On Discriminative Bayesian Network Classifiers and Logistic Regression

Teemu Roos and Hannes Wettig
Complex Systems Computation Group, Helsinki Institute for Information Technology, P.O. Box 9800, FI-02015 HUT, Finland

Peter Grünwald
Centrum voor Wiskunde en Informatica, P.O. Box 94079, NL-1090 GB Amsterdam, The Netherlands

Petri Myllymäki and Henry Tirri
Complex Systems Computation Group, Helsinki Institute for Information Technology, P.O. Box 9800, FI-02015 HUT, Finland

Abstract. Discriminative learning of the parameters in the naive Bayes model is known to be equivalent to a logistic regression problem. Here we show that the same fact holds for much more general Bayesian network models, as long as the corresponding network structure satisfies a certain graph-theoretic property. The property holds for naive Bayes but also for more complex structures such as tree-augmented naive Bayes (TAN) as well as for mixed diagnostic-discriminative structures. Our results imply that for networks satisfying our property, the conditional likelihood cannot have local maxima so that the global maximum can be found by simple local optimization methods. We also show that if this property does *not* hold, then in general the conditional likelihood *can* have local, non-global maxima. We illustrate our theoretical results by empirical experiments with local optimization in a conditional naive Bayes model. Furthermore, we provide a heuristic strategy for pruning the number of parameters and relevant features in such models. For many data sets, we obtain good results with heavily pruned submodels containing many fewer parameters than the original naive Bayes model.

Keywords: Bayesian classifiers, Bayesian networks, discriminative learning, logistic regression

1. Introduction

Bayesian network models are widely used for discriminative prediction tasks such as classification. The parameters of such models are often determined using 'unsupervised' methods such as maximization of the joint likelihood (Friedman et al., 1997). In recent years, it has been recognized, both theoretically and experimentally, that in many situations it is better to use a matching 'discriminative' or 'supervised' learning algorithm such as *conditional likelihood maximization* (Friedman et al., 1997; Greiner et al., 1997; Ng and Jordan, 2001; Kontkanen et al., 2001). In this paper, we show that if the network structure satisfies a certain

simple graph-theoretic condition, then the corresponding conditional likelihood maximization problem is equivalent to logistic regression based on certain statistics of the data—different network structures leading to different statistics. We thereby establish a connection between the relatively new methods of Bayesian network classifier learning and the established statistical method of logistic regression, which has a long history and which has been thoroughly studied in the past (McLachlan, 1992). One of the main implications of this connection is the following: if the condition mentioned above holds for the network structure, the corresponding conditional likelihood of the Bayesian network model can not have local optima so that the global optimum can be found by local optimization methods. Other implications, as well as our additional results are summarized below.

Section 2 reviews Bayesian network models, Bayesian network classifiers and discriminative learning of Bayesian network models, based on the conditional rather than joint likelihood. Consider a network structure (directed acyclic graph, DAG) $\mathcal{B}$ on a tuple of discrete-valued random variables $(X_0, \ldots, X_M)$. In Section 2.4 we define our main concept, the *conditional Bayesian network model* $\mathcal{M}^{\mathcal{B}}$. This is the set of *conditional* distributions of class variable X_0 given the feature variables $X_1, \ldots, X_M$, induced by the Bayesian network model based on structure $\mathcal{B}$ that can be achieved with non-zero parameters. We define for each network structure $\mathcal{B}$ a corresponding *canonical form* $\mathcal{B}^*$ that facilitates comparison with logistic regression. Graph $\mathcal{B}^*$ is simply the *Markov blanket* of the class variable X_0 in $\mathcal{B}$, with arcs added to make all parents of X_0 fully connected. We show that the models $\mathcal{M}^{\mathcal{B}}$ and $\mathcal{M}^{\mathcal{B}^*}$ are identical: even though the set of *joint* distributions on $(X_0, \ldots, X_M)$ corresponding to $\mathcal{B}$ and $\mathcal{B}^*$ may not coincide, the set of *conditional* distributions of X_0 given $(X_1, \ldots, X_M)$ is the same for both graphs.

Section 3 reviews the *multiple logistic regression* model. We provide a reparameterization of conditional Bayesian network models $\mathcal{M}^{\mathcal{B}}$ such that the parameters in the new parameterization correspond to logarithms of parameters in the standard Bayesian network parameterization. In this way, each conditional Bayesian network model is mapped to a logistic regression model. However, *in some cases the parameters of this logistic regression model are not allowed to vary freely.* In other words, the Bayesian network model corresponds to a subset of a logistic regression model rather than a 'full' logistic regression model. This is established in our Theorem 2.

In Section 4 we present our main result (Theorem 3) which provides a general condition on the network structure $\mathcal{B}$ under which the Bayesian network model is mapped to a full logistic regression model

with freely varying parameters. This condition is very simple: it requires the corresponding canonical structure $\mathcal{B}^*$ to be a *perfect* DAG, meaning that all nodes are *moral* nodes. It is satisfied by, for example, the naive Bayes model, the tree-augmented naive Bayes model (TAN), but also for more complicated models in which the class node has parents.

The conditional log-likelihood for logistic regression models is a concave function of the parameters. Therefore, in the new parameterization the conditional log-likelihood becomes a concave function of the parameters that under the perfectness condition are allowed to vary freely over the convex set $\mathbb{R}^k$. This implies that we can find the global maximum in the conditional likelihood surface by simple local optimization techniques such as hill climbing. This result still leaves open the possibility that there are *no* network structures for which the conditional likelihood surface has local, non-global maxima. This would make perfectness a superfluous condition. Our second result (Theorem 4) shows that this is not the case: there are very simple network structures $\mathcal{B}$ whose canonical version $\mathcal{B}^*$ is not perfect, and for which the conditional likelihood can exhibit local, non-global maxima.

Section 5 discusses the various issues arising when implementing local optimization methods for finding the maximum conditional likelihood of Bayesian network models whose structure $\mathcal{B}$ satisfies the perfectness property. These involve choosing the appropriate parameterization, dealing with model selection (parameter/feature pruning), missing data and flat regions of the conditional likelihood surface, which we handle by introducing Bayesian parameter priors. It turns out that in the standard parameterization of a Bayesian network, the conditional likelihood may be non-concave, even if the Bayesian network corresponds to a logistic regression model. Therefore, optimization should be performed in the logistic parameterization in which the likelihood *is* concave. Finally, we introduce the algorithms of our experiments here, including a heuristic algorithm for pruning the conditional naive Bayes model.

Section 6 reports the results of these experiments. One of our main findings is that our pruning strategy leads to considerably simpler models that are competitive against both naive Bayes and conditional naive Bayes with the full set of features in terms of predictive performance.

Viewing Bayesian network models as subsets of logistic regression models has been suggested earlier in papers such as (Heckerman and Meek, 1997a; Ng and Jordan, 2001; Greiner and Zhou, 2002). Also, the concavity of the log-likelihood surface for logistic regression is a well-known result. Our main contribution is to supply the condition under which Bayesian network models correspond to logistic regression with *completely freely varying parameters*. Only then can we guarantee

that there are no local maxima in the likelihood surface. As a direct consequence of our result, we show that the conditional likelihood of, for instance, the tree-augmented naive Bayes (TAN) model has no local non-global maxima.

2. Bayesian Network Classifiers

We start with some notation and basic properties of Bayesian networks. For more information see, e.g., (Pearl, 1988; Lauritzen, 1996).

2.1. Preliminaries and Notation

Consider a discrete random vector $\mathbf{X} = (X_0, X_1, \ldots, X_M)$, where each variable X_i takes values in $\{1, \ldots, n_i\}$. Let $\mathcal{B}$ be a directed acyclic graph (DAG) over $\mathbf{X}$, that factorizes $P(\mathbf{X})$ into

$$P(\mathbf{X}) = \prod_{i=0}^{M} P(X_i \mid \mathbf{Pa}_i), \tag{1}$$

where $\mathbf{Pa}_i \subseteq \{X_0, \ldots, X_M\}$ is the parent set of variable X_i in $\mathcal{B}$. Such a model is usually parameterized by vectors $\theta^{\mathcal{B}}$ with components of the form $\theta^{\mathcal{B}}_{x_i|\mathbf{pa}_i}$ defined by

$$\theta^{\mathcal{B}}_{x_i|\mathbf{pa}_i} := P(X_i = x_i \mid \mathbf{Pa}_i = \mathbf{pa}_i), \tag{2}$$

where $\mathbf{pa}_i$ is any configuration (set of values) for the parents $\mathbf{Pa}_i$ of X_i. Whenever we want to emphasize that each $\mathbf{pa}_i$ is determined by the complete data vector $\mathbf{x} = (x_0, \ldots, x_M)$, we write $\mathbf{pa}_i(\mathbf{x})$ to denote the configuration of $\mathbf{Pa}_i$ in $\mathcal{B}$ given by the vector $\mathbf{x}$. For a given data vector $\mathbf{x} = (x_0, x_1, \ldots, x_M)$, we sometimes need to consider a modified vector where x_0 is replaced by x'_0 and the other entries remain the same. We then write $\mathbf{pa}_i(x'_0, \mathbf{x})$ for the configuration of $\mathbf{Pa}_i$ given by $(x'_0, x_1, \ldots, x_M)$.

We are interested in predicting some *class variable* X_l for some $l \in \{0, \ldots, M\}$ conditioned on all X_i, $i \neq l$. Without loss of generality we may assume that $l = 0$ (i.e., X_0 is the class variable) and that the children of X_0 in $\mathcal{B}$ are $\{X_1, \ldots, X_m\}$ for some $m \leq M$. For instance, in the so-called naive Bayes model we have $m = M$ and the children of the class variable X_0 are independent given the value of X_0. The Bayesian network model corresponding to $\mathcal{B}$ is the set of all distributions satisfying the conditional independencies encoded in $\mathcal{B}$.

Conditional distributions for the class variable given the other variables can be written as

$$\begin{aligned} P(x_0 \mid x_1, \ldots, x_M, \theta^{\mathcal{B}}) &= \frac{P(x_0, x_1, \ldots, x_M \mid \theta^{\mathcal{B}})}{\sum_{x_0'=1}^{n_0} P(x_0', x_1, \ldots, x_M \mid \theta^{\mathcal{B}})} \\ &= \frac{\theta^{\mathcal{B}}_{x_0|\mathbf{pa}_0(\mathbf{x})} \prod_{i=1}^{M} \theta^{\mathcal{B}}_{x_i|\mathbf{pa}_i(\mathbf{x})}}{\sum_{x_0'=1}^{n_0} \theta^{\mathcal{B}}_{x_0'|\mathbf{pa}_0(\mathbf{x})} \prod_{i=1}^{M} \theta^{\mathcal{B}}_{x_i|\mathbf{pa}_i(x_0',\mathbf{x})}}. \end{aligned} \tag{3}$$

2.2. Bayesian Network Classifiers

A Bayesian network model can be used both for probabilistic prediction and for classification. By probabilistic prediction we mean a game where given a *query vector* $(x_1, \ldots, x_M)$, we must output a conditional distribution $\hat{P}(X_0 \mid x_1, \ldots, x_M)$ for the class variable X_0. Under the *logarithmic loss function* (log-loss) we incur $-\ln \hat{P}(x_0 \mid x_1, \ldots, x_M)$ units of loss where x_0 is the actual outcome. If we successively predict class variable outcomes $x_0^1, x_0^2, \ldots, x_0^N$ given query vectors $(x_1, \ldots, x_M)^1, \ldots, (x_1, \ldots, x_M)^N$ using the same $\hat{P}$, then the logarithmic loss function is just minus the conditional log-likelihood of $x_0^1, x_0^2, \ldots, x_0^N$ given the query vectors, a standard statistical measure. By *classification* we mean the scenario where given a query vector we must output a single value, $\hat{x}_0$ of X_0 that we consider the most likely outcome. Under *0/1 loss* the loss is zero if our guess was correct and one otherwise.

Given a Bayesian network model, the corresponding *Bayesian network classifier* (Friedman et al., 1997) uses (3) for both prediction and classification. Given a parameter vector $\theta^{\mathcal{B}}$, under the log-loss we output the conditional distribution given by (3) and under the 0/1 loss we choose as our guess the class value x_0 maximizing (3). If the distribution indexed by the parameter vector is actually the distribution generating the data vectors, then in both cases this is the Bayes optimal choice.

