Canadian AI 2022
30 May–3 June 2022, Toronto

An empirical study of scoring functions for learning Bayesian networks in model averaging

Zhenyu A. Liao[†], Charupriya Sharma[†], Dongshu Luo[†], and Peter van Beek[†, *]
[†] Cheriton School of Computer Science, University of Waterloo

Abstract

Scoring functions for Bayesian network (BN) structure learning can conflict in their rankings and previous work has empirically studied their effectiveness with an aim to provide recommendations on their use. However, previous studies on scoring functions are limited by the small number and scale of the instances used in the evaluation and by a focus on learning a single network. Often, a better alternative to committing to a single network is to learn multiple networks and perform model averaging as this method provides confidence measures for knowledge discovery and improved accuracy for density estimation. In this paper, we empirically study a selection of widely used and also recently proposed scoring functions. We address design limitations of previous empirical studies by scaling our experiments to larger BNs, comparing on an extensive set of both ground truth BNs and real-world datasets, considering alternative performance metrics, and comparing scoring functions on two model averaging frameworks: the bootstrap and the credible set. Contrary to previous recommendations based on finding a single structure, we find that for model averaging the BDeu scoring function is the preferred choice in most scenarios for the bootstrap framework and a recent score called quotient normalized maximum likelihood (qNML) is the preferred choice for the credible set framework.

Keywords: Bayesian networks, model averaging, structure learning

1. Introduction

A Bayesian network (BN) is a widely used probabilistic graphical model. Its structure can be learned from data using the well-known *score-and-search* approach, where a scoring function is used to evaluate the fit of a proposed BN to the data, and the space of directed acyclic graphs is searched for the best-scoring BN. Scoring functions commonly balance goodness of fit to the data with a penalty term for model complexity to avoid overfitting. Common scoring functions for discrete data include AIC [1], BIC/MDL [2–4], and BDeu [5, 6]. More recently, the qBDJ [7] and qNML [8] scoring functions have been proposed.

There are three main aims for learning a Bayesian network [9, Ch. 16.2]: probability density estimation, classification, and knowledge discovery. BNs are still widely used for density estimation, especially in low dimension and data scarce regimes. Its knowledge discovery capability, e.g., representing causal effects and conditional independence relations, is still unmatched. Previous work has empirically studied the best scoring function to use for each of these aims. However, previous studies are limited by the small number and scale of the instances used in the evaluation, and by a focus on learning a single network as opposed to the widely used methodology of learning multiple networks and performing model averaging.

In early work, Van Allen and Greiner [10] compared AIC and BIC for density estimation. Their work learned a single network and only studied randomly generated instances up to 10 variables and two real-world networks: Alarm and Insurance. Carvalho [11] compared AIC, BDeu, and BIC for classification. However, the experimental evaluation was restricted to learning a single tree BN. Yang and Chang [12] compared BIC, BDe (a variant

[*] peter.vanbeek@uwaterloo.ca

of BDeu), and several other scoring functions on density estimation and knowledge discovery. The evaluation focused on small instances with five or fewer variables and learned a single network.

More recently, Silander et al. [13] proposed a new scoring function called fNML and compared AIC, BDeu, BIC, and fNML on density estimation and knowledge discovery. Liu et al. [14] performed an extensive empirical comparison of AIC, BIC, BDeu, and fNML for knowledge discovery and concluded that BIC was overall the preferred choice. However, the evaluation used instances limited to at most 20 variables and did not consider model averaging.

Silander et al. [8] proposed a new scoring function called qNML and compared it against BDeu, BIC, and fNML on density estimation and knowledge discovery. The evaluation used small instances: 11 variables or fewer for evaluating knowledge discovery and 15 variables or fewer for evaluating density estimation. On the dimension of learning algorithms as opposed to scores, Scutari et al., [15] compared constraint-based, score-based, and hybrid learning algorithms. They found that the choice of statistical criteria (scores and their matching criteria) strongly affect the quality of the learned network, and that score-based algorithms and hybrid ones have similar performance with constraint-based ones slightly falling behind. They also used both BIC and BDeu as the scoring functions and found no apparent difference. Broom et al. [16] is, to the best of our knowledge, the only empirical study of scoring functions that considers model averaging, as opposed to learning a single network. Their study uses the bootstrap framework but only performs experiments over two networks: Alarm and Insurance. By using such a limited testbed, they were not able to make any recommendations on which scoring function to prefer in general.

In this paper, we fill the gap in previous empirical studies on scoring functions by scaling up the experiments to cover larger sized networks, by using a much more extensive testbed of instances, and by experimenting with two different model averaging frameworks: the bootstrap framework [17, 18] and the credible set framework [19]. We study five discrete scoring functions for Bayesian network structure learning (BNSL), namely AIC, BDeu, BIC, qBDJ, and qNML, and evaluate their performance on knowledge discovery and density estimation using both the ground truth BNs from bnlearn [18] and real-world datasets from the UCI repository. In addition to structural Hamming distance (SHD) for evaluating knowledge discovery, we also use the F-beta-measure and the weighted error rate, which allows us to study tradeoffs between false positives and false negatives on discovering edges in the network. We use the negative log likelihood as an approximation to the KL divergence in density estimation. We find that the ideal score under the model averaging scheme is very different from previous recommendations resulting from learning a single structure. Based on our empirical evaluation, we conclude that BDeu is the clear preferred choice in most scenarios for the bootstrap framework. For the credible set model averaging framework, we conclude that qNML is the best choice for knowledge discovery, and that AIC is best suited for density estimation with qNML trailing slightly behind.

