Efficient Approximations for the Marginal Likelihood of a Bayesian Network

David Maxwell Chickering*[*] and David Heckerman
Microsoft Research
Redmond 98052-6399, WA
dmax@cs.ucla.edu, heckerma@microsoft.com

Abstract

We examine asymptotic approximations for the marginal likelihood of a Bayesian network. We consider the well-known LaPlace and BIC approximations, as well as approximations proposed by Draper (1993) and Cheeseman and Stutz (1995). Each of these measures can be used to approximate the marginal likelihood of graphical models for discrete, Gaussian, and Gaussian-mixture distributions. In a series of experiments using synthetic data generated from naive-Bayes models having a hidden root node, we show that the CS measure is both an accurate and efficient scoring function, and is likely to be the most cost effective of all the measures.

1 Introduction

There is growing interest in the problem of learning graphical models from data. Simple Bayesian techniques have been developed for learning both directed and undirected models, given a *complete* data set—that is a data set in which each sample contains observations for every variable in the model. More complex approximation techniques have been developed for learning with incomplete data, including situations where some variables are *hidden* or never observed. These methods include Monte Carlo approaches such as Gibbs sampling and importance sampling (Neal, 1993), sequential updating methods (Spiegelhalter & Lauritzen, 1990; Cowell, Dawid, & Sebastiani, 1995), and asymptotic approximations (Kass, Tierney, & Kadane, 1988; Kass & Raftery, 1993; Draper, 1993). Summaries of methods for learning graphical models can be found in Heckerman (1995) and Buntine (1996).

In this paper, we compare the accuracy of various asymptotic approximations for learning graphical models with hidden variables. We examine the LaPlace approximation (Kass et al., 1988; Kass & Raftery, 1993; Azevedo-Filho & Shachter, 1994) and the Bayesian Information Criterion or BIC (Schwarz, 1978), which is equivalent to Risannen's (1987) Minimum-Description-Length measure. In addition, we consider two approximations described by Draper (1993) and Cheeseman and Stutz (1995). Each of these measures can be used to approximate the marginal likelihood of graphical models for discrete, Gaussian, and Gaussian-mixture distributions.

Both theoretical and empirical studies have shown that the LaPlace approximation is more accurate than is the BIC. Furthermore, it is well known that the LaPlace approximation is significantly less efficient than are the BIC, Draper, and Cheeseman-Stutz measures. To our knowledge, however, there have been no theoretical or formal empirical studies that compare the accuracy of the LaPlace approximation with those of Draper and Cheeseman. Here, we describe an experimental comparison of the approaches for learning directed graphical models (Bayesian networks) for discrete variables where one variable is hidden.

2 Background and Motivation

The Bayesian approach for learning Bayesian networks from data is as follows. Given a domain or set of variables $\mathbf{X} = \{X_1, \ldots, X_n\}$, suppose we know that the true joint distribution of $\mathbf{X}$ can be encoded in the Bayesian-network structure S. Let S^h denote the hypothesis that this encoding is possible. Also, suppose that we are uncertain about the parameters of the Bayesian network ($\boldsymbol{\theta}_s$) that determine the true joint distribution. Given a prior distribution over these parameters and a random sample $D = \{\mathbf{X}_1 = \mathbf{x}_1, \ldots, \mathbf{X}_n = \mathbf{x}_n\}$ from the true joint distribution, we can apply Bayes' rule to infer the posterior distribution

[*] Author's primary affiliation: Computer Science Department, University of California, Los Angeles, CA 90024.

of $\boldsymbol{\theta}_s$:

$$p(\boldsymbol{\theta}_s|D, S^h) = c\ p(D|\boldsymbol{\theta}_s, S^h)\ p(\boldsymbol{\theta}_s|S^h) \qquad (1)$$

where c is a normalization constant. Because D is a random sample, the likelihood $p(D|\boldsymbol{\theta}_s, S^h)$ is simply the product of the individual likelihoods

$$p(D|\boldsymbol{\theta}_s, S^h) = \prod_{l=1}^{N} p(\mathbf{x}_l|\boldsymbol{\theta}_s, S^h)$$

Furthermore, given some quantity of interest that is a function of the network structure and its parameters, $f(S, \boldsymbol{\theta}_s)$, we can compute its expectation, given D, as follows:

$$E(f(S, \boldsymbol{\theta}_s)|D, S^h) = \int f(S, \boldsymbol{\theta}_s)\ p(\boldsymbol{\theta}_s|D, S^h)\ d\boldsymbol{\theta}_s \quad (2)$$

Consider the case where the variables $\mathbf{X}$ are discrete. Let $\mathbf{Pa}_i$ denote the set of variables corresponding to the parents of X_i. Let x_i^k and $\mathbf{pa}_i^j$ denote the kth possible state of X_i and the jth possible state of $\mathbf{Pa}_i$, respectively. Also, let r_i and q_i denote the number of possible states of X_i and $\mathbf{Pa}_i$, respectively. Assuming that there are no logical constraints on the true joint probabilities other than those imposed by the network structure S, the parameters $\boldsymbol{\theta}_s$ correspond to the true probabilities (i.e., long-run fractions) associated with the Bayesian-network structure. In particular, $\boldsymbol{\theta}_s$ is the set of parameters θ_{ijk} for all possible values of i, j, and k, where θ_{ijk} is the true probability that $X_i = x_i^k$ given $\mathbf{Pa}_i = \mathbf{pa}_i^j$. We use the notation

$$\boldsymbol{\theta}_{ij} = (\theta_{ijk})_{k=1}^{r_i} \qquad \boldsymbol{\theta}_i = (\boldsymbol{\theta}_{ij})_{j=1}^{q_i} \qquad \boldsymbol{\theta}_s = (\boldsymbol{\theta}_i)_{i=1}^{n}$$