In order to predict and/or classify new instances based on some training data, we need to fix a method to infer good parameter values based on the training data. A commonly used method is to maximize the *likelihood*, or equivalently the log-likelihood, of the training data. Given a complete data-matrix $D = (\mathbf{x}^1, \ldots, \mathbf{x}^N)$, the *(full) log-likelihood*, $LL(D; \theta^{\mathcal{B}})$, with parameters $\theta^{\mathcal{B}}$ is given by

$$LL(D; \theta^{\mathcal{B}}) := \sum_{j=1}^{N} \ln P(x_0^j, \ldots, x_M^j \mid \theta^{\mathcal{B}}) = \sum_{j=1}^{N} \sum_{i=0}^{M} \ln P(x_i^j \mid \mathbf{pa}_i(x^j), \theta^{\mathcal{B}}). \tag{4}$$

Taking the derivative wrt. $\theta^{\mathcal{B}}$ shows that for complete data the maximum is achieved by setting for each x_i and $\mathbf{pa}_i$

$$\hat{\theta}^{\mathcal{B}}_{x_i|\mathbf{pa}_i}(D) = \frac{n_{[x_i,\mathbf{pa}_i]}}{n_{[\mathbf{pa}_i]}}, \tag{5}$$

where $n_{[x_i,\mathbf{pa}_i]}$ and $n_{[\mathbf{pa}_i]}$ are respectively the numbers of data vectors with $X_i = x_i, \mathbf{Pa}_i = \mathbf{pa}_i$ and $\mathbf{Pa}_i = \mathbf{pa}_i$. In case the data contain missing values, there is no closed form solution but iterative algorithms such as the Expectation-Maximization algorithm (Dempster et al., 1977; McLachlan and Krishnan, 1997), or local search methods such as gradient ascent (Russell et al., 1995; Thiesson, 1995) can be applied.

In the *$\mathcal{M}$-closed* case (Bernardo and Smith, 1994), i.e. the case in which the data generating distribution can be represented with the Bayesian network $\mathcal{B}$, maximizing the full likelihood is a consistent method of estimating the joint distribution of X: The joint distribution of X given the maximum likelihood (ML) parameters converges (with growing sample size) to the generating distribution. Thus, also the conditional distributions of the class variable given the other variables and the maximum likelihood parameters converge to the true conditional distribution. As a consequence, the *maximum likelihood plug-in classifier* obtained by plugging (5) into (3) converges to the Bayes optimal classifier. This holds for both the 0/1 loss and the logarithmic loss.

2.3. Discriminative Parameter Learning

In the context of predicting the class variable given the other variables, the full log-likelihood is not the most natural objective function since it does not take into account the *discriminative* or *supervised* nature of the prediction task. A more focused version is the *conditional log-likelihood*, defined as

$$CLL(D;\theta^{\mathcal{B}}) \;:=\; \sum_{j=1}^{N} \ln P(x_0^j \mid x_1^j \ldots, x_M^j, \theta^{\mathcal{B}}), \tag{6}$$

where $P(x_0^j \mid x_1^j \ldots, x_M^j, \theta^{\mathcal{B}})$ is given by (3). It is important to note that (3), appearing in (6), is not one of the terms in (4) unless $\mathbf{Pa}_0 = \{X_1, \ldots, X_M\}$, i.e., unless the class variable has only incoming arcs. If this were the case, the $\theta_{x_0|\mathbf{pa}_0}$ maximizing the full likelihood would also maximize the conditional likelihood. Due to the normalization over possible values of X_0 in (3), the parameters maximizing conditional likelihood (6) are not given by (5). In fact no closed-form solution is known (Friedman et al., 1997).

Maximizing the conditional likelihood is a consistent method for estimating the conditional distributions of the class variable in the $\mathcal{M}$-closed case, just like maximizing the full likelihood. However, there is a crucial difference between the two methods in the case where the generating distribution is not in the model class, i.e., some of the independence assumptions inherent to the Bayesian network structure $\mathcal{B}$ are violated. In this $\mathcal{M}$-*open* case it is clear that no plug-in predictor can be guaranteed to converge to the Bayes optimal classifier. However, in probabilistic prediction under log-loss, maximizing the conditional likelihood converges to the *best possible* distribution in the model class, in that it minimizes the expected conditional log loss. To see this, let Q, P be two distributions on $(X_0, \ldots, X_M)$ and define the *conditional Kullback-Leibler divergence* as

$$D_{\mathrm{cond}}(Q\|P) := \mathrm{E}_{X_0,\ldots,X_M} \ln \frac{Q(X_0 \mid X_1, \ldots, X_M)}{P(X_0 \mid X_1, \ldots, X_M)}, \tag{7}$$

where the expectation is taken with respect to Q. The Kullback-Leibler divergence gives the expected *additional* log-loss over the best possible distribution given by Q (see Appendix A in (Friedman et al., 1997)). The following proposition shows that $P(\cdot \mid \theta_N^{\mathcal{B}})$ converges in probability to the distribution in $\mathcal{B}$ minimizing expected conditional log-loss:

Proposition 1. Let the data be i.i.d. according to any distribution Q with full support, i.e. $Q(x_0, x_1, \ldots, x_M) > 0$ for all vectors $\mathbf{x}$ with components $x_i \in \{1, \ldots, n_i\}$. Then with probability one, for all large N there exists at least one $\theta_N^{\mathcal{B}}$ maximizing the conditional log-likelihood (6). For any sequence of such maximizing $\theta_N^{\mathcal{B}}$ the distribution $P(\cdot \mid \theta_N^{\mathcal{B}})$ converges in probability to the distribution closest to Q in conditional Kullback-Leibler divergence.

This follows from Thm. 1 in (Greiner and Zhou, 2002) which gives in addition a rate of convergence in term of sample size. When maximizing the *full* likelihood in the $\mathcal{M}$-open case, we may *not* converge to the best possible distribution. In the Appendix, we give a very simple concrete case (Example 4) such that with probability 1, the ordinary ML estimator converges to a parameter vector $\tilde{\theta}$ and the conditional ML estimator converges to another vector $\tilde{\theta}_{\mathrm{cond}}$ with

$$D_{\mathrm{cond}}(Q\|P(\cdot \mid \tilde{\theta})) = \ln 2 \quad ; \quad D_{\mathrm{cond}}(Q\|P(\cdot \mid \tilde{\theta}_{\mathrm{cond}})) = \tfrac{1}{2} \ln 2.$$

In classification the situation is not as clear cut as in log loss prediction. Nevertheless, there is both empirical and theoretical evidence that maximizing the conditional likelihood leads to better classification as well, see (Friedman et al., 1997; Ng and Jordan, 2001; Greiner and

Zhou, 2002) and references therein. Theoretically, one can argue as follows. Suppose data are i.i.d. according to some distribution Q not necessarily in $\mathcal{M}$. Given a distribution $P(\cdot \mid \theta^{\mathcal{B}})$, we say $\theta^{\mathcal{B}}$ is *conditionally correct* if $Q(X_0 \mid X_1, \ldots, X_M) = P(X_0 \mid X_1, \ldots, X_M, \theta^{\mathcal{B}})$ for all $(X_0, \ldots, X_M)$ that occur with Q-positive probability. Now suppose that $\mathcal{M}$ contains a $\tilde{\theta}_{\mathrm{cond}}$ that is conditionally correct. Then $\tilde{\theta}_{\mathrm{cond}}$ must also be the distribution that is optimal for classification; that is, the Bayes classifier based on $\tilde{\theta}_{\mathrm{cond}}$ achieves the minimum expected classification error where the expectation is over the distribution Q. As a consequence of Proposition 1, in the large N limit, the conditional ML estimator $\theta^{\mathcal{B}}_N$ converges to $\tilde{\theta}_{\mathrm{cond}}$ and hence, asymptotically, the Bayes classifier based on the ML estimator is optimal. Note that this still holds if $P(\cdot \mid \tilde{\theta}_{\mathrm{cond}})$ is 'unconditionally incorrect' in the sense that the marginal distributions $Q(X_1, \ldots, X_M)$ and $P(X_1, \ldots, X_M \mid \tilde{\theta}_{\mathrm{cond}})$ are very different from each other.

In contrast, when maximizing the unconditional likelihood, the distribution $\tilde{\theta}$ that the ML estimator converges to can only be guaranteed to lead to the optimal classification rule if $\mathcal{M}$ contains a distribution that is *fully* correct, a much stronger condition. Thus, for large training samples, the conditional ML estimator can be shown to be optimal for classification under much weaker conditions than the unconditional ML estimator. This suggests (but of course does not prove) that, for large training samples, conditional ML estimators will often achieve better classification performance than unconditional ML estimators.

2.4. Conditional Bayesian network models

We define the conditional model $\mathcal{M}^{\mathcal{B}}$ as the set of conditional distributions that can be represented using network $\mathcal{B}$ equipped with any strictly positive[1] parameter set $\theta^{\mathcal{B}} > 0$; that is, the set of all functions from $(X_1, \ldots, X_M)$ to distributions on X_0 of the form (3). The model $\mathcal{M}^{\mathcal{B}}$ does not contain any notion of the joint distribution: Terms such as $P(X_i \mid \mathbf{Pa}_i)$, where X_i is not the class variable are undefined and neither are we interested in them. Heckerman and Meek (1997a,1997b) call such models *Bayesian regression/classification* (BRC) models.

Conditional models $\mathcal{M}^{\mathcal{B}}$ have some useful properties, which are not shared by their unconditional counterparts. For example, for many different network structures $\mathcal{B}$, the corresponding conditional models $\mathcal{M}^{\mathcal{B}}$ are equivalent. This is because in (3), all $\theta^{\mathcal{B}}_{x_i|\mathbf{pa}_i}$ with $i > m$ (standing for nodes that are neither the class variable nor any of its children) cancel out, since for these terms we have $\mathbf{pa}_i(\mathbf{x}) \equiv \mathbf{pa}_i(x'_0, \mathbf{x})$ for all

[1] We avoid zero-parameters by introducing priors on the parameters, see Section 5.5.

x'_0. Thus the only relevant parameters for the conditional likelihood are of the form $\theta^{\mathcal{B}}_{x_i|\mathbf{pa}_i}$ with $i \in \{0,\ldots,m\}$, $x_i \in \{1,\ldots,n_i\}$ and $\mathbf{pa}_i$ any configuration of X_i's parents $\mathbf{Pa}_i$. In terms of graphical structure this is expressed in Lemmas 1 and 2 below.

Lemma 1. (Pearl, 1987) Given a Bayesian network $\mathcal{B}$, the conditional distribution of a variable X_i depends on the other variables only through the nodes in the Markov blanket[2] of X_i.

Lemma 2. (Buntine, 1994) Let $\mathcal{B}$ be a Bayesian network and $\mathcal{M}^{\mathcal{B}}$ the corresponding conditional model. If a node X_i and all its parents have their values given (which implies that neither X_i nor any of its parents is the class variable X_0), then the Bayesian network $\mathcal{B}'$ created by deleting all the arcs into X_i represents the same conditional probability model as $\mathcal{B}$: $\mathcal{M}^{\mathcal{B}} = \mathcal{M}^{\mathcal{B}'}$.

Note that although for a node X_i with $i > m$, the parameter $\theta^{\mathcal{B}}_{x_i|\mathbf{pa}_i}$ cancels out in (3), the value of X_i may still influence the conditional probability of X_0 if X_i is a *parent* of X_0 or of some child of X_0. In that case, $X_i \in Pa_j$ where j is such that parameters $\theta^{\mathcal{B}}_{x_j|pa_j}$ do *not* cancel out in (3). Thus, we can restrict attention to the Markov blanket, but not just the class node and its children. Lemma 2, slightly rephrased from (Buntine, 1994) and illustrated by Figure 1, implies that can assume that all parents of the class variable are fully connected—which greatly simplifies our comparison to logistic regression models.

Definition 1. For an arbitrary Bayesian network structure $\mathcal{B}$, we define the corresponding *canonical* structure (for classification) as the structure $\mathcal{B}^*$ which is constructed from $\mathcal{B}$ by first restricting $\mathcal{B}$ to X_0's Markov blanket and then adding as many arcs as needed to make the parents of X_0 fully connected[3].

