Learning the Finer Things: Bayesian Structure Learning at the Instantiation Level

Chase Yakaboski, Eugene Santos, Jr.

Thayer School of Engineering at Dartmouth College, Hanover, NH
{chase.th, esj}@dartmouth.edu

Abstract

Successful machine learning methods require a trade-off between memorization and generalization. Too much memorization and the model cannot generalize to unobserved examples. Too much over-generalization and we risk under-fitting the data. While we commonly measure their performance through cross validation and accuracy metrics, how should these algorithms cope in domains that are extremely under-determined where accuracy is always unsatisfactory? We present a novel probabilistic graphical model structure learning approach that can learn, generalize and explain in these elusive domains by operating at the random variable instantiation level. Using Minimum Description Length (MDL) analysis, we propose a new decomposition of the learning problem over all training exemplars, fusing together minimal entropy inferences to construct a final knowledge base. By leveraging Bayesian Knowledge Bases (BKBs), a framework that operates at the instantiation level and inherently subsumes Bayesian Networks (BNs), we develop both a theoretical MDL score and associated structure learning algorithm that demonstrates significant improvements over learned BNs on 40 benchmark datasets. Further, our algorithm incorporates recent off-the-shelf DAG learning techniques enabling tractable results even on large problems. We then demonstrate the utility of our approach in a significantly under-determined domain by learning gene regulatory networks on breast cancer gene mutational data available from The Cancer Genome Atlas (TCGA).

Introduction

Since popularization by Pearl (1986), learning Bayesian Networks (BNs) has solidified into a steadfast research area for 40 years. It has become an important paradigm for modeling and reasoning under uncertainty and has seen applications from stock market prediction (Malagrino, Roman, and Monteiro 2018) and medical diagnosis (Shih, Choi, and Darwiche 2018) to Gene Regulatory Networks (GRNs) (Sauta et al. 2020). Despite Bayesian Network Structure Learning (BNSL) being NP-hard (Chickering, Heckerman, and Meek 2004) and even simpler structures like polytrees being NP-hard(er) (Dasgupta 1999), new constraints (Grüttemeier and Komusiewicz 2020), improvements (Trösser, de Givry, and Katsirelos 2021), and scalings (Scanagatta et al. 2015) are presented at major AI conferences every year. This is because BNs and affliated structures like Markov (Koller and Friedman 2009) and Dependency Networks (DNs) (Heckerman et al. 2000) offer a quality that other methods such as deep learning do not; explainability (Došilović, Brčić, and Hlupić 2018; Burkart and Huber 2021).

The optimization that occurs in Probabilistic Graphical Model (PGM) structure learning is

$$G^* = \underset{G}{\operatorname{argmax}}\, F(G, D) \quad \text{subject to } G \in \Qoppa$$

where D is a database, G is a graph structure such as a BN, F is a scoring function that yields the goodness of fit of the structure G, and $\Qoppa$ is the set of allowed structures for G; for BNs this would be the space of all possible Directed Acyclic Graphs (DAGs).

Scoring functions are essential to the structure learning problem and should have a theoretical justification in information theory or otherwise. For instance, the most common scoring functions such as Bayesian Information Criteria (BIC) (Schwarz 1978), Minimum Description Length (MDL) (Rissanen 1978), and Akaike Information Criterion (Akaike 1974) are all based on information theoretic criteria or can be viewed from this perspective. While we will spend part of this paper in theoretically justifying our model scoring approach, our goal is *not* to present a better scoring function. Instead, our goal is to illustrate that no matter the scoring function or learning algorithm, an over-generalization is encountered when modeling at the Random Variable (RV) level.

By operating at the RV level, models force a complete distribution, as is the case with BNs. While a complete distribution is often desired, this has an unintended over-generalization consequence, particularly in under-determined domains. This phenomenon even occurs in deep learning systems, and is generally referred to as fooling (Szegedy et al. 2014; Nguyen, Yosinski, and Clune 2015; Kardan and Stanley 2018). However, we will limit our scope to PGMs as our end goal is to analyze and/or hypothesize structural dependency relationships. Given this goal, such over-generalization could yield non-optimal structures, biasing analysis and derived hypotheses leading to misguided conclusions. To illustrate

For appendices visit: https://arxiv.org/abs/2303.04339.
For source code visit: https://github.com/di2ag/pybkb.

this over-generalization and provide intuition for learning at the RV instantiation level, we provide a motivating example taken from real-world data.

Motivating Example It is well known in cancer research that the genes TP53 and TTN have somatic mutations that affect chemotherapy responses (Xue et al. 2021). To demonstrate a real-world effect of BN over-generalization, we learned a simple BN for this interaction over the TCGA (Tomczak, Czerwińska, and Wiznerowicz 2015) mutational dataset as seen in Figure 1a. This BN encodes four possible worlds represented by distinctly styled arrows in Figure 1b. For this example we have reduced the state space of each gene to just mutated or not mutated. Assume our goal is to minimize the entropy or uncertainty of each world or explanation. Then the conditional entropy of the model is the sum over each world's conditional entropy which is inherently direction dependent. Since there exists many possible world edge configurations (RV instantiation dependencies), there may exist a better set of edges than those induced by the BN. Figure 1c shows this is true and illustrates the best collection of minimal entropy inferences for this example.

Contributions To address the over-generalization described we develop a structure learning algorithm leveraging the Bayesian Knowledge Base (BKB) framework as it inherently operates on the RV instantiation level. We accomplish this by detailing a necessary scoring metric to rank BKB models based on an MDL analysis and show theoretically that our MDL score takes over-generalization into account. Leveraging this theoretical result, we then develop our BKB Structure Learning (BKBSL) algorithm to minimize MDL and demonstrate empirically both competitive accuracy and better data fit compared to learned BNs. Further, we show that our algorithm can utilize existing optimization frameworks for DAG learning bringing BKBSL into the realm of these well studied off-the-shelf methods. Lastly, we conclude by utilizing a learned BKB to explain possible gene associations over TCGA breast cancer data.

