Improving Bayesian Network Structure Search with Random Variable Aggregation Hierarchies

John Burge and Terran Lane

University of New Mexico, Department of Computer Science
{lawnguy, terran}@cs.unm.edu

Abstract. Bayesian network structure identification is known to be NP-Hard in the general case. We demonstrate a heuristic search for structure identification based on *aggregation hierarchies*. The basic idea is to perform initial exhaustive searches on composite "high-level" random variables (RVs) that are created via aggregations of atomic RVs. The results of the high-level searches then constrain a refined search on the atomic RVs. We demonstrate our methods on a challenging real-world neuroimaging domain and show that they consistently yield higher scoring networks when compared to traditional searches, provided sufficient topological complexity is permitted. On simulated data, where ground truth is known and controllable, our methods yield improved classification accuracy and structural precision, but can also result in reduced structural recall on particularly noisy datasets.

Keywords: Bayesian network structure search hierarchy fMRI.

1 Introduction

Bayesian networks (BNs) [17] are a widely employed graphical modeling framework used to reason under uncertainty. Their topological structures describe correlational (or possibly causal) relationships among random variables (RVs). This topology may not be known a priori and must be searched for—a process known to be NP-hard in the general case [4]. Instead of directly learning the structure for a BN with a large number of RVs, we propose that searches may first be performed on simpler domains whose RVs are constructed as the aggregation of the original domain's RVs. The results of these searches can then influence searches on the original domain via structural priors or as modifications to search heuristics which allow exhaustive searches on constrained structure spaces.

Our approach is analogous to the statistical problem of blood pooling. Assume that a blood test for some disease must be performed on many patients but is expensive and cannot be applied exhaustively. Instead, blood samples are divided into a small number of groups and pooled into aggregate group samples. Results from the pooled samples can then be used to constrain the application of the test to the individual samples by only testing the individual constituents of a positive group sample.

Just as results on the pooled group samples indicate which individual samples to test, elicited correlations among composite RVs can guide elicitation of correlations

J. Fünrnkranz, T. Scheffer, and M. Spiliopoulou (Eds.): ECML 2006, LNAI 4212, pp. 66 – 77, 2006.

among atomic RVs. To illustrate this, consider an example from a neurological domain. At a fine level, some neuroanatomical databases break up the human brain into approximately 70 regions of interest (ROIs). The search space for a BN with 70 RVs contains on the order of 10^{23} structures and cannot be searched exhaustively. However, the neuroanatomical databases can aggregate these ROIs into roughly 50 ROIs, which can then be further aggregated into 12 and then 7 ROIs. A BN with only seven nodes requires roughly 1,000 structures to be searched and could be performed exhaustively. Results from this search could then be used to constrain searches among finer RVs under the assumption that correlations among those RVs will be observable as correlations among the gross ROIs they compose.

We demonstrate our methods on such a neuroimaging domain, but there are many other domains where RVs may be sensibly aggregated together. E.g., other image analyses where pixel neighborhoods of varying size can be grouped together; geographic data such as cities, states and countries; genetic regulatory network reconstruction where genes can be grouped into families and super-families; document topic hierarchies (e.g., newsgroups); word types in grammar trees; medical diagnoses where diseases and symptoms are grouped into sub-categories; and Fourier and wavelet analyses where coefficients are spatially and temporally related.

Typically, the RV aggregations can be arranged into a hierarchy. To form this hierarchy, two domain-specific questions must be answered. First, which RVs should be aggregated together and second, what function should perform the aggregation? In the neuroimaging domain, we group ROIs together based on spatial and functional locality and aggregate them as a weighted linear combination.

Of course, the assumption that correlations will persist across the aggregation hierarchy will be violated to some degree in most domains. Further, while constraining subsequent structure searches based on previous structure results is intuitive and appealing, straightforward implementations can yield unfavorable results. We empirically demonstrate this and propose a constraint mechanism which performs well. For both generative and class-discriminative scores, our methods consistently yield higher scoring structures than traditional searches on four neuroimaging datasets collected under widely differing paradigms, provided that the search is allowed to produce BNs with sufficient structural complexity—typically two to three parents per node. On a simulated domain, in which ground truth is known and controllable, we demonstrate higher classification accuracy and structural precision, but also lowered structural recall on particularly noisy datasets.

2 Background

Bayesian Networks (BNs) [17] are graphical models that explicitly represent dependencies among RVs. A BN's topological structure, represented as a directed acyclic graph, contains nodes for RVs and directed links between correlated *parent* and *child* nodes. A *family* is composed of a single child and its parents. We assume fully observable discrete RVs so that a family's conditional probability, *P*(*child* | *parents*), can be represented with a conditional probability table (CPT).