2. Background

In this section, we briefly review the necessary background in Bayesian network structure learning (BNSL), scoring functions, model parameters, model averaging frameworks, and pruning rules.

2.1. Bayesian Networks

A Bayesian network (BN) is a probabilistic graphical model that consists of a labeled directed acyclic graph G in which the vertices $\mathcal{X} = \{X_1, \ldots, X_n\}$ correspond to n random variables, the edges represent direct influence of one random variable on another, and each

vertex X_i is labeled with a conditional probability distribution $P(X_i \mid \Pi_i)$ that specifies the dependence of the variable X_i on its set of parents Π_i in G. A BN can alternatively be viewed as a factorized representation of the joint probability distribution over the random variables and as an encoding of the Markov condition on the nodes; i.e., given its parents, every variable is conditionally independent of its non-descendants. Each vertex X_i has state space $\Omega_i = \{x_{i1}, \ldots, x_{ir_i}\}$, where $r_i \geq 2$ is the cardinality of Ω_i and v_{ik} is the k-th value for vertex X_i. Each Π_i has the state space $\Omega_{\Pi_i} = \{\pi_{i1}, \ldots, \pi_{ir_{\Pi_i}}\}$, where r_{Π_i} is the cardinality of Ω_{Π_i} and π_{ij} is the j-th vector of values for Π_i. The parameter set $\Theta = \{\theta_{ijk}\}$ for all $i = \{1, \ldots, n\}, j = \{1, \ldots, r_{\Pi_i}\}$ and $k = \{1, \ldots, r_i\}$ represents a complete parameterization in G where $\theta_{ijk} = P(X_i = x_{ik} \mid \Pi_i = \pi_{ij})$.

The predominant method for Bayesian network structure learning (BNSL) from data is the *score-and-search* method. Let $\mathcal{D} = \{D_1, \ldots, D_N\}$ be a dataset where each instance D_i is an n-tuple that is a complete instantiation of the variables in $\mathcal{X}$. A *scoring function* $\sigma(\mathcal{D} \mid G; \Theta)$ assigns a real value measuring the quality of G given the data $\mathcal{D}$. The BNSL task is to find the acyclic graph G with the highest score.

2.2. **Scoring Functions**

Scoring functions provide the model selection criteria in BNSL. Ideally a scoring function should have the following properties.

- **Consistency** [9, Def. 18.1]. The probability of the true graph $P(G^* \mid D) \to 1$ as $N \to \infty$. In terms of BNSL, the scoring function should choose the true graph G^* given a sufficiently large amount of data.
- **Decomposability** [9, Ch. 17.2.2]. The score of the entire network $\sigma(\mathcal{D} \mid G; \Theta)$ can be decomposed as the sum of local scores associated to each vertex $\sum_{i=1}^{n} \sigma(X_i \mid \Pi_i; \Theta)$.
- **Normality** [7] (score equivalence [20]). BNs of the same equivalence class have identical scores. Two BNs are equivalent if they impose identical conditional independence relations and can be structurally identified using the skeleton and v-structure [21].
- **Regularity** [22]. For two candidate parent sets $\Pi_{ij} \subset \Pi_{ik}$ of the child X_i, if both of them have identical empirical conditional entropy $H(X_i \mid \Pi_{i*})$, the smaller parent set Π_{ij} should have a better score.

Most scoring functions for BNSL are based on either log likelihood or Bayesian Dirichlet marginal likelihood. The log likelihood is the log probability of data $\mathcal{D}$ given a structure G and is often rearranged by vertices $\mathcal{X}$ with their parent sets Π,

$$LL(\mathcal{D} \mid G; \Theta) = \log \prod_{i=1}^{N} P(D_i \mid G; \Theta) = \sum_{i=1}^{n} \sum_{j=1}^{r_{\Pi_i}} \sum_{k=1}^{r_i} n_{ijk} \log \theta_{ijk},$$

where n_{ijk} is the count for $X_i = x_{ik}$ and $\Pi_i = \pi_{ij}$ in $\mathcal{D}$. It is well-known that using the log likelihood alone in BNSL yields the complete network since the likelihood never decreases when an edge is added. Various forms of penalties have been proposed to address the problem and several scoring functions have been derived that hold the desired properties above, including Akaike information criterion (AIC) [1], the Bayesian information criterion (BIC) [2], and a quotient score based on normalized maximum likelihood (qNML) [8]. The scores can be defined generally as $\sigma(*) = LL(\mathcal{D} \mid G; \Theta) - \mathbf{pen}(*)$ for some penalty $\mathbf{pen}(*)$, and we present different penalties below. The penalty for the AIC scoring function in this work is defined as,

$$\mathbf{pen}(\text{AIC}) = \sum_{i=1}^{n} r_{\Pi_i}(r_i - 1).$$

AIC is traditionally used for supervised tasks as it minimizes mean squared error of predictions [23] and is asymptotically equivalent to leave-one-out cross validation [24]. A lower

AIC score means a model is considered to be closer to the truth. The penalty for the BIC scoring function in this work is defined as,

$$\mathbf{pen}(\text{BIC}) = \sum_{i=1}^{n} r_{\Pi_i}(r_i - 1)\frac{\log N}{2}.$$