Related Work and Preliminaries

As the MDL principle will be our guiding force in both theoretical and empirical analysis, we provide a brief review of its applications to directed PGMs, e.g. Bayesian Networks, as these models are most applicable to our study of BKBs. Lam and Bacchus (1994) first presented an MDL learning approach for BNs based on a heuristic search method seeking to spend equal time between simple and more complex BN structures. This was accomplished by extending Chow and Liu's (1968) result on recovering polytrees to general BNs via Kullback-Leibler cross entropy minimization allowing them to develop a weighting function over nodes. Their approach demonstrated that minimizing the MDL of BNs performs an intuitive trade-off between model accuracy and complexity. In their work, they also eluded to a potential subjectivity in choosing a model encoding strategy leading to research for improved MDL scores for BNs (Yun and Keong 2004; Drugan and Wiering 2010). Hansen and Yu (2001) detail a complete review of various MDL formulations.

Empirical evaluation of MDL as a scoring function for BN learning has also been well studied. Yang and Chang (2002) analyzed the performance of five scoring functions: uniform prior score metric (UPSM), conditional uniform prior score metric (CUPSM), Dirichlet prior score metric (DPSM), likelihood-equivalence Bayesian Dirichlet score metric (BDeu), and MDL. They showed that MDL was able to correctly identify ground-truth network structures from a variety of possible candidates, yielding the highest discrimination ability. Liu et al (2012) also performed empirical BN learning analysis over different scoring function, namely: MDL, Akaike's information criterion (AIC), BDeu, and factorized normalized maximum likelihood (fNML). Their approach tested the recovery accuracy of each scoring method over various gold standard networks as compared to the random networks used by Yang and Chang. Their results confirm the utility of MDL as it performed best in recovering the optimal networks when sufficient data was given.

To our knowledge there has been no work in structure learning on the RV instantiation level, likely due to the desire to learn complete distributions. Further, we have limited our comparisons to BNs as they are a predominant model in literature and provide a comparison to judge empirical results.

Bayesian Networks

A BN is a DAG $G = (V, E)$ that represents the factorized joint probability distribution over random variables $X = (X_1, \ldots, X_n)$ of the form:

$$\Pr(X) = \prod_{i}^{n} P(X_i \mid \Pi(X_i)) \tag{1}$$

such that $\Pi(X_i)$, or more concisely π_i, are the structural parents of the random variable X_i according to G and each node $V_i \in V$ correspond directly to a random variable X_i and n is the number of random variables in X. As the BN MDL formulation is well known in the literature, we point the reader to Appendix D.1 for a review.

Bayesian Knowledge Bases

Santos, Jr. and Santos (1999) developed the BKB framework to generalize BNs to the random variable instantiation level and to offer knowledge engineers a semantically sound and intuitive knowledge representation. BKBs unify "if-then" rules with probability theory to provide several modeling benefits compared to BNs, fuzzy logics, etc. First, BKBs do not require complete accessibility and can even leverage incomplete information to develop potentially more representative models. Second, since BKBs operate at the instantiation level, they can handle various types of cyclic knowledge. Lastly, BKBs have both robust tuning algorithms to assist in model correction/validation (Santos, Gu, and Santos 2013; Yakaboski and Santos 2018) and information fusion algorithms to incorporate knowledge efficiently and soundly from disparate knowledge sources (Santos, Wilkinson, and Santos 2011; Yakaboski and Santos 2021).

BKBs consist of two components: instantiation nodes (I-nodes) which represent instantiations of random variables of the form $X_i = x_{ik}$ where k is the k-th state of

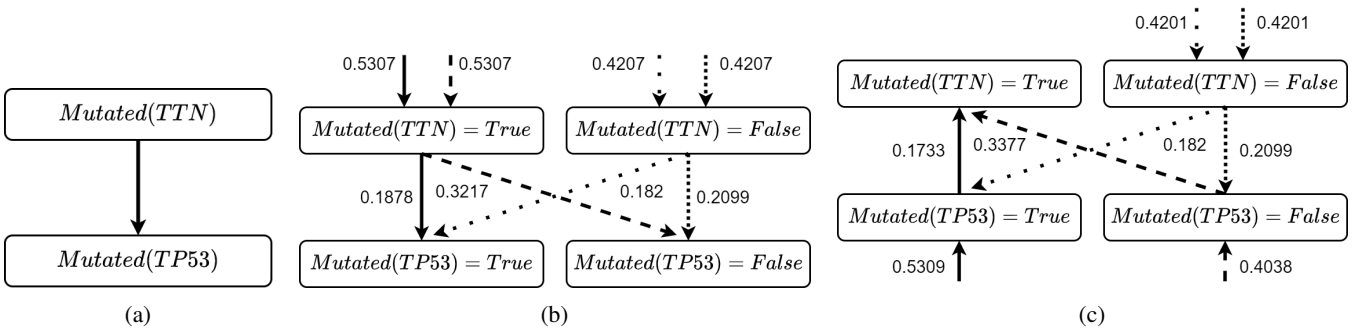


Figure 1: (a) A simple learned BN over the TCGA gene mutation dataset using GOBNILP (Cussens, Haws, and Studeny 2015) where the variable states are either mutated or not mutated. (b) A graph of all corresponding worlds represented in (a) delineated by different line styles. (c) A better orientation of intra-world dependency relationships that lead to a lower total conditional entropy. All values are conditional entropies calculated from the TCGA gene mutational dataset.

X_i, and support nodes (S-nodes) that represent the conditional probabilities between I-node relationships of the form $X_i = x_{ik} \rightarrow q = 0.87 \rightarrow X_j = x_{jl}$. The collection of these (in)dependencies describe the BKB correlation graph. For a precise definition and a graphical depiction see Appendix D.

While BKBs can handle incompleteness and various forms of cyclicity, it is necessary that all S-nodes in a BKB obey mutual exclusivity (mutex) and associated probability semantics. Mutex guarantees that mutually exclusive events cannot be true at the same time. Concretely, we say that two sets of I-nodes, I_1 and I_2, are mutex if there exists an I-node $X_i = x_{ik_1} \in I_1$ and $X_i = x_{ik_2} \in I_2$ such that $k_1 \neq k_2$. We say that two S-nodes are mutex if the two I-node parent sets of each S-node are mutex.

BKBs use a weight function to map S-node weights to associated conditional probabilities associated with the conditional dependence relationship described by the S-node. This weight function is analogous to the conditional probability tables (CPTs) of BNs except that BKBs do not require complete information. This weight function along with the correlation graph defines the BKB.

