# Algorithms for Large Scale Markov Blanket Discovery

**Ioannis Tsamardinos, Constantin F. Aliferis, Alexander Statnikov**

Department of Biomedical Informatics, Vanderbilt University
2209 Garland Ave, Nashville, TN 37232-8340
{ioannis.tsamardinos, constantin.aliferis, alexander.statnikov@vanderbilt.edu}

### Abstract

This paper presents a number of new algorithms for discovering the Markov Blanket of a target variable T from training data. The Markov Blanket can be used for variable selection for classification, for causal discovery, and for Bayesian Network learning. We introduce a low-order polynomial algorithm and several variants that soundly induce the Markov Blanket under certain broad conditions in datasets with thousands of variables and compare them to other state-of-the-art local and global methods with excellent results.

## Introduction

The Markov Blanket of a variable of interest *T*, denoted as *MB(T)*, is a minimal set of variables conditioned on which all other variables are probabilistically independent of the target *T*. Given this property, knowledge of only the values of the *MB(T)* is enough to determine the probability distribution of *T* and the values of all other variables become superfluous. Therefore, the variables in the *MB(T)* are adequate for optimal classification. The strong connection between *MB(T)* and optimal, principled variable selection has been explored in (Tsamardinos and Aliferis 2003).

In addition, under certain conditions (faithfulness to a Bayesian Network; see next section) the *MB(T)* is identical to the direct causes, direct effects, and direct effects of direct causes of T and thus it can be used for causal discovery, e.g., to reduce the number of variables an experimentalist has to consider in order to discover the direct causes of *T*.

Finally, Markov Blanket discovery algorithms can be used to guide Bayesian Network learning algorithms: the *MB(T)* for all *T* are identified as a first step, and then used to guide the construction of the Bayesian Network of the domain; this is the approach taken in (Margaritis and Thrun 1999). Indeed, given the potential uses and significance of the concept of the Markov Blanket "It is surprising … how little attention it has attracted in the context of Bayesian net structure learning for all its being a fundamental property of a Bayesian net" (Margaritis and Thrun 1999).

In this paper we present novel algorithms that soundly induce the *MB(T)* from data and scale-up to thousands of variables. We compare the new algorithms with other state-of-the-art methods for inducting the *MB(T)* with excellent results. The novel algorithms are particularly suited for the cases where the available sample size is enough to perform conditional independence tests condition on the full *MB(T)*.

## Background

Bayesian Networks (BN) (Neapolitan 1990) are mathematical objects that compactly represent a joint probability distribution *J* using a graph *G* annotated with conditional probabilities; *J* and *G* are connected by the Markov Condition property: a node is conditionally independent of its non-descendants, given its parents. The *MB(T)* probabilistically shields *T* from the rest of the variables and graphically it corresponds to a neighborhood of *T* in the BN graph. We will denote the conditional independence of *X*, and *T* given $\mathbf{Z}$, as $\mathbf{I}(X\,;\,T \mid Z) \equiv \mathrm{P}(T \mid X, Z) = \mathrm{P}(T \mid Z)$.

**Definitions**: The Markov Blanket of a variable *T*, *MB(T)*, is a minimal set for which $\mathbf{I}(X\,;\,T \mid MB(T))$, for all $X \in$ V – {T} – *MB(T)* (Margaritis and Thrun 1999). A BN *C* is faithful to a joint probability distribution *J* over the variable set *V* if and only if every dependence entailed by the graph of *C* is also present in *J* (Spirtes *et al*. 2000). A BN *C* faithful if it is faithful to its corresponding distribution *J*.

The Markov Condition ensures that every conditional independence entailed by the graph *G* is also present in probability distribution *J*. Thus, together Faithfulness and the Markov Condition establish a close relationship between the graph *G* and some empirical or theoretical probability distribution *J*. In practical terms, what faithfulness implies is that we can associate statistical properties of the probability distribution *J* with properties of the graph *G* of the corresponding BN. It turns out that in faithful BNs, the set of parents, children, and spouses (i.e., parents of children of T) is the unique *MB(T)*. An example of the Markov Blanket concept is displayed in Figure 1: the *MB(T)* is the set of gray-filled nodes.