BIC estimates the posterior probability of a model being true. It penalizes models more heavily than AIC and requires a sample size much larger than the number of parameters in the model [14]. This definition of BIC is also equivalent to minimum descriptive length (MDL) [4] scoring function under the assumption that $N \to \infty$ and $D_1, \ldots, D_N$ are i.i.d. The qNML score is derived from the factorized normalized maximum likelihood (fNML) [13] that uses the vertex partitions in the normalizing factor. fNML is another log likelihood based score with the penalty defined as the regret, where a possible approximation is $\mathbf{reg}(N, r) \approx N\left(\log(\beta) + (\beta+2)\log(C_\beta) - \frac{1}{C_\beta}\right) - \frac{1}{2}\log\left(C_\beta + \frac{2}{\beta}\right)$, $\beta = \frac{r}{N}$, and $C_\beta = \frac{1}{2} + \frac{1}{2}\sqrt{1+\frac{4}{\beta}}$. However, fNML is not score equivalent in order to maintain decomposability. This drawback is recently addressed by the quotient version dubbed qNML. The penalty for the qNML scoring function in this work is defined as,

$$\mathbf{pen}(\text{qNML}) = \sum_{i=1}^{n} \mathbf{reg}(N, r_{\Pi_i} r_i) - \mathbf{reg}(N, r_{\Pi_i}).$$

Analytically qNML is similar to both AIC and BIC since they are all maximum likelihood based scores, though it has a more forgiving penalty.

From the Bayesian perspective, we can assume that the model parameters θ_{ijk} are independent Dirichlet variables with the priors $\mathrm{A} = \{\alpha_{ijk}\}$. Because the Dirichlet distribution is the conjugate prior distribution of the multinomial distribution, the posteriors $\theta_{ijk} \mid D_{ijk}; \alpha_{ijk} \sim \mathrm{Dir}(\alpha_{ijk} + n_{ijk})$. It follows that,

$$\begin{aligned} LL(\mathcal{D} \mid G, \Theta; \mathrm{A}) &= \log P(\mathcal{D} \mid G, \Theta) P(\Theta; \mathrm{A}) \\ &= \sum_{i=1}^{n}\sum_{j=1}^{r_{\Pi_i}} \frac{\sum_{k=1}^{r_i} \log\Gamma(\alpha_{ijk} + n_{ijk})}{\log\Gamma(\alpha_{ij*} + n_{ij*})} - \frac{\sum_{k=1}^{r_i}\log\Gamma(\alpha_{ijk})}{\log\Gamma(\alpha_{ij*})}, \end{aligned}$$

where $\alpha_{ij*} = \sum_{k=1}^{r_i} \alpha_{ijk}$ and $n_{ij*} = \sum_{k=1}^{r_i} n_{ijk}$. We consider two scores from the Bayesian Dirichlet (BD) family that have different priors. The likelihood-equivalence BD score with uniform priors (BDeu) [5, 25] assigns $\alpha_{ijk} = \frac{\alpha}{r_i r_{\Pi_i}}$ for some equivalent sample size α, whereas the BD score based on Jeffreys' prior (BDJ) [22] assigns $\alpha_{ijk} = 0.5$. The BDeu scoring function has an associated hyperparameter α that must be properly set prior to scoring. Previous work has shown empirically (e.g., [14, 26]) the importance of choosing a suitable value for α. Unfortunately, there is little guidance available for setting α. Recently, Suzuki [22] proves that BDeu is not regular, often yielding unnecessarily complex structures. On the other hand, the BDJ scoring function is regular yet not normal [22]. Therefore, it is not desirable to use BDJ directly in BNSL. Switching the conditional scores $\sigma_{\text{BDJ}}(X_i \mid \Pi_i, \Theta; \mathrm{A})$ in BDJ to the quotient version $\frac{\sigma_{\text{BDJ}}(X_i, \Pi_i \mid \Theta; \mathrm{A})}{\sigma_{\text{BDJ}}(\Pi_i \mid \Theta; \mathrm{A})}$ yields the quotient BDJ (qBDJ) [7] that are both regular and normal. The qBDJ scoring function in this work is defined as,

$$\sigma(\text{qBDJ}) = \sum_{i=1}^{n}\sum_{j=1}^{r_{\Pi_i}} \log\frac{\sum_{k=1}^{r_i}\Gamma(n_{ijk}+0.5)}{\Gamma(n_{ij*}+0.5)} - \sum_{i=1}^{n}\log\frac{\Gamma(0.5 r_i r_{\Pi_i} + N)\Gamma(0.5 r_{\Pi_i})}{\Gamma(0.5 r_{\Pi_i} + N)\Gamma(0.5 r_i r_{\Pi_i})}.$$

By Stirling's approximation, $\sigma(\text{qBDJ}) = LL(\mathcal{D} \mid G; \Theta) + O(1) - \mathbf{pen}(\text{qBDJ})$, where $\mathbf{pen}(\text{qBDJ})$ is exactly the second term in the definition.

Notably BDeu is the only irregular score in our study due to its broad application in BNSL. Other scores in the following experiments hold all four desirable properties.

2.3. **Parameters**

The parameters in log likelihood based scores are derived from maximum likelihood estimates, i.e., $\widehat{\theta_{ijk}} = \frac{n_{ijk}}{n_{ij*}}$. Although they are the closed form solutions to $\max_\Theta LL(\mathcal{D} \mid G; \Theta)$, it is often desirable to apply smoothing to model parameters, especially when some $n_{ijk} = 0$. In this work we use the m-estimate [27] defined as,

$$\widehat{\theta^m_{ijk}} = \frac{n_{ijk} + \frac{m}{r_i r_{\Pi_i}}}{n_{ij*} + \frac{m}{r_{\Pi_i}}}.$$

Recall that for the BD family, the posteriors $\theta_{ijk} \mid D_{ijk}; \alpha_{ijk} \sim \mathrm{Dir}(\alpha_{ijk} + n_{ijk})$. Then the expected value of the posterior $\widehat{\theta^{\mathrm{BD}}_{ijk}}$ is,

$$\widehat{\theta^{\mathrm{BD}}_{ijk}} = \frac{n_{ijk} + \alpha_{ijk}}{n_{ij*} + \alpha_{ij*}}.$$

Coincidentally the m-estimate is the same as the expected value of the posterior parameters for BDeu when $m = \alpha$, where m is also called the equivalent sample size but stems from the idea of additive smoothing.