Bayesian Knowledge Base A Bayesian Knowledge Base (BKB) is a tuple $K = (G, w)$ where G is a correlation graph $G = (I \cup S, E)$ where I and S are the sets of I- and S-nodes, $E \subset \{I \times S\} \cup \{S \times I\}$, and $w : S \rightarrow [0, 1]$ is a weight function, such that the following properties are met:

1. $\forall q \in S$, the set of all incident I-nodes to q, denoted $\Pi(q)$, contains at most one instantiation of each random variable.
2. For distinct S-nodes $q_1, q_2 \in S$ that support the same I-node, denoted $Head_G(q_i)$, the sets $\Pi(q_1)$ and $\Pi(q_2)$ must be mutex.
3. For any $Q \subseteq S$ such that (i) $Head_G(q_1)$ and $Head_G(q_2)$ are mutex, and (ii) $\Pi(q_1)$ and $\Pi(q_2)$ are not mutex, then $\forall q_1, q_2 \in Q, \sum_{q \in Q} w(q) \leq 1$.

The probability of an inference (world) in a BKB, denoted τ, is the product of all S-node weights $w(q)$ consistent with that world (Appendix D.2). The joint probability distribution of a BKB is the sum of all inference probabilities consistent with an evidence set Θ represented as I_Θ and given by

$$P(\Theta) = \sum_{\tau \in I_\Theta} \prod_{q \in \tau} w(q) \tag{2}$$

The BKB Minimum Description Length

To construct a BKB structure learning algorithm, we first need to define a scoring function that can rank BKB structures as well as provide a theoretical justification for its utility. For these reasons we focus our attention on a Minimum Description Length (MDL) score as it is a well studied tenet of learning theory (Rissanen 1978). The idea is that the best model given data should minimize both (1) the encoding length of the model and (2) the length of encoding the data given the model. This is akin to applying Occam's Razor to model selection, i.e., choose the simplest model that still describes the data reasonably well.

Encoding the BKB

The minimum length needed to encode a BKB is related directly to the number of S-nodes modeled by the BKB. The encoding of each S-node will contain a probability value and a parent set of I-nodes. For a problem with n RVs such that each RV can have r_i number of instantiations, to encode all I-nodes we would need $\log_2(m)$ number of bits where $m = \prod_i r_i$. The general BKB MDL is

$$\sum_{q \in S} \Big((|\Pi(q)| + 1) \log_2(m) + \delta \Big) - N \sum_{\tau} p_\tau \log_2(q_\tau) \tag{3}$$

where δ is the number of bits needed to store the probability value and the first term is the BKB model encoding length. From Equation 3 it is clear that we incur a modeling cost for the BKB based on the finer granularity of the model. This cost is derived in Appendix C.1. Therefore, if we know that the distribution factorizes to a BN, there is no reason to use a BKB. However, rarely in practice do we know the true distribution of the data and in most cases the data we learn from is incomplete. This eludes to a natural rationale for using a BKB that will be theoretically justified.

Encoding the Data with a BKB

The task of encoding dataset D is centered on learning a joint distribution over random variables $X = (X_1, \ldots, X_n)$. As we are focused on the discrete case, each variable X_i will have r_i discrete states and every unique choice of random variable instantiations defines a complete world τ of the ground-truth distribution of the data and is assigned an associated probability value, p_τ.

We will make several standard statistical assumptions. We assume that each data instance in D is a complete world such that each data instance specifies a value for every random variable in X[1]. We assume that each data instance is a result of independent random trails and expect that each world would appear in the dataset with a frequency $\approx N p_\tau$.

The main theme of the MDL metric is to efficiently compress N data instances as a binary string such that we minimize the string's encoding length. There are many different coding and compression algorithms we can use, but since we simply care about comparing different MDLs as a metric we can limit our focus to just symbol codes (MacKay 2005). Symbols codes assign a unique binary string to each world in the dataset. For instance, the world $\{X_1 = 1, X_2 = 1, X_3 = 0\}$ might be assigned the symbol 0001. We can then encode the dataset as just the concatenation of all these world symbols and judge our compression effectiveness by calculating the length of this final binary string.

Research in information theory has proved that for symbol codes it is possible to minimize the length of this encoded data string by leveraging the probability/frequency of each world symbol occurring. Specifically, Huffman's algorithm generates optimal Huffman codes (MacKay 2005), i.e., optimal symbol code mappings, that yield a minimum length. The key intuition for Huffman codes is that we should give shorter code words to more frequently occurring worlds and longer ones to less probable worlds. Lam and Bacchus (1994) proved that the encoding length of the data is a monotonically increasing function of the cross-entropy between the distribution defined by the model and the true distribution.

Therefore, if we have a true distribution P and a model distribution Q over the same set of worlds, $\tau_1, \ldots, \tau_t$, where each world τ_i is assigned the probability p_i by P and q_i by Q, then the cross entropy between these distributions is

$$C(P, Q) = \sum_{i=1}^{t} p_i \log_2 \frac{p_i}{q_i} = \sum_{i=1}^{t} p_i (\log_2 p_i - \log_2 q_i) \quad (4)$$

Calculating Equation 4 is not appealing as the number of worlds is combinatorial in the number of variables. Chow and Lui (1968) developed a famous simplification of Equation 4 as just a local computation over low-order marginals when Q has a tree factorization. Lam and Bucchus (1994) extended their result to models that have a general DAG structure. Their main result concludes that $C(P, Q)$ is a monotonically decreasing function of the sum of each random variable's mutual information with their parents, $I(X_i; \Pi(X_i))$. Their exact result is restated in Appendix Theorem 3.

[1]Only for simplicity do we make the assumption that a data instance must be complete, i.e., not contain any missing values.

Further generalizing these results to the instantiation level, we can now show that an optimally learned BKB can encode the distribution as well or better than a BN. Consider again the fundamental MDL problem of learning the dataset encoding. Equation 4 says that we need to minimize the total cross-entropy between P and Q. However, in terms of data encoding, we only need to minimize the difference between each p_i and q_i for unique worlds that are actually in D. In this sense, our database encoding doesn't care about the worlds that aren't in the dataset, but for which a model like a BN naturally defines and generalizes. Therefore, BKBs handling of incompleteness gives us an opportunity to more tightly fit the data we actually know. Consider the following cross-entropy

$$C(P, Q) = \sum_{i=1}^{d} p_i (\log_2 p_i - \log_2 q_i) + \sum_{i=d+1}^{t} p_i (\log_2 p_i - \log_2 q_i) \quad (5)$$

where worlds $\tau_1, \ldots, \tau_d$ are represented by the unique exemplars in D that we hope to encode, i.e., $\{\tau_i, \ldots, \tau_d\} = \{d_1, d_2, d_3, \ldots\}_{\neq} \subseteq D$, and worlds $\tau_{d+1}, \ldots, \tau_t$ are worlds that our model induces. In terms of encoding length we can narrow our focus to only considering worlds present in D. As Lam and Bacchus (1994) proved the encoding length of the data is a monotonically increasing function cross-entropy, it is trivial to prove the following corollary.

