Bounding the False Discovery Rate in Local Bayesian Network Learning

Ioannis Tsamardinos
Dept. of Computer Science, Univ. of Crete, Greece
BMI, ICS, Foundation for Research and Technology, Hellas
Dept. of Biomed. Inf., Vanderbilt Univ., USA
tsamard@ics.forth.gr

Laura E. Brown
Dept. of Biomed. Inf., Vanderbilt Univ., USA
laura.e.brown@vanderbilt.edu

Abstract

Modern Bayesian Network learning algorithms are time-efficient, scalable and produce high-quality models; these algorithms feature prominently in decision support model development, variable selection, and causal discovery. The quality of the models, however, has often only been empirically evaluated; the available theoretical results typically guarantee asymptotic correctness (consistency) of the algorithms. This paper describes theoretical bounds on the quality of a fundamental Bayesian Network local-learning task in the finite sample using theories for controlling the False Discovery Rate. The behavior of the derived bounds is investigated across various problem and algorithm parameters. Empirical results support the theory which has immediate ramifications in the design of new algorithms for Bayesian Network learning, variable selection and causal discovery.

Introduction

State-of-the-art BN-learning algorithms can reconstruct the complete network representing the data distribution in problems with thousands of variables. Their empirical learning quality has been determined by extensive experimentation (Tsamardinos, Brown, and Aliferis 2006). For some of the algorithms, theoretical guarantees of their asymptotic behavior have been proven, e.g., they will converge to the correct network in the sample limit. Unfortunately, there are limited theoretical guarantees for their finite sample behavior. Bootstrapping methods are possible but extremely computationally consuming (see Related Work section for more details). As a result, the practitioner who uses the algorithms is only provided with a point estimate (i.e. a single network) with no indication of how close it is to the true one (other than perhaps its log-likelihood). *The inability of algorithms to accompany their output with some measure of confidence has arguably been a significant deterrent from using BN learning algorithms in many classical analysis settings.*

One of the core problems in BN learning is that of identifying the neighbors of a target variable T in the (unknown) network that represents the data distribution (local learning). We denote the neighbors of T by $\mathbf{N}_T$. If one is able to identify the $\mathbf{N}_T$ sets efficiently and accurately, then one could identify all edges in a Bayesian Network by estimating the neighbors for all variables. Several BN-learning algorithms first identify and then piece together these neighbor sets, before proceeding with orienting the edges of the network for the final result. Other BN-learning algorithms use the neighbor sets to selectively reconstruct parts of the network of particular interest, if time does not allow complete reconstruction (Tsamardinos et al. 2003; Peña, Björkegren, and Tegnér 2005). Two state-of-the-art algorithms for identifying $\mathbf{N}_T$ are MMPC and HITON-PC (Tsamardinos, Brown, and Aliferis 2006; Aliferis, Tsamardinos, and Statnikov 2003).

The set $\mathbf{N}_T$ has the following important property: any variable not in $\mathbf{N}_T$ can be made probabilistically independent of T conditioned on some variable subset of $\mathbf{N}_T$. In other words, for any *other* variable V, there is a variable context in which V carries no information for T. Intuitively then, $\mathbf{N}_T$ is an important set for the prediction of T. It has been shown (Tsamardinos and Aliferis 2003) that $\mathbf{N}_T$ is part of the set of the *strongly-relevant* variables as defined by Kohavi and John and the Markov Blanket of T, i.e., the minimum variable set required for optimal prediction of T. Successful variable selection algorithms for prediction such as HITON (Aliferis, Tsamardinos, and Statnikov 2003) depend on first identifying $\mathbf{N}_T$ in subroutines.

Finally, under some general assumptions, the $\mathbf{N}_T$ has a *causal* interpretation: the set of direct causes and effects of T (Spirtes, Glymour, and Scheines 2000), where direct means that no other variable measured in the data causally intervenes between a variable in $\mathbf{N}_T$ and the target T.

In this paper, we provide theoretical bounds on the quality of neighbor identification, $\mathbf{N}_T$, from finite sample and for a broad family of algorithms. Specifically, we provide a bound on the expected proportion of false positives in the estimated $\hat{\mathbf{N}}_T$. The bound is based on theories of controlling the False Discovery Rate (Benjamini and Hochberg 1995). The basic idea of the method is threefold, to: (i) express a BN-learning task as a multiple testing problem, (ii) approximate or bound the p-value of a complex hypothesis by the p-values of primitive tests independence, and (iii) use a statistical method for bounding the multiple testing error. We accompany the theory with corroborating empirical results and we investigate the behavior of the derived bounds in terms of various problem and algorithm parameters.