Searching for a BN's topology is accomplished by proposing as many hypothesis structures as possible, guided by a search heuristic, while measuring the goodness of fit between the structures and the data via a structure scoring function. Iterative hill climbing heuristics are commonly employed. For example, starting with a topology with no links, score all legal modifications to the topology where a legal modification is the addition, removal or reversal of a link not resulting in a cycle. Choose the modification that results in the highest score and iterate until no modifications yield improvements. We refer to this as a flat structure search.

Structure scoring functions typically come in two varieties: generative and class-discriminative. Generative scores select structures that increase the posterior likelihood of the data given the structure. Common examples include MDL [13], BIC [18], BDe [10], etc. Discriminative scores select structures that increase the class discriminative ability of learned BNs. Examples include the class-conditional likelihood (CCL) [8] and the approximate conditional likelihood (ACL) [2]. With the notable exception of CCL, most scores are decomposable, i.e., a family's contribution to the score is independent of all other families' topologies.

We use the following notation. Let ***X*** represent a set of n RVs, $\{X_1, X_2, \ldots, X_n\}$ with arities $r_1, r_2, \ldots, r_n$. A data point is a fully observable assignment of values to ***X***. A BN, B, over ***X*** is described by the pair $\langle B_S, B_\Theta \rangle$. B_S is the DAG representing the BN's structural topology. X_i's parent set is denoted $Pa(X_i)$. q_i is the number of configurations for the RVs in $Pa(X_i)$. $B_\Theta = \{\Theta^B_{i,j,k} : 1 \le i \le n, 1 \le j \le r_i, 1 \le k \le q_i\}$ is the set of CPT parameters where $\Theta^B_{i,j,k} = P(X_i = j \mid Pa(X_i) = k)$. $I_{X',Y'}$ is an indicator function which equals one iff there exists a link between RVs in the sets ***X'*** and ***Y'***. Finally, $I_{\varnothing,\varnothing} = I_{X',\varnothing} = I_{\varnothing,X'} = 1$ and $\overline{I}_{X',Y'} = 1 - I_{X',Y'}$.

2.1 Aggregation Hierarchies

Decomposing a complex model into a series of hierarchically related components has been shown to be helpful in many domains. For example, Fine, Singer and Tishby [7] introduce a hierarchical abstraction of hidden Markov models; Gyftodimos and Flatch [9] introduce a hierarchical abstraction of BNs in general and Sutton, Precup & Singh [19] incorporate hierarchies within reinforcement learning.

As in this previous work, we hierarchically decompose a domain into multiple models of varying complexity. Setting our work apart from much of the prior work, we use structural results learned in one model to guide learning in subsequent models. To our knowledge, we are the first to do this with BN structure search, though similar methods for BN parameter learning have been proposed. E.g., Anderson, Domingos and Weld [1] and McCallum et al. [15] use shrinkage to improve parameter learning by combining varying levels of bias and variance in hierarchically related models.

To form our hierarchies, we create *composite RVs* as aggregations of a domain's original *atomic RVs*. Let $\hat{X} = \{X_1, \ldots, X_\tau\} \in$ ***X***. A scalar function of $\hat{X}$, $Y = \xi(\hat{X})$, is an *aggregation function* where Y is a RV whose distribution reflects some aspect of the joint distribution of $\hat{X}$. Common examples of aggregations include *max*, *count*, *variance*, etc. For our neuroimaging domain, we employ the *weighted mean aggregate*, $\xi(\hat{X}) = \sum_{i=1}^{\tau} \alpha_i X_i$, where the α_i's are set by a neuroanatomical database.

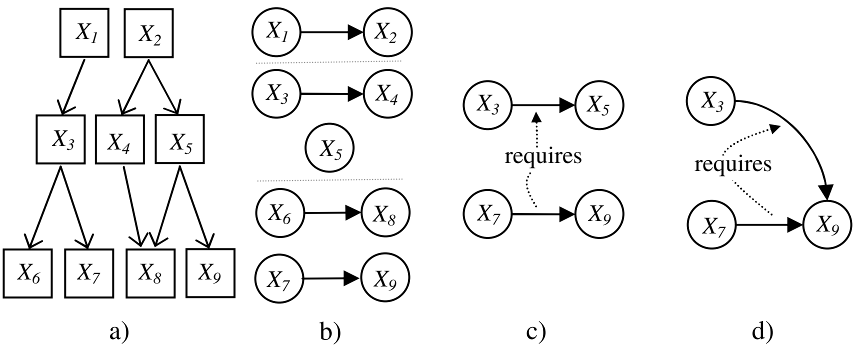

Fig. 1. a) An example hierarchy detailing the hierarchical trellis for the RVs in $X = \{X_1, \ldots, X_9\}$. Boxes are used to emphasize this is not a BN. b) An example BN defined for X. The dotted lines indicate a division between RVs in different hierarchical levels. The link between X_7 and X_9 does not satisfy either hierarchical assumption. c) For the family-wise assumption, the $X_7 \rightarrow X_9$ link requires the existence of the $X_3 \rightarrow X_5$ link. d) For the parent-wise assumption, the $X_7 \rightarrow X_9$ link requires the existence of the $X_3 \rightarrow X_9$ link.