The likelihood for a random sample with no missing observations is given by

$$p(D|\boldsymbol{\theta}_s, S^h) = \prod_{i=1}^{n}\prod_{j=1}^{q_i}\prod_{k=1}^{r_i} \theta_{ijk}^{N_{ijk}}$$

where N_{ijk} are the sufficient statistics for the likelihood—the number of samples in D in which $X_i = x_i^k$ and $\mathbf{Pa}_i = \mathbf{pa}_i^j$. Consequently, we can compute the posterior distribution of $\boldsymbol{\theta}_s$ using Equation 1. This computation is especially simple when (1) the parameter sets $\boldsymbol{\theta}_{ij}$ are mutually independent—an assumption we call *parameter independence*—and (2) the prior distribution for each parameter set $\boldsymbol{\theta}_{ij}$ is a Dirichlet distribution

$$p(\boldsymbol{\theta}_{ij}|S^h) = c \prod_{k=1}^{r_i} \theta_{ijk}^{\alpha_{ijk}-1} \qquad (3)$$

where c is a normalization constant and the $\alpha_{ijk} > 0$ are determined from prior knowledge.

Making the problem more difficult, suppose that we are also uncertain about which structure encodes the true distribution. Given a prior distribution over the possible network-structure hypotheses, we can compute the corresponding posterior distribution using Bayes rule:

$$\begin{aligned} p(S^h|D) &= c\ p(S^h)\ p(D|S^h) \qquad (4) \\ &= c\ p(S^h) \int p(D|\boldsymbol{\theta}_s, S^h)\ p(\boldsymbol{\theta}_s|S^h)\ d\boldsymbol{\theta}_s \end{aligned}$$

Given some quantity of interest, $f(S, \boldsymbol{\theta}_s)$, we can compute its expectation, given D:

$$E(f(S, \boldsymbol{\theta}_s)|D) = \sum_{S^h} p(S^h|D) \int f(S, \boldsymbol{\theta}_s)\, p(\boldsymbol{\theta}_s|D, S^h)\ d\boldsymbol{\theta}_s$$

This full Bayesian approach is an example of what statisticians call *model averaging.* This approach is often prohibitive in cost, in which case, we can select one model S—for example, the network structure with the largest posterior probability—and use Equation 2 as an approximation for the true expectation of $f(S, \boldsymbol{\theta}_s)|D)$. This approximate approach is an example of *model selection.*

Whether we average over models or select a single model, the key computation is that of $p(D|S^h)$, known as the *marginal likelihood of D given S*, or simply the marginal likelihood of S. The marginal likelihood, when multiplied by the structure prior $p(S^h)$ serves as a "scoring function" for model selection. To simplify the remainder of our discussion, we assume that the structure prior is uniform, and refer to the marginal likelihood alone as a scoring function.

When the random sample D is complete, parameters are independent, and parameters priors are Dirichlet, the computation of the marginal likelihood is straightforward:

$$p(D|S^h) = \prod_{i=1}^{n}\prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_{ij})} \cdot \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})} \qquad (5)$$

This formula was first derived by Cooper and Herskovits (1992). Heckerman et al. (1995) refer to this formula in conjunction with the structure prior as the *B*ayesian *D*irichlet (BD) scoring function.

When the random sample D is incomplete, the exact computation of the marginal likelihood is intractable for real-world problems (e.g., see Cooper & Herskovits, 1992). Thus, approximations are required. In this paper, we consider asymptotic approximations.

One well-known asymptotic approximation is the *LaPlace* or *Gaussian* approximation (Kass et al., 1988; Kass & Raftery, 1993; Azevedo-Filho & Shachter, 1994). The idea behind the LaPlace approximation is that, for large amounts of data, $p(\boldsymbol{\theta}_s|D, S^h) \propto p(D|\boldsymbol{\theta}_s, S^h) \cdot p(\boldsymbol{\theta}_s|S^h)$ can often be approximated as a multivariate Gaussian distribution. Consequently,

$$p(D|S^h) = \int p(D|\boldsymbol{\theta}_s, S^h)\ p(\boldsymbol{\theta}_s|S^h)\ d\boldsymbol{\theta}_s \qquad (6)$$

can be evaluated in closed form. In particular, let

$$g(\boldsymbol{\theta}_s) \equiv \log(p(D|\boldsymbol{\theta}_s, S^h) \cdot p(\boldsymbol{\theta}_s|S^h))$$

Let $\tilde{\boldsymbol{\theta}}_s$ be the (vector) value of $\boldsymbol{\theta}_s$ for which the posterior probability of $\boldsymbol{\theta}_s$ is a maximum:

$$\tilde{\boldsymbol{\theta}}_s = \arg\max_{\boldsymbol{\theta}_s} \{p(\boldsymbol{\theta}_s|D, S^h)\} = \arg\max_{\boldsymbol{\theta}_s} \{g(\boldsymbol{\theta}_s)\}$$

