Discriminative Model Selection for Belief Net Structures

Yuhong Guo and **Russ Greiner**
Department of Computing Science
University of Alberta
Edmonton, Alberta T6G 2E8
{ yuhong, greiner }@cs.ualberta.ca

Abstract

Bayesian belief nets (BNs) are often used for classification tasks, typically to return the most likely class label for a specified instance. Many BN-learners, however, attempt to find the BN that maximizes a different objective function — viz., likelihood, rather than classification accuracy — typically by first using some *model selection criterion* to identify an appropriate graphical structure, then finding good parameters for that structure. This paper considers a number of possible criteria for selecting the best structure, both generative (*i.e.*, based on likelihood; *BIC*, *BDe*) and discriminative: Conditional *BIC* (*CBIC*), resubstitution Classification Error (*CE*) and Bias^2+Variance (*BV*). We empirically compare these criteria against a variety of different "correct BN structures", both real-world and synthetic, over a range of complexities. We also compare (1) using the *entire* training sample first to learn the best parameters and then to evaluate the models, versus (2) using only a partition for parameter estimation and another partition for evaluation (cross-validation). Our results show that the discriminative *BV* is one of the best measures for identifying the optimal structure, but the discriminative *CBIC* performs poorly; and that it is typically better to cross-validation here.

1 Introduction

While belief networks (BNs, a.k.a. Bayesian networks, graphical models) are generative models, capable of modeling a joint probability distribution over a set of variables, they are typically used *discriminatively* for some classification task — *e.g.*, to predict the probability of some disease, given some evidence about the patient. This has motivated the growing body of work on learning an effective *BN-classifier* from a datasample.

In general, learning an effective BN-classifier requires first finding a good BN structure (a.k.a. model, which represents the direct dependencies among the variables) then determining appropriate parameters for this model. The first step requires searching through a space of models, seeking the element that optimizes some *model selection criterion*. This paper investigates a number of criteria, towards determining which works best in practice — *i.e.*, which will minimize classifier error on unseen data.

This is not a trivial challenge. While one can typically improve classification performance *on the training data* by increasing the complexity of the model, this usually increases the number of parameters that must be estimated. This typically increases parameter variance, which leads to inferior generalization performance — *i.e.*, worse performance on unseen data. A *model selection criterion* attempts to operationalize this balance between complexity and goodness of fit to training data, by providing a single number for each network structure. A good model selection criterion is especially important when have limited training data, which is the standard case.

Earlier work [VG00] evaluated several standard *generative* criteria, where the goal is a structure that produces the best fit to the underlying distribution (using likelihood, Eqn 3). Our current paper considers two: *BIC* [Sch78] and *BDe* [CH92].

As noted above, our overall goal is different, as we are seeking a structure that leads to good *discriminative* performance, *i.e.*, which has the best classification performance on unseen testing data. We therefore consider several alternative *discriminative* critera: Conditional *BIC* (*CBIC*), resubstitution Classification Error (*CE*), and Bias^2+Variance (*BV*).

When deciding on an appropriate structure, we need to consider how we will instantiate its parameters (CPtables). In particular, each learner uses a corpus of training data, both to find the best parameters for each structure, and also to evaluate the quality of this instantiated model. Should it use the *same* data for both tasks, or should it instead partition the training sample into two subsamples, and use the first for parameter instantiation, and the second for model selection, perhaps in a cross-validation fashion?

The rest of this section discusses related work. Section 2 provides the framework for this paper, describing belief networks, model selection criteria and parameter learning. Section 3 presents our experimental setup and results: As our preliminary experimental results, on data from a real-world distribution, suggest the performance of each criterion may be related to complexity of the Markov blanket around the query variable, we therefore systematically explore the effectiveness of various model selection criteria across generative models with a wide range of Markov blanket complexities. The webpage [Guo04] contains additional information about the experiments reported here, as well as other related results.[1]

1.1 Related Work

There is a considerable literature on structure learning of belief nets, but most focus on generative learning; see [Hec98] for a detailed overview on this subject. MDL is used frequently to evaluate candidate structures [LB94, Suz78, FGG97]. [FY96] examined the sample complexity of the MDL-based belief net learning. [VG00] provides a comprehensive comparison when learning belief network structures *generatively*. While we borrow some of the techniques from these projects, recall our goal is learning the structure that is best for a *discriminative* classification task.