$\boldsymbol{X}$ includes both the atomic RVs as well as the composite RVs. An aggregation hierarchy over $\boldsymbol{X}$ (Figure 1a) can be graphically represented as a *trellis*. A trellis is a relaxation of a forest such that each node may have multiple parents. Let Λ_X be a trellis over $\boldsymbol{X}$ whose leaves are atomic RVs and whose internal nodes are aggregations of their children. Let Λ_{X_i} denote the children of X_i, Λ^{X_i} denote the parents of X_i, Λ_i denote the integer-valued level at which X_i is located in and $\Lambda_{X_i}^{X_i} = \Lambda_{X_i} \cup \Lambda^{X_i}$. Λ_{X_i} = $\varnothing$ for leaves and Λ^{X_i} = $\varnothing$ for the root(s). If X_i is not a leaf node then $X_i = \xi(\Lambda_{X_i})$. $\Lambda(v)$ is the set of RVs at the v^{th} level in the hierarchy and is referred to as an *h-level*. *levels*(Λ_X) returns the number of h-levels in Λ_X. Relationships among RVs, such as *parent* and *child*, are prefixed with *h*- when used in the context of a hierarchy.

2.2 Hierarchical Bayesian Network Structure Search

Exhaustive structure searches can be employed at the highest h-levels where few RVs reside. Searches at lower h-levels cannot normally be performed exhaustively, but can be constrained by the previous h-level's search results. One possible constraint mechanism is based on the assumption that links among high-level RVs will be manifested as links among the low-level RVs they were constructed from. Exhaustive search strategies can then be employed for nodes in the lower h-levels on the space of structures that only contain links obeying this assumption.

Take the BN and the hierarchy in Figure 1 for example. There are two nodes at the highest h-level and structure search is trivial. Assume that a structure search found the $X_1 \rightarrow X_2$ link. According to the hierarchy, $\{X_3\}$ and $\{X_4, X_5\}$ are the h-children of X_1 and X_2. As a correlation between X_1 and X_2 was observed, correlations among their constituents should be searched for. That search could yield, for instance, the $X_3 \rightarrow X_4$ link and then, at the next h-level, the $X_6 \rightarrow X_8$ link. Of course, limiting searches based on this assumption results in links that will not be searched. E.g., the $X_7 \rightarrow X_9$ link will not be searched since there is no link between X_7 and X_9's h-parents, $\{X_3\}$ and $\{X_5\}$. We call this assumption the *family-wise assumption* as the relationships detailed in a family at one level are manifested as families at lower h-levels.

Definition. The *family-wise assumption.* $\forall$ X_i and X_j, if $I_{\Lambda^{X_i},\Lambda^{X_j}} = 0$ or $\Lambda_i \neq \Lambda_j$, then $X_i \notin Pa\left(X_j\right)$,

This assumption does not allow a node's parent set to include its h-parents and h-children, which, given that a node is constructed from its h-children, are likely to be significant. We relax this assumption to allow this.

Definition. *Relaxed family-wise assumption.* $\forall$ X_i and X_j, if $I_{\Lambda^{X_i},\Lambda^{X_j}} = 0$ and $X_i \notin \Lambda_{X_j}^{X_j}$, then $X_i \notin Pa\left(X_j\right)$

The relaxed assumption does not limit candidate parent sets as effectively as the unrelaxed assumption (particularly in dense trellises) and is less likely to allow for exhaustive searches. Ultimately, this will lead to poor search performance. Hence, we introduce the *parent-wise assumption* which only requires a correlation between two RVs to manifest as a correlation between the child and one of the parent's h-parents.

Definition. *Parent-wise assumption.* $\forall$ X_i, X_j, if $I_{\Lambda^{X_i},X_j} = 0$, then $X_i \notin Pa\left(X_j\right)$.

This requirement is not as strict as the family-wise assumption and has the distinct advantage of easily incorporating a node's h-relatives as candidate parents while still effectively restricting structure spaces. Figures 1c and 1d illustrate the different link requirements for the family-wise and parent-wise assumptions.