The quantity $\tilde{\boldsymbol{\theta}}_s$ is known as the maximum *a posteriori* probability (MAP) of $\boldsymbol{\theta}_s$. Expanding $g(\boldsymbol{\theta}_s)$ about $\tilde{\boldsymbol{\theta}}_s$, we obtain

$$g(\boldsymbol{\theta}_s) \approx g(\tilde{\boldsymbol{\theta}}_s) + -\frac{1}{2}(\boldsymbol{\theta}_s - \tilde{\boldsymbol{\theta}}_s)^t A(\boldsymbol{\theta}_s - \tilde{\boldsymbol{\theta}}_s) \qquad (7)$$

where A is the negative Hession of $g(\boldsymbol{\theta}_s)$ evaluated at $\tilde{\boldsymbol{\theta}}_s$. Substituting Equation 7 into Equation 6, integrating, and taking the logarithm of the result, we obtain the LaPlace approximation:

$$\begin{aligned} \log p(D|S^h) \quad &\approx \quad \log p(D|\tilde{\boldsymbol{\theta}}_s, S^h)\ + \log p(\tilde{\boldsymbol{\theta}}_s|S^h) \\ &\quad + \frac{d}{2}\log(2\pi) - \frac{1}{2}\log|A| \qquad (8) \end{aligned}$$

where d is the dimension of $g(\boldsymbol{\theta}_s)$. For a Bayesian network with discrete variables, this dimension is typically the number of parameters of the network structure, $\prod_{i=1}^{n} \prod_{j=1}^{q_i} q_i(r_i - 1)$.[1] Kass et al. (1988) have shown that, under certain regularity conditions, errors in this approximation are $O(1/N)$, where N is the number of samples in D.

A more efficient but less accurate approximation is obtained by discarding $O(1)$ terms from Equation 8. We obtain

$$\log p(D|S^h) \approx \log p(D|\hat{\Theta}_S, S^h)\ - \frac{d}{2}\log N \qquad (9)$$

where $\hat{\Theta}_S$, is the maximum likelihood (ML) value of $\boldsymbol{\theta}_s$: the (vector) value of $\boldsymbol{\theta}_s$ for which $p(D|\boldsymbol{\theta}_s, S^h)$ is a maximum. This approximation is called the *Bayesian information criterion* (BIC), and was first derived by Schwarz (1978).

The BIC approximation is interesting in several respects. First, it does not depend on the prior. Consequently, we can use the approximation without assessing a prior.[2] Second, the approximation is quite intuitive. Namely, it contains a term measuring how well the model predicts the data ($\log p(D|\hat{\Theta}_S, S^h)$) and term that punishes the complexity of the model ($d/2\ logN$). Third, the BIC approximation is exactly the additive inverse of the Minimum Description Length (MDL) scoring function described by Rissanen (1987). The MDL of a network structure is the sum of the number of bits required to encode the data given the model (which decreases with increasing model complexity) and the number of bits required to encode the model (which increases with increasing model complexity).

Draper (1993) suggests another $O(1)$ approximation to Equation 8, in which the term $\frac{d}{2}\log(2\pi)$ is retained:

$$\log p(D|S^h) \approx \log p(D|\hat{\Theta}_S, S^h)\ - \frac{d}{2}\log N + \frac{d}{2}\log(2\pi) \qquad (10)$$

He mentions that, according to his experience, this approximation is better than the BIC. We shall refer to Equation 10 as the *Draper* scoring function.

To compute the LaPlace approximation, we must compute the negative Hession of $g(\boldsymbol{\theta}_s)$ evaluated at $\tilde{\boldsymbol{\theta}}_s$. Meng and Rubin (1991) describe a numerical technique for computing the second derivatives. Raftery (1995) shows how to approximate the Hession using likelihood-ratio tests that are available in many statistical packages. Thiesson (1995) demonstrates that, for discrete variables, the second derivatives can be computed using Bayesian-network inference.

When computing any of these approximations, we must determine $\tilde{\boldsymbol{\theta}}_s$ or $\hat{\boldsymbol{\theta}}_s$. One technique for finding a maximum is gradient ascent, where we follow the derivatives of $g(\boldsymbol{\theta}_s)$ or the likelihood to a local maximum. Russell et al. (1995) and Thiesson (1995) discuss how to compute derivatives of the likelihood for a Bayesian network with discrete variables.

A more efficient technique for identifying a local MAP or ML value of $\boldsymbol{\theta}_s$ is the EM algorithm (Dempster, Laird, & Rubin, 1977). Applied to Bayesian networks for discrete variables, the EM algorithm works as follows. First, we assign values to $\boldsymbol{\theta}_s$ somehow (e.g., at random). Next, we compute the *expected sufficient statistics* for the missing entries in the data:

$$E(N_{ijk}|\boldsymbol{\theta}_s, S^h) = \sum_{l=1}^{N} p(x_i^k, \mathbf{pa}_i^j|\mathbf{x}_l, \boldsymbol{\theta}_s, S^h) \qquad (11)$$

When X_i and all the variables in $\mathbf{Pa}_i$ are observed in sample $\mathbf{x}_l$, the term for this sample requires a trivial computation: it is either zero or one. Otherwise, we can use any Bayesian network inference algorithm to evaluate the term. This computation is called the *E step* of the EM algorithm.

[1] Sometimes, when insufficient data is observed, this dimension can be lower. See Geiger et al. (1996) for a discussion.

[2] One of the technical assumptions used to derive this approximation is that the prior be non-zero almost everywhere.