We have for all network structures $\mathcal{B}$ that $\mathcal{M}^{\mathcal{B}}$ and $\mathcal{M}^{\mathcal{B}^*}$ are equivalent. This follows from Lemmas 1 and 2. Thus, instead of the graph in Figure 1.b. we will use the graph in Figure 1.a.

[2] The *Markov blanket* of a node X_i consists of the parents of X_i, the children of X_i, and the parents of the children of X_i.

[3] The addition can always be done without introducing any cycles (Lauritzen, 1996), usually in several different ways all of which are equivalent for our purposes.

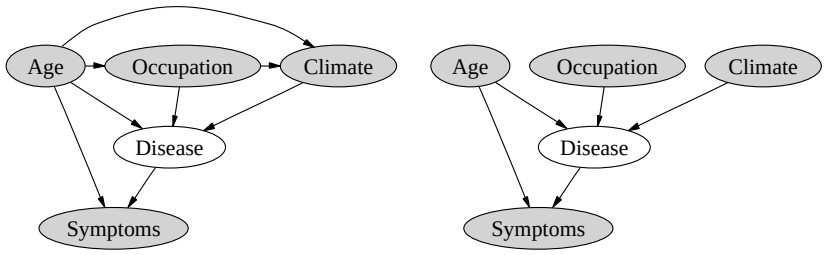

Figure 1. Networks (a) and (b) are equivalent in terms of conditional distributions for variable *Disease* given the shaded variables. Reproduced from (Buntine, 1994).

3. Bayesian Network Classifiers and Logistic Regression

We can think of the conditional model obtained from a Bayesian network also as a predictor that combines the information of the observed variables to update the distribution of the target (class) variable. Thus, we view the Bayesian network as a *discriminative* rather than a *generative* model (Dawid, 1976; Heckerman and Meek, 1997a; Ng and Jordan, 2001; Jebara, 2003). In order to make this view concrete, we now introduce logistic regression models that have been extensively studied in statistics, see, e.g., (McLachlan, 1992). We will then (Section 3.2) see that any conditional Bayesian network model may be viewed as a subset of a logistic regression model.

3.1. Logistic Regression Models

Let X_0 be a random variable with possible values $\{1, \ldots, n_0\}$, and let $\mathbf{Y} = (Y_1, \ldots, Y_k)$ be a real-valued random vector. The *multiple logistic regression model with dependent variable* X_0 *and covariates* $Y_1, \ldots, Y_k$ is defined as the set of conditional distributions

$$P(x_0 \mid \mathbf{y}, \beta) := \frac{\exp \sum_{i=1}^{k} \beta_{x_0,i}\, y_i}{\sum_{x_0'=1}^{n_0} \exp \sum_{i=1}^{k} \beta_{x_0',i}\, y_i}, \tag{8}$$

where the parameter vector β with components of the form $\beta_{x_0,i}$ with $x_0 \in \{1, \ldots, n_0\}, i \in \{1, \ldots, k\}$ is allowed to take on all values in $\mathbb{R}^{n_0 \cdot k}$. Figure 2 is a graphical representation of the model: the intermediate nodes η_i correspond to the linear combinations $\sum_{i=1}^{k} \beta_{x_i,i} y_i$ of the covariates which are converted to predicted class probabilities p_i by the *normalized exponential* or *softmax* (Bishop, 1995) function as in (8).

For all values of the class variable $r \in \{1, \ldots, n_0\}$ and all covariates $s \in \{1, \ldots, k\}$, the components of the gradient vector, i.e., the partial

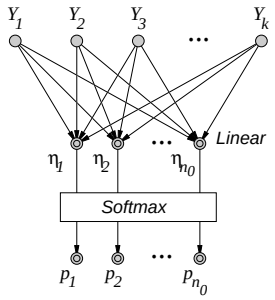

Figure 2. A logistic regression model.

derivatives of the log-likelihood are given by

$$\frac{\partial \ln P(x_0 \mid \mathbf{y}, \beta)}{\partial \beta_{r,s}} = y_s \left(\mathbf{I}_{[r=x_0]} - P(r \mid \mathbf{y}, \beta) \right), \tag{9}$$

where $\mathbf{I}_{[\cdot]}$ is the indicator function taking value 1 if the argument is true, 0 otherwise. The Hessian matrix, H, of second derivatives has entries given by

$$\frac{\partial^2 \ln P(x_0 \mid \mathbf{y}, \beta)}{\partial \beta_{r,s} \partial \beta_{t,u}} = -y_s y_u P(r \mid \mathbf{y}, \beta) \left(\mathbf{I}_{[r=t]} - P(t \mid \mathbf{y}, \beta) \right), \tag{10}$$

where $r, t \in \{1, \ldots, n_0\}$ and $s, u \in \{1, \ldots, k\}$. The following theorem is well-known in statistics, see e.g., (McLachlan, 1992).

Theorem 1. The Hessian matrix is negative semidefinite.

The theorem is a direct consequence of the fact that logistic regression models are exponential families, see e.g., (McLachlan, 1992, p. 260), (Barndorff-Nielsen, 1978). The model is extended to several outcomes under the i.i.d. assumption by defining the log-likelihood

$$CLL_L(D; \beta) \; := \; \sum_{j=1}^{N} \ln P(x_0^j \mid \mathbf{y}^j, \beta). \tag{11}$$

Given a data set, the entries in the gradient vector and the Hessian matrix are sums of terms given by (9) and (10) respectively. By Theorem 1, the Hessian for each data vector is negative semidefinite, thus (8) is concave. Therefore, log-likelihood (11) as a sum of concave functions is also concave, so that we have:

Corollary 1. The log-likelihood (11) is a concave function of the parameters.

Corollary 2. Concavity, together with the fact that the parameter vector β varies freely in a convex set ($\mathbb{R}^{n_0 \cdot k}$) guarantees that there are no local, non-global, maxima in the likelihood surface of a logistic regression model.

The conditions under which a global maximum exists and the maximum likelihood estimators do not diverge are discussed in, e.g., (McLachlan, 1992) and references therein. A possible solution in cases where no maximum exists is to assign a prior on the model parameters and maximize the 'conditional posterior' instead of the likelihood, see Section 5.5. The prior can also resolve problems in optimization caused by the well-known fact that the parameterization of the logistic model is not one-to-one and the log-likelihood surface is not *strictly* concave.

3.2. A Logistic Representation of Bayesian Networks

In order to create a logistic model corresponding to a Bayesian network structure, we introduce a new set of covariates derived from the original variables. First, for all parent configurations $\mathbf{pa}_0$ of X_0, set

$$Y_{\mathbf{pa}_0} := \mathbf{I}_{[\mathbf{Pa}_0 = \mathbf{pa}_0]}. \tag{12}$$

Denote the parameters associated with such covariates by $\beta^{\mathcal{B}}_{x_0,\mathbf{pa}_0}$. Next, define $\mathbf{Pa}_i^+ := \mathbf{Pa}_i \setminus \{X_0\}$, i.e., the parent set of X_i with the exclusion of the class variable X_0. Now, for $i \in \{1,\ldots,m\}$, $x_i \in \{1,\ldots,n_i\}$ and $\mathbf{pa}_i^+ \in \mathrm{dom}(\mathbf{Pa}_i^+)$, set

$$Y_{x_i,\mathbf{pa}_i^+} := \mathbf{I}_{[X_i = x_i, \mathbf{Pa}_i^+ = \mathbf{pa}_i^+]}. \tag{13}$$

Denote the parameters associated for such covariates by $\beta^{\mathcal{B}}_{x_0,x_i,\mathbf{pa}_i^+}$.

Example 1. Consider the Bayesian network in Figure 1.a. The covariates of type (12) correspond to all combinations of the values of *Age, Occupation* and *Climate*. Covariates of type (13) correspond to all combinations of the values of *Age* and *Symptoms*. The logistic model obtained from the network in Figure 1.b. is identical. ◇

For convenience, but without loss of generality we can use indexing of the form $Y_{\mathbf{pa}_0}$ and $Y_{x_i,\mathbf{pa}_i^+}$ instead of Y_i. With these notations (8)

can be written as a function of variables $X_1, \ldots, X_M$:

$$P(x_0 \mid x_1, \ldots, x_M, \beta^{\mathcal{B}}) := \frac{\exp\left(\sum_{\mathbf{pa}_0} \beta^{\mathcal{B}}_{x_0,\mathbf{pa}_0} y_{\mathbf{pa}_0} + \sum_{i=1}^{m} \sum_{x_i=1}^{n_i} \sum_{\mathbf{pa}_i^+} \beta^{\mathcal{B}}_{x_0,x_i,\mathbf{pa}_i^+} \, y_{x_i,\mathbf{pa}_i^+}\right)}{\sum_{x_0'=1}^{n_0} \exp\left(\sum_{\mathbf{pa}_0} \beta^{\mathcal{B}}_{x_0',\mathbf{pa}_0} y_{\mathbf{pa}_0} + \sum_{i=1}^{m} \sum_{x_i=1}^{n_i} \sum_{\mathbf{pa}_i^+} \beta^{\mathcal{B}}_{x_0',x_i,\mathbf{pa}_i^+} \, y_{x_i,\mathbf{pa}_i^+}\right)}.$$

Because most of the indicator variables take value zero, the equation simplifies to

$$P(x_0 \mid x_1, \ldots, x_M, \beta^{\mathcal{B}}) = \frac{\exp\left(\beta^{\mathcal{B}}_{x_0,\mathbf{pa}_0(\mathbf{x})} + \sum_{i=1}^{m} \beta^{\mathcal{B}}_{x_0,x_i,\mathbf{pa}_i^+(\mathbf{x})}\right)}{\sum_{x_0'=1}^{n_0} \exp\left(\beta^{\mathcal{B}}_{x_0',\mathbf{pa}_0(\mathbf{x})} + \sum_{i=1}^{m} \beta^{\mathcal{B}}_{x_0',x_i,\mathbf{pa}_i^+(\mathbf{x})}\right)}. \tag{14}$$

Let the conditional model $\mathcal{M}_L^{\mathcal{B}}$ be the set of conditional distributions for X_0 that can be represented with the logistic regression model corresponding to $\mathcal{B}$. It turns out that the logistic regression conditional model $\mathcal{M}_L^{\mathcal{B}}$ is very closely related to the corresponding conditional BN model $\mathcal{M}^{\mathcal{B}}$: Theorem 2 shows that all conditional distributions representable with the Bayesian network can be mapped to distributions of the logistic model.

Theorem 2. Let $\mathcal{M}^{\mathcal{B}}$ be the set of conditional distributions that can be represented by a Bayesian model with network structure $\mathcal{B}$ and strictly positive parameters, and let $\mathcal{M}_L^{\mathcal{B}}$ be the conditional model defined by the logistic regression model with covariates (12) and (13). Then $\mathcal{M}^{\mathcal{B}} \subseteq \mathcal{M}_L^{\mathcal{B}}$.

Proof. Let $\theta^{\mathcal{B}}$ be an arbitrary parameter vector. The theorem is equivalent to there being a parameter vector $\beta^{\mathcal{B}}$ for the logistic regression model such that the two models represent the same conditional distributions for X_0. Set

$$\beta^{\mathcal{B}}_{x_0,\mathbf{pa}_0} = \ln \theta^{\mathcal{B}}_{x_0|\mathbf{pa}_0} \quad ; \quad \beta^{\mathcal{B}}_{x_0,x_i,\mathbf{pa}_i^+} = \ln \theta^{\mathcal{B}}_{x_i|\mathbf{pa}_i} \text{ for } 1 < i \leq m, \tag{15}$$

where $\mathbf{pa}_i$ is the combination of $\mathbf{pa}_i^+$ and x_0. Plugging these into (14) gives the same conditional distributions as (3). □

Given data D, define the conditional log-likelihood of the logistic model as

$$CLL_L(D;\beta^{\mathcal{B}}) := \sum_{j=1}^{N} \ln P(x_0^j \mid x_1^j \ldots, x_M^j, \beta^{\mathcal{B}}), \tag{16}$$

where $P(x_0^j \mid x_1^j \ldots, x_M^j, \beta^{\mathcal{B}})$ is given by (14). Note that (16) has the same properties as (11). In particular, Corollaries 1 and 2 apply to (16) as well: there are no local maxima in the log-likelihood surface (16) of any logistic regression model $\mathcal{M}_L^{\mathcal{B}}$.