## The IAMB Algorithm and Variants

In this section several novel algorithms for discovering the *MB(T)* are presented that are sound under the following assumptions: (i) the data are generated by processes that can be faithfully represented by BNs, and (ii) there exist reliable statistical tests of conditional independence and measures of associations for the given variable distribution, sample size, and sampling of the data. We discuss the rationale and justification of the assumptions in the Discussion section.

**IAMB Description:** Incremental Association Markov Blanket (**IAMB**) (Figure 2) consists of two phases, a forward and a backward one. An estimate of the *MB(T)* is kept in the set *CMB*. In the forward phase all variables that belong in *MB(T)* and possibly more (false positives) enter *CMB* while in the backward phase the false positives are identified and removed so that *CMB* = *MB(T)* in the end. The heuristic used in **IAMB** to identify potential Markov Blanket members in phase I is the following: start with an empty candidate set for the *CMB* and admit into it (in the next iteration) the variable that maximizes a heuristic function *f(X ; T | CMB)*. Function *f* should return a non-zero value for every variable that is a member of the Markov Blanket for the algorithm to be sound, and typically it is a measure of association between *X* and *T* given *CMB*. In our experiments we used as *f* the Mutual Information similar to what suggested in (Margaritis and Thrun 1999, Cheng *et al*. 1998): *f(X ; T | CMB)* is the Mutual Information between *S* and *T* given *CMB*. It is important that *f* is an informative and effective heuristic so that the set of candidate variables after phase I is as small as possible for two reasons: one is time efficiency (i.e., do not spend time considering irrelevant variables) and another is sample efficiency (do not require sample larger than what is absolutely necessary to perform conditional tests of independence). In backward conditioning (Phase II) we remove one-by-one the features that do not belong to the *MB(T)* by testing whether a feature *X* from *CMB* is independent of *T* given the remaining *CMB*.

**IAMB Proof of Correctness (sketch):** If a feature belongs to *MB(T)*, then it will be admitted in the first step because it will be dependent on *T* given any subset of the feature set because of faithfulness and because the *MB(T)* is the minimal set with that property. If a feature is not a member of *MB(T)*, then conditioned on *MB(T)*, or any superset of *MB(T)*, it will be independent of *T* and thus will be removed from *CMB* in the second phase. Using this argument inductively we see that we will end up with the unique *MB(T)*.

**InterIAMBnPC Description:** The smaller the conditioning test given a finite sample of fixed size, the more accurate the statistical tests of independence and the measure of associations. The **InterIAMBnPC** algorithm uses two methods to reduce the size of the conditioning sets: (a) it interleaves the admission phase of **IAMB** (phase I) with the backward conditioning (phase II) attempting to keep the size of *MB(T)* as small as possible during all steps of the algorithm's execution. (b) it substitutes the backward conditioning phase as implemented in **IAMB** with the PC algorithm instead (Spirtes *et al*. 2000), a Bayesian Network learning algorithm that determines direct edges between variables in a more sample–efficient manner, and that is sound given the stated assumptions (see next section); thus, **interIAMBnPC** is expected to be more sample-efficient than **IAMB**. In addition, **interIAMBnPC** is still practical because PC is running only on small sets of variables, not the full set of variables.

**InterIAMBnPC Proof of Correctness (sketch):** All parents and children of *T* will enter *CMB* by the property of *f* mentioned above. Since **PC** is sound, it will never remove these variables. Since all effects enter *CMB*, conditioned on them, all the spouses (parents of children) of *T* will be dependent with *T* given *CMB* and enter *CMB* at some point. Again, because **PC** is sound, it will not permanently remove them (they may be removed temporarily but will enter *CMB* at a subsequent iteration; we do not elaborate due to space limitations), and they will be included in the final output.