Background

We denote random variables as V_i except for the special variable of interest that is denoted by T. Quantities related to a variable V_i or T use i or T as an index, e.g., $\mathbf{EN}_i$. We denote sets of variables by upper-case, bold-face letters. We use calligraphic fonts for special sets of variables such as the set of all variables considered $\mathcal{V}$. We denote *the independence of two random variables* X and Y *conditioned* on a set $\mathbf{Z}$ in the unknown data distribution P as $I(X;Y|\mathbf{Z})$ and the dependence as $\neg I(X;Y|\mathbf{Z})$.

Definition 1. Let P be a joint probability distribution of the random variables (interchangeably called nodes) in some set $\mathcal{V}$ and $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ be a Directed Acyclic Graph (DAG). We call $\langle \mathcal{G}, P \rangle$ a *Bayesian network* if $\langle \mathcal{G}, P \rangle$ satisfies the Markov Condition: every variable is independent of any subset of its non-descendant variables conditioned on its parents (Spirtes, Glymour, and Scheines 2000).

The graph of a network in conjunction with the Markov Condition directly encodes some of the independencies of the probability distribution and entails others. The faithfulness condition below, asserts that the conditional independencies observed in the distribution of a network are not accidental properties of the distribution, but instead due to the structure of the network.

Definition 2. If all and only the conditional independencies true in the distribution P are entailed by the Markov condition applied to $\mathcal{G}$, we will say that $\langle \mathcal{G}, P \rangle$ is faithful, and $\mathcal{G}$ is faithful to P (Spirtes, Glymour, and Scheines 2000). A distribution P is *faithful* if there exists a graph, $\mathcal{G}$, to which it is faithful.

Notice that, there are distributions P for which there is no faithful Bayesian network $\langle \mathcal{G}, P \rangle$ (however, these distributions are "rare"; see (Meek 1995) for details). Also, there may be more than one graph faithful to the same distribution P. We define the set of neighbors of T in a graph $\mathcal{G}$ as the set of variables sharing an edge with T and denote it with $\mathbf{N}_T^{\mathcal{G}}$. The following theorem (Spirtes, Glymour, and Scheines 2000) is arguably the cornerstone of learning BNs from data using tests of conditional independence:

Theorem 1. *In a faithful BN* $\langle \mathcal{G}, P \rangle$ *on variables* $\mathcal{V}$*:* $V_i \notin \mathbf{N}_T^{\mathcal{G}} \Leftrightarrow \exists \mathbf{Z}_k \subseteq \mathcal{V} \setminus \{V_i, T\}$*, s.t.* $I(V_i; T|\mathbf{Z_k})$

From the above theorem, it is easy to see that if $\langle \mathcal{G}, P \rangle$ and $\langle \mathcal{G}', P \rangle$ are two faithful Bayesian networks (to the same distribution), then for any variable T, it is the case that $\mathbf{N}_T^{\mathcal{G}} = \mathbf{N}_T^{\mathcal{G}'}$. Thus, the set of neighbors of T is unique among all Bayesian networks faithful to the same distribution and so we will drop the superscript and denote it simply as $\mathbf{N}_T$.

Single Tests of Independence

In the context of this paper, single hypothesis testing works as follows. The null hypothesis of each test of independence, denoted as $H_{i,k}$ is that V_i is independent of T conditioned on $\mathbf{Z}_k$, i.e., "$H_{i,k} : I(V_i; T|\mathbf{Z}_k)$". Each such test, denoted as $T(V_i; T|\mathbf{Z}_k)$, computes a test statistic; for several algorithms working on discrete i.i.d. data this is the G^2 statistic; see (Tsamardinos, Brown, and Aliferis 2006) for details. We will denote with $G^2_{i,k}$ the statistic calculated during a test of the form $T(V_i; T|\mathbf{Z}_k)$ (for some indexing of the subsets of variables $\mathbf{Z}_k$). $G^2_{i,k}$ is a random variable since it depends on the random sample. We denote the observed value of $G^2_{i,k}$ in the given sample with $g_{i,k}$. For a test $T(V_i; T|\mathbf{Z}_k)$ the corresponding p-value $p_{i,k}$ of an observed statistic $g_{i,k}$ is defined as:

$$p_{i,k} = P(G^2_{i,k} \geq g_{i,k} | H_{i,k})$$

Intuitively, the p-value is the probability of observing a test statistic as extreme or more extreme than the one observed in the given data under the null hypothesis. If this probability is lower than a specified significance level α, we reject the null hypothesis $I(V_i; T|\mathbf{Z}_k)$ and accept the alternative $\neg I(V_i; T|\mathbf{Z}_k)$. Rejecting the null hypothesis when it is true is called a Type I error or a *False Discovery*. Accepting the null hypothesis when the alternative is true is called a Type II error or a false negative. Thus, in the above setting the probability of Type I error is less than α; we say that the Type I error rate is *controlled* at the α level. To calculate a p-value we need to know the distribution of the test statistic. The $G^2_{i,k}$ *when the corresponding independence holds* is distributed with a χ^2 distribution of certain degrees of freedom that depend on the dimensionality of T, V_i and $\mathbf{Z}_k$.