The global conditional maximum likelihood parameters obtained from training data can be used for prediction of future data. In addition, as discussed by Heckerman and Meek (1997a), they can be used to perform model selection among several competing model structures using, e.g., the Bayesian Information Criterion (BIC) (Schwarz, 1978) or approximations of the Minimum Description Length (MDL) criterion (Rissanen, 1996). Heckerman and Meek (1997a) state that for general conditional Bayesian network models $\mathcal{M}^{\mathcal{B}}$, "although it may be difficult to determine a global maximum, gradient-based methods [...] can be used to locate local maxima". Corollary 2 shows that if the network structure $\mathcal{B}$ is such that the two models are equivalent, $\mathcal{M}^{\mathcal{B}} = \mathcal{M}_L^{\mathcal{B}}$, we can find even the *global* maximum of the conditional likelihood by using the logistic model and using some local optimization method. Therefore, it becomes a crucial problem to determine the exact condition under which the equivalence holds.

4. Theoretical Results

In the preceding sections we gave a logistic representation of Bayesian networks and showed that all conditional distributions for the class variable can be represented with the logistic model. Here we show that in general the converse of this statement is not true, which means that the two conditional models are not equivalent. However, we give a condition on the network structure under which the conditional models are equivalent.

4.1. A Condition on the Network Structure

It was shown in the previous section that by using parameters given by (15), it follows that each distribution in $\mathcal{M}^{\mathcal{B}}$ is also in $\mathcal{M}_L^{\mathcal{B}}$ (Theo-

rem 2). This suggests that by doing the reverse transformation

$$\theta^{\mathcal{B}}_{x_i|\mathbf{pa}_i} = \begin{cases} \exp \beta^{\mathcal{B}}_{x_0,\mathbf{pa}_0}, & \text{if } i = 0, \\ \exp \beta^{\mathcal{B}}_{x_0,x_i,\mathbf{pa}_i^+}, & \text{if } 1 < i \leq m, \end{cases} \tag{17}$$

we could also show that distributions in $\mathcal{M}_L^{\mathcal{B}}$ are also in $\mathcal{M}^{\mathcal{B}}$. However, since the parameters of the logistic model are free, this will in some cases violate the sum-to-one constraint, i.e., for some $\beta^{\mathcal{B}} \in \mathbb{R}^{n_0 \cdot k}$ we get after the transformation (17) parameters such that $\sum_{x_i=1}^{n_i} \theta^{\mathcal{B}}_{x_i|\mathbf{pa}_i^+} \neq 1$ for some $i \in \{0, \ldots, M\}$ and $\mathbf{pa}_i$. Such parameters are *not* valid Bayesian network parameters. Note that simply renormalizing them (over x_i, not x_0!) could change the resulting distributions. But, since the parameterization of the logistic model is not one-to-one, it may still be the case that the *distribution* indexed by parameters $\beta^{\mathcal{B}}$ is in $\mathcal{M}^{\mathcal{B}}$. Indeed, it turns out that for some network structures $\mathcal{B}$, the corresponding $\mathcal{M}_L^{\mathcal{B}}$ is such that each distribution in $\mathcal{M}_L^{\mathcal{B}}$ can be expressed by a parameter vector such that the mapping (17) gives valid Bayesian network parameters. In that case, we *do* have $\mathcal{M}^{\mathcal{B}} = \mathcal{M}_L^{\mathcal{B}}$. Our main result is that this is the case if $\mathcal{B}$ is such that its canonical version $\mathcal{B}^*$ (Definition 1) is *perfect* (Definition 2).

Definition 2. (Lauritzen, 1996) A directed graph in which all nodes having a common child are connected is called *perfect*.

Example 2. Consider the Bayesian networks depicted in Figure 3. Neither $\mathcal{B}_1$ nor $\mathcal{B}_2$ are perfect. The canonical version $\mathcal{B}_1^*$ of $\mathcal{B}_1$ has an added arrow between X_1 and X_2. This makes $\mathcal{B}_1^*$ perfect. However, network $\mathcal{B}_2$ cannot be made perfect without changing the conditional model $\mathcal{M}^{\mathcal{B}}$. Theorem 4 (below) shows that for $\mathcal{B}_2$ the conditional likelihood surface of $\mathcal{M}^{\mathcal{B}}$ can have local maxima, implying that in this case $\mathcal{M}^{\mathcal{B}} \neq \mathcal{M}_L^{\mathcal{B}}$. ◇

Examples of network structures $\mathcal{B}$ that are perfect are the naive Bayes (NB) and the tree-augmented naive Bayes (TAN) models (Friedman et al., 1997). (Proof straightforward and omitted.) The latter is a generalization of the former in which the children of the class variable are allowed to form tree-structures; see Figure 4. Perfectness of $\mathcal{B}$ also implies that the class X_0 must be a 'moral node', i.e., it cannot have a common child with a node it is not directly connected to. Even if X_0 is moral, sometimes perfectness may be violated as exemplified by Figure 4.

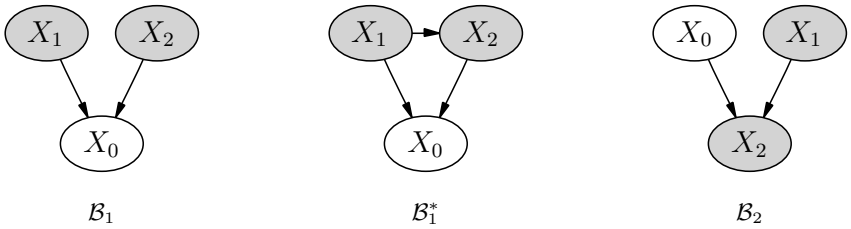

Figure 3. $\mathcal{B}_1$ is a simple Bayesian network (the class variable is denoted by X_0) that satisfies our condition: its canonical form $\mathcal{B}_1^*$ has X_1 and X_2 connected, making the structure perfect; $\mathcal{B}_2$ is a network that remains unchanged under the canonical transformation and remains imperfect.

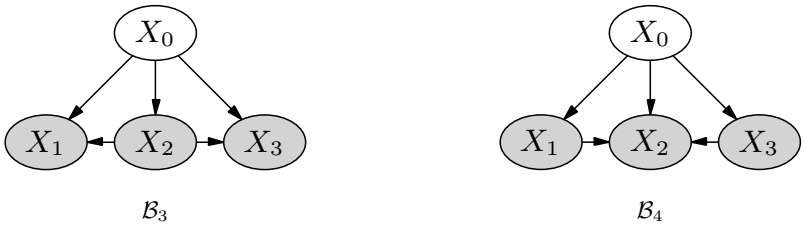

Figure 4. $\mathcal{B}_3$ is a tree-augmented naive Bayes (TAN) model with a perfect graph; and $\mathcal{B}_4$ a network in canonical form that is neither TAN nor perfect.

The condition is also automatically satisfied if X_0 only has incoming arcs[4] ('diagnostic' models discussed in, e.g., (Kontkanen et al., 2001)). For Bayesian network structures $\mathcal{B}$ for which the condition does not hold, we can always add some arrows to arrive at a structure $\mathcal{B}'$ for which the condition does hold (for instance, add an arrow from X_1 to X_3 in $\mathcal{B}_4$ of Figure 4.). Therefore, $\mathcal{M}^{\mathcal{B}}$ is always a subset of a larger model $\mathcal{M}^{\mathcal{B}'}$ for which the condition holds.

4.2. Main Result

We now present our main result stating that the conditional models $\mathcal{M}^{\mathcal{B}}$ and $\mathcal{M}_L^{\mathcal{B}}$ of a Bayesian network $\mathcal{B}$ and the corresponding logistic regression model respectively, are equivalent if $\mathcal{B}^*$ is perfect:

Theorem 3. If $\mathcal{B}$ is such that its canonical version $\mathcal{B}^*$ is perfect, then $\mathcal{M}^{\mathcal{B}} = \mathcal{M}_L^{\mathcal{B}}$.

The proof is based on the following proposition:

Proposition 2. (Lauritzen, 1996) Let $\mathcal{B}$ be a perfect DAG. A distribution $P(\mathbf{X})$ admits a factorization of the form (1) with respect to $\mathcal{B}$

[4] As noted in Section 2, in that case the maximum conditional likelihood parameters may even be determined analytically.

if and only if it factorizes as

$$P(\mathbf{X}) = \prod_{c \in \mathcal{C}} \phi_c(\mathbf{X}), \tag{18}$$

where $\mathcal{C}$ is the set of cliques in $\mathcal{B}$ and the $\phi_c(\mathbf{X})$ are non-negative functions that depend on X only through the variables in clique c.

Recall that a *clique* is a fully connected subset of nodes. The set of cliques $\mathcal{C}$ appearing in (18) contains both maximal and non-maximal cliques (e.g., consisting of single nodes).

Proof. (of Theorem 3) We need to show that for an arbitrary parameter vector $\beta^{\mathcal{B}}$ for the logistic model, there are Bayesian network parameters that index the same distribution as the logistic model. Let $\theta^{\mathcal{B}}_{x_i|\mathbf{pa}_i}$ be the parameters obtained from (17), and define the normalizing constant $Z = \sum_x \prod_{i=0}^m \theta^{\mathcal{B}}_{x_i|\mathbf{pa}_i(\mathbf{x})}$. Define

$$\phi_i(\mathbf{x}) = \begin{cases} Z^{-1}\theta^{\mathcal{B}}_{x_0|\mathbf{pa}_0} & \text{if } i = 0 \\ \theta^{\mathcal{B}}_{x_i|\mathbf{pa}_i} & \text{if } 1 < i \leq m. \end{cases} \tag{19}$$

Consider the joint distribution $P_Z(\mathbf{X})$ defined by

$$P_Z(\mathbf{x} \mid \theta^{\mathcal{B}}) = \prod_{i=0}^m \phi_i(\mathbf{x}) = \frac{1}{Z} \prod_{i=0}^m \theta^{\mathcal{B}}_{x_i|\mathbf{pa}_i(\mathbf{x})}. \tag{20}$$

Note that, even though the product $\theta^{\mathcal{B}}_{x_i|\mathbf{pa}_i(\mathbf{x})}$ may not sum to one over all data vectors x, by introducing the normalizing constant Z, we ensure that the resulting $P_Z(\mathbf{x} \mid \theta^{\mathcal{B}})$ defines a probability distribution for $(X_1, \ldots, X_m)$. Distribution (20) induces the same conditionals for X_0 as the logistic model with parameter vector $\beta^{\mathcal{B}}$ given by (14). Each function $\phi_i(\mathbf{x})$ is non-negative and a function of the set $\{X_i\} \cup \mathbf{Pa}_i$, which is a clique by assumption. Thus (20) is a factorization of the form (18) and Proposition 2 implies that $P_Z(\mathbf{X})$ admits a factorization of the form (1) (the usual Bayesian network factorization). Thus there are Bayesian network parameters that satisfy the sum-to-one constraint and represent distribution $P_Z(\mathbf{X})$. In particular, those parameters give the same conditional distributions for X_0 as the logistic model. □

The proof is not constructive in that it does not explicitly give the Bayesian network parameters that give the same conditional distributions as the logistic model. A constructive proof is given by Wettig et al. (2003). We omitted it here because the present proof is much shorter, easier to understand and clarifies the connection to perfectness.

Together with Corollary 2 (stating that the conditional log-likelihood is concave), Theorem 3 shows that perfectness of $\mathcal{B}^*$ suffices to ensure that the conditional likelihood surface of $\mathcal{M}^{\mathcal{B}}$ has no local (non-global) maxima. This further implies that, for example, the conditional likelihood surface of TAN models has no local maxima. Therefore, a global maximum can be found by local optimization techniques.

But what about the case in which $\mathcal{B}^*$ is not perfect? Our second result, Theorem 4 (proven in the Appendix) says that in this case, there can be local maxima:

Theorem 4. There exist network structures $\mathcal{B}$ whose canonical form $\mathcal{B}^*$ is not perfect, and for which the conditional likelihood (6) has local, non-global maxima.

The theorem implies that $\mathcal{M}_L^{\mathcal{B}} \neq \mathcal{M}^{\mathcal{B}}$ for some network structures $\mathcal{B}$. In fact, it implies the stronger statement that for some structures $\mathcal{B}$, *no* logistic model indexes the same conditional distributions as $\mathcal{M}^{\mathcal{B}}$. The proof of this stronger statement is by contradiction: if $\mathcal{M}^{\mathcal{B}}$ coincided with some logistic regression model, the conditional likelihood surface in $\mathcal{M}^{\mathcal{B}}$ would not have local maxima - contradiction.