As noted earlier, belief nets are often used for this classification task. This dates back (at least) to NaïveBayes classifiers [DH73], and has continued with various approaches that include feature selection [LS94], and alternative structures [FGG97, CG99]. [KMST99] compared several model selection criteria (unsupervised/supervised marginal likelihood, supervised prequential likelihood, cross validation) on a subset of Bayesian networks regarded as "pruned Naive Bayes". [GD04] presented an algorithm for discriminative learn-

[1](1) There are many reasons to select some specific criteria, some of which relate more to prior assumptions and constraints, than to performance. In this paper, however, we are *only* concerned with eventual classification performance, as measured by Eqn 1.
(2) While most of these criteria are known to be asymptotically correct, our interest is with the practical use of these criteria. We therefore focus on small sample sizes.
(3) Our goal is to better understand model selection criteria, divorced with the search issues associated with learning itself. We therefore follow the standard framework for evaluating criteria [VG00]: consider only a small set of models, small enought that *each* can be evaluated.

ing belief networks that used the conditional likelihood of the class variable given the evidence (Eqn 2) as the model selection criterion. Our work differs by proposing several new discriminative model selection criteria (including a variant of a generative criteria (*CBIC*), and another (*BV*) motivated by the classification task in general [Rip96]), and by providing a comprehensive comparison between classical generative model selection criteria and discriminative criteria on the task of learning good structures for a BN-classifier.

2 Framework

2.1 Belief Network Classifiers

We assume there is a stationary underlying distribution $P(\cdot)$ over n (discrete) random variables $\mathcal{V} = \{V_1, \ldots, V_n\}$, which we encode as a "(Bayesian) belief net" (BN) — a directed acyclic graph $B = \langle \mathcal{V}, A, \Theta \rangle$, whose nodes $\mathcal{V}$ represent variables, and whose arcs A represent dependencies. Each node $D_i \in \mathcal{V}$ also includes a conditional-probability-table (CPtable) $\theta_i \in \Theta$ that specifies how D_i's values depend (stochastically) on the values of its immediate parents. In particular, given a node $D \in \mathcal{V}$ with immediate parents $\mathbf{F} \subset \mathcal{V}$, the parameter $\theta_{d|\mathbf{f}}$ represents the network's term for $P(\, D\!=\!d \,|\, \mathbf{F}\!=\!\mathbf{f}\,)$ [Pea88].

The user interacts with the belief net by asking *queries*, each of the form "What is $P(\, C\!=\!c \,|\, \mathbf{E}\!=\!\mathbf{e}\,)$?" — *e.g.*,

"What is $P($`Cancer = true|Gender=female, Smoke=true`$)$?"

— where $C \in \mathcal{V}$ is a single "query variable", $\mathbf{E} \subset \mathcal{V}$ is the subset of "evidence variables", and c (resp., $\mathbf{e}$) is a legal assignment to C (resp., $\mathbf{E}$).

Given any unlabeled instance $\mathbf{E} = \mathbf{e}$, the belief net B will produce a distribution over the values of the query variable; perhaps $B($`Cancer = true`$|\,\mathbf{E} = \mathbf{e}\,) = 0.3$ and $B($`Cancer = false`$|\,\mathbf{E} = \mathbf{e}\,) = 0.7$. In general, the associated H_B classifier system will then return the value $H_B(\mathbf{e}) = \operatorname{argmax}_c\{B(C\!=\!c|\mathbf{E}\!=\!\mathbf{e})\}$ with the largest posterior probability — here return $H_B(\,\mathbf{E} = \mathbf{e}\,) =$ false as $B($`Cancer = false`$|\,\mathbf{E} = \mathbf{e}\,) >$ $B($`Cancer = true`$|\,\mathbf{E}\!=\!\mathbf{e}\,)$.