Two other **IAMB** variants we experimented with are **interIAMB** and **IAMBnPC** which are similar to **interIAMBnPC** but they employ only either interleaving the first two phases or using PC for the backward phase, respectively. Even though **IAMB** provides theoretical guarantees only in the sample limit, the quality of the output and the approximation of the true *MB(T)* degrades gracefully in practical settings with finite sample (see experimental section). **IAMB** and its variants are expected to perform best in problems where the *MB(T)* is small relatively to the available data samples, but the domain may contain hundreds of thousands of variables.

**Time Complexity**: Typically, the performance of BN-induction algorithms based on tests of conditional independence is measured in the number of association calculations and conditional independence tests executed (both operations take similar computation effort and we will not distinguish between the two) (Spirtes *et al*. 2000, Cheng *et al*. 1998, Margaritis and Thrun 1999). Phase II performs O(|*CMB*|) conditional independence tests. Phase I performs *N* association computations for each variable that enters *CMB*, where *N* is the number of variables, and so the algorithm performs O(|*CMB*|×*N*) tests. In the worst case |*CMB*|=*N* giving an order of O($N^2$). In all experiments of **IAMB** we observed |*CMB*|=O(*MB(T)*) giving an average case order of O(*MB(T)* ×*N*) tests. For Mutual Information there exists an algorithm linear to the size of the data (Margaritis and Thrun 1999). The other **IAMB** variants have higher worst-case time complexity (since for example the PC is exponential to the number of variables) trading-off computation for higher performance. Nevertheless, since in our experiments we observed that the size of the *CMB* is relative small to the total number of variables, the additional time overhead of the variants versus the vanilla **IAMB** was minimal.

## Other Markov Blanket algorithms

To our knowledge, the only other algorithm developed explicitly for discovering the *MB(T)* and that scales-up is the Grow-Shrink (**GS**) algorithm (Margaritis and Thrun 1999). It is theoretically sound but uses a static and potentially inefficient heuristic in the first phase. **IAMB** enhances **GS** by employing a dynamic heuristic. The Koller-Sahami algorithm (**KS**) (Koller and Sahami 1996) is the first algorithm for feature selection to employ the concept of the Markov Blanket. **KS** is a heuristic algorithm and provides no theoretical guarantees.

The **GS** algorithm is structurally similar to **IAMB** and follows the same two-phase structure. However, there is one important difference: GS statically orders the variables when they are considered for inclusion in phase I, according to their strength of association with *T* given the empty set. It then admits into *CMB* the next variable in that ordering that is not conditionally independent from *T* given *CMB*. One problem with this heuristic is that when the *MB(T)* contains spouses of *T*. In that case, the spouses are typically associated with *T* very weakly given the empty set and are considered for inclusion in the *MB(T)* late in the first phase (associations between spouses and *T* are only through confounding /common ancestors variables, thus they are weaker than those ancestors' associations with *T*). In turn, this implies that more false positives will enter *CMB* at phase I and the conditional tests of independence will become unreliable much sooner than when using **IAMB**'s heuristic. In contrast, conditioned on the common children, spouses may have strong association with *T* and, when using **IAMB**'s heuristic, enter the *CMB* early. An analogous situation is in constraint satisfaction where dynamic heuristics typically outperform static ones. We provide evidence to support this hypothesis in the experiment section. We would also like to note that the proof of correctness of GS is indeed correct only if one assumes faithfulness, and not just the existence of a unique *MB(T)* as is stated in the paper: a non-faithful counter example is when *T* is the exclusive or of *X* and *Y* on which the GS will fail to discover the *MB(T)*, even though it is unique.