When the null $H_{i,k}$ cannot be rejected, this is either due to the alternative (conditional dependence) holding, or insufficient statistical power. Most algorithms do not perform the test $T(V_i; T|\mathbf{Z}_k)$ unless they are confident about the power of the test. In that case, default rules determine whether to accept or reject the null without direct use of the data.

Multiple Tests of Independence

When testing multiple hypotheses, defining and measuring the overall error rate becomes more complicated. An approach of increasing importance was proposed by Benjamini and Hochberg (1995) and is based on what is called the False Discovery Rate (FDR). *FDR is the expected proportion of false positive findings among all the rejected hypotheses.* Benjamini and Hochberg (1995) provided a sequential p-value method to control the FDR. Their method works as follows: consider testing $H_1, \ldots, H_m$ with p-values in the set $\mathcal{P} = \{p_1, \ldots, p_m\}$. Let $p_{(1)} \leq \ldots \leq p_{(m)}$ be the ordered p-values and A the desired level for controlling the FDR. Define

$$F(\mathcal{P}, A) = \max\{p_{(i)} \in \mathcal{P}, \text{ s.t. } p_{(i)} \leq \frac{i}{m} A\}$$

Under certain broad assumptions, Benjamini and Hochberg (1995) show that rejecting all hypotheses H_i, s.t., $p_i \leq F(\mathcal{P}, A)$ guarantees that the FDR $\leq A$. We can invert the procedure so that given a threshold t to reject the hypotheses, it returns the minimum FDR level A guaranteed. We denote this procedure with $F^{-1}(\mathcal{P}, t)$. It is easy to see that for $p_{(i)}$ the largest p-value less or equal to t, $F^{-1}(\mathcal{P}, t) = p_{(i)} \frac{m}{i}$.

Discovering the Set of Neighbors $\mathbf{N}_T$

Typical algorithms that identify the $\mathbf{N}_T$ reject the independence (null) $H_{i,k} : I(V_i; T|\mathbf{Z}_k)$, if and only if $p_{i,k} \leq \alpha$. For

each node $V_i \in \mathcal{V} \setminus \{T\}$ they search for a *certificate of exclusion* from $\mathbf{N}_T$, i.e., a subset $\mathbf{Z}_k$ such that the independence $I(V_i; T|\mathbf{Z}_k)$ holds. If one such subset is found, then according to Theorem 1 above, $V_i \notin \mathbf{N}_T$; otherwise, V_i is assumed to belong in $\mathbf{N}_T$. The estimated set of neighbors $\hat{\mathbf{N}}_T$ is then returned. If all subsets $\mathbf{Z} \subseteq \mathcal{V} \setminus \{V_i, T\}$ are tested for each node, then by Theorem 1 an algorithm should return the correct $\mathbf{N}_T$ assuming $H_{i,k} \Leftrightarrow p_{i,k} > \alpha$ for all i, k.

An algorithm that considers all subsets $\mathbf{Z} \subseteq \mathcal{V} \setminus \{V_i, T\}$ requires performing an exponential number of tests for each variable V_i. This implies that the procedure is prohibitively computationally expensive on anything but very small problems. In addition, when sample is finite, the more tests one performs the higher the probability a $p_{i,k}$ value will obtain a large value even when the alternative hypothesis holds; in turn this increases the probability of false negatives and lowers the statistical power. Thus, it is highly desirable for both quality and time-efficiency purposes to minimize the number of tests required by such procedures. This is achieved by the following theorem (Aliferis et al. 2007) (see Supplemental Material for proof at http://www.dsl-lab.org):

Theorem 2. *Let $\mathbf{EN}_T, \mathbf{EN}_i$ be any subsets of variables such that $\mathbf{EN}_T \supseteq \mathbf{N}_T$ and $\mathbf{EN}_i \supseteq \mathbf{N}_i$. Then,*

$$V_i \notin \mathbf{N}_T \Leftrightarrow \exists \mathbf{Z}_k \in 2^{\mathbf{EN}_T \setminus \{V_i, T\}} \cup 2^{\mathbf{EN}_i \setminus \{V_i, T\}}, \text{ s.t. } I(V_i; T|\mathbf{Z_k})$$

The theorem dictates that we only need to search for certificates of exclusion within supersets of $\mathbf{N}_T$ and $\mathbf{N}_i$; we call these supersets the *extended neighbors* (EN) sets. Obviously, the smaller the sets $\mathbf{EN}_T$ and $\mathbf{EN}_i$, the fewer tests are required to use the above theorem and provide a final estimated $\hat{\mathbf{N}}_T$. In the worst case, $\mathbf{EN}_T = \mathcal{V}$ and there are no savings relative to using Theorem 1. Typical and state-of-the-art algorithms employ the theorem to return an estimated neighbors' set as shown in the general algorithmic template of Algorithm 1 (Aliferis et al. 2007).