A good belief net classifier is one that produces the appropriate answers to these unlabeled queries. We will use "classification error" (aka "0/1" loss) to evaluate the resulting B-based classifier H_B

$$\mathrm{err}(\,B\,) \quad = \sum_{\langle \mathbf{e}, c\rangle : H_B(\mathbf{e}) \neq c} P(\,\mathbf{e}\,,\,c\,) \tag{1}$$

Our goal is a belief net B^* that minimizes this score, with respect to the true distribution $P(\cdot)$. While we do not know this distribution *a priori*, we can use a sample drawn from this distribution to help determine which belief net is optimal. We will use a training set S of $m = |S|$ complete instances, where the ith instance is represented as $\langle c^i,\, e^i_1, \ldots, e^i_n \rangle$. This paper focuses on the task of learning the BN-structure $G = \langle \mathcal{V}, A \rangle$ that allows optimal classification performance on unseen examples.

Conditional Likelihood: Given a sample S, the empirical "log conditional likelihood" of a belief net B is

$$\mathit{LCL}^{(S)}(\,B\,) \quad = \quad \frac{1}{|S|} \sum_{\langle \mathbf{e}, c\rangle \in S} \log(\,B(\,c \,|\, \mathbf{e}\,)\,) \tag{2}$$

where $B(\,c \,|\, \mathbf{e}\,)$ represents the conditional probability produced by the belief network B. [MN89, FGG97] note that maximizing this score will typically produce a classifier that comes close to minimizing the classification error (Eqn 1).

While this $\mathit{LCL}^{(S)}(\,B\,)$ formula closely resembles the (empirical) "log likelihood" function

$$\mathit{LL}^{(S)}(\,B\,) \quad = \quad \tfrac{1}{|S|} \textstyle\sum_{\langle \mathbf{e}, c\rangle \in S} \log(\,B(\,c, \mathbf{e}\,)\,) \tag{3}$$

used as part of many *generative* BN-learning algorithms, it is significantly different [FGG97].

We will measure the complexity of the BN B as the number of free parameters in the network

$$k(B) \quad = \quad \sum_{i=1}^{n}(|V_i|-1) \prod_{F \in \mathrm{Pa}(\,V_i\,)} |F| \tag{4}$$

where n is the number of variables, $|V|$ is the number of values of any variable V, and $\mathrm{Pa}(\,V\,)$ is the set of immediate parents of the node V.

For a belief network structure, given a completely instantiated tuple, a variable C is only dependent on the variables in its Markov Blanket [Pea88], which is defined as the union of C's direct parents, C's direct children and all direct parents of C's direct children. We define $k_C(\,B\,)$ as the number of parameters in C's Markov blanket, within B, using an obvious analogue to Eqn 4.

2.2 Generative Model Selection Criteria

Most of the *generative* criteria begin with the average empirical log likelihood of the data, Eqn 3, as $LL^{(S')}(\,B\,)$ on *unseen* data S' is useful as an unbiased estimate of the average generative quality of the distribution B. To avoid overfitting, *BIC* adds a "regularizing" term that penalizes complex structures, as an embodiment of the trade-off between model simplicity and goodness of fit to the training data.

$$BIC^{(S)}(\,B\,) \quad = \quad -LL^{(S)}(\,B\,) + \frac{k(B)\,\log m}{2m}$$

Another generative model selection criterion is the marginal likelihood — averaged over all possible CPtable values (in the Bayesian framework):

$$BDe^{(S)}(\,B\,) = \prod_{i=1}^{n}\prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij}+a_{ij})} \prod_{k=1}^{|E_i|} \frac{\Gamma(\alpha_{ijk}+a_{ijk})}{\Gamma(\alpha_{ijk})},$$

where $q_i = \prod_{Y \in \mathrm{Pa}(\,E_i\,)} |Y|$ is the number of states of the parents of variable E_i, α_{ijk} are the Dirichlet prior parameters (here set to 1), $\alpha_{ij} = \sum_{k=1}^{|E_i|} \alpha_{ijk}$, and a_{ijk} are the empirical counts — *i.e.*, the number of instances in the datasample S where the ith variable E_i takes its kth value and its parents have their jth value.