Next, we use the expected sufficient statistics as if they were actual sufficient statistics from a complete random sample D'. If we are doing a MAP calculation, we compute the values of $\boldsymbol{\theta}_s$ that maximize $p(\boldsymbol{\theta}_s|D', S^h)$:

$$\theta_{ijk} = \frac{E(N_{ijk}|\boldsymbol{\theta}_s) + \alpha_{ijk}}{E(N_{ij}|\boldsymbol{\theta}_s) + \alpha_{ij}}$$

If we are doing an ML calculation, we compute the values of $\boldsymbol{\theta}_s$ that maximize $p(D'|\boldsymbol{\theta}_s, S^h)$:

$$\theta_{ijk} = \frac{E(N_{ijk}|\boldsymbol{\theta}_s)}{E(N_{ij}|\boldsymbol{\theta}_s)}$$

This assignment is called the *M step* of the EM algorithm. Dempster et al. (1977) showed that, under certain regularity conditions, iteration of the expectation and maximization steps will converge to a local maximum. The EM algorithm assumes parameter independence,[3] and is typically used whenever the expected sufficient statistics can be computed efficiently (e.g., discrete, Gaussian, and Gaussian-mixture distributions).

In the EM algorithm, we treat expected sufficient statistics as if they we actual sufficient statistics. This use suggests another approximation to the marginal likelihood:

$$\log p(D|S^h) \approx \log p(D'|S^h) \tag{12}$$

where D' is an imaginary data set consistent with the expected sufficient statistics computed in the last iteration of the EM algorithm. We call this scoring function the *marginal likelihood of the expected data* or MLED. For discrete variables, MLED is given by the logarithm of the right-hand-side of Equation 5, where N_{ijk} is replaced by $E(N_{ijk}|\hat{\boldsymbol{\theta}}_s)$.

One difficulty with this scoring function is that it does not necessarily converge to the BIC for large data sets. That is, this scoring function is not asymptotically correct. A simple modification of this scoring function that does converge is given by

$$\begin{aligned}\log p(D|S^h) \quad \approx \quad & \log p(D'|S^h) + \log p(D|\hat{\boldsymbol{\theta}}_s, S^h) \\ & - \log p(D'|\hat{\boldsymbol{\theta}}_s, S^h)\end{aligned} \tag{13}$$

Equation 12 was first proposed by Cheeseman and Stutz (1995) as a scoring function for AutoClass, an algorithm for data clustering. We shall refer to Equation 13 as the *Cheeseman-Stutz* (CS) scoring function. We note that both the MLED and SC scoring functions can easily be extended to the directed Gaussian-mixture models described in (Lauritzen & Wermuth, 1989) and to undirected Gaussian-mixture models.

3 Experimental Design

In our experiments, we evaluated the relative accuracy of the CS, Draper, MLED, and BIC scoring functions as approximations to the marginal likelihood, using synthetic models containing a single hidden variable. In our evaluation, we used the LaPlace approximation as the gold standard. That is, we compared the marginal likelihood as approximated by each scoring function with the marginal likelihood as approximated by the LaPlace approximation. We used the LaPlace approximation as a gold standard, because it is provably more accurate than the BIC and Draper measures. We note, however, that no theoretical work has been to do show that the CS or MLED approximations are better or worse than the LaPlace approximation, and our experiments did not rule out either the possibility. Furthermore, the regularity conditions under which the LaPlace approximation is valid (i.e., accurate to order $1/N$) may have been violated in some of our experiments.

For reasons discussed in Section 4, we limited our synthetic networks to naive-Bayes models for discrete variables. A naive-Bayes model for variables $\{C, X_1, \ldots, X_n\}$ encodes the assertion that $X_1, \ldots, X_n$ are mutually independent, given C. The network structure for this model contains the single root node C and leaf nodes X_i each having only C as a parent. (We use the same notation to refer to a variable and its corresponding node in the network structure.) We generated a variety of naive-Bayes models by varying the number of states of C (c) and the number of observed variables n (all of which are binary). We determined the parameters of each model by sampling from the uniform (Dirichlet) distribution ($\alpha_{ijk} = 1$).

We sampled data from a model so as to make the root node C a hidden variable. Namely, we sampled data from a model using the usual Monte-Carlo approach where we first sampled a state $C = c$ according to $p(C)$ and then sampled a state of each X_i according to $p(X_i|C = c)$. We then discarded the samples of C, retaining only the samples of $X_1, \ldots, X_n$.

In a single experiment, we first generated a model for a given n and c, and subsequently a data set for a given sample size N. Next, we approximated the marginal likelihood for that data set given a series of *test models* that were identical to the synthesized model, except we allowed the number of states of the hidden variable to vary. Finally, we compared the different approximations of the marginal likelihood.

As described by Equation 8, we evaluated the LaPlace approximation at the MAP of $\boldsymbol{\theta}_s$. To simplify the computations, we also evaluated the CS, MLED, Draper,

[3] Actually, some parameter sets may be equal, provided these sets are mutually independent.

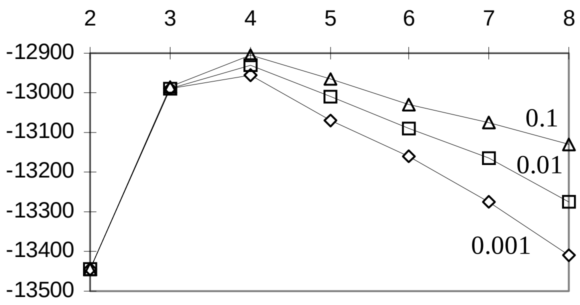

Figure 1: Log marginal likelihood (determined by the LaPlace approximation) as a function of the number of states of the hidden variable for $\epsilon = 0.1, 0.01$, and 0.001. ($n = 64, c = 4, N = 400$.)

and BIC measures at the MAP. We used the method of Thiesson (1995) to evaluate the negative Hession of $g(\boldsymbol{\theta}_s)$.