Algorithms following the template may differ in the heuristics of initializing $\widehat{\mathbf{EN}}_T$ (Line 2), the way they alternate between the steps in the loop, the way they search for a certificate of exclusion (Line 4), and the way they select the next variable to insert to $\widehat{\mathbf{EN}}_T$ (Line 6). To see that such procedures are correct (in the sample limit where $p_{i,k} > \alpha \Leftrightarrow I(V_i; T|\mathbf{Z}_k)$), notice that in $AlgEN_T$, if $V_i \in \mathbf{N}_T$, it will enter $\widehat{\mathbf{EN}}_T$ and never be removed, since by Theorem 1 there exist no certificate of exclusion. Thus, for the returned set it holds that $\widehat{\mathbf{EN}}_T \supseteq \mathbf{N_T}$. Also, if $V_i \in \widehat{\mathbf{EN}}_T$, there is no certificate of exclusion within $\widehat{\mathbf{EN}}_T$ (Line 4). If also $T \in \widehat{\mathbf{EN}}_i$, there is no subset for which $I(V_i; T|\mathbf{Z}_k)$ for all $\mathbf{Z}_k \in \widehat{\mathbf{EN}}_i$. Thus, there is no certificate for exclusion for V_i in $\widehat{\mathbf{EN}}_T$ nor in $\widehat{\mathbf{EN}}_i$. By Theorem 2, V_i should belong in $\mathbf{N}_T$ and so it is retained in $\hat{\mathbf{N}}_T$, otherwise it is removed (Line 15).

Algorithm 1 Identify $\mathbf{N}_T$

1: **procedure** $AlgEN_T(T, \mathcal{D}, \alpha)$
Input: target variable T; data $\mathcal{D}$, significance level α.
2: Initialize $\widehat{\mathbf{EN}}_T \subseteq \mathcal{V} \setminus \{T\}$
3: **repeat** % Arbitrarily alternate steps (a) & (b)
(a) % Search for a certificate of exclusion
4: **if** $\exists V_i \in \widehat{\mathbf{EN}}_T, \mathbf{Z}_k \subseteq \widehat{\mathbf{EN}}_T$, s.t., $p_{i,k} > \alpha$
5: $\widehat{\mathbf{EN}}_T \leftarrow \widehat{\mathbf{EN}}_T \setminus \{V_i\}$ % Remove V_i
(b) % Insert one more variable into $\widehat{\mathbf{EN}}_T$
6: $\widehat{\mathbf{EN}}_T \leftarrow \widehat{\mathbf{EN}}_T \cup \{V_i\}$, where V_i has never entered $\widehat{\mathbf{EN}}_T$ before.
7: **until** there is no change in $\widehat{\mathbf{EN}}_T$ and all variables have entered $\widehat{\mathbf{EN}}_T$ at least once
8: **return** $\widehat{\mathbf{EN}}_T$
9: **end procedure**

10: **procedure** $AlgN_T(T, \mathcal{D}, \alpha)$
11: $\widehat{\mathbf{EN}}_T = AlgEN_T(T, \mathcal{D}, \alpha)$
12: $\hat{\mathbf{N}}_T = \widehat{\mathbf{EN}}_T$
13: **for** $V_i \in \hat{\mathbf{N}}_T$
14: $\widehat{\mathbf{EN}}_i = AlgEN_T(V_i, \mathcal{D}, \alpha)$
15: **if** $T \notin \widehat{\mathbf{EN}}_i$ **then** remove V_i from $\hat{\mathbf{N}}_T$
16: **return** $\hat{\mathbf{N}}_T$
17: **end procedure**

Bounding the FDR of Identifying the $\mathbf{N}_T$

In this section, we provide procedures for bounding the expected FDR of any algorithm $AlgN_T$ following the template above. All constraint-based algorithms for identifying the neighbor set, that we are aware of, belong in this class.

A False Discovery (of a neighbor) occurs when the hypothesis "$H_i : V_i \notin \mathbf{N}_T$" holds, but it is falsely rejected. This is a complex hypothesis, but it can be reduced to a logical statement involving only primitive hypotheses of independence "$H_{i,k} : I(V_i; T|\mathbf{Z_k})$": by Theorem 2 "$H_i : \exists \mathbf{Z}_k \in 2^{\mathbf{EN}_T \setminus \{V_i, T\}} \cup 2^{\mathbf{EN}_i \setminus \{V_i, T\}}$, s.t. $I(V_i; T|\mathbf{Z_k})$". Were we able to calculate the p-values of such hypotheses, we could then directly pass them to an FDR procedure. Unfortunately, this calculation is a difficult task but the next theorem bounds these p-values based on the p-values $p_{i,k}$ of the primitive hypotheses:

Theorem 3. *Consider the hypothesis "$H_i : \exists \mathbf{Z}_k \in \mathcal{S}$, such that $I(V_i; T|\mathbf{Z_k})$", where $\mathcal{S}$ is a set of subsets of variables of $\mathcal{V}$. Then, for the corresponding p-value p_i it holds that: $p_i \leq p_i^*$, where $p_i^* = max\{p_{i,k}, \text{ s.t. } \mathbf{Z}_k \in \mathcal{S}\}$.*

Proof. If we assume the null hypothesis H_i holds, there exist at least one $\mathbf{Z}_k \in \mathcal{S}$ such that the independence $I(V_i; T|\mathbf{Z}_k)$ holds. Then, for any such k we get:

$$p_i = P(G^2_{i,1} \geq g_{i,1}, \ldots, G^2_{i,n} \geq g_{i,n}|H_i) \quad (1)$$
$$\leq P(G^2_{i,k} \geq g_{i,k}|H_i) \quad (2)$$
$$= P(G^2_{i,k} \geq g_{i,k}|I(V_i; T|\mathbf{Z}_k)) \quad (3)$$
$$= p_{i,k} \quad (4)$$

or equivalently,

$$p_i \leq p_{i,k}, \text{ for any } k \text{ s.t. } I(V_i; T|\mathbf{Z}_k)$$

Algorithm 2 Find an FDR bound

1: **procedure** B-FDR(T, $\mathcal{D}$, $AlgN_T$, α)
Input: target variable T; data $\mathcal{D}$, method to find neighbors of T, $AlgN_T$, significance level α.
2: $\hat{\mathbf{N}}_T = AlgN_T(T, \mathcal{D}, \alpha)$
3: **for** each variable $V_i \in \hat{\mathbf{N}}_T$
4: $\widehat{\mathbf{EN}}_i = AlgEN_T(V_i, \mathcal{D}, \alpha)$
5: $p_i^* = \max p_{i,k}$, for all $\mathbf{Z}_k \in 2^{\hat{\mathbf{N}}_T \setminus \{V_i, T\}} \cup 2^{\widehat{\mathbf{EN}}_i \setminus \{V_i, T\}}$
6: FDR-bound = $F^{-1}(\{p_i^* : V_i \in \hat{\mathbf{N}}_T\}, \alpha)$
7: **return** FDR-bound
8: **end procedure**

Thus, we could bound p_i by the *minimum* of $p_{i,k}$ obtained when conditioning on a subset $\mathbf{Z}_k$ for which the independence $I(V_i, T|\mathbf{Z}_k)$ holds. Obviously however, we do not know for which subsets the independence holds and we are forced instead to use as a looser and more conservative bound the maximum $p_{i,k}$ over all possible subsets: $p_i \leq p_i^*, p_i^* = \max_k p_{i,k}$ $\square$

Based on this, we could use the bounds p_i^* instead of the actual unknown p-values and form the set $\mathcal{P} = \{p_i^*\}$. Subsequently, we can provide a bound on the FDR based on the p_i^*'s. Since, the latter ones are conservative, the FDR bound should also be looser than had we used the actual p-values. Algorithm 2 returns an FDR bound on the output of any algorithm $AlgN_T$ following the template in Algorithm 1.

A few comments on the above procedures. $AlgN_T$ controls the false discovery rate at the significance level α of each primitive hypothesis of independence. It *does not control* the FDR of the complex hypotheses $H_i : V_i \notin \mathbf{N}_T$. B-FDR on the other hand, provides a bound on the FDR of the complex discoveries H_i's.

The algorithm B-FDR only calculates the p-value bounds p_i^* for the variables in $\hat{\mathbf{N}}_T$ and not for all variables in $\mathcal{V}$. The call to F^{-1} (Line 6) uses only these p-values. This is justified as follows: Recall that $F^{-1}(\{p_i^* : V_i \in \hat{\mathbf{N}}_T\}, \alpha) = p_{(i)}^* \frac{m}{i}$, for the largest i s.t. $p_{(i)}^* \leq \alpha$. Since, for all $V_i \notin \hat{\mathbf{N}}_T$, $p_i^* > \alpha$, there is no need to compute and include these p-value bounds in the call of the F^{-1} procedure because we would obtain the same answer.

The basic assumptions of the algorithms are that the data distribution is faithful, the tests of independence return a p-value, and the FDR procedure used is correct. There are no assumptions regarding the shape of the distribution such as normality.

