Constrained Local Latent Variable Discovery

Tian Gao
Rensselaer Polytechnic Institute, Troy NY
and IBM Watson Research Center,
Yorktown Heights NY
tgao@us.ibm.com

Qiang Ji
IBM Watson Research Center,
Yorktown Heights NY
jiq@rpi.edu

Abstract

For many applications, the observed data may be incomplete and there often exist variables that are unobserved but play an important role in capturing the underlying relationships. In this work, we propose a method to identify local latent variables and to determine their structural relations with the observed variables. We formulate the local latent variable discovery as discovering the Markov Blanket (MB) of a target variable. To efficiently search the latent variable space, we exploit MB topology to divide the latent space into different subspaces. Within each subspace, we employ a constrained structure expectation-maximization algorithm to greedily learn the MB with latent variables. We evaluate the performance of our method on synthetic data to demonstrate its effectiveness in identifying the correct latent variables. We further apply our algorithm to feature discovery and selection problem, and show that the latent variables learned through the proposed method can improve the classification accuracy in benchmark feature selection and discovery datasets.

1 Introduction

Latent variable models have been extensively studied in the literature. Learning latent variables can lead to the discovery of previously unknown factors to a certain phenomenon, and has many real-world applications such as human behavior analysis and medical diagnosis [Anandkumar *et al.*, 2013]. Latent variables can predict causal relationships and interpret the hidden effects, and also provide more compact representation of the data and a simpler model by which to make better generalizations [Elidan and Friedman, 2005]. The vast majority of latent variable models assumes that the number of latent variables and their exact locations with other variables are pre-determined, including many successful graphical models such as Latent Dirichlet Allocation [Blei *et al.*, 2003] and Hidden Conditional Random Fields [Quattoni *et al.*, 2007]. Deep architectures of graphical models with latent variables, such as Restricted Boltzmann Machines [Hinton *et al.*, 2006], have recently shown promising performance. A more challenging case of learning with latent variables considers the problem where the number of latent variables are known but their exact locations are unknown. To learn their structural relationships with observed variables, the most common structure learning method is the structural expectation-maximization (SEM) algorithm [Friedman and others, 1997; Elidan and Friedman, 2005; Borchani *et al.*, 2008]. When both the exact number and locations of latent variables are unknown, identifying and learning with latent variables becomes the most challenging, since it can create an infinite search space of possible structures. In graphical models, several works aim to detect the possible existence of structural latent variables by studying structural signatures such as cliques [Elidan *et al.*, 2000; He *et al.*, 2014]. Since it has been observed that the learned graph structures tend to become more densely connected at the presence of latent variables, it is possible to find some densely-connected sub-graphs and introduce some latent variables to simplify the graph structures.

To constrain such a difficult problem, we aim to discover local latent variables with respect to one target variable (such as the class label in classification). We specifically formulate such a problem as MB learning with latent variables, or latent MB learning for brevity. Except that assuming latent variables must appear in the MB of the target variable, we make no further assumptions on the number of latent variables and their specific locations inside the MB. Existing MB discovery algorithms are generally constraint-based or score-based approaches. Constraint-based approaches use independence tests to infer the MB [Koller and Sahami, 1996; Tsamardinos *et al.*, 2003; Aliferis *et al.*, 2003]. Score-based approaches, in particular the Score-Based Local Learning algorithm (SLL) [Niinimaki and Parviainen, 2012], use exact BN structure learning algorithms [Chickering, 2002; Cussens, 2011] with score criteria to find the local structures and thus the MB. However, standard MB discovery algorithms assume all the variables are observed. In comparison, we focus on learning a better MB of a target by considering latent variables: compared to the MB learned from observed variables, the jointly learned MB of the target variable should consist of both observed and latent variables, be more compact, and contain more mutual information about the target variable.

2 Markov Blanket Learning with Latent Variables

2.1 Preliminaries

A Bayesian Network for a set of random variables $\mathbf{V}$ is represented by a pair (G, θ). The network structure G is a directed acyclic graph (DAG) with nodes corresponding to the random variables in $\mathbf{V}$. The parameters θ indicate the conditional probability distribution of each node given its parents. If there is a directed path from X to Y, then X is an *ancestor* of Y and Y is a *descendant* of X. Two nodes are adjacent if they are connected by an edge. If nonadjacent X and Y have a common child, X and Y are *spouses* of each other.