From the NML principle, we have yet another estimation called conditional NML predictive probability [28] (sequential NML [8]),

$$\widehat{\theta^{\mathrm{sNML}}_{ijk}} = \frac{(n_{ijk} + 1)e(n_{ijk})}{\sum_{k=1}^{r_i}(n_{ijk} + 1)e(n_{ijk})},$$

where $e(n) = (1 + 1/n)^n$ and $e(0) = 1$. It has been shown [28] that sNML parameter converges to Krichevsky-Trofimov predictive probability, a special cases of the m-estimate when $m = r_{\Pi_i}$. Nevertheless, sNML provides an optimality guarantee in terms of regret [13], whereas the m-estimate has no known optimality property.

2.4. **Model Averaging**

We consider two model averaging frameworks for BNSL—the bootstrap and the credible set frameworks—as these two methods have been shown to scale the best among all available model averaging methods.

Bootstrapping is regarded as a general, flexible tool to provide confidence measures to statistics estimates. In the context of structure learning in Bayesian networks, Friedman et al. [17] proposed bootstrapping with thresholds to determine the existence of edges and other features. In particular, the non-parametric approach samples the original dataset with replacement and then heuristically learns a structure using the re-sampled data. After repeating such procedure many times, we can get the empirical probabilities of all edges by averaging on the learned structures. A threshold is finally applied to get the averaged structure.

In the credible set approach, all networks that are optimal or near-optimal in score are learned [19]. Note that the optimization problem defined by a scoring function and a dataset is to find the maximum-score BN. Let OPT be the score of the optimal BN. The set of networks learned from a dataset, denoted the *credible set*, is given by,

$$\{G \mid score(G) \geq OPT - \log BF\},$$

where the difference between the optimal score and the score of a network under consideration is proportional to the logarithm of the Bayes factor (BF), a well-known criteria for selecting between two models. Each network in the credible set can then be aggregated to form a combined structure weighted by their score, where the scores of the networks in the credible set are normalized to sum to 1 and the best model has the highest weight. Alternatively, the networks can be equally weighted when averaged.

Table 1. UCI datasets (*left*, *middle*) and bnlearn Bayesian networks (*right*), where n is the number of variables in the dataset or network, and N is the number of instances in the original UCI dataset.

UCI dataset	n	N
shuttle	10	58,000
census income	14	48,842
letter	17	20,000
online shopping	18	12,330
lymphography	19	148
hepatitis	20	155
parkinsons	23	195
credit card	24	30,000

UCI dataset	n	N
robot navigation	25	5,456
horse colic	27	368
steel	28	1,941
flags	29	194
breast cancer	31	569
soybean	36	683
biodeg	42	1,055
spectf heart	45	267

network	n
sachs	11
child	20
insurance	27
water	32
mildew	35
alarm	37
barley	48
hailfinder	56
heparII	70
win95pts	76

2.5. **Pruning**

Applying a scoring function to a dataset is a computationally intensive task, as many candidate parent sets need to be considered and scored. Fortunately, effective pruning rules have been developed for some scoring functions that preserve optimality but significantly reduce the candidate parents sets that need to be considered.

One of the most effective pruning rules for AIC and BIC is an upper-bound $\lceil \log_2(N) \rceil$ on the size of parent sets based on the sample size N. The rule is originally proposed in [29] for the optimal BNSL problem and generalized in [19] for the credible set approach. This rule enables AIC and BIC to scale much better than other scores under consideration. As we will show in our experiments, in scores other than AIC and BIC we often need to manually restrict the allowable maximum number of parents in order to score larger datasets within reasonable resource limits. Another effective family of pruning rules can eliminate certain parent sets and their supersets. Such rules for AIC/BIC and BDeu are originally proposed in [29] for the optimal BNSL problem and generalized to credible sets in [19].

3. **Experimental Methodology**

In this section, we describe the methodology we followed to experimentally study and compare scoring functions for Bayesian network structure learning in model averaging. We explain construction of the datasets (Section 3.1), scoring the datasets and learning the Bayesian network structures (Section 3.2), and the performance evaluation metrics (Section 3.3). The scoring computations were conducted on SHARCNET[1] and the structure learning experiments were conducted on a shared server with 346 GB RAM and Intel Xeon Gold 6148 at 2.4 GHz. For scoring the datasets memory usage was limited to 64 GB and for structure learning a limit of 128 GB was imposed. For both scoring and learning, a computation time limit of 24 hours was imposed for each instance.

3.1. **Datasets**

To empirically study the scoring functions, we considered a wide selection of datasets from the UCI repository[2] and networks from the bnlearn Bayesian network repository[3] (see Table 1). We preprocessed the UCI datasets using a k-nearest neighbor imputation algorithm, with $k = 5$, to fill in missing values and a supervised discretization method [30] based on the MDL principle to discretize continuous variables. For evaluating the scoring

[1] https://www.sharcnet.ca
[2] https://archive.ics.uci.edu/ml
[3] https://www.bnlearn.com/bnrepository/

functions on the task of density estimation, each UCI dataset was then randomly partitioned to a training set and a test set by a 70% to 30% ratio.