Another important assumption is that there is enough statistical power at the α level, so that when $V_i \in \mathbf{N}_T$, $p_{i,k} < \alpha$ and so V_i will not be removed from $\widehat{\mathbf{EN}}_T$. When this is true we obtain that $\widehat{\mathbf{EN}}_T \supseteq \mathbf{N}_T$ and $\widehat{\mathbf{EN}}_i \supseteq \mathbf{N}_i$ as Theorem 2 requires. In other words, the false discoveries (Type I error) of the complex hypotheses "$H_i : V_i \notin \mathbf{N}_T$", depend *both* on the Type I error and on the Type II error of the underlying primitive hypotheses "$H_{i,k} : I(V_i; T|\mathbf{Z}_k)$". Algorithm B-FDR guarantees the FDR only by assuming the Type II error rate at level α is zero.

As already mentioned, to ensure enough power, algorithms $AlgN_T$ typically do not perform the test $T(V_i; T|\mathbf{Z}_k)$ unless there is enough sample per degrees of freedom. Thus, another assumption required for the bounds to be accurate is that when $V_i \notin \mathbf{N}_T$ the independence $I(V_i; T|\mathbf{Z}_k)$ holds for a test that is actually going to be performed. For large enough sample and sparse networks this assumption is typically true.

In terms of the time-complexity of the algorithm, notice that all required $p_{i,k}$ values are computed during the execution of the algorithms $AlgN_T$ and $AlgEN_T$. Thus, with smart caching of these values computing the maximums p_i^* requires *no extra cost*, while computing F^{-1} requires in the worst case time linear to the number of variables. The time complexity of $AlgN_T$ provided some smart optimizations are in place is $O(|\mathcal{V}| \cdot |\mathbf{EN}|^{l+1})$, where l is the maximum size allowed for any conditioning set $\mathbf{Z}_k$ and $\mathbf{EN}$ the maximum-size set of extended neighbors computed (see (Tsamardinos, Brown, and Aliferis 2006) for details).

Experimental Results

For the experiments that follow we have selected the networks Alarm, Child, Insurance, Gene, and Pigs (37, 20, 27, 801, and 441 variables respectively) from real decision support systems. We have sampled 5 datasets from the distribution of each network and run B-FDR on all datasets targeting every variable in the first three networks and 37 nodes randomly chosen from the latter two for a total of 790 runs: 158 variables $\times$ 5 samplings. The underlying instantiation of $AlgN_T$ is the algorithm $MMPC$ (Tsamardinos, Brown, and Aliferis 2006).

Figure 1 plots the *average* true FDR found for a variable (over all samplings), i.e. the percentage of false positives in $\hat{\mathbf{N}}_T$, versus the *average* FDR *bound* returned by B-FDR (158 points per graph). We expect the bound to be larger than the observed FDR and so all points in the graphs to fall below the diagonal, minus some statistical fluctuations. Ideally, we would also desire the bound to be tight and for each point to fall exactly onto the diagonal. Theoretically, this is not expected to be the case, since we are unable to compute the exact p-values of the complex hypotheses.

In the first set of experiments (first row, Figures 1(a)-(c)) the parameter α is set to 0.05, while the sample size ranges in the set $\{10000, 5000, 1000\}$. In the second set of experiments (second row, Figures 1(d)-(f) and Figure 1(b)) we have fixed the sample size to 5000 and varied the α parameter within the set $\{0.01, 0.05, 0.10, 0.15\}$.

Focusing on the first row, we see that in Figures 1(a) and 1(b), the FDR bound is indeed accurate and most points fall below or close to the diagonal. Certain small deviations are expected due to statistical fluctuations, since the theoretical bounds regard the average (expected) behavior. For sample size 1000 however, Figure 1(c) we see that the FDR bound calculated fails on many cases. In close inspection of these results we discovered that for several variables the algorithm would not perform all necessary tests to remove variables