Thus, perfectness of $\mathcal{B}^*$ is not a superfluous condition. We may now ask whether it is a *necessary* condition for having $\mathcal{M}_L^{\mathcal{B}'} = \mathcal{M}^{\mathcal{B}}$ for some logistic model $\mathcal{M}_L^{\mathcal{B}'}$, with a (possibly different) network structure $\mathcal{B}'$. We plan to address this intriguing open question in future work.

5. Technical Issues

At this point we have all the tools together in order to build a logistic regression model equivalent to any given Bayesian classifier with underlying network structure $\mathcal{B}$ such that $\mathcal{B}$'s canonical version $\mathcal{B}^*$ is perfect. Its parameters may be determined using hill-climbing or some other local optimization method, such that the conditional log-likelihood is maximized. This results in a prediction method that in most cases outperforms the corresponding Bayesian classifier with ordinary maximum likelihood parameters. In practice, however, we find that there is a number of questions yet to be answered. In the following, we address some crucial technical details and outline the algorithms implemented.

5.1. Standard or logistic parameterization?

Because the mapping from the Bayesian parameters to the logistic model in the proof of Theorem 2 is continuous, it follows (with some

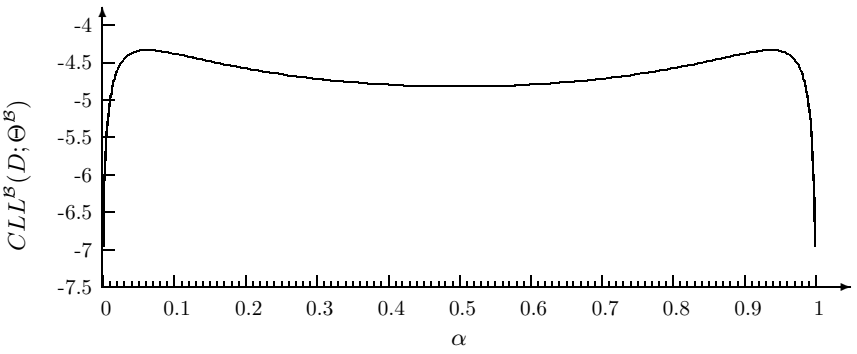

Figure 5. The conditional log-likelihood of the conditional Bayesian network of Example 3 peaks twice along a line defined by $\alpha \in (0, 1)$.

calculus) that if $\mathcal{M}^{\mathcal{B}} = \mathcal{M}_L^{\mathcal{B}}$, then all maxima of the (concave) conditional likelihood in the logistic parameterization are global (and connected) maxima also in the standard Bayesian parameterization. Thus we may also in the standard parameterization optimize the conditional log-likelihood locally to obtain a global maximum.

Nevertheless, as demonstrated in Example 3 below, the log-likelihood surface (6) as a function of $\theta^{\mathcal{B}}$ has some unpleasant properties: it is *not* concave in general and, what is worse, can have 'wrinkles': by these we mean convex *subsets* of the parameter space in which the likelihood surface does exhibit local, non-global maxima. This suggests that it is computationally preferable to optimize in the logistic parameterization rather than in the original Bayesian network parameterization, which is what we will do.

Example 3. Consider a Bayesian network where the class variable X_0 has only one child, X_1, and both variables take values in $\{1, 2\}$. Let the training data be given by $D = ((1, 1), (1, 2), (2, 1), (2, 2))$. Set the parameters $\theta^{\mathcal{B}}$ as follows:

$$\theta^{\mathcal{B}}_{x_0} = \begin{cases} 0.1 & \text{if} \quad x_0 = 1, \\ 0.9 & \text{if} \quad x_0 = 2, \end{cases} \quad ; \quad \theta^{\mathcal{B}}_{x_1|x_0} = \begin{cases} 0.5 & \text{if} \quad x_0 = 1, x_1 = 1, \\ 0.5 & \text{if} \quad x_0 = 1, x_1 = 2, \\ \alpha & \text{if} \quad x_0 = 2, x_1 = 1, \\ 1 - \alpha & \text{if} \quad x_0 = 2, x_1 = 2. \end{cases}$$

Figure 5 shows the conditional log-likelihood $CLL(D; \theta^{\mathcal{B}})$ given data D as a function of α. Note that the log-likelihood peaks twice along a straight line, contradicting concavity.

5.2. Continuous Predictor Variables

Bayesian classifiers have a built-in difficulty in handling continuous data since all conditional probability distributions are represented as *tables* of the form $\theta^{\mathcal{B}}_{x_i|\mathbf{pa}_i}$, see Section 2.2. Opposed to that, logistic regression models have no natural way of handling discrete data. But this shortcoming can easily be ironed out by introducing a covariate $Y_{x_i,\mathbf{pa}_i}$ for each instantiation $\mathbf{pa}_i$ of the parent set of a variable X_i as we have seen in Section 3.2.

For our experiments we discretized any continuous features based on the training data only, using the entropy-based method of Fayyad and Irani (1993). This way, the methods we compare will have the same (discrete) input.

Note, that logistic models are much more flexible than this and we have not yet exploited all their advantages. We could have fed them the original continuous values just as well, they would have not had any difficulty in combining this information with that of other, discrete attributes; we could have even handled mixed features (e.g. "died at age"$\in \{\text{alive}, 0, 1, \ldots\}$ etc.). It is also possible to discretize on the fly, i.e. to generate covariates of the form $Y_{\leq x_i} := \mathbf{I}_{[X_i \leq x_i]}$ as this seems beneficial for the discriminative model. One might even use a combination of the original continuous value of a feature and its discretized version by introducing a piece-wise log-linear function into the model. We leave the pursuit of these ideas as an objective for future research.

5.3. Model Selection

Usually we are not provided with a Bayesian network structure $\mathcal{B}$, but we are only given a data sample D, and somehow must choose a model (structure) $\mathcal{B}$ ourselves. How should we do this? This is a hard problem already when modelling the joint data distribution (Buntine, 1994; Heckerman, 1996; Myllymäki et al., 2002). Finding a good conditional model can be much harder than joint modelling, and many of the tools for modelling the joint distribution can no longer be used. For example, methods such as cross-validation or prequential validation become computationally highly demanding, since for each candidate network one has to re-optimize the model parameters. While optimizing the conditional log-likelihood of a single model is quite feasible for reasonable size data, optimization seems too computationally demanding for the task model selection. Using the joint data likelihood as a criterion is easier but may yield very poor results, while improvements have been achieved with heuristic 'semi-supervised' methods that mix supervised and unsupervised learning (Keogh and Pazzani, 1999; Kont-

kanen et al., 1999; Cowell, 2001; Shen et al., 2003; Madden, 2003; Raina et al., 2003; Grossman and Domingos, 2004).

For these reasons, we take a different approach. We start out with the naive Bayes classifier, the simplest model taking into account all data entries given. Our predictions are of the form (14), extended by independence. Since for naive Bayes models, the set $\mathbf{pa}_i^+$ is empty for all nodes X_i, we may denote $\beta_{x_0} := \beta_{x_0,\mathbf{pa}_0(\mathbf{x})}$ and $\beta_{x_0,x_i} := \beta_{x_0,x_i,\mathbf{pa}_0^+(\mathbf{x})}$. This becomes our first algorithm, *conditional naive Bayes* (cNB). Our second algorithm, *pruned naive Bayes* (pNB) will become a submodel of cNB, with a parameter selection scheme as described below. Both the cNB and the pNB models are parameterized in the logistic regression fashion.

We stress that the scope of our experiments is rather limited. Implementing a fully supervised, yet computationally feasible model selection method without severe restrictions on the range of network structures is a challenging open research topic.

5.4. Missing Data

Most standard classification methods—including Bayesian network classifiers and logistic regression—have been designed for the case where all training data vectors are complete, in the sense that they have no missing values. Yet in real-world data, missing data (feature values) appear to be the rule than the exception. Therefore we need to explicitly deal with this problem. The most general way of handling the issue is to treat 'missing' as a legitimate value for all features X_i whose value is missing in one or more data vectors (Myllymäki et al., 2002). This however, makes our model larger and thus may result in more overfitting, and therefore worse classification performance for small samples.

Instead, it is often assumed that the patterns of 'missingness' in the data do not provide any information about the class values. Mathematically, this can be expressed as follows: we assume that the true data generating distribution satisfies, for all class values x_0, all vectors $\mathbf{x}_{-i}$ with the ith element missing,

$$P(x_0 \mid \mathbf{x}_{-i}, x_i = \textit{'missing'}) = \sum_{x_i} P(x_i \mid \mathbf{x}_{-i}) P(x_0 \mid \mathbf{x}), \qquad (21)$$

where the sum is over all 'ordinary' values for X_i (excluding the value 'missing'). The same equation can be trivially extended to multiple missing values in $\mathbf{x}$.

While the assumption (21) is typically wrong, it often leads to acceptable results in practice. The related notion of 'missing completely at random' (MCAR) (see e.g. (Little and Rubin, 1987)) is a strictly

stronger requirement since it requires that x_i being missing gives no information about the joint distribution of $(X_0, \ldots, X_M)$, whereas we only require this to hold for the class variable X_0.

The proper way to implement (21) would to integrate out the missing entries. However, this makes the parameter learning (search) problem NP-complete. Therefore, we adjusted our learning method so as to achieve an approximation of (21), by effectively ignoring ('skipping') parameters corresponding to missing information, during both inference and prediction. More precisely, we introduce constraints of the form

$$\sum_{x_i} P(x_i \mid \mathbf{x}_{-i}) \beta_{x_0,i} = 0, \tag{22}$$

where we estimate the terms $P(x_i \mid \mathbf{x}_{-i})$ from the training data. As a result, our models will respect an approximation to (21), namely its logarithmic version

$$\log P(x_0 \mid \mathbf{x}_{-i}, x_i = \textit{'missing'}, \beta) = \sum_{x_i} P(x_i \mid \mathbf{x}_{-i}) \log P(x_0 \mid \mathbf{x}, \beta), \tag{23}$$

This way, skipping all parameters corresponding to missing values results in *logarithmically unbiased* predictive distributions, which judged on the basis of our experiments reported in Section 6 seem to be a good enough approximation in practice.

5.5. Priors

In practical applications, a sample D will typically include zero frequencies, i.e. $n_{[x_0,x_i]} = 0$ for some x_0, x_i. In that case, the conditional log-likelihood $CLL(D; \beta)$ given by (16) will have no maximum over β, but some β_{x_0,x_i} will diverge to $-\infty$. The same problem can arise in more subtle situations as well, see Example 4 in (Wettig et al., 2002).

We can avoid such problems by introducing a Bayesian *prior distribution* on the conditional model $\mathcal{B}$ (Bernardo and Smith, 1994). Kontkanen et al. (2000) have shown that for ordinary, unsupervised naive Bayes, whenever we are in danger of over-fitting the training data—usually with small sample sizes—prediction performance can be *greatly* improved by imposing a prior on the parameters. Since the conditional model cNB is inclined to worse over-fitting than unsupervised naive Bayes (Ng and Jordan, 2001), this should hold also in our case.

We impose a strictly concave prior that goes to zero as the absolute value of any parameter approaches infinity. We choose the closest we can find to the *uniform*, least informative prior in the usual parameterization where parameters take values between zero and one. We retransform the parameters β_{x_0} and β_{x_0,x_i} back into the space of

probability distributions (by taking their normalized exponentials) and define their prior probability density, $P(\beta)$, to be proportional to the product of all entries in the resulting distributions.

Instead of the conditional log-likelihood CLL we optimize the 'conditional posterior' (Grünwald et al., 2002):

$$CLL^+(D;\beta) := CLL(D;\beta) + \ln P(\beta). \tag{24}$$

Using this prior also yields *strict* concaveness of CLL^+ as the sum of a concave and a strictly concave function, which guarantees a unique maximum.

5.6. Algorithms

We now fill in the missing details of our algorithms, and explain how the actual optimization is performed. The conditional naive Bayes algorithm maximizes $CLL^+(D;\beta)$ using component-wise binary search until convergence. Although we can compute all first and second derivatives, (9) and (10), we found that this takes so much computation time that the benefit in convergence speed obtained by using more sophisticated methods such as Newton-Raphson or conjugate gradient ascent is lost. Minka (2001) suggests and compares a number of algorithms for this task. In our case the simplest of them, coordinate-wise line search, seems to suffice.