We initialized the EM algorithm as follows. First, we initialized 64 copies of the parameters $\boldsymbol{\theta}_s$ at random, and ran one E and M step. Then, we retained the 32 copies of the parameters for which $g(\boldsymbol{\theta}_s)$ was largest, and ran two EM iterations. Next, we retained the 16 copies of the parameters for which $g(\boldsymbol{\theta}_s)$ was largest, and ran 4 EM iterations. We continued this procedure four more times, until only one set of parameters remained.

To guarantee convergence of the EM algorithm, we performed 200 EM iterations following the initialization phase. We checked convergence by examining the relative change of $g(\boldsymbol{\theta}_s)$ between successive iterations. In all of experiments, the relative change fell below 0.0001 in less than 125 iterations. In 70 out of our 85 experiments, the relative change fell below 0.0001 in less than 30 iterations.

We assigned Dirichlet priors to each parameter set $\boldsymbol{\theta}_{ij}$. We used the almost uniform prior $\alpha_{ijk} = 1+\epsilon$, because it produced local maxima in the interior of the parameter space. (The traditional LaPlace approximation is not valid at the boundary of a parameter space.) To test the sensitivity of our results to ϵ, we generated 400 samples from a naive-Bayes model with a 4-state hidden root node and 64 observed binary variables. We then computed the log marginal likelihood of test models with 2 to 8 hidden states, using the LaPlace approximation. Figure 1 shows the results for $\epsilon = 0.1, 0.01$, and 0.001. The results are relatively insensitive to ϵ; and we used $\epsilon = 0.01$ in all subsequent experiments.

All experiments were run on a P5 100MHz machine under the Windows NT$^{\mathrm{TM}}$ operating system. The algorithms were implemented in C++.

4 Results and Discussion

We conducted three sets of comparisons for different values of c (number of states of the hidden variable), n (number of observed variables), and N (sample size). In our first set of experiments, we fixed $c = 64$ and $N = 400$ and varied n. In particular, we generated 400-sample data sets from four naive-Bayes models with 8, 16, 32, and 64 observed variables, respectively, each model having a hidden variable with four states. Figure 2 shows the approximate log marginal likelihood of the data given test models having hidden variables with two to eight states. (Recall that each test model has the same number of observed variables as the corresponding generative model.)

In our second set of experiments, we fixed $n = 64$ and $N = 400$, and varied c. In particular, we generated 400-sample data sets from four naive-Bayes models with $c = 32$, 16, 8, and 4 hidden states respectively, each model having 64 observed variables. Figure 3 shows the approximate log marginal likelihood of the data for test models having values of c that straddle the value of c for the generative model.

In our third set of experiments, we fixed $n = 32$ and $c = 4$, and varied N. In particular, from a naive-Bayes model with $n = 32$ and $c = 4$, we generated four databases with sample sizes (N) 100, 200, 400, and 800, respectively. Figure 4 shows the approximate log marginal likelihood of the data for test models having hidden variables with two to eight states.

Overall, the CS and LaPlace measures were extremely close for all values of n, c, and N; the MLED and LaPlace measures were close except for small values of n; the Draper and LaPlace measures were close except for large values of c; and the BIC and LaPlace measures were the most different.

These accuracy results must be balanced against the computational costs of the various approximations. The computational complexities of CS, MLED, Draper, and BIC are dominated by the complexity of the MAP computation, given by $O(dN)$. Assuming $N > d$, the computational complexity of LaPlace is dominated by that of Hessian computation, given by $O(d^2N)$. To appreciate the constants of computational cost, the run times for the experiment $n = 64, c = 32, N = 400$ are shown in Table 1. Thus, according to our experiments, the CS measure is the most cost effective.

The trends in the marginal-likelihood curves as a function of n, c, and N are not surprising. For each approximation, the curves become more peaked about