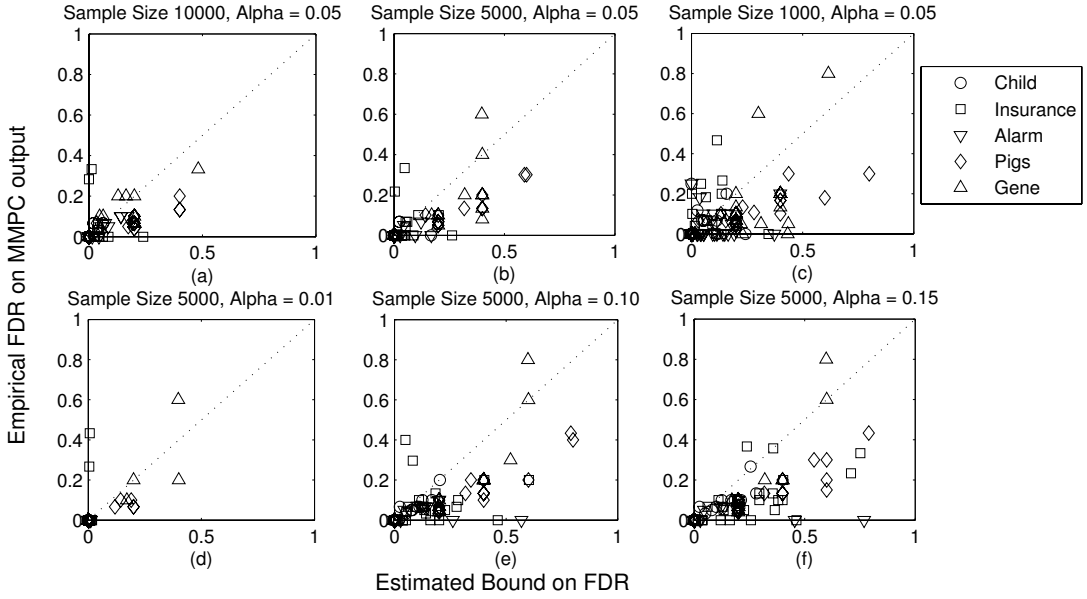

Figure 1: The average true FDR vs. the average computed FDR bound returned by B-FDR for 158 variables from real BNs. Minus statistical fluctuations, points should fall below the diagonal. In Figures (a)-(c) sample size varies. The assumptions of the method are violated for sample size 1000 with a direct, visual degradation of performance. When the assumptions are restored, the bound's accuracy improves (Figure 2). In Figures (d)-(f) and (b) the α-threshold values of the underlying tests of independence vary while sample size is held constant at 5000. The results are summarized numerically in Table 1.

from the neighbor set. This is due to $MMPC$ determining that the sample is insufficient relative to the overall degrees of freedom of the test to obtain a reasonable statistical power. Not performing all tests violates the assumptions of the B-FDR algorithm. In such cases, the theory still holds for a looser definition of a false discovery: a false discovery regards a variable $V_i \in \hat{\mathbf{N}}_T$ that can be made independent of T given a subset $\mathbf{Z}$ of either $\widehat{\mathbf{EN}}_T$ or $\widehat{\mathbf{EN}}_i$ for which the test $T(V_i; T|\mathbf{Z}_k)$ can actually be performed. When we keep only the variables where all the required tests could be performed, the situation is improved as shown in Figure 2.

There are other minor violations of the assumptions of B-FDR. In all graphs shown, there are two nodes from the Insurance network that are systematically above the diagonal (compare for example Figures 1(a) and 1(b); these are the two points between 0.2 and 0.4 true FDR, and above the diagonal in both figures). For these two variables we determined the existence of determinism in the Insurance network and the violation of the faithfulness assumption.

The lower the value of α, the easier it becomes to accept $H_i : V_i \notin \mathbf{N}_T$ and so, as we observe in the figures the true FDR is lower for lower values of α. As α increases both the true FDR and its bound increase; the cloud of points is moving towards point (1,1) in the graph, while staying below the diagonal.

While useful, the graphs of Figure 1 may be somewhat misleading since several points fall onto each other, while outliers visually stick out. We now present two statistics that summarize and quantify the information in the graphs. Let $tFDR$ be the true FDR of a run of the B-FDR and $bFDR$ the FDR bound estimated. We define the average *error* of the bound as the average $\max(tFDR - bFDR, 0)$ and the average *slack* of the bound as the average $\max(bFDR - tFDR, 0)$ over all 790 runs. The error and the slack is the average underestimation and overestimation of the bound, respectively. The results are in Table 1.

Both the graphs and the statistics indicate that, provided the assumptions hold, the algorithm returns bounds that are accurate across different sample sizes and values of the α threshold. For example, within the scope of our experiments, for sample size 5000, and $\alpha = 0.05$ a practitioner should expect on average overestimating the true FDR by about 0.5% and underestimating it by 3.5%. Some obvious trends observed in the table are that the slack decreases with sample size and increases with α. In terms of computational complexity, the average time over all runs to compute $\hat{\mathbf{N}}_T$ and its bound is 33 seconds. The Pigs network was the most demanding; excluding it, the average time drops to 12sec.

Related Work

Most early work on estimating errors in BN learning is based on bootstrapping that is extremely computationally demanding. In Listgarten and Heckerman (2007), an algorithm for determining the FDR of edge detection in BNs is presented. The method is based on permutation testing which is more efficient than bootstrapping but still requires multiple learning rounds. In addition, the way permutation testing is performed requires one to learn the complete network (global

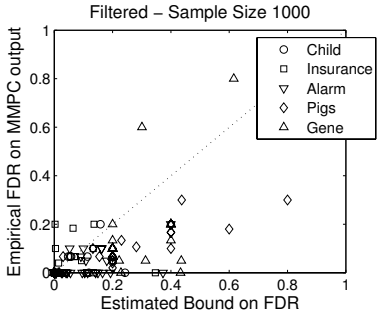

Figure 2: The average true FDR versus the average FDR bound for sample size 1000 when variables violating the assumptions are removed.