Corollary 1. *Let's Define the cross-entropy* $C_D(P, Q) = \sum_{i=1}^{d} p_i (\log_2 p_i - \log_2 q_i)$ *between two distributions* P *and* Q *over the same set of worlds* $\tau_1, \ldots, \tau_d$ *s.t. these worlds must be included in a dataset* D*. Then the encoding length of the data* D *is monotonically increasing function of* C_D*.*

Combining Corollary 1 with Lam and Bacchus mutual information theorem (Appendix D Theorem 3) we arrive at our main theoretical result.

Theorem 1. $C_D(P, Q)$ *is a monotonically decreasing function of*

$$\sum_{\tau \in D_{\neq}} \sum_{i=1}^{n} p(x_{i\tau}, \pi_{i\tau}) \log_2 \frac{p(x_{i\tau}, \pi_{i\tau})}{p(x_{i\tau}) p(\pi_{i\tau})} \quad (6)$$

where $x_{i\tau}$ *is the instantiation of random variable* X_i *determined by data instance* τ*,* $\pi_{i\tau}$ *is the parent set instantiation of random variable* X_i *governed by* τ*, and* $D_{\neq}$ *is the set of unique data instances (worlds) represented in* D*. Therefore,* $C_D(P, Q)$ *will be minimized when Equation 6 is maximized.*

We leave the proof of this theorem to Appendix B.1 as the intuition is fairly straightforward from Lam and Bacchus' theorem (Appendix D Theorem 3). We have established that the encoding length of the data is a increasing function solely of C_D and that maximizing Equation 6 minimizes C_D and thereby the encoding length. With these results, we can deduce the existence of a theoretical BKB that will have an equal to or better data encoding length than the induced worlds of a BN given D.

Theorem 2. *Given a BN G learned over a dataset D as to maximize the weight $W_G = \sum_i I(X_i; \Pi(X_i))$ given the structure G, there exists a BKB K with a weight W_K according to Equation 6 such that $W_K \geq W_G$.*

We defer a detailed proof of Theorem 2 to Appendix B.2 and provide a more concise proof sketch. The key insight is that any BN G can be transformed into a corresponding BKB K_G as BKBs subsume BNs. We are only interested in the data encoding length which can now be calculated over K_G by summing over the complete worlds represented in G. Consider a single random variable X_i and it's associated parent set $\Pi(X_i)$. For each instantiation of X_i and $\Pi(X_i)$ there will be an associated S-node created in K_G with an instantiated weight according to Equation 6. Since the choice of parent set for each random variable instantiation in $\Pi(X_i) \cup \{X_i\}$ is governed by G, we don't consider other S-node configurations of the same instantiated random variable set that may have greater weight. The BN structure constrains our S-node structures. Therefore, if we start with a optimal BN, transform it into a BKB K_G, and analyze every permutation of each S-node's possible configurations taking the permutation that maximizes the instantiated weight, we will end up with a BKB K with the same number of S-nodes that has a total weight equal to or greater than the BN representation K_G. This result allows us to also state the following corollary based on the fact that $I(X_i; \Pi(X_i)) = H(X_i) - H(X_i|\Pi(X_i))$.

Corollary 2. *Since $I(X_i; Pi(X_i)) = H(X_i) - H(X_i|\Pi(X_i)) \geq 0$. We can maximize Equation 6 by minimizing the instantiated conditional entropy $H(x_{i\tau}|\pi_{i\tau}) = p(x_{i\tau}, \pi_{i\tau}) \log_2 \frac{p(x_{i\tau}, \pi_{i\tau})}{p(\pi_{i\tau})}$.*

BKB Structure Learning

Theorem 1 dictates that for every random variable instantiation $x_{i\tau}$ in a data instance (world) $\tau \in D_{\neq}$ where $D_{\neq}$ is the set of unique data instances in D, we should assign an instantiated parent set $\pi_{i\tau}$ such that the instantiated conditional entropy is minimized according to Corollary 2. The key insight of our structure learning approach is that we can decompose our learning over the worlds represented in the data. In each world, we will have at most a single instantiation of each RV and our goal is to select a set of S-nodes with a structure that minimizes instantiated conditional entropy for that world. We can view each world in the data as a separate complete inference which form an acyclic subgraph of their respective encompassing BKB. A precise definition of a BKB inference can be found in Appendix D.

Our structure learning algorithm reduces to finding a directed acyclic inference graph for each world that minimizes $\sum_\tau \sum_i H(x_{i\tau}|\pi_{i\tau}) = p(x_{i\tau}, \pi_{i\tau}) \log_2 \frac{p(x_{i\tau}, \pi_{i\tau})}{p(\pi_{i\tau})}$. Further, we can use any off-the-shelf DAG learning algorithm to accomplish this step so far as our scoring function inherently minimizes instantiated conditional entropy and BKB encoding length. There has been significant advancements in field of BN and DAG learning and we make no attempt in covering all such procedures. Instead we will focus on the state-of-the-art exact BN (DAG) solver GOBNILP (Cussens 2012; Cussens, Haws, and Studeny 2015).

Algorithm 1: BKB Structure Learning

Input: Dataset D, Source Reliabilities R, DAG learning algorithm f and hyperparameters Θ

```
K ← ∅
for τ ∈ D_≠ do
    G_τ ← f(τ, R, Θ)
    K ← K ∪ {G_τ}
end for
return BKB-Fusion(K, R)
```

Upon learning each minimal entropy inference, we then need a method for merging this knowledge together that is semantically sound. A standard union type operation will not generally be supported as the unioned BKB would likely incur many mutex violations as seen in Appendix Figure 4a. Instead, we can employ a well-studied BKB fusion (Santos, Wilkinson, and Santos 2011; Yakaboski and Santos 2021) algorithm that supports the fusion of an arbitrary number of BKB Fragments (BKFs) by attaching source I-nodes to every S-node corresponding to the data instance from which the inference graph originated. A graphical example of this approach is depicted in Appendix Figure 4b along with additional information regarding BKB fusion in Appendix D.3. This procedure ensures that that no mutual exclusion violations are present in the fused BKB maintaining a consistent probability distribution over the data and leading to model generalization. Appendix D.3 provides a detailed explanation of generalization in fused BKBs.