2.3 Discriminative Model Selection Criteria

The *CBIC* (conditional *BIC*) criterion is a discriminative analogue of the generative *BIC* criterion, which differs by using log *conditional* likelihood to measure "training error" and by using $k_C(\,B\,)$ rather than $k(B)$ as the number of parameters.

$$CBIC^{(S)}(\,B\,) \quad = \quad -LCL^{(S)}(\,B\,) + \frac{k_C(\,B\,)\log m}{2\,m}$$

As we use classification error on testing data to measure a BN-classifier's performance, we decided to include its classification error (*CE*) on *training data* as a discriminative model selection criterion.

$$CE^{(S)}(\,B\,) \quad = \quad \frac{|\{\langle \mathbf{e}, c\rangle \in S \mid H_B(\mathbf{e}) \neq c\}|}{|S|}$$

[Rip96] proves that the expected L_2 error of a classifer corresponds to "Bias^2+Variance"

$$BV^{(S)}(\,B\,) \quad = \quad \frac{1}{|S|} \sum_{\langle c,\mathbf{e}\rangle \in S} [\,t(\,c\,|\,\mathbf{e}\,) \;-\; B(\,c\,|\,\mathbf{e}\,)]^2 \;+\; \hat{\sigma}^2[B(\,c\,|\,\mathbf{e}\,)]$$

where the "true" response $t(\,c\,|\,\mathbf{e}\,)$ corresponds to the empirical frequency within the training data:

$$t(\,c\,|\,\mathbf{e}\,) \quad = \quad \frac{\#_S(C\!=\!c, \mathbf{E}\!=\!\mathbf{e})}{\#_S(\mathbf{E}\!=\!\mathbf{e})}$$

where $\#_S(\mathbf{E} = \mathbf{e})$ is the number of instances in training set S that match this (partial) assignment, and we use the (Bayesian) variance estimate provided in [VGH01]:

$$\hat{\sigma}^2[B(\, c \,|\, \mathbf{e}\,)] \quad = \quad \sum_{\theta_{D|\mathbf{f}} \in \Theta} \frac{1}{n_{D|\mathbf{f}}} \left[\begin{array}{l} \displaystyle\sum_{d \in D} \frac{1}{\theta_{d|\mathbf{f}}} [B(\, d, \mathbf{f}, c \,|\, \mathbf{e}\,) - B(\, c \,|\, \mathbf{e}\,) \, B(\, d, \mathbf{f} \,|\, \mathbf{e}\,)]^2 \\ - \; (B(\, \mathbf{f}, c \,|\, \mathbf{e}\,) \; - \; B(\, c \,|\, \mathbf{e}\,) \, B(\, \mathbf{f} \,|\, \mathbf{e}\,)\,)^2 \end{array} \right]$$

which requires summing over the CPtable rows $\theta_{D=d|\mathbf{F}=\mathbf{f}}$, and uses $n_{D|\mathbf{F}=\mathbf{f}} = 1 + |D| + \#_S(\mathbf{F}=\mathbf{f})$ as the "effective sample size" of the conditioning event for this row.

2.4 How to Instantiate the Parameters

As mentioned above, a BN includes both a structure *and a set of parameters* for that structure. Given complete training data, the standard parameter learning algorithm sets the value of each parameter to its empirical frequency in the datasample, with a Laplacian correction:

$$\theta_{D=d|\mathbf{F}=\mathbf{f}} \quad = \quad \frac{\#_S(D=d, \mathbf{F}=\mathbf{f}) + 1}{\#_S(\mathbf{F}=\mathbf{f}) \; + \; |D|}$$

[CH92] prove these generative values corresponds to the mean posterior of a distribution whose prior was a uniform Dirichlet; moreover, they optimize the likelihood of the data, Eqn 3, for the given structure.

The learner has access to a training sample S, to use as it wishes when producing the optimal structure. A simple model selection process will use the "undivided sample" approach: use *all* of S when finding the appropriate instantiation of the parameters. It will then compute a score for this instantiated structure, based again on S. Note this "1Sample" approach was the motivation for many of the scoring criteria! We will compare this to the obvious "cross-validation" approach (5CV): first partition the data into 5 subsets, and for each subset S_i, use the other 4 subsets to fit parameters then compute the score of the result on S_i.

3 Empirical Studies

This section reports on our empirical studies that compare the 5 model selection criteria mentioned above, to help determine when (if ever) to use each, and also whether to partition the training data or not. We therefore asked each of the criteria to identify the appropriate structure, across a range of situations. Section 3.1 first explains how we will run each experiment, and how we will evaluate the results. Section 3.2 presents our first study, on a real-world distribution. This data suggests that the complexity of the generative model may determine which criteria works best. The remaining subsections explore this. Section 3.3 (resp., 3.4) considers the performance of the selection criteria on a set of synthetic models with a range of complexity, using 1Sample (resp., 5CV sample).