Our second method, the pruned naive Bayes classifier pNB aims at preventing over-fitting. We prune the full naive Bayes model cNB by maximizing the same objective but with the additional freedom to exclude some parameters from the model. These are being ignored in the same way any parameter is when the corresponding data entry is missing:

$$CLL^+(D;\beta) = \sum_j \ln \left(\frac{\exp(\beta_{x_0} + \sum_{i:\beta_{x_0,x_i^j} \in \beta^j} \beta_{x_0,x_i^j})}{\sum_{x_0'=1}^{n_0} \exp(\beta_{x_0'} + \sum_{i:\beta_{x_0',x_i^j} \in \beta^j} \beta_{x_0',x_i^j})} \right) + \ln P(\beta), \tag{25}$$

where β^j is defined to be the set of parameters that apply to vector x^j: $\beta^j := \{\beta_{x_0,x_i} \in \beta : x_i^j \neq \textit{'missing'}\}$.

Note that although zero valued parameters have no effect on the conditional likelihood, the corresponding prior term is not zero, so that any parameter that is chosen to be part of the model has an associated cost to it. This defines a natural threshold: a parameter β_{x_0,x_i} that does

not improve the conditional *log-likelihood* by at least $-\ln P(\beta_{x_0,x_i}) \geq -\ln P(0)$ will be removed from the model, since it will deteriorate the *log-posterior*.

The pNB algorithm is quite simple. We start out with the full cNB model and eliminate parameters one at a time until no improvement is achieved. To speed up the process, we choose the next parameter to be dropped so that $CLL^+(D;\beta)$ is maximized without re-optimizing the remaining parameters. We re-optimize only after choosing which parameter to drop. When there is no parameter left whose exclusion from the model yields direct gain, we choose the parameter causing the least loss. If after re-optimizing the system the objective still has not improved, we undo the last step and the algorithm terminates.

6. Empirical Results

We compare our methods against conventional naive Bayes and against each other. Our experimental methodology resembles that of Madden (2003) and Friedman et al. (1997). We split each data set into disjunct train and test sets at random such that the training set contains 80% of the original data set and the test set the remaining 20%. This we do 20 times independently of the previous splits. We report both the average log-loss and 0/1-loss scores achieved and their 95 percent confidence intervals.

As mentioned in Section 5.1, we discretize continuous features using the entropy-based method of Fayyad and Irani (1993). This is done using the training data only, so that for each random split this results in possibly different discretizations. Data vectors with missing entries were included in the tests. As a test bed we took 18 data sets from the UCI Machine Learning Repository (Blake and Merz, 1998), ten of which contain missing data. In the tables below, these are marked by an asterisk '$*$'.

Table I lists the data sets used, their sizes and the number of parameters used by the different algorithms. The NB and cNB models, although parameterized differently, obviously contain the same number of parameters. Variance in this number is due to individual discretizations on each training set. Note the drastic pruning performed by pNB.

We list the log-scores achieved by the algorithms NB, cNB and pNB in Table II. For comparison we also report the results of the default predictor (class node independent of everything else) which—as also the NB algorithm—has been equipped with a uniform prior on its parameters. The default gives some clue about how hard it is to learn

Table I. Data sets used, and the numbers of parameters in models learned.

Data set	Size	#Classes	NB/cNB	pNB
Balance Scale	625	3	63	32.25±3.81
BC Wisconsin*	699	2	180	13.15±4.07
Congr. Voting*	435	2	66	7.30±1.32
CRX*	690	2	106.30±1.61	10.65±1.95
Ecoli	336	8	124.40±16.24	22.05±5.76
Glass Ident.	214	6	126.90±15.78	16.95±6.57
HD Cleveland*	303	5	171.75±5.52	14.60±5.78
HD Hungarian*	294	2	62.10±.74	8.25±2.06
HD Switzerland*	123	5	158.00±6.75	4.55±1.25
HD VA*	200	5	157.75±7.30	6.40±1.80
Hepatitis*	155	2	78.80±1.97	4.30±1.70
Iris	150	3	33.45±1.81	5.10±1.59
Mushrooms*	8124	2	234	29.00±15.40
Pima Diabetes	768	2	34.80±1.97	7.80±2.48
Postoperative*	90	4	96	2
Tic-Tac-Toe	958	2	56	37.65±0.81
Waveform	5000	3	215.70±8.61	96.40±30.49
Wine Rec.	178	3	88.20±6.48	9.45±3.27

from the predictors; where it is hard to even beat the default predictor, there may not be much information in the features about the class. On the other hand, high variance in the default may indicate great effect of the random splits on how hard the prediction task will be. The winning scores are typeset in boldface.

In terms of log-loss, both discriminative models cNB and pNB clearly outperform standard naive Bayes. In some cases standard naive Bayes does slightly better while more often it is outperformed by the supervised methods with much greater margin. This is the case especially on large data sets (e.g., Mushrooms, Waveform) and whenever the independence assumptions of the naive Bayes model are badly violated (e.g., Balance Scale, Congressional Voting, Tic-Tac-Toe). This behavior is natural and has been reported already by Greiner and Zhou (2002) and Wettig et al. (2002). Figures 6.a. and 6.c. illustrate the numerical results.

Observe how the pNB algorithm chooses only about one out of six parameter candidates, while its performance is comparable to that of cNB; see also Figure 6.e. In addition, pNB seems to be more robust in that it tends to yield better results where cNB can be assumed to over-fit (i.e. cNB loses against NB or even the default), while losing little in those cases where cNB has better performance. Note also that, regardless of its suboptimal search method, pNB is also more stable in terms of variance in its performance on different data splits. With

Table II. Predictive accuracies with respect to logarithmic loss.

Data set	Default	NB	cNB	pNB
Balance Scale	.925±.057	.528±.078	**.214**±.067	.228±.081
BC Wisconsin*	.646±.047	.261±.195	.140±.078	**.113**±.050
Congr. Voting*	.675±.033	.577±.359	**.112**±.081	.121±.064
CRX*	.688±.012	.396±.134	**.332**±.099	.339±.078
Ecoli	1.545±.149	.448±.227	**.445**±.236	.488±.214
Glass Ident.	1.546±.181	.828±.271	**.784**±.184	.937±.222
HD Cleveland*	1.304±.185	**1.023**±.322	1.125±.350	1.053±.232
HD Hungarian*	.647±.041	.403±.226	**.366**±.128	.407±.211
HD Switzerland*	1.392±.157	1.468±.299	1.652±.349	**1.367**±.149
HD VA*	1.542±.074	**1.500**±.166	1.613±.207	1.544±.135
Hepatitis*	.532±.160	.434±.296	**.409**±.267	.413±.177
Iris	1.110±.018	**.091**±.123	.112±.091	.111±.075
Mushrooms*	.693±.001	.129±.029	**.001**±.001	.002±.001
Pima Diabetes	.647±.036	**.433**±.059	.442±.043	.456±.041
Postoperative*	.714±.213	.808±.310	.927±.337	**.707**±.219
Tic-Tac-Toe	.650±.029	.554±.045	**.090**±.030	**.090**±.030
Waveform	1.099±.001	.634±.082	**.296**±.027	.299±.025
Wine Rec.	1.104±.048	**.014**±.034	.028±.033	.053±.051

very few parameters the pruned logistic model achieves good results. Interestingly, on the Postoperative data-set, from which it seems to be very hard to learn anything about its class, pNB constantly chooses only two parameters which leads to better performance than using the full model. On the other hand, for data-set Tic-Tac-Toe, pNB chooses a large fraction of the available parameters behaving no different from cNB on all splits.

Table III is the counterpart of Table II, reporting the results of the same test runs in terms of 0/1-loss. Figures 6.b., 6.d. and 6.f. compare the classification errors of the three algorithms. Naive Bayes seems to do relatively better under the 0/1-loss, but note that again it wins by smaller margins than those it loses by on other data sets. The logistic models cNB and pNB are quite comparable also in terms of classification accuracy.

7. Conclusions

The focus of this paper is in discriminative learning of models from sample data, where the goal is to determine the model parameters maximizing the conditional (supervised) likelihood instead of the commonly used joint (unsupervised) likelihood. In the theoretical part of the paper we showed that for Bayesian network models satisfying a

Table III. Percentages of correct predictions.

Data set	Default	NB	cNB	pNB
Balance Scale	42.36±4.08	90.24±3.27	**93.20**±3.56	92.60±3.98
BC Wisconsin*	65.61±7.19	**97.22**±1.87	95.82±2.03	95.04±2.68
Congr. Voting*	60.12±6.44	89.89±5.71	**95.92**±3.33	95.75±3.01
CRX*	55.65±5.35	**87.36**±4.30	85.98±4.93	85.11±4.12
Ecoli	43.16±10.09	**85.29**±8.60	85.00±8.22	82.87±8.46
Glass Ident.	31.51±8.90	**69.54**±11.57	**69.54**±11.64	65.12±13.93
HD Cleveland*	54.51±10.85	**58.77**±9.87	58.61±11.54	58.69±10.74
HD Hungarian*	65.59±7.89	**85.34**±7.02	84.41±6.56	83.22±7.67
HD Switzerland*	**38.40**±12.88	36.00±14.16	34.60±13.55	34.00±14.40
HD VA*	23.50±10.54	**33.75**±13.04	30.00±13.07	25.38±9.83
Hepatitis*	78.23±11.27	**86.78**±8.24	84.68±9.09	81.94±9.95
Iris	27.17±5.42	**95.67**±5.93	**95.67**±5.07	**95.67**±5.07
Mushrooms*	51.69±1.78	95.51±0.92	**100.00**	99.98±0.11
Pima Diabetes	65.26±5.73	**79.16**±4.69	78.51±3.89	78.15±3.61
Postoperative*	**72.22**±13.91	68.89±11.64	63.89±13.43	**72.22**±13.91
Tic-Tac-Toe	64.79±4.57	69.72±4.42	**98.10**±1.48	**98.10**±1.48
Waveform	33.50±1.79	82.20±1.84	**86.69**±1.69	86.60±1.68
Wine Rec.	37.36±12.71	**99.58**±1.68	99.17±2.15	99.03±2.69

simple graph-theoretic condition, this problem is equivalent to a logistic regression problem. Bayesian network structures satisfying this condition include the naive Bayes model and the tree-augmented naive Bayes model, but the condition allows also other non-trivial network structures. It remains an interesing open problem whether the condition is also necessary so that Bayesian networks violating the condition can not be represented as any logistic regression model.

In the empirical part of the paper we exploited the theoretical results obtained and experimented with two discriminative models. The first model was a conditional version of the naive Bayes model with the parameters optimized with respect to the conditional likelihood. The second model added a heuristic procedure for selecting the set of parameters and relevant features used. In both cases the theoretical results offer a parameterization under which the conditional likelihood has only one global maximum so that finding the maximizing discriminative parameters was in principle easy, although computationally more demanding than using parameters maximizing the joint likelihood.