Aside from forming a mutual exclusive BKB, fusion also presents us with another degree of freedom during learning. If each data instance was generated by an i.i.d. process, it is natural to assume a normalized probability over all source nodes. However, many processes do not generate truly i.i.d or representative samples. Therefore, if we view these source S-nodes as reliabilities that can be tuned, we may be able to correct errors in higher order inference calculations that arise due to under-fitting or over-generalization. We leave such analysis to future work. Combining each of the steps presented so far, we outline our general BKB structure learning procedure in Algorithm 1.

Empirical Results

To demonstrate the utility of both our proposed algorithm as well as our learned BKB models we conducted 40 experiments on benchmark datasets comparing BKBSL and BNSL in terms of MDL and complexity performance. We then conducted 22 cross validation classification experiments to compare accuracy performance with learned BNs as well as a use-case studying the under-determined bioinformatics domain of structural dependency analysis among single-nucleotide polymorphism (SNPs) in breast cancer.

Benchmark Analysis

When comparing MDL between our learned BKBs and BNs, we are only concerned with comparing the *data* encoding length, as the model encoding length is only used to penalize more complex models. Our *data MDL* results in Appendix

Table 1 demonstrate that a BKB learned using our BKBSL algorithm finds a tighter data fit than the best BN in all 40 dataset. Intuitively, this is because the BN must generalize away from potentially good instantiated scores in favor of the entire random variable score.

Our MDL experiments also demonstrates a practical strength of BKBSL over BNSL related to the number of calls to a joint probability calculator or estimator. In order to calculate the necessary scores for an exact DAG learning algorithm like GOBNILP, we needed to calculate empirical joint probabilities from each dataset. For all experiments we tracked the number of unique calls to this function by our BKBSL algorithm and traditional BNSL algorithm. Since BNSL operates at the RV level, it had to calculate all joint probabilities governed by a given parent set configuration. However, BKBSL did not need to calculate the full CPTs as it operates at the RV instantiation level and decomposes over each data instance, reducing the number of calls to this calculator. We feel that this is a more representative complexity performance metric as it is agnostic to equipment configurations. This effect is detailed in Appendix Table 1, and we can see strong correlation between performance savings over BNs and the number of features (Pearson $r = -0.5994$, p-value $= 4.363 \times 10^{-5}$) as well as the number of I-nodes ($r = -0.4916$, p-value $= 0.0013$) in the dataset. All learned BKBs and BNs are hosted on Github and can be viewed within in a Jupyter Notebook for easier visualization.

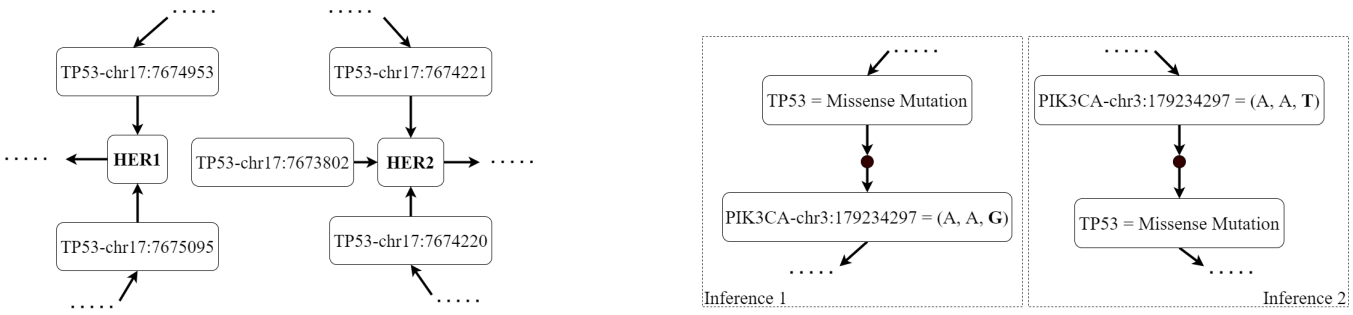
To finalize our BKBSL benchmark analysis we performed accuracy comparisons between our learned BKBs and traditionally learned BNs using GOBNILP and standard MDL scoring. We performed a 10-fold classification cross validation on a subset of only 22 datasets due to the increased learning and reasoning time incurred by running cross validation analysis. We can see from Appendix Table 2 that our BKBSL algorithm is very competitive with BNs in terms of precision, recall and F1-score. Further, our BKBSL models even beat BNs in 63% of cases in terms of precision with greater degradation of performance in terms of recall and F1-score. The alternative hypothesis that either traditionally learned BNs or our learned BKBs will outperform each other in all accuracy categories (Chi^2 Statistic $\chi = 1.0$, p-value $= 0.3173$) is *not* statistically significant. Therefore, we fail to reject the null that learned BNs or BKBs perform better in these cases owing to approximately equal total performance.

This raises the question: Why does our learned BKB perform better in some datasets and not in others? While no feature of the datasets provided any statistically significant predictor of superior performance and leaving more in-depth analysis to future work, we do hypothesize an explanation. It is a well-known problem that real world datasets are often unfaithful to DAGs, e.g. BNs, due to the existence of multiple information equivalent Markov Boundaries (MBs) (Statnikov, Lemeir, and Aliferis 2013; Wang and Wang 2020). Since our BKBSL focuses on learning an optimal structure for every unique exemplar τ, we can view each learned BKF as an equivalent inference from a hypothetical BN whose dependency structure matches that of the associated BKF. As we are only concerned with the specific instantiations of τ, the hypothetical BN and BKF will yield the same probability for this world as their parameters are governed by the same dataset. As our prediction task is to determine the most likely state of a response variable Y given a complete set of evidence E, e.g., $y^* = \text{argmax}_y Q(Y = y \mid E)$, then the closer our joint probability $Q(Y = y, E)$ is to the true data distribution $P(Y = y, E)$ the more accurate our classifier. This is due to the fact when comparing all conditional probabilities $Q(Y = y_i \mid E) = \frac{Q(Y=y_i,E)}{Q(E)}$ the denominator cancels out and we are only concerned with the accuracy of $Q(Y = y_i, E)$. If we imagine our learned BKFs deriving from various hypothetical BNs each with uniquely induced MBs for every RV, our fused BKB essentially incorporates a multiplicity of MBs choices for each of these hypothetical BNs and selects the best performing choice for every world of interest, i.e., prediction class given evidence. We hypothesize that our BKBSL will then perform better on datasets that induce more information equivalent MBs since a BN must select only one and our BKBSL can incorporate multiple in its predictions. Whereas in datasets with fewer MBs, our BKBSL performance may degrade due to overfitting. We intend to study this area further as it may yield clear indications about when to use BKBSL over BNSL in practice.