For evaluating the scoring functions on the task of structure learning, we used a total of 90 ground truth BNs: 10 ground truth BNs came from the bnlearn repository and a further 80 ground truth BNs were constructed following a similar approach to Liu et al. [14] by (i) scoring each of the 16 UCI datasets using each of the five scoring functions AIC, BDeu, BIC, qBDJ, and qNML in turn, (ii) learning an optimal network structure from each scored dataset, and (iii) and fitting the parameters to each structure to give a final Bayesian network. Given the 90 ground truth BNs, we used the logic sampling function `rbn` from the bnlearn R package [18] to generate random samples of sizes $N = 50$, 100, 500, 1,000, 5,000, and 10,000 from the bif files. We collected three samples for each dataset size N, for a total of 18 samples for each ground truth BN. The number of variables n used in our experiments, ranging from 11 to 76, pushes the limits of both the bootstrap and the credible set model averaging approaches, especially when using scoring functions such as BDeu and qNML that do not have as effective of pruning rules.

3.2. **Scoring and Structure Learning**

To evaluate the scoring functions within the bootstrap model averaging framework, we used the implementation available as the function `boot.strength` from the bnlearn R package [18]. We used the default replication factor of 200 and the tabu search algorithm, as in preliminary experiments it performed better than the alternative hill climbing algorithm. Due to score availability in bnlearn, we only consider AIC, BDeu, and BIC in the bootstrap experiments. A total of 4,320 bootstrap experiments were performed.

To evaluate the scoring functions within the credible set model averaging framework, we implemented the scoring functions AIC, BDeu, BIC, qBDJ, and qNML in `Python` to ensure a fair comparison[4]. The code takes a CSV file as input and generates a pruned score file iteratively for each parent set size. Saving the intermediate scoring files that guarantee optimality up to some parent set size is important since we do not limit the size a priori. As we stated above, the pruning rules for AIC and BIC are far more effective than those for other scores since an upper bound on the number of parents can be placed without losing optimality. For other scores, we have to abort the scoring generation at the end of the 24-hour limit. We note that similar experiments in [8, 14] have 20 variables as a computational limit for exact algorithms using scores other than AIC and BIC. Once the score files were generated, we used the `eBNSL` package [19], an extended version of `GOBNILP` [31], for collecting the credible networks. All networks falling within a Bayes factor (BF) of 150 were collected with a counting limit of 100,000. We also set the equivalent sample size $\alpha = 1$ for BDeu while the other scores do not have hyperparameters. We use a constant threshold 0.6 to determine whether an edge is present, and we have verified that the score ranking does not change with other reasonable thresholds. A total of 5,940 credible set experiments were performed.

3.3. **Performance Evaluation Metrics**

We evaluated the scoring functions based on their performance on knowledge discovery and density estimation. The former compares the learned structure with the ground truth BN in terms of directed and undirected edges and the latter compares the inference ability of the learned BNs. Each BN is weighted by its scores when the evaluation is conducted on a credible set using model averaging. For scoring functions based on posterior probabilities such as those in our experiments, the difference between two scores is proportional to the

[4] https://github.com/alisterl/hipss/releases/tag/v0.1.0

logarithm of the BF for the underlying models. The choice of the BF also has implications in model averaging since the worst model in the credible set will have a weight of $\frac{1}{\text{BF}}$ when the best model has a weight of 1.

To evaluate scoring functions on knowledge discovery, we use the structural Hamming distance (SHD). Given a ground truth network and a learned network, the SHD measures the distance between the CPDAG representations of the networks, where a CPDAG captures the equivalence class to which a network belongs (see [32]). Although the SHD has been widely used in evaluating structure learning, it has a number of significant limitations. First, it gives equal weight in case of a missing edge (FN), an extra edge (FP), and an edge in the wrong direction. However, in many applications of knowledge discovery, one does not wish to treat FP and FN as being of equal weight but rather wishes to specify an application-specific tradeoff. Second, adding an edge can increase the SHD by more than 1 if it makes other edges not compelled anymore, and thus SHD tends to penalize FP more than FN.

To address the limitations of SHD, we use two additional *cost sensitive* metrics: F-beta-measure and weighted error rate (see, e.g., [33]). F-beta-measure is a generalization to the F-measure, the harmonic mean of precision and recall. Recall that precision = TP / (TP + FP) and recall = TP / (TP + FN) where TP indicates the undirected edge is present in both learned BN and ground truth BN. These metrics are particularly useful for BNSL since there is a large number of true negatives, i.e., both the ground truth BN and the learned BN are sparse. The F-beta-measure is defined as,

$$F_\beta = (1+\beta^2) \times \frac{\mathit{precision} \times \mathit{recall}}{(\beta^2 \times \mathit{precision}) + \mathit{recall}}.$$

When $\beta = 1$, the F_β measure is the same as the F-measure, frequently referred to as the F1 score. When $\beta < 1$, the F_β measure gives more weight to precision and vice versa. The cost sensitive weighted error rate is defined as $(\alpha \times \text{FN} + \text{FP})/(n(n-1))$, where n is the number of variables in the dataset. When $\alpha > 1$, the error rate gives more weight to FN and thus penalizes missing edges over extra edges. The F-beta-measure and the weighted error rate allow us to evaluate the performance on a spectrum with different tradeoffs between precision and recall and between FP and FN. As we will show in our results, the rankings of scoring functions will fluctuate as different weights are place on FP and FN.