The empirical results were contrasted to those obtained with the standard naive Bayes classifier. The results demonstrate that the discriminative models typically give better predictive accuracy, in particular with respect to the logarithmic loss. What is more, the parameter pruning algorithm introduced yields models that are much simpler

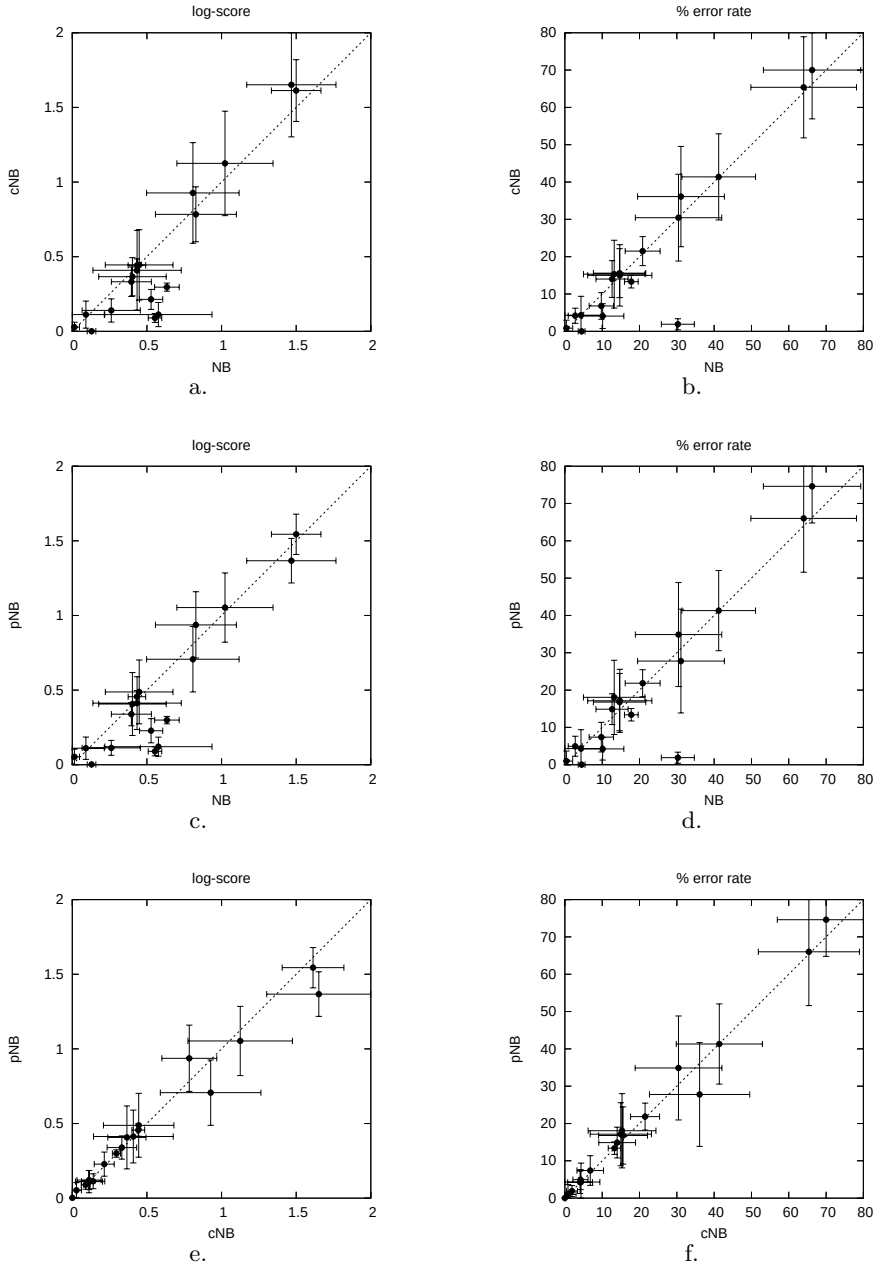

Figure 6. Pairwise comparison of algorithms NB and cNB, (a) and (b), NB and pNB, (c) and (d), and cNB and pNB, (e) and (f), in terms of log-loss, (a), (c), (e), and misclassification percentage, (b), (d), (f), for the 18 UCI data sets used as our test bed.

than the naive Bayes classifier or its discriminative version, without a significant decrease in the accuracy.

The fact that for many interesting Bayesian network structures, the conditional likelihood function has only one global maximum, is practically important as it means that the discriminative parameters can be found by local optimization methods. Hence there is no need to apply computationally elaborate techniques for finding these parameters. Furthermore, the result suggests that if one wishes to use the naive Bayes classifier as a straw man method to which alternative approaches are to be compared, as is often the case, one could equally well use an even better straw man method offered by the supervised version of the naive Bayes model.

On the other hand, the results may have implications with respect to the model selection problem in supervised domains: many model selection criteria typically contain the data likelihood as one of the factors of the criterion—cf. for example, the Bayesian Information Criterion (BIC) (Schwarz, 1978) or approximations of the Minimum Description Length (MDL) criterion (Rissanen, 1996). Therefore, it is natural to assume that the conditional likelihood plays an important role in the supervised versions of the model selection criteria. This aspect will be addressed more formally in our future work.

Acknowledgments

The authors thank the anonymous reviewers for useful comments. This work was supported in part by the Academy of Finland under project Cepler and the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778. This publication only reflects the authors' views.

Appendix: Proofs

Proof (sketch) of Theorem 4. Use the rightmost network in Figure 3 with structure $X_0 \rightarrow X_2 \leftarrow X_1$. Let the data be

$$D = ((1,1,1),(1,1,2),(2,2,1),(2,2,2)) \,. \tag{26}$$

We are interested in predicting the value of X_0 given X_1 and X_2. The parameter defining the distribution of X_1 has no effect on conditional predictions and we can ignore it. For the remaining five parameters we

use the following notation:

$$\begin{aligned} \theta_2 &:= P(X_0 = 2), \\ \theta_{2|1,1} &:= P(X_2 = 2 \mid X_0 = 1, X_1 = 1), \\ \theta_{2|1,2} &:= P(X_2 = 2 \mid X_0 = 1, X_1 = 2), \\ \theta_{2|2,1} &:= P(X_2 = 2 \mid X_0 = 2, X_1 = 1), \\ \theta_{2|2,2} &:= P(X_2 = 2 \mid X_0 = 2, X_1 = 2). \end{aligned} \tag{27}$$

Idea of the Proof. In the empirical distribution based on data D, X_0 is highly (even perfectly) dependent on X_1, even given the value of X_2. Such a perfect dependence cannot be represented by any of the distributions in $\mathcal{M}^{\mathcal{B}}$, since the network structure implies that conditioned on X_2, X_1 and X_0 must be independent. However, the parameters $\theta_{2|1,2}$ and $\theta_{2|2,1}$ correspond to contexts that do not occur in D and this fact can be exploited: by setting $\theta_{2|2,1}$ to 0 and $\theta_{2|1,2}$ to 0, we can represent distributions θ with $P(X_0 = X_1 \mid X_2 = 2, \theta) = 1$ and $P(X_0 = X_1 \mid X_2 = 1, \theta) = 0.5 \pm \epsilon$, for any $\epsilon > 0$. These distributions represent some of the dependence between X_0 and X_1 after all and, for $\epsilon \to 0$, converge to the maximum conditional likelihood. However, by setting $\theta'_{2|2,1}$ to 1 and $\theta'_{2|1,2}$ to 1, we can represent distributions θ with $P(X_0 = X_1 \mid X_2 = 1, \theta') = 1$ and $P(X_0 = X_1 \mid X_2 = 2, \theta') = 0.5 \pm \epsilon$ which also converge to the maximum conditional likelihood as $\epsilon \to 0$. In Part I of the proof we formalize this argument and show that with data (26), there are four non-connected suprema of the conditional likelihood. In Part II, we sketch how the argument can be extended to allow for non-global maxima (rather than suprema).

Part I. The conditional log-likelihood can be written as

$$CLL(D; \theta^{\mathcal{B}}) = g(1 - \theta_2, \theta_{2|1,1}, \theta_{2|2,1}) + g(\theta_2, \theta_{2|2,2}, \theta_{2|1,2}), \tag{28}$$

where

$$g(x, y, z) := f(x, y, z) + f(x, 1 - y, 1 - z), \tag{29}$$

and

$$f(x, y, z) := \ln \frac{xy}{xy + (1 - x)z}. \tag{30}$$

Figure 7 illustrates function $g(x, y, z)$ at $x = 0.5$. In (28) each parameter except θ_2 appears only once. Thus, for a fixed θ_2 we can maximize each term separately. We can now apply Lemma 3 below with $\alpha = 1/2$, so that $g_\alpha(x, y, z) = 2g(x, y, z)$ for the g defined in (29). It follows from the lemma that the supremum of the log-likelihood with θ_2 fixed is $\ln(1-\theta_2)+\ln(\theta_2)$, which achieves its maximum value $-2 \ln 2$ at $\theta_2 = 0.5$. Furthermore, the lemma shows that the log-likelihood approaches its

supremum when $\theta_{2|2,1} \in \{0,1\}$, $\theta_{2|1,2} \in \{0,1\}$, $\theta_{2|1,1} \to \theta_{2|2,1}$, and $\theta_{2|2,2} \to \theta_{2|1,2}$. Moreover, by item (ii) of the lemma, these suprema are separated by areas where the log-likelihood is smaller, i.e., the suprema are local and not connected.

Part II. To conclude the proof we still need to address two issues: (a) the four local suprema give the same conditional log-likelihood $-2\ln 2$, and (b), they are suprema, not maxima (not achieved by any $\theta^{\mathcal{B}}$). We now roughly sketch how to extend the argument to deal with these issues. Concerning (a), fix some $0 < \alpha < 1$ and consider a sample D' consisting of N data vectors, with $\alpha N/2$ repetitions of $(1,1,1)$, $\alpha N/2$ repetitions of $(2,2,1)$, $(1-\alpha)N/2$ repetitions of $(1,1,2)$ and $(1-\alpha)N/2$ repetitions of $(2,2,2)$. Using Lemma 3 in the same way as before, we find that the conditional log-likelihood has four local suprema which are not connected. Moreover, if $\alpha \neq 1/2$, then at least two of these suprema are not equal.

Concerning (b), let D'' be the sample D' but with four extra 'barrier' data vectors $(1,2,1),(2,1,1),(1,2,2),(2,1,2)$ added. Let C'' be the corresponding conditional log-likelihood, $C''(\theta^{\mathcal{B}}) := CLL(D''; \theta^{\mathcal{B}})$. If either of the five parameters $\theta_2, \theta_{2|1,1}, \theta_{2|2,1}, \theta_{2|2,2}, \theta_{2|1,2} \in \{0,1\}$, then $C''(\theta^{\mathcal{B}}) = -\infty$. On the other hand, C'' is continuous and finite for $(\theta_2, \theta_{2|1,1}, \theta_{2|2,1}, \theta_{2|2,2}, \theta_{2|1,2}) \in (0,1)^5$, so C'' must achieve its maximum or maxima for each N. On the other hand, for large N, the influence of the barrier data vectors C'' at each individual point $\vec{\theta}^{\mathcal{B}}$ becomes negligible. Using Lemma 3, item (iii), these two facts can be exploited to show that for large N, C'' has (at least) two maxima, which, if $\alpha \neq 1/2$, are neither equal nor connected. We omit further details.

Lemma 3. Define $g_\alpha(x,y,z) := \alpha f(x,y,z) + (1-\alpha) f(x, 1-y, 1-z)$, where $f(x,y,z)$ is defined as in (30). Fix some $0 < x < 1$ and $0 < \alpha < 1$. With y and z both varying between 0 and 1, we have:

i. The global supremum of $g_\alpha(x,y,z)$ satisfies

$$\sup_{0 \leq y,z \leq 1} g_\alpha(x,y,z) = \sup\{\alpha \ln x, (1-\alpha)\ln x\}.$$

ii. The local suprema are at $z = 0, y \downarrow 0$ and at $z = 1, y \uparrow 1$:

$$\lim_{y \downarrow 0} g_\alpha(x,y,0) = (1-\alpha)\ln x \quad ; \quad \lim_{y \uparrow 1} g_\alpha(x,y,1) = \alpha \ln x.$$

iii. For restricted z we have:

$$\begin{aligned} \sup_{0 \leq y,z \leq 1, z=y} g_\alpha(x,y,z) &= \ln x \\ \sup_{0 \leq y,z \leq 1, z<y} g_\alpha(x,y,z) &= (1-\alpha)\ln x \\ \sup_{0 \leq y,z \leq 1, z>y} g_\alpha(x,y,z) &= \alpha \ln x. \end{aligned}$$

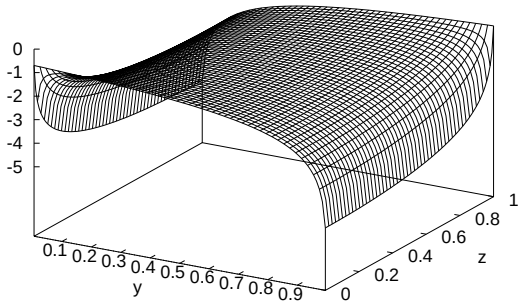

Figure 7. Function g(x,y,z) given by (29) with $x = 0.5$.