3.1 Experimental Setup

In each experiment, we have a specific "true" distribution $P(\cdot)$ — *i.e.*, correct BN-structure and parameters — which we either download, or produce. We generate a number of complete datasamples from this $P(\cdot)$, of various sizes. We also produce a set of possible models by modifying the true structure; see below. For each training sample we then run each of the selection criteria (in the appropriate context: 1Sample vs 5CV). Each criteria produces a single number for each candidate structure. Figure 1(a) shows this, in the context of the ALARM [BSCC89] network (Section 3.2).[2] Each criteria then identifies the structure it considers best, which is the one with the lowest score. Here, for example, *CBIC* would select the structure labeled "−7", *BIC* would pick "−9", *BV* would select "0" and *BDe*, "1".

[2] Each measure is normalized to fit between 0 (best) and 1.

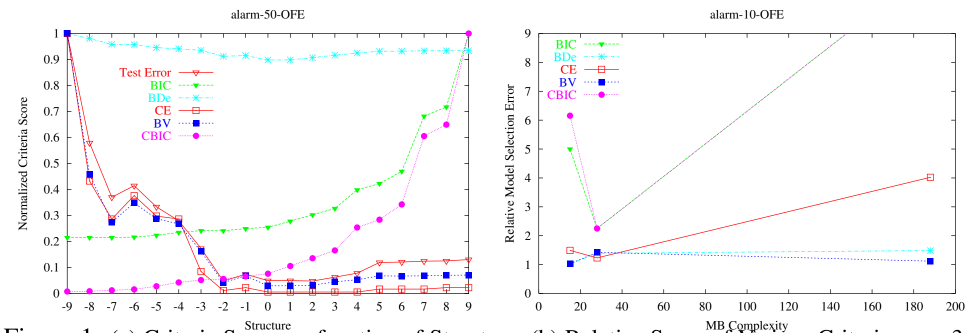

Figure 1: (a) Criteria Score, as function of Structure; (b) Relative Score of Various Criteria, over 3 queries in ALARM Network.

(These numbers correspond to the number of edges added, or deleted, to the initial structure. Hence, the original structure is the one labeled "0".) For each criteria χ, let B^χ be this selected structure, instantiated appropriately. We then compute the error of each B^χ, based on a hold-out sample S' of size $|S'| = 1000$, generated from $P(\cdot)$ — *i.e.*, $err^{(S')}(\,B^\chi\,)$.

We also determine which of the structures $B^* = \text{argmin}_B\{err^{(S')}(\,B\,)\}$ really was the best — *i.e.*, had the smallest error. (See the "Test Error" line in Figure 1(a); notice this picks "-2". That is, the structure that is best for a particular sample need not be the original structure!) The score for criterion χ is the ratio $err^{(S')}(\,B^\chi\,)/err^{(S')}(\,B^*\,)$. For each sample size, we compute the average over 20 repeated trials, each time using a different training set. This ratio will be 1 for a perfect criteria; in general, it will be higher.

Proper model selection is most challenging, and hence more relevant, when given limited training data; this paper therefore focuses on a very small training sample, of 20 instances. (The data for other sizes was similar; see [Guo04].)

Generating Sequence of Structures: Given a true BN-structure G^*, we generate a sequence of BN-structure candidates with increasing complexity, as follows:

1. Starting from the original structure, sequentially remove one randomly-selected edge from the Markov blanket (MB) of the class variable, to generate a series of structures whose class variable has decreasing MB size.

2. Starting from the original structure, sequentially add one randomly-selected edge to the Markov blanket of the class variable, to generate a series of structures whose class variable has increasing MB size.

3.2 Experiment I: Real-World Distr'n, 1Sample

Our preliminary investigations examined several real-world belief nets; here we focus on ALARM [BSCC89]; see [Guo04] for the others. We considered three different variables to serve as the class variable, which produced three different query forms, whose Markov blankets had a wide range in size: 15, 28, 188; see Figure 1(b).