For density estimation, we use negative log likelihood on a test set as an approximation to KL distance [9, Ch. 16.2.1]. Let P be the ground truth probability distribution and $\hat{P}$ be the probability distribution represented by a learned BN G over the same space $\mathcal{X}$. The KL divergence (relative entropy) is given by $\mathcal{KL}(P \parallel \hat{P}) = -H(P) - \sum_{x \in \mathcal{X}} P(x) \log \hat{P}(x)$. The first term is fixed and can be dropped when comparing BNs with the same assumed true distribution. The second term can be estimated by the negative log likelihood $-\frac{1}{m} \sum_{i=1}^{m} log\ \hat{P}(D_i \mid G; \Theta)$ on a test set $\mathcal{D}_{test} = \{D_1, \ldots, D_m\}$.

4. **Experimental Results and Discussion**

In this section, we present the results of our experimental study and discuss their implications. The experimental results are aggregated using the Borda count. In the Borda count, in each trial (for a fixed dataset and model averaging method) the scoring functions are ranked according to the performance metric with ties allowed and each scoring function is awarded points corresponding to the number of scoring functions strictly lower in the ranking. Thus, the lowest ranked scoring function always gets 0 points and the highest ranked scoring function gets at most k points (exactly k if there are no ties for highest ranked), where k is the number of scoring functions under consideration. The Borda count was chosen to aggregate the results as it is known to select broadly acceptable options.

Table 2. Comparison of scoring functions using structural Hamming distance for the bootstrap (*left*) and credible set (*right*) model averaging approaches. At each row, the aggregated Borda count is shown when comparing the scoring functions on a set of experiments that consist of three samples from each ground truth network and dataset sample sizes of $N = 50$, 100, 500, 1,000, 5,000, 10,000.

Ground truth	Scoring function AIC	BDeu	BIC
bnlearn	**259**	145	85
UCI-AIC	225	**266**	96
UCI-BDeu	168	**335**	102
UCI-BIC	172	**248**	121
UCI-qBDJ	222	**226**	121
UCI-qNML	214	**246**	94
Total	1,260	**1,466**	619

Ground truth	Scoring function AIC	BDeu	BIC	qNML
bnlearn	249	209	**270**	235
UCI-AIC	354	234	234	**378**
UCI-BDeu	306	330	249	**376**
UCI-BIC	258	240	300	**339**
UCI-qBDJ	370	188	270	**430**
UCI-qNML	344	184	267	**410**
Total	1,881	1,385	1,590	**2168**

We present the results of knowledge discovery using the bootstrap approach in Table 2 (*left*) for SHD and in Figure 1 for the weighted error rate and the F-beta-measure. In the table, UCI-AIC, for example, refers to the ground truth set of networks that were constructed by using the AIC scoring function to find the optimal structure for each UCI dataset and then fitting the parameters to the structure to obtain a Bayesian network. Recall that the alpha value indicates the tradeoff between FP and FN, whereas the beta value indicates the tradeoff between recall and precision. In the bootstrap approach, we observe that BDeu dominates both AIC and BIC in weighted error rate and F-beta-measure with varying values of α and β. We have a similar observation for the SHD performance metric except for the ground truth BNs from bnlearn. A finer-grained analysis of the SHD results for the bootstrap approach reveals that the values are dominated by missing edges, and thus AIC, with its lower penalty on complexity, produced structures with fewer missing edges.

We present the results of knowledge discovery using the credible set approach in Table 2 (*right*) for SHD and in Figure 2 for the weighted error rate and the F-beta-measure. The scoring function qBDJ is omitted from the presented results, as in extensive preliminary experiments it was dominated by qNML. The figures clearly show that BIC is a high precision low recall score since its Borda count is much higher as β or α decreases from 1. This is consistent with the fact that BIC imposes the most strict penalty on the number

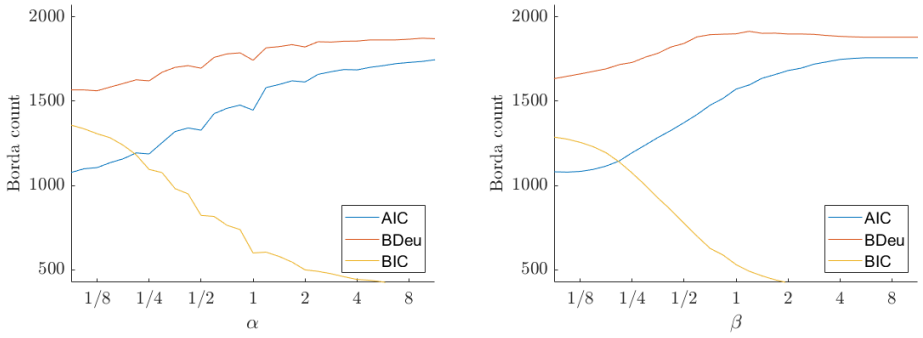

Figure 1. Comparison of scoring functions using error rate (*left*) and F_β score (*right*) on *undirected* edges for the bootstrap model averaging approach. At each α, β, the aggregated Borda count is shown when comparing the scoring functions on a set of experiments that consist of three samples from each bnlearn benchmark and dataset sample sizes of $N = 50$, 100, 500, 1,000, 5,000, 10,000.