Proof. Differentiating twice wrt. z gives

$$\frac{\partial^2}{\partial^2 z} g_\alpha(x,y,z) = \frac{\alpha(1-x)^2}{(xy+(1-x)z)^2} + \frac{(1-\alpha)(1-x)^2}{(x(1-y)+(1-x)(1-z))^2},$$

which is always positive and the function achieves its maximum value either at $z = 0$ or $z = 1$ or both. At these two points differentiating wrt. y yields

$$\frac{\partial}{\partial y} g_\alpha(x,y,0) = \frac{(1-\alpha)(x-1)}{(1-y)(1-xy)} \; ; \; \frac{\partial}{\partial y} g_\alpha(x,y,1) = \frac{\alpha(1-x)}{y(xy+1-x)}. \tag{31}$$

Since in the first case the derivative is always negative, and in the second case the derivative is always positive, $g_\alpha(x,y,0)$ increases monotonically as $y \to 0$, and $g_\alpha(x,y,1)$ increases monotonically as $y \to 1$. Denoting

$$L_0 = \lim_{y \downarrow 0} g_\alpha(x,y,0) = (1-\alpha)\ln x \quad \text{and} \quad L_1 = \lim_{y \uparrow 1} g_\alpha(x,y,1) = \alpha \ln x,$$

this implies that for each (y_0, z_0) with $0 \leq y_0, z_0 \leq 1$, either $L_0 > g_\alpha(y_0,z_0)$ or $L_1 > g_\alpha(y_0,z_0)$. Item (i) now follows. By inspecting the first derivative of $g_\alpha(x,y,z)$ wrt. z, we see that there exists some $y^* > 0$ (depending on x and α) such that, for all $0 \leq y < y^*$, $g_\alpha(x,y,z)$ increases monotonically as $z \to 0$. Since by (31), $g_\alpha(x,y,0)$ increases monotonically as $y \to 0$, we have for all (x,y) in a neighborhood of $(0,0)$ that $g_\alpha(x,y,z) < L_0$. It follows that L_0 is a local supremum. The proof that $g_\alpha(x,y,z) < L_1$ in a neighborhood of $(1,1)$ is analogous, concluding item (ii).

The first equation of item (iii) follows upon calculating $g_\alpha(x, y, y)$ and noting that the result $\ln x$ is independent of y. The second equation follows since, as we already showed, the second derivative of $g_\alpha(x, y, z)$ wrt. z is always positive, so that, for fixed y, $\sup_{0 \leq z < y} g_\alpha(x, y, z) = \max\{g_\alpha(x, y, 0), g_\alpha(x, y, y)\}$. The supremum of this expression over y is achieved for $y \downarrow 0$ and equal to L_0. The third equation is proved similarly. □

Example 4. In order to see what conditional predictions result from the maximum likelihood parameters for data set (26), consider, as in the proof of Theorem 4, the parameter configuration $\tilde{\theta}_{\text{cond}} = (\theta_2, \theta_{2|1,1}, \theta_{2|1,2}, \theta_{2|2,1}, \theta_{2|2,2})$ with $\theta_2 = 0.5, \theta_{2|2,1} = \theta_{2|1,2} = 0$, and $\theta_{2|1,1} = \theta_{2|2,2} = \epsilon$ with $\epsilon > 0$ small. If $X_2 = 1$, the conditional probability that $X_0 = X_1$, given X_1 and X_2, is close to 0.5, whereas if $X_2 = 2$ the conditional probability of $X_0 = X_1$ is one. In contrast, with the *unconditional* maximum likelihood parameters $\tilde{\theta}$, the conditional probability of $X_0 = X_1$ is always 0.5.

Suppose now data are i.i.d. according to some distribution Q which puts uniform probability 1/4 on each of the data vectors in (26). By the law of large numbers, with Q-probability 1, the unconditional ML parameters for the sample $D = (\mathbf{x}^1, \ldots, \mathbf{x}^N)$ will converge to $\tilde{\theta}$, whereas the conditional ML parameters will achieve one of their maxima at $\tilde{\theta}_{\text{cond}}$ with ϵ infinitesimally close to 0. The arguments of the proof of Theorem 4 show that:

$$D_{\text{cond}}(Q \| P(\cdot \mid \tilde{\theta})) = \ln 2 \; ; \quad D_{\text{cond}}(Q \| P(\cdot \mid \tilde{\theta}_{\text{cond}})) = \frac{1}{2} \ln 2,$$

with D_{cond} denoting the conditional KL-divergence, defined as in (7). This example shows that there exist data generating distributions for which the conditional likelihood is far superior to the unconditional likelihood.

References

Barndorff-Nielsen, O. (1978). *Information and Exponential Families in Statistical Theory.* New York, NY: John Wiley & Sons.

Bernardo, J., & Smith, A. (1994). *Bayesian theory.* New York, NY: John Wiley & Sons.

Bishop, C. (1995). *Neural Networks for Pattern Recognition.* Oxford, UK: Oxford University Press.

Blake, C., & Merz, C. (1998). UCI repository of machine learning databases. http://www.ics.uci.edu/∼mlearn/MLRepository.html. University of California, Irvine, Department of Information and Computer Science.

Buntine, W. (1994). Operations for learning with graphical models. *Journal of Artificial Intelligence Research, 2*, 159–225.

Cowell, R. (2001). On searching for optimal classifiers among Bayesian networks. In T. Jaakkola, & T. Richardson (Eds.), *Proceedings of the Eight International Workshop on Artificial Intelligence and Statistics* (pp. 175–180). San Francisco, CA: Morgan Kaufmann.

Dawid, A. (1976). Properties of diagnostic data distributions. *Biometrics, 32:3*, 647–658.

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

Fayyad, U., & Irani, K. (1993). Multi-interval discretization of continuous-valued attributes for classification learning. In R. Bajcsy (Ed.), *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence* (pp. 1022–1027). San Francisco, CA: Morgan Kaufmann.

Friedman, N., Geiger, D., & Goldszmidt, M. (1997). Bayesian network classifiers. *Machine Learning, 29:2*, 131–163.

Greiner, R., Grove, A., & Schuurmans, D. (1997). Learning Bayesian nets that perform well. In D. Geiger, & P. P. Shenoy (Eds.), *Proceedings of the Thirteenth Annual Conference on Uncertainty in Artificial Intelligence* (pp. 198–207). San Francisco, CA: Morgan Kaufmann.

Greiner, R., & Zhou, W. (2002). Structural extension to logistic regression: discriminant parameter learning of belief net classifiers. In R. Dechter, M. Kearns, & R. Sutton (Eds.), *Proceedings of the Eighteenth National Conference on Artificial Intelligence* (pp. 167–173). Cambridge, MA: MIT Press.

Grossman, D., & Domingos, P. (2004). Learning Bayesian network classifiers by maximizing conditional likelihood. In C. E. Brodley (Ed.), *Proceedings of the Twenty-first International Conference on Machine Learning* (pp. 361–368). Madison, WI: Omnipress.

Grünwald, P., Kontkanen, P., Myllymäki, P., Roos, T., Tirri, H., & Wettig, H. (2002). Supervised posterior distributions. Presented at the Seventh Valencia International Meeting on Bayesian Statistics, Tenerife, Spain.

Heckerman, D. (1996). A tutorial on learning with Bayesian networks. Technical Report MSR-TR-95-06, Microsoft Research, Redmond, WA.

Heckerman, D., & Meek, C. (1997a). Embedded Bayesian network classifiers. Technical Report MSR-TR-97-06, Microsoft Research, Redmond, WA.

Heckerman, D., & Meek, C. (1997b). Models and selection criteria for regression and classification. In D. Geiger, & P. P. Shenoy (Eds.), *Proceedings of the Thirteenth Annual Conference on Uncertainty in Artificial Intelligence* (pp. 198–207). San Francisco, CA: Morgan Kaufmann.

Jebara, T. (2003). *Machine Learning: Discriminative and generative*. Boston, MA: Kluwer Academic Publishers.

Keogh, E., & Pazzani, M. (1999). Learning augmented Bayesian classifiers: a comparison of distribution-based and classification-based approaches. In D. Heckerman, & J. Whittaker (Eds.), *Proceedings of the Seventh International Workshop on Artificial Intelligence and Statistics* (pp. 225–230). San Francisco, CA, Morgan Kaufmann.

Kontkanen, P., Myllymäki, P., Silander, T., & Tirri, H. (1999). On supervised selection of Bayesian networks. In K. B. Laskey, & H. Prade (Eds.), *Proceedings of the Fifteenth International Conference on Uncertainty in Artificial Intelligence* (pp. 334–342). San Francisco, CA: Morgan Kaufmann Publishers.

Kontkanen, P., Myllymäki, P., Silander, T., Tirri, H., & Grünwald, P. (2000). On predictive distributions and Bayesian networks. *Statistics and Computing 10:1*, 39–54.

Kontkanen, P., Myllymäki, P., & Tirri, H. (2001). Classifier learning with supervised marginal likelihood. In J. S. Breese, & D Koller (Eds.), *Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence* (pp. 277-284). San Francisco, CA: Morgan Kaufmann.

Lauritzen, S. (1996). *Graphical Models*. Oxford, UK: Oxford University Press.

Little, R., & Rubin, D. (1987), *Statistical Analysis with Missing Data*. New York, NY: John Wiley & Sons.

Madden, M. (2003). The performance of Bayesian network classifiers constructed using different techniques. In *Working Notes of the ECML/PKDD-03 Workshop on Probabilistic Graphical Models for Classification* (pp. 59–70).

McLachlan, G. (1992). *Discriminant Analysis and Statistical Pattern Recognition*. New York, NY: John Wiley & Sons.

McLachlan, G. & Krishnan, T. (1997). *The EM Algorithm and Extensions*. New York, NY: John Wiley & Sons.

Minka, T. (2001). Algorithms for maximum-likelihood logistic regression. Technical Report 758, Carnegie Mellon University, Department of Statistics. Revised Sep. 2003.

Myllymäki, P., Silander, T., Tirri, H., & Uronen, P. (2002). B-Course: a web-based tool for Bayesian and causal data analysis. *International Journal on Artificial Intelligence Tools 11:3*, 369–387.

Ng, A., & Jordan, M. (2001). On discriminative vs. generative classifiers: a comparison of logistic regression and naive Bayes. In T. G. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in Neural Information Processing Systems 14* (pp. 605–610). Cambridge, MA: MIT Press.

Pearl, J. (1987). Evidential reasoning using stochastic simulation of causal models. *Artificial Intelligence 32:2*, 245–258.

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA: Morgan Kaufmann.

Raina, R., Shen, Y., Ng, A. Y., & McCallum, A. (2003). Classification with hybrid generative/discriminative models. In S. Thrun, L. Saul, & B. Schölkopf (Eds.), *Advances in Neural Information Processing Systems 16*. Cambridge, MA: MIT Press.

Rissanen, J. (1996). Fisher information and stochastic complexity. *IEEE Transactions on Information Theory 42:1*, 40–47.

Russell, S., Binder, J., Koller, D., & Kanawaza, K. (1995). Local learning in probabilistic networks with hidden variables. In S. Thrun, & T. M. Mitchell (Eds.), *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence* (pp. 1146–1152). San Francisco, CA: Morgan Kaufmann Publishers.

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics 6:2*, 461–464.

Shen, B., Su, X., Greiner, R., Musilek, P., & Cheng, C. (2003). Discriminative parameter learning of general Bayesian network classifiers. In *Proceedings of the 15th IEEE International Conference on Tools with Artificial Intelligence* (pp. 296–305). Los Alamitos, CA: IEEE Computer Society Press.

Thiesson, B. (1995). Accelerated quantification of Bayesian networks with incomplete data. In U. M. Fayyad, & R. Uthurusamy (Eds.), *Proceedings of the First International Conference on Knowledge Discovery and Data Mining* (pp. 306–311). Cambridge, MA: MIT Press.

Wettig, H., Grünwald, P., Roos, T., Myllymäki, P., & Tirri, H. (2002). Supervised naive Bayes parameters. In P. Ala-Siuru, & S. Kaski (Eds.), *Proceedings of the Tenth Finnish Artificial Intelligence Conference* (pp. 72–83). Oulu, Finland: Finnish Artificial Intelligence Society.

Wettig, H., Grünwald, P., Roos, T., Myllymäki, P., & Tirri, H. (2003). When discriminative learning of Bayesian network parameters is easy. In G. Gottlob, & T. Walsh (Eds.), *Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence* (pp. 491–496). San Francisco, CA: Morgan Kaufmann Publishers.