Gene Regulatory Network (GRN) Application in Breast Cancer

We applied our approach to somatic mutation profiles of breast cancer cases in TCGA (Tomczak, Czerwińska, and Wiznerowicz 2015) to study whether our learned model could still offer utility in this extremely under-determined domain. Since prediction accuracy would not be a reliable metric of success in this dataset, we focused our analysis on hypothesizing potentially significant mutational interactions in cancer. However, if we are to trust any structural hypotheses generated by our approach, we need to ensure the model captures two fundamental biological concepts: (1) We can extract two- or three-hit interactions that are supported in the literature (Knudson 1971, 2001; Segditsas et al. 2009), and (2) we can identify and (possibly) handle genomic instability (Bai et al. 2014; Croessmann et al. 2017).

Given the well-regarded two- and three-hit hypotheses for understanding the role of genetic mutations in cancer development, a model attempting to describe a mutational GRN should be able to capture this concept. The premise behind the two- or three-hit hypotheses is that because many cancers are driven by mutations in various tumor suppressor genes and these loss-of-function mutations are recessive in nature, in general, at least two mutations in these genes are needed to develop cancer. There are certain tumor suppressors genes such as TP53 (Kandoth et al. 2013; Muller and Vousden 2014) that are common in all cancer sub-types and are likely the first hit for non-hereditary cancers. Looking at the dependence relationship subgraph in our learned BKB between a first hit tumor suppressor gene such as TP53 and a second hit tumor suppressor gene related to breast cancer such as a HER1 or HER2 (Osborne, Wilson, and Tripathy 2004), we should observe a directed dependency relationship from TP53 to HER1 or HER2. Figure 2a shows we observe this relationship adding to the biological validity of our model.



(a) (b)

Figure 2: (a) RV level graph of learned BKB over TCGA breast cancer subgraphed on TP53 SNP relationships with HER genes. (b) Genomic instability evidence from the RV level cycle between PIK3CA SNP and general TP53 gene variable. To capture multiple levels of granularity, we included features related to exact positional SNPs as well as including gene level features and general variant classifications. For details about our naming conventions and feature selection process see Appendix A.

It is well known that cancer is also driven by genomic instability (Negrini, Gorgoulis, and Halazonetis 2010), the increased tendency for gene mutations to accumulate, disrupting various biological pathways and in turn causing more mutations. An artifact that we discovered in our BKB network analysis and illustrative of our first example is that in some cases there exist cyclic relationships on the random variable level. This is due to separate inferences learning opposing dependency relationships due to the additional context of their respective inference. This effect can be seen in Figure 2b. Here we have a cyclic random variable relationship between TP53 and the SNP located at chromosome 17 position 179234297 of the PIK3CA gene. TP53 and PIK3CA are known drivers of genomic instability (Bai et al. 2014), affecting the mutational states of many other genes in their inference contexts. Since our inference level parent set limit was set to one, our algorithm cannot reliably extract mutual dependency relationship between TP53 and PIK3CA, thereby causing different inferences to have different directionalities. This cannot be captured by a BN. Further, this result is supported by in-vitro research regarding the joint effect of PIK3CA and TP53 on genomic instability in breast cancer (Croessmann et al. 2017) and is expected from this model given that both genes drive many other downstream mutations. Overall we found 12 of these cyclic relationships in our learned BKB. Of these 12 associations we found literature support for four of them, namely, PIK3CA and TP53 (Croessmann et al. 2017), OBSCN and TP53 (Sjöblom et al. 2006; Perry et al. 2013), MAP3K1 and PIK3CA (Avivar-Valderas et al. 2018), CAD and PIK3CA (Wu et al. 2014).

Limitations and Future Work

The primary limitation of our approach is that we need to learn a BKB fragment for every instance in the training dataset. While we have some complexity benefits from making fewer calls to a given joint probability calculator, this benefit could be neutralized due to the requirement of having to run the DAG learning algorithm multiple times. For instance, once all scores are calculated, BNSL only requires a single run of the respective DAG learning algorithm, whereas our BKBSL approach requires running this algorithm N times. We leave to future work the exploration of this trade off while hypothesizing that there may be a DAG learning formulation in which all fragments can be learned in a single pass by reusing/sharing local scores/structures between fragments.

While our BKBSL algorithm seems to generalize well to unobserved examples in our benchmark experiments, we still see instances with significant accuracy degradation. It is a largely unanswered question as to why machine learning algorithms perform better or worse on particular datasets (Pham and Triantaphyllou 2008) and we have detailed a possible Markov blacket explanation to be explored in future work. Further, we could also address accuracy degradation by tuning the source node reliabilities in our model. Such an approach yields another degree of freedom to adjust the model and also may highlight the importance/significance of individual data instances in relation to over model accuracy. We also leave this direction for future research.

Conclusions

We have presented a new approach for performing Bayesian structure learning at the random variable instantiation level by leveraging Bayesian Knowledge Bases. We have detailed a theoretical justification for our algorithm and learned model as being the fusion of minimal entropy inferences or explanations over each training exemplar. We demonstrated empirically that our BKBSL algorithm finds a superior BKB than an equivalent BN (scored as BKB) on 40 benchmark datasets using our MDL formulation. Further, we demonstrated the practical utility of our approach by presenting statistically competitive accuracy with learned BNs over 22 benchmark datasets using a 10-fold cross validation. This provides evidence that our algorithm adequately generalizes to unseen data based on known knowledge rather than over-generalizing to force a complete distribution. Lastly, we conducted a structural analysis over a gene regulatory network learned from breast cancer mutation data taken from TCGA. This analysis resulted in finding dependency relationships that matched biological intuition and also revealed associations that are well known in the bioinformatics community.