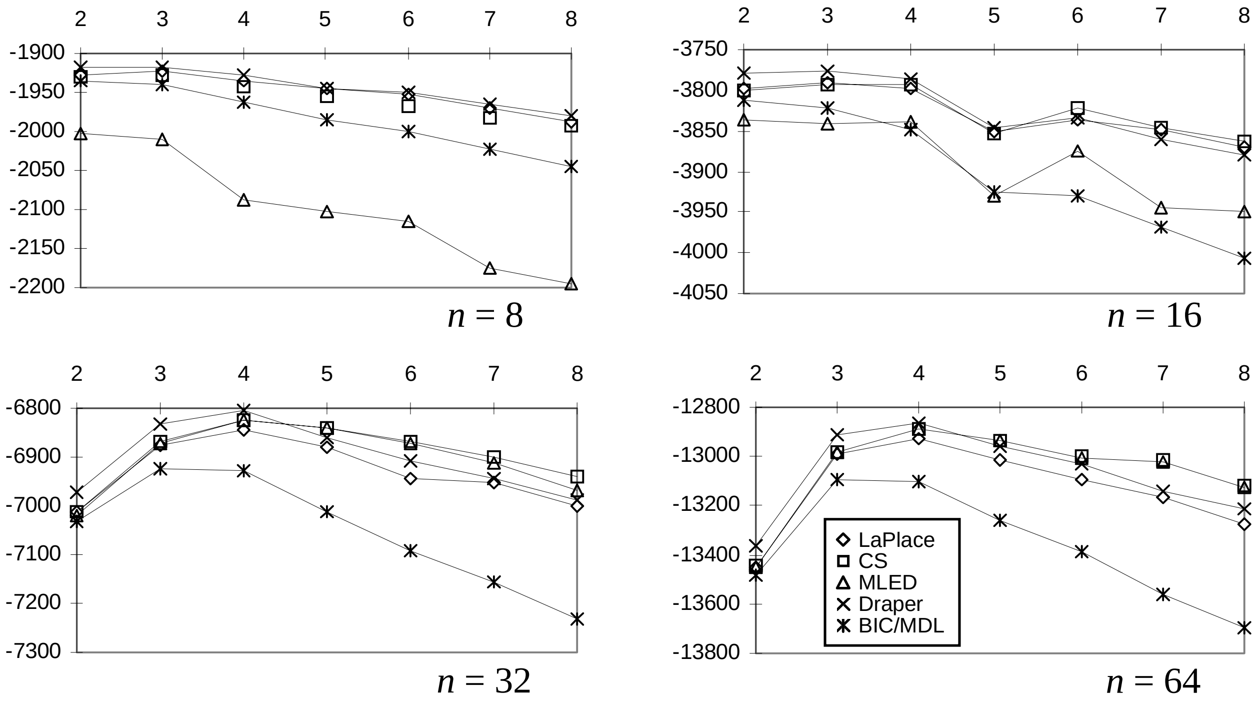

Figure 2: Approximate log marginal likelihood of the data given a test model as a function of the number of hidden states in the test model. The 400 sample data sets were generated from naive-Bayes models with n observed variables and 4 hidden states.

Table 1: Algorithm runtime (seconds) as a function of the number of hidden states of the test model (h). ($n = 64, c = 32$, and $N = 400$).

h	LaPlace	CS	MLED	Draper	BIC
?	?	?	?	?	?
?	?	?	?	?	?
?	?	?	?	?	?
?	?	?	?	?	?
?	?	?	?	?	?
?	?	?	?	?	?

the value of c (the number of hidden states in the generative model) as N and n increase, and as c decreases. The first result says that learning improves as the amount of data increases. The second result is a reflection of the fact that larger numbers of observed variables provide more evidence for the identify of the hidden variable. The third result says that it becomes more difficult to learn as the number of hidden states increases.

In our analysis, we have used the LaPlace approximation as a gold standard under the assumption that this scoring function is the most accurate of the measures. To investigate this assumption, we reanalyzed the data using an alternative gold standard. In particular, for each experiment, we computed the difference between the number of states of the hidden variable in the generative model (c), and the number of states of the hidden variable in the test model having the largest approximate marginal likelihood. This difference, which we call Δc, reflects the error made in performing model selection with a particular scoring function. The results are shown in Table 2. To our surprise, the CS and MLED measures selected the true number of hidden states more often than did the LaPlace measure, suggesting that these measure may be more accurate than the LaPlace approximation. An alternative explanation of these results is that the LaPlace approximation is more accurate, but that errors introduced by the gold standard cancel errors in the CS and MLED measures. Namely, it may be that—due to the noise in the data—the test model with the largest (correct) marginal likelihood has fewer hidden states than does the generative model, but the CS and MLED measures are punishing model complexity too little. Nonetheless, these results suggest that the theoretical properties of the CS and MLED measures should be examined. Another interesting observation in Table 2 is that all differences are non-positive. This observation suggests that either there are errors introduced by the gold standard, or the asymptotic approximations tend to punish model complexity too much.