Markov Condition [Pearl, 1988] states that a node in a BN is independent of its non-descendant nodes, given its parents. A DAG G and a joint distribution $\mathcal{P}$ are *faithful* to each other if all and only the conditional independencies true in $\mathcal{P}$ are entailed by G. The faithfulness condition has been assumed in existing BN structure learning and Markov Blanket discovery algorithms. *Markov Blanket* [Pearl, 1988] of a target variable T, $\mathbf{MB}_T$, is the minimal set of nodes conditioned on which all other nodes are independent of T, denoted as $X \perp\!\!\!\perp T|\mathbf{MB}_T, \forall X \subseteq \mathbf{V} \setminus T \setminus \mathbf{MB}_T$. The minimal set means that none of a MB's proper subsets satisfy the MB's property. Given an unknown distribution $\mathcal{P}$ that satisfies the Markov condition with respect to an unknown DAG G^0, Markov Blanket Discovery is the process of estimating the MB of a target node in G^0, from independently and identically distributed (i.i.d) data D of $\mathcal{P}$. If a DAG G and a distribution $\mathcal{P}$ are faithful to each other, then $\mathbf{MB}_T$, $T \in \mathbf{V}$, is unique [Pearl, 1988] and is the set of parents, children, and spouses of T. In addition, the parents and children set of T, $\mathbf{PC}_T$, is also unique. For example, in Figure 1, nodes P and C form $\mathbf{PC}_T$. $\mathbf{MB}_T$ contains its parent node P, its child node C, and its spouses $S1\&S2$. All other nodes are independent of T, given $\mathbf{MB}_T$.

Score-based Markov Blanket discovery algorithms rely on some score criteria to learn a best-fitting DAG G of the data and then extract the MB of a target variable from G. A *score* of a DAG structure G measures the goodness of fit of G on data D. Let D be a set of data consisting of i.i.d. samples from some distribution $\mathcal{P}$. Let G be any BN structure and G' be the same structure as G but with an extra edge from a node T to a node X. Let $\mathbf{Pa}_X^G$ be the parent set of X in G. A score criterion s is *consistent* if, as the size of the data D goes to infinity, the following two properties hold true: 1) if the structure G contains $\mathcal{P}$ and another structure G' does not, then $s(G, D) > s(G', D)$. 2) if G and G' both contain $\mathcal{P}$ but G has fewer parameters, then $s(G, D) > s(G', D)$. A score criterion is *decomposable* if it is a sum of each node's individual score that depends on only this node and its parents.

2.2 Problem Statement

We treat local latent variable learning as the latent MB learning of a target in directed graphical models (i.e., the learned MB must satisfy the DAG constraint). This procedure is different from other DAG structure learning methods with latent variables, as we consider only local latent variables to the target. The target is often the class label variable. Let $\mathcal{V}$ be the observed variable space and $\mathcal{H}$ be the latent variable space. Let T be a target variable in $\mathcal{V}$. When some variables in the MB of a target variable T from an unknown DAG are not observed, the learned MB from observed variables, $\mathbf{MB}_T^V \subseteq \mathcal{V}$, will contain some false positive observed nodes. For example, for the BN in Figure 1(a), if variable C is latent, $\mathbf{MB}_T^V$ would consist of P, $S1$, $S2$, B, and F by its definition as shown in Figure 1(b). The ideal MB of T with latent variables should be P, $S1$, $S2$, and C as shown in Figure 1(c).

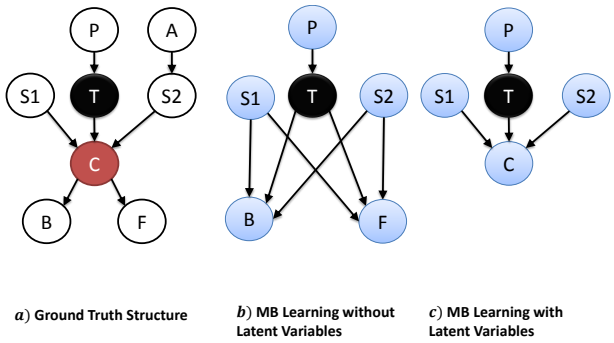

Figure 1: a)A Sample Bayesian Network. Black node T is the target node. b) If C is not observed, the learned MB set of T consists of P, $S1$, $S2$, B, and F. c) The ground truth MB set of T consists of P, $S1$, $S2$, and C.

Lemma 1. ***Latent MB Advantage***. *When latent variables exist in the MB of T, there exists a $\mathbf{MB}_T$ that includes latent variables and contains higher mutual information than $\mathbf{MB}_T^V$, i.e., $I(\mathbf{MB}_T; T) > I(\mathbf{MB}_T^V; T)$.*