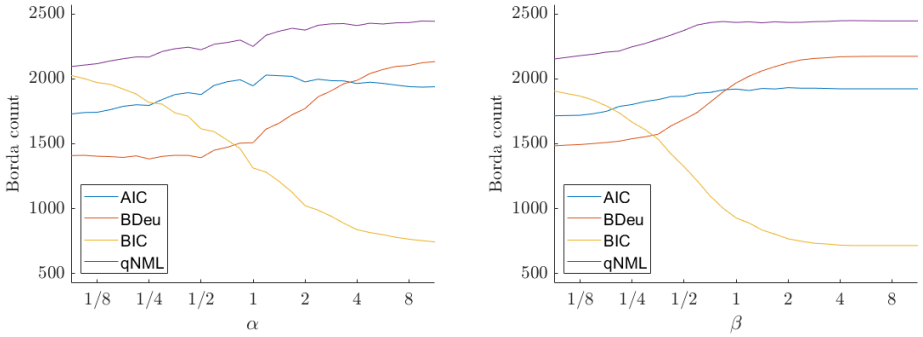

Figure 2. Credible set. Comparison of scoring functions using error rate (*left*) and F_β score (*right*) on *undirected* edges for the credible set model averaging approach. At each α, β, the aggregated Borda count is shown when comparing the scoring functions on a set of experiments that consist of three samples from each bnlearn benchmark and dataset sample sizes of $N = 50$, 100, 500, 1,000, 5,000, 10,000.

of parameters in the network. The aggregated results for the credible set approach also suggests that qNML dominates the other scoring functions with varying tradeoffs between either precision and recall or FP and FN. Considering SHD, we observe that qNML is the best score on UCI datasets with ground truth generated using all 5 scores, whereas BIC is the best score on the ground truth BNs from bnlearn. A finer-grained analysis of the SHD results for the credible set approach reveals that the values here are dominated by extra edges, and thus BIC comes out ahead due to its heavy penalty on complexity. We note that the best score in the aggregated results is not always the best choice for individual datasets but it is the overall winner if we consider all experiments.

In both the bootstrap and credible set approaches, our observations are different from those in Liu et al. [14] where the conclusion using only the optimal network leads to BIC being the dominant score. This difference can be attributed to (i) our evaluation is conducted with model averaging and (ii) we use a much more extensive set of datasets both in network sizes and in sample sizes in our experiments.

The results of density estimation are summarized in Table 3. Again, we use Borda count to aggregate the results on all datasets from the UCI repository. For the log likelihood based scores (AIC, BIC, and qNML), we learn their parameters using both the sNML method and the smoothed maximum likelihood method with $m = 1$, though the two methods have similar performance on the test data. The credible sets learned by BDeu and qBDJ are parameterized by their assumed Dirichlet distributions with $\alpha = 1$ and $\alpha_{ijk} = 0.5$. Note that the BDeu parameters are equivalent to the smoothed maximum likelihood ones since $m = \alpha = 1$. The negative log likelihood is calculated on a held-out test set from a 70%-30% train-test split ratio and the results indicate that AIC is the clear winner in inference with qNML trailing slightly behind. AIC's advantage in inference is less apparent when we only consider large BNs or large datasets, but BIC remains the worst performer for inference in almost all cases. This observation suggests that BIC should not be used when density estimation is the intended usage of the learned BN.

The runtime of the structure learning task for each score is reflective of the pruning rules available to each score. In particular, AIC and BIC can complete the scoring task with almost all datasets while the other scoring functions require limits to be set on the maximum number of parents for $n \geq 20$ variables. The advantage of pruning rules for AIC and BIC, however, does not show up in the metrics used for both tasks in our study. When we put a

Table 3. Borda score comparison on inference task using the set of credible networks learned from UCI datasets; e.g., the entry at column (AIC, snml) represents the Borda score for the combination of AIC as scoring function and snml as parameter estimation method.

AIC		BDeu	BIC		qBDJ	qNML	
m	snml	bdeu	m	snml	bdj	m	snml
85	73	49	29	22	66	68	56

limit on the size of the parent set, all scoring functions have similar runtime, suggesting that such a limit is the defining factor in efficiency. The scale of our experiments push the limits of model averaging approaches for Bayesian network structure learning. Although approximation methods that find a single high-quality network have been extended to thousands of variables [34], in contrast to model averaging approaches, such single-network methods cannot provide confidence measures for knowledge discovery and improved accuracy for density estimation.

5. Conclusion

Scoring functions can conflict in their rankings and previous work has empirically studied their effectiveness with an aim to provide recommendations on their use. However, previous studies on scoring functions are limited by the small number and scale of the instances used in the evaluation and by a focus on learning a single network. We have studied five discrete scoring functions for BNSL, namely AIC, BIC, qNML, BDeu, and qBDJ, scaled our experiments to large BNs using an extension to GOBNILP, and evaluated the scores with confidence measures on structure discovery and density estimation. We have addressed previous design limits by considering multiple metrics for structure discovery including the SHD, the F-beta-measure, and the weighted error rate. The cost sensitive metrics present a full picture with varying tradeoffs between precision vs. recall and FP vs. FN. We also evaluated scores on negative log likelihood in density estimation. We used both the ground truth BNs from bnlearn and real world UCI datasets in our structure learning tasks, and we are the first to provide an extensive experimental study of scoring functions in a model averaging framework.

Contrary to previous recommendations in [14], we find that qNML is the best contender for knowledge discovery using the exact credible set approach, and BDeu using bootstrapping, in most real world scenarios. We also find that AIC is best suited for density estimation with qNML trailing slightly behind. Our empirical study provides an insightful look at discrete score functions for Bayesian network structure learning and closes the gap in evaluating BN structures with confidence measures.

References

[1] H. Akaike. “Information theory and the maximum likelihood principle”. In: *Proc. of the International Symposium on Information Theory*. 1973, pp. 267–281.