The **KS** algorithm (Koller, Sahami 1996) is the first one that employed the concept of the Markov Blanket for feature selection. The algorithm accepts two parameters: (i) the number *v* of variables to retain and (ii) a parameter *k* which is the maximum number of variables the algorithm is allowed to condition on. For *k*=0 **KS** is equivalent to univariately ordering the variables and selecting the first *v*. The Koller-Sahami paper does not explicitly claim to identify the *MB(T)*; however, if one could guess the size of the *MB(T)* and set the parameter *v* to this number then ideally the algorithm should output *MB(T)*. Viewed this way we treated the KS algorithm as an algorithm for approximating the *MB(T)* using only *v* variables. Unlike **IAMB**, the **IAMB** variants, and **GS**, the **KS** algorithm does not provide any theoretical guarantees of discovering the *MB(T)*.

**PC** (Spirtes *et al*. 2000) is a prototypical BN learning algorithm that is sound given the stated set of assumptions. **PC** learns the whole network (and so it does not scale-up well) from which the *MB(T)* can be easily extracted as the set of parents, children, and spouses of *T*. The **PC** algorithm starts with a fully connected unoriented Bayesian Network graph and has three phases. In phase I the algorithm finds undirected edges by using the criterion that variable *A* has an edge to variable *B iff* for all subsets of features there is no subset *S*, s.t. $\mathbf{I}(A \; ; B \mid S)$. In phases II and III the algorithm orients the edges by performing global constraint propagation. **IAMBnPC** could be thought of as improving **GS** by employing a more efficient, but still sound, way (i.e., **PC**) for the backward phase and a dynamic heuristic for the forward phase, or as improving **PC** by providing an admissible first phase heuristic that focuses **PC** on a local neighborhood.

We now provide a hypothetical trace of **IAMB** on the BN of Figure 1. We assume the reader's familiarity with the *d*-separation criterion (Spirtest *et al*. 2000) which is a graph-theory criterion that implies probabilistic conditional independence. In the beginning *CMB* is empty and the variable mostly associated with *T* given the empty set will enter *CMB*, e.g. *W*. In general, we expect the variables closer to *T* to exhibit the highest univariate association. Conditioned on *W*, the associations of all variables with *T* are calculated. It is possible that *O* will be the next variable to enter, since conditioned on *W*, *O* and *T* are dependent. After both *W* and *O* are in CMB*, Q* is independent of *T* and cannot enter *CMB*. Let us suppose that *R* enters next (a false positive). It is guaranteed that both *U* and *V* will also enter the *CMB* because they are dependent with *T* given any subset of the variables. In the backwards phase, *R* will be removed since it is independent of *T* given both *U* and *V*. Notice that in **GS**, *O* and *Q* are the last variables to be considered for inclusion, since they have no association with *T* given the empty set. This increases the probability that a number of false positives will have already entered *CMB* before *O* is considered, making the conditional independence tests unreliable.

## Experimental Results

In order to measure the performance of each algorithm, we need to know the real *MB(T)* to use it as a gold standard, which in practice is possible only in simulated data.

**Experiment Set 1**: BNs from real diagnostic systems (Table 1). We tested the algorithm on the ALARM Network (Beinlich *et al*. 1989), which captures the structure of a medical domain having 37 variables, and on Hailfinder, a BN used for modeling and predicting the weather, published in (Abramson *et al*. 1996), with 56 variables. We randomly sampled 10000 training instances from the joint probability that each network specifies. The task was to find the Markov Blanket of certain target variables. For ALARM the target variables were all variables, on which we report the average performance, while for Hailfinder there were four natural target nodes corresponding to weather forecasting, on which we report the performance individually. The performance measure used is the area under the ROC curve (Metz 1978). The ROCs were created by examining various different thresholds for the statistical tests of independence. For the **PC** algorithm thresholds correspond to the significance levels of the $G^2$ statistical test employed by the algorithm, whereas for the **GS** and the **IAMB** variants we consider $\mathbf{I}(X \; ; T \mid CMB)$ *iff* Mutual-Info$(X \; ; T \mid CMB)$ < *threshold*. For the **KS** we tried all the possible values of the parameter *v* of the variables to retain to create a very detailed ROC curve, and all values *k* that have been suggested in the original paper.