Proof. Let latent variable set $\mathbf{X}$ exists in the $\mathbf{MB}_T$. By definition $X \not\perp\!\!\!\perp T|\mathbf{MB}_T^V$. Therefore, $I(\mathbf{MB}_T; T|\mathbf{MB}_T^V) > 0$. Also by the MB definition of $\mathbf{MB}_T$, $I(\mathbf{MB}_T^V; T|\mathbf{MB}_T) = 0 \implies I(\mathbf{MB}_T;\ T|\mathbf{MB}_T^V) > I(\mathbf{MB}_T^V; T|\mathbf{MB}_T)$. Since $I(\mathbf{MB}_T^V \cup \mathbf{MB}_T; T) = I(\mathbf{MB}_T^V; T) + I(\mathbf{MB}_T; T|\mathbf{MB}_T^V) = I(\mathbf{MB}_T; T) + I(\mathbf{MB}_T^V; T|\mathbf{MB}_T)$, $I(\mathbf{MB}_T; T) > I(\mathbf{MB}_T^V; T)$ hold. □

Since we can always add unknown $\mathbf{X}$ to $\mathbf{MB}_T^V$ without decreasing its mutual information to T, the problem would become trivial. Hence, we are more interested in the case where the size of latent MB set $\mathbf{MB}_T$ is equal or smaller than that of $\mathbf{MB}_T^V$, i.e., $\mathbf{X}$ in $\mathbf{MB}_T$ have to replace some observed variables in $\mathbf{MB}_T^V$[1]. Since the latent variables are still dependent of T given $\mathbf{MB}_T^V$, $I(\mathbf{MB}_T; T) > I(\mathbf{MB}_T^V; T)$ holds. For example, in Figure 1, $\mathbf{MB}_T$ should resemble the ground truth MB set $\{P, C, S1, S2\}$ and be smaller than or equal to $\mathbf{MB}_T^V$. It is our goal to learn and recover node C and the ground truth MB set.

[1]False negative MB nodes can exist when they are spouses, but only if their common children with the target variable has no descendants. Since in this case the latent MB set size would be bigger than the observed MB set, we do not consider this case by the problem definition.

2.3 Problem Formulation

We propose to learn the latent MB by using a score-based method, fitting $\mathbf{MB}_T$ to the data of observed variables so that a certain scoring criterion of $\mathbf{MB}_T$ increases from that of $\mathbf{MB}_T^V$. Since MB represents the local structure of the target, we can use decomposable and consistent DAG learning scores like Bayesian information Criterion (BiC) or Bayesian scores to evaluate different possible MB structures in the entire search space. Let $\mathbf{H}$ be the latent variables in $\mathbf{MB}_T$. Let $s_T(\mathbf{MB}_T, D)$ be the score of a MB set $\mathbf{MB}_T$ with respect to a target T, based on observed data samples D. Exploiting the decomposable property of the score, we have $s_T(\mathbf{MB}_T, D) = \sum_{X_i \in \mathbf{MB}_T} s_T(X_i, D)$, and each X_i depends on T and a small subset of $\mathcal{J} = \mathcal{V} \cup \mathcal{H}$ (i.e., only X_i's parent set in $\mathcal{J}$). Then we have the following formulation for MB discovery with latent variables H:

$$\mathbf{MB}_T^*, \theta_H^* = \underset{\mathbf{MB}_T, \theta_H}{\text{argmax}} \sum_{X_i \in \mathbf{MB}_T} s_T(X_i, D) \quad \text{subject to} \quad \mathbf{H} \subseteq \mathbf{MB}_T \tag{1}$$

where $\mathbf{MB}_T$ may contain subsets of variables from $\mathcal{V}$ and $\mathcal{H}$ and θ_H^* represents the unknown probability distribution of H.

It is easy to see that $\mathbf{MB}_T$ would score higher than $\mathbf{MB}_T^V$ in Equation 1: since H is in $\mathbf{MB}_T$, the dependency between H and T would make $\mathbf{MB}_T$ to have a higher scoring function than $\mathbf{MB}_T^V$ by the score consistency property. By the problem definition, we also enforce that the size of $\mathbf{MB}_T$ should have smaller or equal to that of $\mathbf{MB}_T^V$.

2.4 Latent MB Learning with Constraints

Equation 1 with the constraints creates a difficult formulation to solve directly, with multiple variables to be optimized in a nonlinear nonconvex objective function. Instead we propose a divide-and-conquer iterative approach for latent MB learning with constrained structure EM algorithm (LMB-CSEM), by discovering and learning the latent variables one at a time in different MB subspaces. The approach primarily divides the MB search space from Equation 1 into several non-overlapping subspaces and learns in each subspace separately. Then the optimal MBs within each subspace are compared to each other to obtain the final optimal MB. The overall methodology is summarized in Figure 2. Base on the outline from Figure 2, the proposed LMB-SEM is shown in Algorithm 1.