[2] G. Schwarz. “Estimating the dimension of a model”. In: *The Annals of Statistics* 6 (1978), pp. 461–464.

[3] W. Lam and F. Bacchus. “Learning Bayesian belief networks: An approach based on the MDL principle”. In: *Computational Intelligence* 10 (1994), pp. 269–293.

[4] J. Rissanen. “Modeling by shortest data description”. In: *Automatica* 14 (1978), pp. 465–471.

[5] W. L. Buntine. “Theory refinement of Bayesian networks”. In: *Proc. of UAI*. 1991, pp. 52–60.

[6] D. Heckerman, D. Geiger, and D. M. Chickering. “Learning Bayesian networks: The combination of knowledge and statistical data”. In: *Machine Learning* 20 (1995), pp. 197–243.

[7] J. Suzuki and J. Kawahara. "Branch and bound for regular Bayesian network structure learning". In: *Proc. of UAI*. 2017.

[8] T. Silander, J. Leppä-aho, E. Jääsaari, and T. Roos. "Quotient normalized maximum likelihood criterion for learning Bayesian network structures". In: *Proc. of AISTATS*. 2018.

[9] D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. The MIT Press, 2009.

[10] T. Van Allen and R. Greiner. "Model selection criteria for learning belief nets: An empirical comparison". In: *Proc. of ICML*. 2000, pp. 1047–1054.

[11] A. M. Carvalho. *Scoring functions for Bayesian networks*. INESC-ID Tech. Rep. 54. 2009.

[12] S. Yang and K.-C. Chang. "Comparison of score metrics for Bayesian network learning". In: *IEEE Transactions on Systems, Man and Cybernetics* 32 (2002), pp. 419–428.

[13] T. Silander, T. Roos, and P. Myllymäki. "Learning locally minimax optimal Bayesian networks". In: *International J. of Approximate Reasoning* 51.5 (2010), pp. 544–557.

[14] Z. Liu, B. Malone, and C. Yuan. "Empirical evaluation of scoring functions for Bayesian network model selection". In: *BMC Bioinformatics* 13.Suppl 15 (2012).

[15] M. Scutari, C. E. Graafland, and J. M. Gutiérrez. "Who learns better Bayesian network structures: Accuracy and speed of structure learning algorithms". In: *International J. of Approximate Reasoning* 115 (2019), pp. 235–253.

[16] B. M. Broom, K.-A. Do, and D. Subramanian. "Model averaging strategies for structure learning in Bayesian networks with limited data". In: *BMC Bioinformatics* 13.Suppl 13 (2012).

[17] N. Friedman, M. Goldszmidt, and A. Wyner. "Data Analysis with Bayesian Networks: A Bootstrap Approach". In: *Proc. of UAI*. 1999, pp. 196–205.

[18] M. Scutari. "Learning Bayesian Networks with the bnlearn R Package". In: *Journal of Statistical Software* 35.3 (2010), pp. 1–22.

[19] Z. A. Liao, C. Sharma, J. Cussens, and P. van Beek. "Finding all Bayesian network structures within a factor of optimal". In: *Proc. of AAAI*. Vol. 33. 2019, pp. 7892–7899.

[20] D. M. Chickering. "Learning equivalence classes of Bayesian network structures". In: *J. of Machine Learning Research* 2 (2002), pp. 445–498.

[21] T. Verma and J. Pearl. "Equivalence and synthesis of causal models". In: *Proc. of UAI*. 1990, pp. 220–227.

[22] J. Suzuki. "A theoretical analysis of the BDeu scores in Bayesian network structure learning". In: *Behaviormetrika* 44.1 (2017), pp. 97–116.

[23] K. P. Burnham and D. R. Anderson. *Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach*. 2nd. Springer, 2002.

[24] M. Stone. "An asymptotic equivalence of choice of model by cross-validation and Akaike's criterion." In: *J. of the Royal Statistical Society Series B* 39 (1977), pp. 44–47.

[25] D. Heckerman. *A tutorial on learning Bayesian networks*. Tech. rep. MSR-TR-95-06. Microsoft Research, 1995.

[26] T. Silander, P. Kontkanen, and P. Myllymäki. "On sensitivity of the MAP Bayesian network structure to the equivalent sample size parameter". In: *Proc. of UAI*. 2007.

[27] T. M. Mitchell. *Machine Learning*. McGraw-Hill, 1997.

[28] J. Rissanen and T. Roos. "Conditional NML universal models". In: *Information Theory and Applications Workshop*. IEEE. 2007, pp. 337–341.

[29] C. P. de Campos and Q. Ji. "Efficient structure learning of Bayesian networks using constraints". In: *J. of Machine Learning Research* 12 (2011), pp. 663–689.

[30] U. Fayyad and K. Irani. "Multi-interval discretization of continuous-valued attributes for classification learning". In: *Proc. of IJCAI*. 1993, pp. 1022–1029.

[31] M. Bartlett and J. Cussens. "Advances in Bayesian network learning using integer programming". In: *Proc. of UAI*. 2013, pp. 182–191.

[32] I. Tsamardinos, L. E. Brown, and C. F. Aliferis. "The max-min hill-climbing Bayesian network structure learning algorithm". In: *Machine learning* 65.1 (2006), pp. 31–78.

[33] N. Japkowicz and M. Shah. *Evaluating Learning Algorithms: A Classification Perspective*. Cambridge University Press, 2011.

[34] M. Scanagatta, C. P. de Campos, G. Corani, and M. Zaffalon. "Learning Bayesian networks with thousands of variables". In: *Proc. of NeurIPS*. 2015, pp. 1864–1872.