Table 1: Summary statistics. Number UE is the number of runs (out of 790 = 158 nodes $\times$ 5 samplings) where the bound in under-estimated.

SS	Alpha	Fig.	Num UE	Ave. Error	Ave. Slack
1000	0.05	1(c)	59	0.018	0.068
1000	0.05	2	26	0.009	0.068
5000	0.05	1(b)	18	0.005	0.034
10000	0.05	1(a)	22	0.006	0.025
5000	0.01	1(d)	12	0.006	0.005
5000	0.05	1(b)	18	0.005	0.034
5000	0.10	1(e)	20	0.006	0.064
5000	0.15	1(f)	16	0.004	0.087

learning) and the orientation of the edges, and is geared toward search-and-score learning algorithms. The permutation testing procedure presented in the above paper has some theoretical problems that lead to overestimation of the FDR. Other related recent work includes (Peña 2008). The algorithm in that paper learns a Markov Blanket of a variable T (a superset of $\mathbf{N}_T$), concerns only Gaussian Graphical Models, and finally, it requires conditioning on sets of size almost equal to the network area identified; the latter implies that the general sample requirements of the algorithm are exponential to the number of variables returned. Finally, interesting related work is presented in (Nilsson et al. 2007). The problem is to identify all multivariately differentially expressed genes while controlling FDR and it involves only unconditional statistical tests; nevertheless it employs similar statistical techniques. We intend to further explore the relations to the above algorithms in future work.

Discussion and Conclusions

We have presented an algorithm for bounding the False Discovery Rate of identifying the set of neighbors of a variable of interest T in any Bayesian Network faithful to a sample distribution. Preliminary empirical results corroborate the theoretical properties of the algorithm across a range of sample sizes and threshold values of the underlying independence test. One limitation empirically illustrated is that the method's accuracy degrades abruptly when the assumptions are violated; this is due to the hard-heuristic rules of $MMPC$ and all similar constraint-based algorithms that completely refuse to perform a test of independence unless it passes some ad-hoc criteria for determining sufficient statistical power. The idea of the method could possibly be extended to encompass other BN-learning algorithms and tasks; it could also be augmented with techniques that bound the false negative rate or a weighted average of Type I and Type II errors using more advanced and recent statistical theories. The algorithm presented has immediate applications in designing BN-learning algorithms, Markov-Blanket based (constraint-based) feature selection methods for prediction, and causal discovery methods that we intend to explore.

References

Aliferis, C. F.; Statnikov, A.; Tsamardinos, I.; Mani, S.; and Koutsoukas, X. D. 2007. Local causal and markov blanket induction algorithms for causal discovery and feature selection for classification. Technical Report DSL TR-07-02, DBMI, Vanderbilt Univ.

Aliferis, C. F.; Tsamardinos, I.; and Statnikov, A. 2003. HITON, a novel markov blanket algorithm for optimal variable selection. In *AMIA*, 21–25.

Benjamini, Y., and Hochberg, Y. 1995. Controlling the false discovery rate - a practical and powerful approach to multiple testing. *J Roy Stat Soc B Methods* 57(1):289–300.

Listgarten, J., and Heckerman, D. 2007. Determining the number of non-spurious arcs in a learned dag model: Investigation of a bayesian and a frequentist approach. In *23rd Conference on Uncertainty in Artificial Intelligence*.

Meek, C. 1995. Strong completeness and faithfulnes in bayesian networks. In *Conference on Uncertainty in Artificial Intelligence*, 411–418.

Nilsson, R.; Peña, J.; Björkegren, J.; and Tegnér, J. 2007. Detecting multivariate differentially expressed genes. *BMC Bioinformatics* 8(150).

Peña, J.; Björkegren, J.; and Tegnér, J. 2005. Growing bayesian network models of gene networks from seed genes. *Bioinformatics* 21, Suppl 2:ii224–ii229.

Peña, J. 2008. Learning gaussian graphical models of gene networks with false discovery rate control. In *6th European Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics*.

Spirtes, P.; Glymour, C.; and Scheines, R. 2000. *Causation, Prediction, and Search*. MIT Press, 2nd edition.

Tsamardinos, I., and Aliferis, C. F. 2003. Towards principled feature selection: Relevancy, filters and wrappers. In *AI&Stats 2003*.

Tsamardinos, I.; Aliferis, C. F.; Statnikov, A.; and Brown, L. E. 2003. Scaling-up bayesian network learning to thousands of variables using local learning techniques. Tech. Report DSL 03-02, 2003, DBMI, Vanderbilt Un.

Tsamardinos, I.; Brown, L.; and Aliferis, C. 2006. The Max-Min Hill-Climbing bayesian network structure learning algorithm. *Machine Learning* 65(1):31–78.