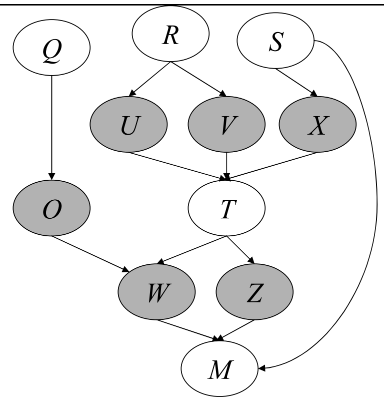

**Figure 1**: A example of a graph of a Bayesian Network. The gray-filled nodes are the MB(T).

| | ***ALARM*** | ***HAILFINDER*** | | | | |
|---|---|---|---|---|---|---|
| | | *Target 1* | *Target 2* | *Target 3* | *Target 4* | ***Average*** |
| **IAMB** | 86.70 | 96.30 | 96.23 | **97.12** | 78.04 | **90.88** |
| **interIAMB** | 86.70 | 96.30 | 96.23 | **97.12** | 78.04 | **90.88** |
| **interIAMBnPC** | **90.50** | **100.00** | **100.00** | **97.12** | 78.04 | **93.13** |
| **IAMBnPC** | 89.30 | **100.00** | **100.00** | **97.12** | 78.04 | **92.89** |
| **GS** | 80.59 | 96.30 | 77.67 | 72.12 | 68.04 | **78.94** |
| **KS, k=0** | 82.82 | **100.00** | 92.31 | 88.73 | **97.60** | **92.29** |
| **KS, k=1** | 80.56 | 70.28 | 47.76 | 82.84 | 67.40 | **69.77** |
| **KS, k=2** | 82.14 | 99.53 | 42.95 | 45.59 | 75.00 | **69.04** |
| **PC** | 95.20 | 99.07 | 98.11 | 81.73 | 96.08 | **94.04** |

**Table 1**: Experiments on Bayesian Networks used in real Decision Support Systems.

```
Phase I (forward)
CMB = ∅,
While CMB has changed
    Find the feature X in V-CMB-{T} that maximizes f(X ; T | CMB))
    If not I(X ; T | CMB )
        Add X to CMB
    End If
End While
Phase II (backwards)
Remove from CMB all variables X, for which I(X ; T | CMB-{X})
Return CMB
```

**Figure 2:** The IAMB algorithm.

| | MB with one spouse, three parents,and two children | | | | MB with four spouses, one parent, and two children | | | |
|---|---|---|---|---|---|---|---|---|
| | 50 Vars | 200 Vars | 1000 Vars | Average | 50 Vars | 200 Vars | 1000 Vars | **Average** |
| **IAMB** | 94.53 | 91.00 | 91.43 | **92.32** | 85.05 | 87.11 | **87.90** | **86.68** |
| **interIAMB** | 91.93 | 91.00 | 91.43 | **91.46** | 85.05 | 87.11 | **87.90** | **86.68** |
| **interIAMBnPC** | 93.67 | 94.43 | 88.77 | **92.29** | 87.71 | **88.01** | 73.69 | **83.14** |
| **IAMBnPC** | 94.43 | 91.60 | 91.67 | **92.57** | **90.48** | 85.63 | 85.70 | **87.27** |
| **GS** | 86.36 | 90.46 | 83.07 | **86.63** | 74.58 | 74.57 | 73.51 | **74.22** |
| **KS, k=0** | **95.93** | **96.17** | **96.15** | **96.08** | 74.72 | 73.39 | 73.06 | **73.72** |
| **KS, k=1** | 79.91 | 71.13 | 73.37 | **74.80** | 85.94 | 79.92 | 79.08 | **81.65** |
| **KS, k=2** | 86.11 | 87.35 | 86.94 | **86.80** | 85.88 | 82.24 | 81.40 | **83.17** |
| **PC** | 95.60 | - | - | **-** | 96.43 | - | - | **-** |