As outlined above, we computed the relative score for each criteria, $err^{(S')}(\,B^\chi\,)/err^{(S')}(\,B^*\,)$. Figure 1(b) is the result when we used a sample of size $m = 20$. We found that *BV* perform well throughout, with *BDe* being very close; but the other measures were generally inferior. ([Guo04] shows similar performances on other sample sizes, and for other networks.)

3.3 Experiment II: Synthetic Distribution, 1Sample

We observed different behavior of the various selection criteria as we varied the complexity of the Markov blanket around the query variable. To further explore this, we generated a set of synthetic networks, whose query variables could have arbitrary Markov blanket complexity. We will use the networks here and below.

We first randomly generated six groups of belief network structures with varying Markov

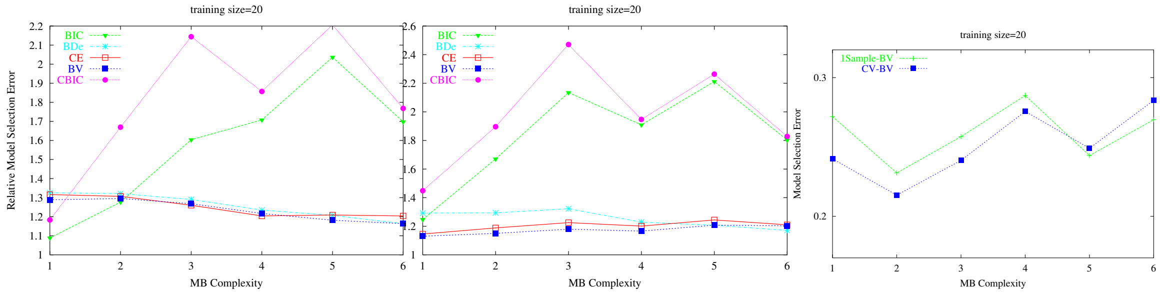

Figure 2: Experiments on 7-Variable BN, $m = 20$: (a) Experiment II: 1Sample; (b) Experiment III: 5CV; (c) Average Raw Performance (*BV*)

blanket complexity, where each group includes 30 structures. Each of these became the gold standard, used to generate datasamples.

We used the experimental apparatus described in the previous section to test the behavior of each criterion, across a spectrum of complexities and a range of sample sizes. The graph in Figure 2(a) show the results for belief networks with seven variables, over a sample of size $m = 20$, using 1Sample. The complexity (on the X axis, from 1 to 6) represents the six group of structures, with increasing generative complexity.

This plot shows that the *BV*, *CE* and *BDe* criteria perform comparably across the generative complexity, and each is typically (far) superior to *BIC* and *CBIC*. ([Guo04] shows this holds for many training sizes as well.)

The *BIC* and *CBIC* criteria perform well only when the generative complexity is very small, otherwise, they perform very poorly. Our experiments reveal why: These criteria have too strong a preference for simple structures, as they almost always pick the simplest structure in the sequence, irrespective of the data. (Notice this data *was* sufficient to tell the other measures to prefer other larger structure.) This is consistent with the [VG00] observation that this criterion seriously underfits — indeed, for small samples, it almost invariably produced no edges. This suggests the complexity penalty term for *BIC*/*CBIC* may be too big, and not appropriate for belief network on most cases.

3.4 Experiment III: Synthetic Distribution, 5CV

Figure 2(b) shows the results of the 5CV variant. In general, we can see that *BV* is often the best, closely followed by *CE*, then often *BDe*. Once again, we see that *BIC* and *CBIC* perform significantly worse; even with 5CV, they continue to select the simplest structure in almost all cases.

Here, the *CE* score here corresponds to the standard 5-fold CV. Note that it does not always produce the best response; (5fold) *BV* is better!

Figure 2(c) compares the 1Sample vs 5CV approaches — showing the average absolute 0/1 classification error (Eqn 1) obtained when using the *BV* criterion. We see that 5CV does better than the 1Sample variant when the generative model has low complexity, but the situation reverses as the model becomes more complex. ([Guo04] shows similar behavior for the other criteria.)