Acknowledgements

This research was funded in part by the National Center for Advancing Translational Sciences (NCATS), National Institutes of Health, through the Biomedical Data Translator program (NIH award OT2TR003436), Air Force Office of Scientific Research Grant No. FA9550-20-1-0032 and DURIP Grant No. N00014-15-1-2514. Special thanks to Joseph Gormley and Eugene Hinderer for their comments and encouragement.

References

Akaike, H. 1974. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6): 716–723.

Avivar-Valderas, A.; McEwen, R.; Taheri-Ghahfarokhi, A.; Carnevalli, L. S.; Hardaker, E. L.; Maresca, M.; Hudson, K.; Harrington, E. A.; and Cruzalegui, F. 2018. Functional significance of co-occurring mutations in PIK3CA and MAP3K1 in breast cancer. *Oncotarget*, 9(30): 21444–21458.

Bai, X.; Zhang, E.; Ye, H.; Nandakumar, V.; Wang, Z.; Chen, L.; Tang, C.; Li, J.; Li, H.; Zhang, W.; Han, W.; Lou, F.; Zhang, D.; Sun, H.; Dong, H.; Zhang, G.; Liu, Z.; Dong, Z.; Guo, B.; Yan, H.; Yan, C.; Wang, L.; Su, Z.; Li, Y.; Jones, L.; Huang, X. F.; Chen, S.-Y.; and Gao, J. 2014. PIK3CA and TP53 Gene Mutations in Human Breast Cancer Tumors Frequently Detected by Ion Torrent DNA Sequencing. *PLOS ONE*, 9(6): e99306.

Burkart, N.; and Huber, M. F. 2021. A Survey on the Explainability of Supervised Machine Learning. *Journal of Artificial Intelligence Research*, 70: 245–317.

Chickering, D. M.; Heckerman, D.; and Meek, C. 2004. Large-Sample Learning of Bayesian Networks is NP-Hard. *Journal of Machine Learning Research*, 5: 44.

Chow, C.; and Liu, C. 1968. Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, 14(3): 462–467.

Croessmann, S.; Wong, H. Y.; Zabransky, D. J.; Chu, D.; Rosen, D. M.; Cidado, J.; Cochran, R. L.; Dalton, W. B.; Erlanger, B.; Cravero, K.; Button, B.; Kyker-Snowman, K.; Hurley, P. J.; Lauring, J.; and Park, B. H. 2017. PIK3CA mutations and TP53 alterations cooperate to increase cancerous phenotypes and tumor heterogeneity. *Breast Cancer Research and Treatment*, 162(3): 451–464.

Cussens, J. 2012. Bayesian network learning with cutting planes. *arXiv:1202.3713*.

Cussens, J.; Haws, D.; and Studeny, M. 2015. Polyhedral aspects of score equivalence in Bayesian network structure learning. *arXiv:1503.00829*.

Dasgupta, S. 1999. Learning Polytrees. In *Fifteenth Conference on Uncertainty in Artificial Intelligence*, 134–141. ArXiv: 1301.6688.

Došilović, F. K.; Brčić, M.; and Hlupić, N. 2018. Explainable artificial intelligence: A survey. In *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 0210–0215.

Drugan, M. M.; and Wiering, M. A. 2010. Feature selection for Bayesian network classifiers using the MDL-FS score. *International journal of approximate reasoning*, 51(6): 695–717.

Grüttemeier, N.; and Komusiewicz, C. 2020. Learning Bayesian Networks Under Sparsity Constraints: A Parameterized Complexity Analysis. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, 4245–4251. Yokohama, Japan. ISBN 978-0-9992411-6-5.

Hansen, M. H.; and Yu, B. 2001. Model selection and the principle of minimum description length. *Journal of the American Statistical Association*, 96(454): 746–774.

Heckerman, D.; Chickering, D. M.; Meek, C.; Rounthwaite, R.; and Kadie, C. 2000. Dependency networks for inference, collaborative filtering, and data visualization. *Journal of Machine Learning Research*, 1(Oct): 49–75.

Jr, E. S.; and Santos, E. S. 1999. A framework for building knowledge-bases under uncertainty. *J. Exp. Theor. Artif. Intell.*, 11(2): 265–286.

Kandoth, C.; McLellan, M. D.; Vandin, F.; Ye, K.; Niu, B.; Lu, C.; Xie, M.; Zhang, Q.; McMichael, J. F.; Wyczalkowski, M. A.; Leiserson, M. D. M.; Miller, C. A.; Welch, J. S.; Walter, M. J.; Wendl, M. C.; Ley, T. J.; Wilson, R. K.; Raphael, B. J.; and Ding, L. 2013. Mutational landscape and significance across 12 major cancer types. *Nature*, 502(7471): 333–339.

Kardan, N.; and Stanley, K. O. 2018. Fitted Learning: Models with Awareness of their Limits. *arXiv:1609.02226*.

Knudson, A. G. 1971. Mutation and cancer: statistical study of retinoblastoma. *Proceedings of the National Academy of Sciences*, 68(4): 820–823.

Knudson, A. G. 2001. Two genetic hits (more or less) to cancer. *Nature Reviews Cancer*, 1(2): 157–162.

Koller, D.; and Friedman, N. 2009. *Probabilistic graphical models: principles and techniques*. MIT press.

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

Liu, Z.; Malone, B.; and Yuan, C. 2012. Empirical evaluation of scoring functions for Bayesian network model selection. *BMC Bioinformatics*, 13(S15): S14.

MacKay, D. J. C. 2005. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press.

Malagrino, L. S.; Roman, N. T.; and Monteiro, A. M. 2018. Forecasting stock market index daily direction: A Bayesian Network approach. *Expert Systems with Applications*, 105: 11–22.

Muller, P.; and Vousden, K. 2014. Mutant p53 in Cancer: New Functions and Therapeutic Opportunities. *Cancer Cell*, 25(3): 304–317.

Negrini, S.; Gorgoulis, V. G.; and Halazonetis, T. D. 2010. Genomic instability — an evolving hallmark of cancer. *Nature Reviews Molecular Cell Biology*, 11(3): 220–228.

Nguyen, A.; Yosinski, J.; and Clune, J. 2015. Deep Neural Networks are Easily Fooled: High Confidence Predictions for Unrecognizable Images.

