chickering. Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 20(3):197–243, 1995.
20. David Heckerman. A tutorial on learning Bayesian networks. Technical Report MSR-TR-95-06, Microsoft Research, 1995.
21. R. A. Johnson and D. W. Wichern. *Applied Multivariate Statistical Analysis*. Prentice Hall, 2007.
22. D. Koller and N. Friedman. *Probabilistic Graphical models: Principles and techniques*. MIT Press, 2009.
23. P. M. Lewis. Approximating probability distributions to reduce storage requirements. *Inormation and Control*, 2:214–225, 1959.
24. J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.
25. S. V. Pemmaraju and S. S. Skiena. *Computational discrete mathematics: Combinatorics and graph theory with Mathematica*. Cambridge University Press, 2003.
26. T. Silander, T. Roos, P. Kontkanen, and P. Myllymäki. Bayesian network structure learning using factorized NML universal models. In *Proc. PGM'08*, pages 257–264, 2008.
27. J. Su and H. Zhang. Full Bayesian network classifiers. In *Proc. ICML'06*, pages 897–904, 2006.

411 28. J. Su, H. Zhang, C. X. Ling, and S. Matwin. Discriminative parameter learning for Bayesian
412 networks. In *Proc ICML'08*, pages 1016–1023, 2008.
413 29. T. Verma and J. Pearl. Equivalence and synthesis of causal models. In *Proc. UAI'90*, pages
414 255–270, 1990.
415 30. E. Wingender, X. Chen, E. Fricke, R. Geffers, R. Hehl, I. Liebich, M. Krull, V. Matys, H. Michael,
416 R. Ohnhuser, M. Prss, F. Schacherer, S. Thiele, and S. Urbach. The TRANSFAC system on gene
417 expression regulation. *Nucleic Acids Research*, 29(1):281–283, 2001.
418 31. S. Yang and K.-C. Chang. Comparison of score metrics for Bayesian network learning. *IEEE*
419 *Transactions on Systems, Man, and Cybernetics, Part A*, 32(3):419–428, 2002.

422