We performed similar experiments with othersample sizes (*e.g.*, $m = 10$, $m = 50$), and also on a set of larger belief networks (*e.g.*, with 15 variables) and obtained similar results. We also considered *AIC* and its analogue *CAIC*, and found they behaved essentially the same as *BIC* (*CBIC*). We also considered learning when using the parameters that optimize *conditional* likelihood, rather than simple *likelihood* [GZ02], and again found similar results. All of these results appear in [Guo04].

4 Conclusions

Belief nets are often used as classifiers. When learning the structure for such a BN-classifier, it is useful to have a criteria for evaluating the different candidate structures. This paper investigates various criteria to determine which is appropriate.

We proposed a number of novel *discriminative* model selective criteria, one (*CBIC*) an analogue of a standard generative criterion (commonly used when learning generative models), and another (*BV*) motivated by the familiar discriminative approach of decomposing error into bias and variance components. We then evaluated these methods, along with the generative ones, across a number of different situations: over queries of different complexities and different ways to use the training sample (and in [Guo04], over different sample sizes and different ways to instantiate the parameters — generatively vs discriminatively).

As our underlying task is discriminative, we had anticipated that perhaps all of the discriminative methods would work well. This was only partly true: while one discriminate method *BV* is amoungst the best criteria, another *CBIC* performed very poorly. We also expected 5CV to be uniformly superior to the 1Sample approach; our empirical evidence show that this, too, was not always true. (However, even when 5CV was inferior, it was never much worse.)

Our main contributions are defining the *BV* criteria, and providing empirical evidence that it does perform effectively, even for small samples. (While the *CBIC* criterion is also discriminative, our empirical evidence argues strongly against using this measure.) Our data also supports our recommendation for using the standard 5CV approach in this discriminative context.

References

[BSCC89] I. Beinlich, H. Suermondt, R. Chavez, and G. Cooper. The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks. In *European Conference on Artificial Intelligence in Medicine*, 1989.

[CG99] J. Cheng and R. Greiner. Comparing bayesian network classifiers. In *UAI99*, 1999.

[CH92] G. Cooper and E. Herskovits. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning Journal*, 9:309–347, 1992.

[DH73] R. Duda and P. Hart. *Pattern Classification and Scene Analysis*. Wiley, New York, 1973.

[FGG97] N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. *Machine Learning Journal*, 29, 1997.

[FY96] N. Friedman and Z. Yakhini. On the sample complexity of learning Bayesian networks. In *UAI96*, 1996.

[GD04] D. Grossman and P Domingos. Learning bayesian network classifiers by maximizing conditional likelihood. In *ICML2004*, 2004.

[Guo04] 2004. http://www.cs.ualberta.ca/∼yuhong/DiscriminantModelSelection.

[GZ02] R. Greiner and W. Zhou. Structural extension to logistic regression: Discriminant parameter learning of belief net classifiers. In *AAAI-02*, 2002.

[Hec98] D. Heckerman. A tutorial on learning with Bayesian networks. In *Learning in Graphical Models*, 1998.

[KMST99] P. Kontkanen, P. Myllymäki, T. Silander, and H. Tirri. On supervised selection of bayesian networks. In *UAI99*, 1999.

[LB94] Wai Lam and Fahiem Bacchus. Learning Bayesian belief networks: An approach based on the MDL principle. *Computation Intelligence*, 10(4):269–293, 1994.

[LS94] P. Langley and S. Sage. Induction of selective bayesian classifiers. In *UAI-94*, 1994.

[MN89] P. McCullagh and J. Nelder. *Generalized Linear Models*. Chapman and Hall, 1989.

[Pea88] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.

[Rip96] B. Ripley. *Pattern Recognition and Neural Networks*. Cambridge University, 1996.

[Sch78] G. Schwartz. Estimating the dimension of a model. *Annals of Statistics*, 6:461–464, 1978.

[Suz78] J. Suzuki. Learning bayesian belief networks based on the mdl principle: An efficient algorithm using the branch and bound technique. *Annals of Statistics*, 6:461–464, 1978.

[VG00] T. Van Allen and R. Greiner. Model selection criteria for learning belief nets: An empirical comparison. In *ICML'00*, 2000.

[VGH01] T. Van Allen, R. Greiner, and P. Hooper. Bayesian error-bars for belief net inference. In *UAI01*, 2001.