These assumptions may be incorporated directly into the BN scoring function. Scoring functions include terms (or can generally be modified to include terms) that probabilistically weight structures based on prior knowledge. Formulating domain knowledge as a structural prior is advantageous as it can be easily incorporated into many structure scores. $P_{rf}(B_S)$ and $P_p(B_S)$ give the relaxed family-wise and parent-wise assumptions as structural priors, respectively:

$$P_{rf}(B_S) = \frac{Z}{1+\alpha v_{rf}}, v_{rf} = \sum_{i \times j} I_{\{X_i\},\{X_j\}} \bar{I}_{\Lambda^{X_i},\Lambda^{X_j}} \bar{I}_{X_i,\Lambda_{X_i}^{X_i}}, \quad P_p(B_S) = \frac{Z}{1+\alpha v_p}, v_p = \sum_{i \times j} I_{\{X_i\},\{X_j\}} \bar{I}_{\Lambda^{X_i},\{X_j\}},$$

where Z is a normalization constant, α is a penalty scale factor, and v_{rf} and v_p are the number of links that violate the relaxed family-wise and parent-wise assumptions. When α is sufficiently large, the prior probability of a structure with any violating links can be treated as zero. Incorporation of the hierarchical assumptions can then be equivalently realized as modifications to structure search heuristics by limiting candidate parent sets. It is this case we investigate in this paper, though, future work investigating the case where α is relatively small is also promising.

For the relaxed family-wise assumption, structure search begins by exhaustively searching for the optimal parent sets for each $X_i \in \Lambda(1)$. Structure searches for the remaining h-levels are then iteratively performed with the structural results of prior h-levels constraining candidate parent sets (CPSs) at subsequent h-levels. The runtime of hierarchical structure searches will typically be longer than that of flat searches but ultimately depends on the CPS limits where exhaustive searches are allowed. We have found that it is reasonable to exhaustively search for a node's optimal parent set when its CPS has less than 20 parents, to use a simulated annealing search when its CPS has less than 40 parents and to resort to a hill climbing search otherwise. We refer to this search as the *RFW-Hier* search.

Unlike searches based on the family-wise assumption, searches based on the parent-wise assumption would require RVs to have many simultaneous parents—far more than would be allowed due to overfitting and computational limitations. This can be addressed by searching for the optimal candidate parents for a node one h-level at a time using the following heuristic. For each RV X_i, exhaustively search for the highest scoring set of n legal parents from $\Lambda(1)$. Record and remove these parents. Then, find the best set of n legal parents for each X_i from each subsequent h-level where the recorded results from the prior h-level constrain the parent sets. This results in a final set of (at most) $n \times levels(\Lambda)$ recorded parents. Perform one last search through this set for the final parent set. As in the RFW-Hier search, we use a combination of exhaustive, simulated annealing and greedy searches. We refer to this search as the *PW-Hier* search.

When searching through CPSs one node at a time, cycles could be introduced into the topology. We address this by placing constraints that ensure no cycles can exist (see Section 3). Other methods for dealing with the introduction of cycles exist, e.g., a *repair* operator that removes cycles that have been introduced [14].

3 Experiments

We test on both simulated and real-world neuroimaging domains. The neuroimaging data is temporal and BNs that explicitly represent time are referred to as *dynamic Bayesian Networks* (DBNs). The simulated data is generated from DBNs.

In the most general case, DBNs include one column of RVs for every time step and one node in each column for every RV. For most real world problems, such DBNs are intractably large. We make the *stationary* and *Markov order 1* assumptions, resulting in a topology of two columns: one for time t and one for time $t+1$. The nodes do not represent absolute time points but instead represent RV correlations averaged across time. Links originate in the left column and terminate in the right. DBNs may also include isochronal links, which we omit as temporal links are of primary interest. Thus, all link additions are guaranteed to be acyclic.

Notation for DBNs is slightly modified from BNs in general. X_i^t and X_i^{t+1} represent the i^{th} RV in columns t and $t+1$ and $\boldsymbol{X} = \{X_i^t, X_i^{t+1}: 1 \le i \le n\}$. The parameters for a node's CPT, $P_B(X_i^{t+e} \mid Pa(X_i^{t+e}) = j)$, are denoted $\Theta^B_{e,i,j,k}$, $e \in \{0,1\}$.

We gauge the efficacy of our heuristics using both generative and discriminative scores. For a generative score, we use the BDe metric [10], a commonly employed metric with a strong mathematical underpinning. Its parameter priors are themselves parameterized by the *equivalent sample size* (ESS), which has the effect of controlling for structural complexity. For a discriminative score, we use the *approximate conditional likelihood* (ACL) score [2], a decomposable alternative to CCL.

3.1 Simulated Domain

Simulated data is created from a pair of DBNs whose topologies are selected at random but comply with the parent-wise hierarchical assumption (structures consistent with the relaxed family-wise assumption were omitted due to space constraints, but results were qualitatively very similar). We test the ability of both flat and hierarchical searches to find the underlying generative structure. Three experimental paradigms are used: an IID case in which data is generated from DBNs with varying magnitudes of differences, a noisy case in which the IID assumptions are violated and a case where hierarchical assumptions are violated.