**Table 2**: Experiments on randomly generated Bayesian Networks

**Experiment Set 2:** Random BNs (Table 2). We generated three random BNs with 50, 200, and 1000 nodes each, such that the number of the parents of each node was randomly and uniformly chosen between 0 and 10 and the free parameters in the conditional probability tables were drawn uniformly from (0, 1). The Markov Blanket of an arbitrarily chosen target variable *T* contained 6 variables (three parents, two children, and one spouse) and was held fixed across the networks so that consistent comparisons could be achieved among different-sized networks. Each network adds more variables to the previous one without altering the *MB(T)*. We ran the algorithms for sample sizes in the set {1000, 10000, 20000} and report the average areas under the ROCs curve in Table 2. We remind the reader that the released version of the PC algorithm does not accept more than 100 variables, hence the missing cells in the figure. We see that the **IAMB** variants scale very well to large number of variables both in performance, and in computation time (**IAMB** variants took less than 20 minutes on the largest datasets, except **interIAMBnPC** which took 12 hours; the other methods took between one and five hours; all experiments on an Intel Xeon 1.8 and 2.4 GHz Pentium). We also generated another three BNs using the same approach as before, but this time the *MB(T)* contained four spouse nodes (instead of one), one parent, and two children nodes (for a total of seven nodes).

**Interpretation:** The results are shown in Tables 1 and 2. The best performance in its column is shown in bold (PC is excluded since it does not scale-up). We did not test whether the faithfulness assumption holds for any of the above networks, thus the results are indicative of the performance of the algorithms on arbitrary BNs. Whenever applicable, we see that **PC** is one of the best algorithms. Experiment Set 1 (Figure 3(a)): **IAMBnPC** and **interIAMBnPC** were the best algorithms on average. All **IAMB** variants are better than **GS**, implying that a dynamic heuristic for selecting variables is important. **KS** for *k*=0 is equivalent to ordering the variables according to univariate association with the target, a standard and common technique used in statistical analysis. This algorithm performs well in this set; however, the behavior of **KS** is quite unstable and non-monotonic for different values of *k* which is consistent with the results in the original paper (Koller, Sahami 1996). Experiment Set 2 (Figure 3(b)): We expect the simple static heuristic of **GS**, and **KS** for k=0, to perform well in cases were most members of *MB(T)* have strong univariate association with *T*, which is typically the case when there are no spouses of *T* in *MB(T)*. Indeed, in the first random BN, where there is only one spouse, both of these algorithms perform well (Figure 3(b)). However, in the second random BN there are four spouses of *T*, which seriously degrades the performance of **KS** for *k*=0 and **GS** (Figure 3(b)). **KS** for *k*=1,2 has unpredictable behavior, but it always performs worse than the **IAMB** variants. The **IAMB** variants and the **PC** algorithm still perform well even in this trickier case.

**Other Results**: Due to space limitations it is impossible to report all of our experiments. Other experiments we ran provide evidence to support another important hypothesis: **IAMB**'s dynamic heuristic is expensive in the data sample, therefore it is possible that for small sample sizes the simplest heuristics of **KS** for *k*=0 and GS will perform better, especially when there are not that many spouses in the *MB(T)*. Other experiments, suggest that the performance of the **PC** significantly degrades for small (less than 100 instances) data samples. This is explained by the fact that **PC** has a bias towards sensitivity: it removes an edge only if it can prove it should be removed, and retains it otherwise. Below a certain sample size the **PC** is not able to remove most edges and thus reports unnecessarily large Markov Blankets.