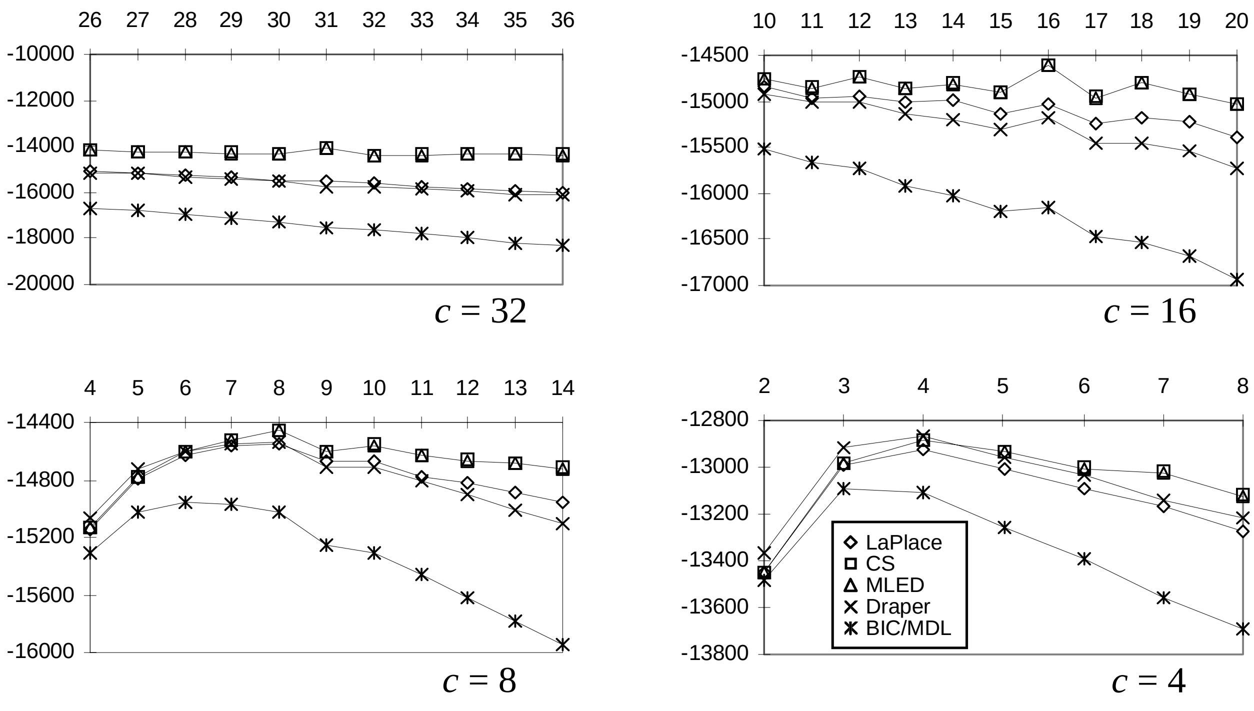

Figure 3: Approximate log marginal likelihood of the data given a test model as a function of the number of hidden states in the test model. The 400 sample data sets were generated from naive-Bayes models with 64 observed variables and c hidden states.

Our findings are valid only for naive-Bayes models with a hidden root node. These results are important, because they apply directly to the AutoClass algorithm, which is growing in popularity. Also, it is likely that our results will extend to models for discrete variables and data sets where each variable that is unobserved has an observed Markov blanket. Under these conditions, each Bayesian inference required by the scoring functions (e.g., Equation 11) reduces to a naive-Bayes computation. Furthermore, we see no reason, *a priori*, why our results will not generalize to arbitrary network structures and Gaussian-mixture distributions. Nonetheless, more detailed experiments are warranted to address models with more general structure and non-discrete distributions.

5 Reality Check

In our analysis of scoring functions for hidden-variable models, we have made an important assumption. Namely, we have assumed that, when the true model contains a hidden variable, it is better to learn by searching over models with hidden variables than those without hidden variables. This assumption is not trivially correct. Given a naive-Bayes model for the variables $C, X_1, \ldots, X_n$, the joint distribution for these variables can be encoded by a Bayesian network without hidden variables. (Assuming there are no accidental cancellations in the probabilities, this Bayesian network will be completely connected.) Thus, we can attempt to learn a model containing no hidden variables, and this model may be more accurate than that learned by searching over naive-Bayes models having a hidden root node.

We tested our assumption as follows. First, we generated a naive-Bayes model with $n = 12$ and $c = 3$. From this model we generated a database of size 800, discarding the observations of the variable C. Second, we learned a single naive-Bayes model containing a hidden root node using our experimental technique described in the previous section. In particular, we varied the number of hidden states of the naive-Bayes model, and selected the one with the largest (approximate) marginal likelihood. (In this case, all scoring functions yielded the same model: one with three hidden states). Third, we learned a single model containing no hidden variables using the approach described in Heckerman et al. (1995). In particular, we used the BD scoring function with a uniform prior over the parameters in conjunction with a greedy search algorithm (in directed-graph space) initialized with an empty graph. Finally, we measured the cross entropies $H(p_g; p_h)$ and $H(p_g; p_n)$, where p_g, p_h, and p_n are the joint distributions over the observed vari-

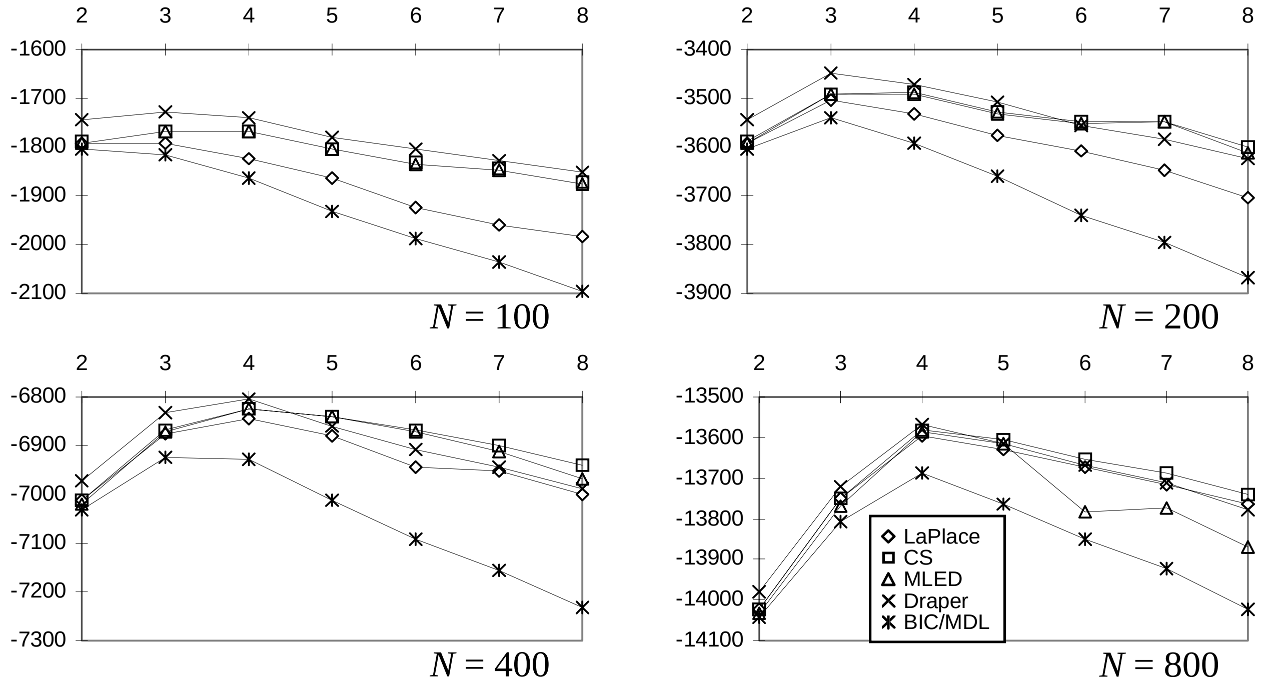

Figure 4: Approximate log marginal likelihood of the data given a test model as a function of the number of hidden states in the test model. The data sets of size N were generated from naive-Bayes models with 32 observed variables and 4 hidden states.