In all cases, a single hierarchy, Λ_X, over RVs $\boldsymbol{X} = \{X_1, \ldots, X_{57}\}$, is created with 3 h-levels containing 3, 9 and 45 nodes. The hierarchy is a perfectly balanced tree with each node in Λ(1) linking to three unique node in Λ(2), each of which, in turn, links to five unique nodes in Λ(3). The two generating DBNs, G_1 and G_2, are constructed with nodes $\{X_1^t, X_1^{t+1}, \ldots, X_{57}^t, X_{57}^{t+1}\}$. Fifteen links—one between nodes in Λ(1), four between nodes in Λ(2) and ten between nodes in Λ(3)—are created between 15 parents in the *t* column and 15 unique children in the *t+1* column.

The *correlational strength* for a link, measured via a normalized mutual information score (NMIS), is determined by the CPT generated for the child node. At an NMIS of zero, a parent is completely uncorrelated with its child and at an NMIS of one, it is completely correlated. A node with no parents is parameterized by a normalized information score (NIS). At an NIS of zero, the CPT is completely non-uniform and at one its uniform.

The method for generating a CPT for a node X_i^{t+e} that conforms to a NIS or a set of NMISs is outside the scope of this paper. We will refer to it as the distribution $P(\Theta_{e,i}^B \mid S_{e,i}^B)$, where $e \in \{0,1\}$ and $S_{e,i}^B$ is a set containing a single NIS if X_i has no parents, or is a list of NMIS's, with an NMIS for each of the *p* parents. $\Theta_{e,i}^B$ can be modified to produce a new CPT, $\Theta_{e,i}^{\prime B}$, compliant with a different NIS or list of NMIS's, $S_{e,i}^{\prime B}$. This generator is the distribution $P(\Theta_{e,i}^{\prime B} \mid \Theta_{e,i}^B, S_{e,i}^{\prime B})$. The closer $S_{e,i}^{\prime B}$ is to $S_{e,i}^B$, the smaller the KL divergence between $\Theta_{e,i}^{\prime B}$ and $\Theta_{e,i}^B$ will be. If $S_{e,i}^{\prime B}$ = $S_{e,i}^B$, then $\Theta_{e,i}^{\prime B} = \Theta_{e,i}^B$.

The CPTs in G_1 for nodes with no parents are generated from the $P(\Theta_{e,i}^B \mid \{0.9\})$ distribution, yielding fairly uniform CPTs. Nodes with parents are generated from the $P(\Theta_{e,i}^B \mid \{0.1\})$ distribution so that a child's value is only loosely correlated with the parent's value. G_1 and G_2 share an identical structure and all the CPTs in G_2 are copies of those in G_1. Thus, initially G_1 and G_2 represent the same distribution.

The overall process for an experiment is as follows. First, the CPT parameters in G_1 and/or G_2 are modified in accordance to a particular experimental paradigm. Twenty training and twenty testing data points are generated for each class. A DBN is then learned for each class with the BDe score. Classification is performed on a testing data point by selecting the DBN with the largest posterior probability. *Structural precision*, the fraction of links in the learned DBNs present in the generating DBNs, and *structural recall*, the fraction of links in the generating DBNs also found in the learned DBNs, are measured. Each point listed in the resulting graphs in Figures 2 and 3 are calculated as the average of 120 runs of the experiment. Significance tests were computed via the *t*-test for dependent samples.

3.2 Neuroscience Domain

Functional magnetic resonance imaging (fMRI) is widely used in the study and diagnosis of mental illness. It is a non-invasive technique measuring the activity of small cubic regions of brain tissue (voxels). Psychologists frequently use fMRI data to test hypotheses about the changing neural activity underlying mental illness.

There are too many voxels in each 3D fMRI image to model directly, so voxels are marginalized to *regions of interest* (ROIs) via the widely employed Talairach database [12]. Thus, each image is represented as the activation of 147 ROIs. Then, the time series for each ROI is modeled with a temporal RV. Data for each class of patient, healthy vs. diseased, is grouped together and each class is modeled with a DBN containing the nodes $\boldsymbol{X} = \{X_i^t,\ X_i^{t+1} : 1 \leq i \leq 147\}$. The 147 ROIs are hierarchically related via the *mean aggregate function* given in Section 2.1.

We analyze four fMRI datasets collected under widely differing experimental paradigms on different patient populations suffering from different illnesses. The first was collected by Buckner et al. [3] for analysis of senile dementia, the second and third datasets were collected by the Clark et al. [5] and The Mind Institute [16] for schizophrenia and the fourth dataset was collected by Kiehl [11] and also focused on schizophrenic patients. We will refer to these datasets as the *demented*, *schizoM1, schizoM2* and *schizoK* datasets, respectively.

4 Results

The first set of simulated experiments measures how each search performs under IID conditions (Figure 2, top left). The CPTs in G_2 for the nodes with parents are redrawn from the $\mathrm{P}(\Theta_{1,i}^{G_2} \mid \Theta_{1,i}^{G_2},\ \{0.1 \pm c\})$ distribution where c determines the magnitude of difference between G_2's and G_1's CPTs. Addition versus subtraction is chosen at random. As c increases, the difference between classes increases. When $c = 0$, classification is impossible and when $c = 0.02$, classification is trivial.