Osborne, C.; Wilson, P.; and Tripathy, D. 2004. Oncogenes and Tumor Suppressor Genes in Breast Cancer: Potential Diagnostic and Therapeutic Applications. *The Oncologist*, 9(4): 361–377.

Pearl, J. 1986. Fusion, propagation, and structuring in belief networks. *Artificial Intelligence*, 29(3): 241–288.

Perry, N. A.; Ackermann, M. A.; Shriver, M.; Hu, L.-Y. R.; and Kontrogianni-Konstantopoulos, A. 2013. Obscurins: Unassuming Giants Enter the Spotlight. *IUBMB life*, 65(6): 479–486.

Pham, H. N. A.; and Triantaphyllou, E. 2008. The Impact of Overfitting and Overgeneralization on the Classification Accuracy in Data Mining. In Maimon, O.; and Rokach, L., eds., *Soft Computing for Knowledge Discovery and Data Mining*, 391–431. Boston, MA: Springer US. ISBN 978-0-387-69935-6.

Rissanen, J. 1978. Modeling by shortest data description. *Automatica*, 14(5): 465–471.

Santos, E.; Gu, Q.; and Santos, E. E. 2013. Bayesian knowledge base tuning. *International Journal of Approximate Reasoning*, 54(8): 1000–1012.

Santos, E.; Wilkinson, J. T.; and Santos, E. E. 2011. Fusing multiple Bayesian knowledge sources. *International Journal of Approximate Reasoning*, 52(7): 935–947.

Sauta, E.; Demartini, A.; Vitali, F.; Riva, A.; and Bellazzi, R. 2020. A Bayesian data fusion based approach for learning genome-wide transcriptional regulatory networks. *BMC bioinformatics*, 21: 1–28. Publisher: Springer.

Scanagatta, M.; de Campos, C. P.; Zaffalon, M.; and Corani, G. 2015. Learning Bayesian Networks with Thousands of Variables. *NIPS*.

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

Segditsas, S.; Rowan, A. J.; Howarth, K.; Jones, A.; Leedham, S.; Wright, N. A.; Gorman, P.; Chambers, W.; Domingo, E.; Roylance, R. R.; Sawyer, E. J.; Sieber, O. M.; and Tomlinson, I. P. M. 2009. APC and the three-hit hypothesis. *Oncogene*, 28(1): 146–155.

Shih, A.; Choi, A.; and Darwiche, A. 2018. Formal Verification of Bayesian Network Classifiers. *Proceedings of Machine Learning Research*, 72: 427–438.

Sjöblom, T.; Jones, S.; Wood, L. D.; Parsons, D. W.; Lin, J.; Barber, T. D.; Mandelker, D.; Leary, R. J.; Ptak, J.; Silliman, N.; Szabo, S.; Buckhaults, P.; Farrell, C.; Meeh, P.; Markowitz, S. D.; Willis, J.; Dawson, D.; Willson, J. K. V.; Gazdar, A. F.; Hartigan, J.; Wu, L.; Liu, C.; Parmigiani, G.; Park, B. H.; Bachman, K. E.; Papadopoulos, N.; Vogelstein, B.; Kinzler, K. W.; and Velculescu, V. E. 2006. The consensus coding sequences of human breast and colorectal cancers. *Science (New York, N.Y.)*, 314(5797): 268–274.

Statnikov, A.; Lemeir, J.; and Aliferis, C. F. 2013. Algorithms for discovery of multiple Markov boundaries. *The Journal of Machine Learning Research*, 14(1): 499–566.

Szegedy, C.; Zaremba, W.; Sutskever, I.; Bruna, J.; Erhan, D.; Goodfellow, I.; and Fergus, R. 2014. Intriguing properties of neural networks. *arXiv:1312.6199*.

Tomczak, K.; Czerwińska, P.; and Wiznerowicz, M. 2015. The Cancer Genome Atlas (TCGA): an immeasurable source of knowledge. *Contemporary oncology*, 19(1A): A68.

Trösser, F.; de Givry, S.; and Katsirelos, G. 2021. Improved Acyclicity Reasoning for Bayesian Network Structure Learning with Constraint Programming. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence*, 4250–4257. Montreal, Canada: International Joint Conferences on Artificial Intelligence Organization. ISBN 978-0-9992411-9-6.

Wang, Y.; and Wang, L. 2020. Causal inference in degenerate systems: An impossibility result. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, 3383–3392. PMLR.

Wu, X.; Renuse, S.; Sahasrabuddhe, N. A.; Zahari, M. S.; Chaerkady, R.; Kim, M.-S.; Nirujogi, R. S.; Mohseni, M.; Kumar, P.; Raju, R.; Zhong, J.; Yang, J.; Neiswinger, J.; Jeong, J.-S.; Newman, R.; Powers, M. A.; Somani, B. L.; Gabrielson, E.; Sukumar, S.; Stearns, V.; Qian, J.; Zhu, H.; Vogelstein, B.; Park, B. H.; and Pandey, A. 2014. Activation of diverse signaling pathways by oncogenic PIK3CA mutations. *Nature communications*, 5: 4961.

Xue, D.; Lin, H.; Lin, L.; Wei, Q.; Yang, S.; and Chen, X. 2021. TTN/TP53 mutation might act as the predictor for chemotherapy response in lung adenocarcinoma and lung squamous carcinoma patients. *Translational Cancer Research*, 10(3).

Yakaboski, C.; and Santos, E. 2018. Bayesian Knowledge Base Distance-Based Tuning. In *2018 IEEE/WIC/ACM International Conference on Web Intelligence, WI 2018, Santiago, Chile, December 3-6, 2018*, 64–72. IEEE Computer Society.

Yakaboski, C.; and Santos, E. 2021. Efficient Reasoning upon Fusion of Many Data Sources. In Bell, E.; and Keshtkar, F., eds., *Proceedings of the Thirty-Fourth International Florida Artificial Intelligence Research Society Conference, North Miami Beach, Florida, USA, May 17-19, 2021*.

Yang, S.; and Chang, K.-C. 2002. Comparison of score metrics for Bayesian network learning. In *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, volume 32, 419–428.

Yun, Z.; and Keong, K. 2004. Improved MDL score for learning of Bayesian networks. In *Proceedings of the International Conference on Artificial Intelligence in Science and Technology, AISAT*, 98–103.