ables as determined by the generative, hidden, and non-hidden models respectively. Repeating this experiment five times, we obtained $H(p_g; p_h) = 0.023 \pm 0.003$ and $H(p_g; p_n) = 1.28 \pm 0.45$, respectively. In additional experiments, we found that differences between $H(p_g; p_h)$ and $H(p_g; p_n)$ increased as we increased the size of the models.

6 Conclusions

We have shown that the CS scoring function is an accurate and efficient approximation for the marginal likelihood of a Bayesian network. Although we conducted our experiments on discrete-variable naive-Bayes models, the CS measure is easily generalized to any (directed or undirected) graphical model for Gaussian-mixture distributions, and it is not unreasonable to expect that our conclusions will hold for these more general models.

Acknowledgments

We thank Koos Rommelse who helped with system implementation.

References

Azevedo-Filho, A., & Shachter, R. (1994). Laplace's method approximations for probabilistic inference in belief networks with continuous variables. In de Mantaras, R. L., & Poole, D. (Eds.), *Proceedings of Tenth Conference on Uncertainty in Artificial Intelligence,* Seattle, WA, pp. 28–36. Morgan Kaufmann, San Mateo, CA.

Buntine (1996). A guide to the literature on learning graphical models. *IEEE KDE, to appear.*

Cheeseman, P., & Stutz, J. (1995). Bayesian classification (AutoClass): Theory and results. In Fayyad, U., Piatesky-Shapiro, G., Smyth, P., & Uthurusamy, R. (Eds.), *Advances in Knowledge Discovery and Data Mining*, p. ?? AAAI Press, Menlo Park, CA.

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

Cowell, R., Dawid, A., & Sebastiani, P. (1995). A comparison of sequential learning methods for incomplete data. Tech. rep. 135, Department of Statistical Science, University College London.

Table 2: Errors in model selection.

Experiment			Δc				
n	c	N	LaPlace	CS	MLED	Draper	BIC
8	4	400	-1	-1	≤ -2	≤ -1	≤ -2
16	4	400	-1	-1	≤ -2	≤ -1	≤ -2
32	4	400	0	0	0	0	-1
64	4	400	0	0	0	0	-1
64	32	400	≤ -6	0	0	≤ -6	≤ -6
64	16	400	≤ -6	0	0	≤ -6	≤ -6
64	8	400	0	0	0	0	-2
32	4	400	0	0	0	0	-1
32	4	100	-1	0	-1	-1	≤ -2
32	4	200	-1	0	0	-1	-1
32	4	400	0	0	0	0	-1
32	4	800	0	0	0	0	0

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

Draper, D. (1993). Assessment and propagation of model uncertainty. Tech. rep. 124, Department of Statistics, University of California, Los Angeles.

Geiger, D., Heckerman, D., & Meek, C. (1996). Asymptotic model selection for directed networks with hidden variables. Tech. rep. MSR-TR-96-??, Microsoft, Redmond, WA.

Heckerman, D. (1995). A tutorial on learning Bayesian networks. Tech. rep. MSR-TR-95-06, Microsoft, Redmond, WA. Revised January, 1996.

Heckerman, D., Geiger, D., & Chickering, D. (1995). Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, *20*, 197–243.

Kass, R., & Raftery, A. (1993). Bayes factors and model uncertainty. Tech. rep. 571, Department of Statistics, Carnegie Mellon University, PA.

Kass, R., Tierney, L., & Kadane, J. (1988). Asymptotics in Bayesian computation. In Bernardo, J., DeGroot, M., Lindley, D., & Smith, A. (Eds.), *Bayesian Statistics 3*, pp. 261–278. Oxford University Press.

Lauritzen, S., & Wermuth, N. (1989). Graphical models for associations between variables, some of which are qualitative and some quantitative. *Annals of Statistics*, *17*, 31–57.

Meng, X., & Rubin, D. (1991). Using EM to obtain asymptotic variance-covariance matrices: The sem algorithm. *Journal of the American Statistical Association*, *86*, 899–909.

Neal, R. (1993). Probabilistic inference using Markov chain Monte Carlo methods. Tech. rep. CRG-TR-93-1, Department of Computer Science, University of Toronto.

Raftery, A. (1995). Bayesian model selection in social research. In Marsden, P. (Ed.), *Sociological Methodology*. Blackwells, Cambridge, MA.

Rissanen, J. (1987). Stochastic complexity (with discussion). *Journal of the Royal Statistical Society, Series B*, *49*, 223–239 and 253–265.

Russell, S., Binder, J., Koller, D., & Kanazawa, K. (1995). Local learning in probabilistic networks with hidden variables. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, Montreal, QU, pp. 1146–1152. Morgan Kaufmann, San Mateo, CA.

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

Spiegelhalter, D., & Lauritzen, S. (1990). Sequential updating of conditional probabilities on directed graphical structures. *Networks*, *20*, 579–605.

Thiesson, B. (1995). Score and information for recursive exponential models with incomplete data. Tech. rep., Institute of Electronic Systems, Aalborg University, Aalborg, Denmark.