PW-Hier's accuracy is significantly higher than the flat search's over a wide range. This is due to the increased structural precision of the PW-Hier search. Since PW-Hier decreases the candidate parent space for each node, many candidate parents are omitted which would have only contributed noise. Thus, the flat search is much more likely to add a superfluous node as a parent. Approximately one parent was added per child on average in the flat search compared to only 0.3 in the PW-Hier search.

The magnitude of the structural precision increase is due to the BDe equivalent sample size (ESS), which was set to 500. Figure 2 (top right) gives the results of experiments with c fixed at 0.005 and the ESS varying from 50 to 1,000. As the ESS increases, the search is more likely to add noisy parents. This decreases precision for both classifiers, however, PW-Hier's additional constraints counteract this tendency and its structural precision drops less quickly than the flat search's does.

For most domains, assuming data points are drawn from a noiseless process is unrealistic. The second set of experiments measures a score's tolerance to intra-class noise (Figure 2, bottom left). G_1 and G_2 are treated as *base-line* models, but each data point is generated from a modified version of them. Both G_1 and G_2 are generated as in the first experiment with $c = 0.005$ and the ESS = 500. G_α^g, the generator for the

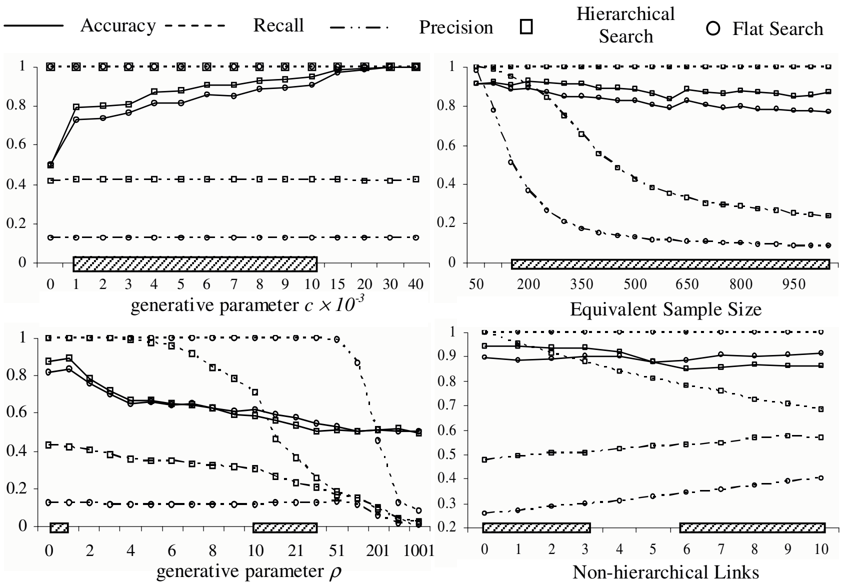

Fig. 2. Classification accuracy, structural recall and structural precision for simulated data experiments. Hierarchical results shown are for the PW-Hier search. RFW-Hier results on simulated data are omitted due to space constraints, but are qualitatively very similar. Shaded boxes on the axis indicate ranges where classification accuracy differences are statistically significant, as measured by the standard *t* test for dependent samples.

g^{th} generated data point for class α, starts as a copy of G_α. For each X_i, ρ random $\langle j,k \rangle$ tuples are chosen, $1 \le j \le r_i$, $1 \le k \le q_i$, and 0.1 is added to $\Theta^{G^g_\alpha}_{1,i,j,k}$. As ρ increases, intra-class differences increase and class discrimination and the base models' true RV correlations becomes more difficult to elicit.

Initially, when ρ equals *1* or *2,* PW-Hier's accuracy is significantly higher than the flat classifier's accuracy. As more randomizations occur, the flat classifier's accuracy catches up and eventually surpasses PW-Hier's. While the structural precision of the PW-Hier search always dominates the flat search, its structural recall begins to diminish significantly before that of the flat classifier. This is because losing the ability to identify a single link can cause a cascade of failures to identify other links. Not recognizing a link at high levels in the hierarchy automatically results in missing all links that depend on it. In the flat classifier, losing any particular link does not increase the risk of losing further links. So in particularly noisy datasets, PW-Hier's structural precision advantage may be overwhelmed by a decrease in structural recall.