Given the above empirical results, we would suggest to the practitioners to apply the algorithms mostly appropriate for the available sample and variable size, i.e., the **PC** algorithm for sizes above 300 training instances and for variable size less than 100, **GS** and **KS** for *k*=0 (i.e., univariate association ordering) for sizes less than 300, and the **IAMB** variants for everything else.

## Discussion and Conclusion

**Discussion:** The *MB(T)* discovery algorithms can also be used for causal discovery. If there exists at least one faithful BN that captures the data generating process then the *MB(T)* of any such BN has to contain the direct causes of *T*. It thus significantly prunes the search space for an experimentalist who wants to identify such direct causes. In fact, other algorithms can post-process the *MB(T)* to direct the edges and identify the direct causes of *T* without any experiments, e.g. the **PC** of the **FCI** algorithm; the first assumes causal sufficiency while the second does not (Spirtes *et al*. 2000). In (Spirtes *et al*. 2000) specific conditions are discussed under which faithfulness gets violated. These situations are relatively rare in the sample limit as supported by the work of (Meek 1995). Most BN learning or *MB(T)* identification algorithms explicitly or implicitly assume faithfulness, e.g., **PC** and **GS**, but also (implicitly) BN score-and-search for most scoring metrics (see (Heckerman *et al*. 1997)).

**Conclusions:** In this paper we took a first step towards developing and comparing Markov Blanket identification algorithms. The concept of the Markov Blanket has strong connections with principled and optimal variable selection (Tsamardinos and Aliferis 2003), has been used as part of Bayesian Network learning (Margaritis and Thrun 1999), and can be used for causal discovery. We presented novel algorithms that are sound under broad conditions, scale-up to thousands of variables, and compare favorably with all the rest state-of-the-art algorithms that we have tried. We followed a principled approach that allowed us to interpret the empirical results and identify appropriate cases of usage of each algorithm. There is much room for improvement to the algorithms and hopefully the present work will inspire other researchers to address this important class of algorithms.

## References

Abramson, B., Brown, J., Edwards, W., Murphy, A., and Winkler R. L., "Hailfinder: A Bayesian system for forecasting severe weather", *International Journal of Forecasting*, 12 (1996), 57-71

Beinlich, I.A., *et al*. "The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks". *In Proc. of the Second European Conference on Artificial Intelligence in Medicine*, London, England. 1989.

Cheng J, Bell D., and Liu W., "Learning Bayesian Networks from Data: An Efficient Approach Based on Information Theory", Technical Report, 1998, URL http://www.cs.ualberta.ca/~jcheng/Doc/report98.pdf

Heckerman, D., Meek, C., and Cooper, G., "A Bayesian Approach to Causal Discovery", *Technical Report, Microsoft Research, MSR-TR-97-05,* 1997.

Koller, D. and M. Sahami. "Toward Optimal Feature Selection", *In Proc. of the Thirteenth International Conference in Machine Learning*. 1996.

Margaritis, D., and Thrun, S. "Bayesian Network Induction via Local Neighborhoods, Carnegie Mellon University", *Technical Report CMU-CS-99-134*, August 1999.

Meek, C., "Strong Completeness and Faithfulnes in Bayesian Networks", *In Proc. of Uncertainty in Artificial Intelligence* (UAI), 1995, 411-418.

Metz C. E. (1978) "Basic principles of ROC analysis", *Seminars in Nuclear Medicine*, 8, 283-298.

Neapolitan, R.E., "Probabilistic Reasoning in Expert Systems: Theory and Algorithms". 1990: John Wiley and Sons.

Spirtes, P., C. Glymour, and R. Scheines. "Constructing Bayesian network models of gene expression networks from microarray data". *In Proc. of the Atlantic Symposium on Computational Biology, Genome Information Systems & Technology*, 2000.

Tsamardinos, I, and C. F. Aliferis, "Towards Principled Feature Selection: Relevancy, Filters, and Wrappers*", In Proc. of Ninth International Workshop on Artificial Intelligence and Statistics*, 2003.