Further, in real-world data, it is possible that the candidate parents that PW-Hier omits would be useful. The final set of simulated experiments (Figure 2, lower right) demonstrate what occurs as the number of links in the generative DBNs that *do not* conform to the parent-wise assumption are added. As expected, as the number of violating links increase, the accuracy of the flat classifier catches up and surpasses that of the PW-Hier classifier. At roughly five violating links, corresponding to 20%

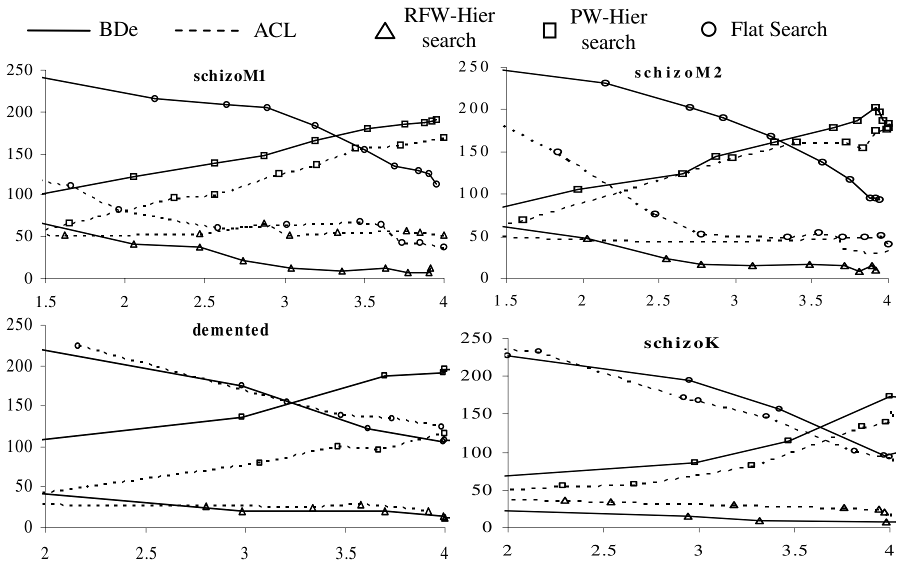

Fig. 3. The number of families for which a search returned the highest scoring parent set. The x-axis gives the average number of parents per family. The PW-Hier search outperformed the flat search in all but the demented dataset with the ACL score provided a certain threshold of complexity was achieved (ranging from 2 to 3.5 average parents).

of the total generative links, the flat classifier and the hierarchical classifier's accuracy are identical. Importantly, as the number of violations increase, PW-Hier's performance degrades gradually, indicating robustness to violations.

4.1 Neuroscience Domain Results

DBNs learned from fMRI data can be employed for several tasks, including the elicitation of correlations among ROIs, classification, creation of simulated surrogate datasets, specific hypothesis testing, etc. For all these tasks, learned DBNs are found by maximizing a scoring metric. In this section, we focus on high scoring networks as a proxy for the myriad of tasks that those networks may eventually be used for.

Figure 3 shows the results for DBNs learned with varying levels of structural complexity (where complexity is measured as the average number of parents per node). For networks learned with BDe, complexity is controlled by the ESS. For networks learned with ACL, complexity is controlled by a minimum description length (MDL) penalty term. Each point in the graph represents the number of families for which the corresponding search returned the highest score. For example, if the highest scoring set of parents for a child node X_i found with a flat search resulted in a ACL score of 15.6, but the *PW-Hier* search found a set of parents for X_i that resulted in a ACL score of 21.7, one point would be added on the y-axis for the *PW-Hier* search results. (Graphs that directly plot structure scores are not shown as they are complicated by complexity penalty trends, but such graphs do not qualitatively differ from those given in Figure 3.)

The results are consistent across each of the datasets. Initially, when the structural complexity is low, the flat search yields family structures with higher scores than the hierarchical search. This is because the hierarchical assumptions are restricting candidate parent sets too dramatically. However, after a certain critical threshold of complexity is reached, around 3.4 parents for BDe and anywhere from 2 to 4 parents for ACL, PW-Hier searches find higher scoring structures than flat searches.

The RFW-Hier search is almost always outperformed. The RFW-Hier search was simply incapable of restricting candidate parents to small enough sets where exhaustive strategies could be used, a key advantage in limiting parent sets to begin with. On the other hand, the PW-Hier search was capable of restricting candidate parents to smaller sets, benefited from exhaustive searches and was capable of outperforming typical flat structure searches on both generative and class discriminative scoring functions.

5 Conclusions

Employing hierarchically related models of varying complexity has proven to be useful in many machine learning applications. We have applied this concept to Bayesian network structure search by aggregating *atomic* random variables (RVs) into a hierarchy of *composite* RVs. Structural results of searches on high-level composite RVs are used to constrain searches on lower-level atomic RVs, allowing exhaustive searches for many of the BN's families.

We introduced two constraint heuristics for restricting searches at one h-level based on the search results at the previous h-level. On both a generative score, BDe [10], and a class-discriminative score, ACL [2], we demonstrated use of these heuristics on multiple datasets in a challenging real-world neuroimaging domain. We empirically showed that the intuitively reasonable *family-wise* search performed poorly while the *parent-wise* search significantly and consistently outperformed traditional, flat structure searches in finding high-scoring families. Results from a simulated domain, in which ground truth was known and controllable, indicated that hierarchical searches increased structural precision and yielded significant improvements to classification. Though, on particularly noisy datasets, a decrease in structural recall was observed which led to decreased classification accuracy.

Our empirical results primarily focused on domains where links between atomic and composite RVs were desirable. This may not be the case in all domains. Unfortunately, the parent-wise search is not useful in such domains, and the family-wise search may not yield desirable results due to its inability to adequately constrain candidate parent sets given dense trellises (such as those used in our neuroimaging domain). Additional work to determine if the family-wise search benefits domains with sparser trellises is warranted, however, as experiments on simulated data indicated similar benefits to the parent-wise search. Another avenue for future work lies in applying our methods to structure searches in relational learning paradigms, whose models contain hierarchies of RVs related with *is-a* and *has-a* relationships.

Acknowledgements. We would like to thank the anonymous reviewers for their valuable feedback as well as the Dartmouth fMRI data center [6], Buckner et al. [3], The Mind Institute [16], Clark et al. [5] and Kiehl [11] for providing us access to their fMRI datasets. This project was partially funded through grant DA012852 from the NIDA, NIH, from NIMH grant number 1R01MH076282-01 as part of the NSF/NIH Collaborative Research in Computational Neuroscience Program and The MIND Institute, DOE Grant DE-FG02-99ER62764.

References

[1] Anderson, C., Domingos, P. and Weld, D. Relational Markov models and their application to adaptive web navigation. *KDD*, 143-152, 2002.

[2] Burge, J., Lane, T. Learning Class-Discriminative Dynamic Bayesian Networks. *ICML*, 22:97-104, 2005.

[3] Buckner, R. L., Snyder, A., Sanders, A., Marcus, R., Morris, J. Functional Brain Imaging of Young, Nondemented, and Demented Older Adults. *Journal of Cognitive Neuroscience*, 12, 2. 24-34, 2000.

[4] Chickering, D., Geiger, D., Heckerman, D. Learning Bayesian Networks is NP-Hard. Technical Report MSR-TR-94-17, Microsoft, 1994.

[5] Clark, V.P., Friedman, L., Manoach, D., Ho, B.C., Lim, K., Andreasen, N. A collaborative fMRI study of the novelty oddball task in schizophrenia: Effects of illness duration. *Society for Neuroscience Abstracts*, 474.9, 2005.

[6] Dartmouth fMRI Data Center, The. http://www.fmridc.org/f/fmridc , 2006.

[7] Fine, S., Singer, Y., Tishby, N. The Hierarchical Hidden Markov Model: Analysis and Applications. *Machine Learning*. vol. 32, 41-62, 1998.

[8] Grossman, D., Domingos, P. Learning Bayesian Network Classifiers by Maximizing Conditional Likelihood. *ICML*, 21, 361-368, 2004.

[9] Gyftodimos, E., Flach, P. Hierarchical Bayesian Networks: An Approach to Classification and Learning for Structured Data. *Proceedings of Methods and Applications of Artificial Intelligence, Third Hellenic Conference in AI.* 291-300, 2004.

[10] Heckerman, D., Geiger, D., Chickering, D.M. Learning Bayesian Networks: the Combination of Knowledge and Statistical Data. *Machine Learning*, 20, 197-243, 1995.

[11] Kiehl, K. An event-related functional magnetic resonance imaging study of an auditory oddball task in schizophrenia. *Schizophrenia Research*, 48:159-171, 2001.

[12] Lancaster J.L., Woldorff M.G., Parsons L.M., Liotti M., Freitas C.S., Rainey L., Kochunov PV, Nickerson D., Mikiten S.A., Fox P.T. Automated Talairach Atlas labels for functional brain mapping. *Human Brain Mapping* 10,120-131, 2000.

[13] Lam, W., Bacchus, F. Learning Bayesian belief networks: An approach based on the MDL principle. *Computational Intelligence*, 10:269-293, 1994.

[14] Margaritis, D., and Thrun S. Bayesian Network Induction via Local Neighborhoods. *Advances in Neural Information Processing Systems 12*, Denver, CO, 1999.

[15] McCallum, A., Rosenfeld, R., Mitchell, T. and Ng, A. Y. Improving Text Classification by Shrinkage in a Hierarchy of Classes. *ICML*, 1998.

[16] MIND Institute, The. http://www.themindinstitute.org/ , 2006.

[17] Pearl, J. Fusion, Propagation and Structuring in Belief Networks. *AI*, 29, 3, 241-288, 1986.

[18] Schwarz, G. Estimating the dimension of a model. *Annals of Stats.*, 6, 461-464, 1978.

[19] Sutton, R. S., Precup, D., and Singh, S. Between MDPs and semi-MDPs: A Framework for Temporal Abstraction in Reinforcement Learning. *AI*, v. 112. pg. 181-211, 1999.