LMB-CSEM Algorithm

LMB-CSEM has three major steps. In the first step of LMB-CSEM (Line 3), we use a standard MB discovery algorithm to find a MB of a target T from observed variables only, $\mathbf{MB}_T^V$. In the second step, we employ a constrained structure EM (CSEM) algorithm to discovery and learn a MB with one latent variable within each subspace. Then in the third step, we compare MBs obtained from each space. If the learned MB with one latent variable in one of three sub-cases, *Constraint Set A, B, and C*, scores higher than that of *Baseline* from Figure 2, we can make the learned MB as the new baseline MB MB^V, and repeat Step 2 $\sim$ 3 to learn another latent variable. We repeat this process until adding more latent variables into the learned MB no longer improves the MB score or violates the size constraint. Below we provide details for each of the three steps.

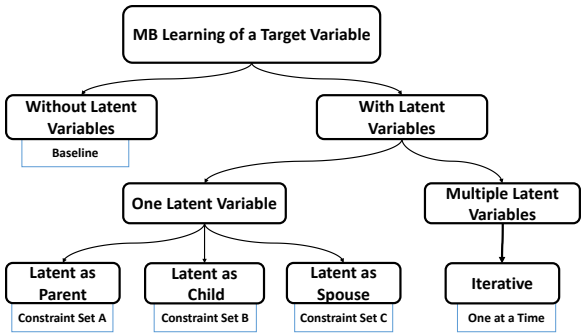

Figure 2: Breakdown of the proposed divide-and-conquer approach to learn the MB of a target variable with latent variables in the LMB-CSEM algorithm. Each subspace with different constraints is learned with CSEM.

Step 1: Learning the MB from Observed Variables Only

In the first step of LMB-CSEM, we learn MB_T^V, which serves as the *Baseline* case. When there are no latent variables, standard MB learning procedure such as the previously proposed SLL [Niinimaki and Parviainen, 2012] can be used. SLL is conjectured sound and complete, using an iterative procedure to find all and only the MB variables of a target one at a time.

In addition, baseline MB_T^V can also help the latent MB learning in other subspaces, as all observed variables in the highest scoring $\mathbf{MB}_T$ from Equation 1 must exist in $\mathbf{MB}_T^V$ by Lemma 2. This enables us to find the latent MB based on $\mathbf{MB}_T^V$, reducing the search space of $\mathbf{MB}_T$.

Lemma 2. ***Independence Consistency***. *Let $\mathbf{MB}_T$ be the highest scoring latent MB from Equation 1. Under the faithfulness assumption and using consistent scores, all the observed variables $X \in \mathbf{MB}_T$ are in $\mathbf{MB}_T^V$.*

Proof. Consider $X \in \mathbf{MB}_T$ but $X \notin \mathbf{MB}_T^V$, then $X \perp\!\!\!\perp T | \mathbf{MB}_T^V$ by the MB definition and using consistent scores, which means all path between T and X are blocked by $\mathbf{MB}_T^V$. As the latent variables H are in the $\mathbf{MB}_T$ by the problem definition, there is an unblocked path from T to H given $\mathbf{MB}_T^V$. Hence, $\mathbf{MB}_T^V$ also blocks all paths from X to H, otherwise there would be an unblocked path from X to H to T, violating the MB definition[2]. Hence, $X \perp\!\!\!\perp H | \mathbf{MB}_T^V$. By the faithfulness assumption and the composition property [Pearl, 1988], $X \perp\!\!\!\perp \{T, H\} | \mathbf{MB}_T^V \implies X \perp\!\!\!\perp T | \mathbf{MB}_T^V \cup H$. Hence, X cannot be the true positive PC node of T. Also by the MB definition, the true PC set must be a subset of $\{\mathbf{MB}_T^V \cup H\}$. Hence, X cannot be a true positive

[2]The exception of this statement occurs when X is a spouse and H is a latent child with no descendants, which is not considered based on the problem definition.

Algorithm 1 Latent MB Learning with Constrained Structure EM(LMB-CSEM) Algorithm

1: **Input:** $\mathcal{D}$: Data; T: the target variable;
2: **Output:** MB: the learned local DAG of T
{Step 1: Learn the Observed MB}
3: $[MB^V, PC^V] \leftarrow$ Learn the MB of T without H;
4: $N \leftarrow$ variable size of MB^V;
{Step 2: Divide-and-conquer}
5: Create one H;
6: $N \leftarrow N + 1$;
{Learn as H is a parent of T }
7: $Constraint \leftarrow zeros(N, N)$;
8: $Constraint(H, T) \leftarrow 1$; //strictly enforced
9: $[MB^P, score] \leftarrow CSEM(\mathcal{D}, T, Constraint, MB^V)$;
{Learn as H is a child of T }
10: $Constraint \leftarrow zeros(N, N)$;
11: $Constraint(T, H) \leftarrow 1$; //strictly enforced
12: $Constraint(X, H) \leftarrow -1, \forall X \in PC^V$; //strictly prohibited
13: $[MB^C, score] \leftarrow CSEM(\mathcal{D}, T, Constraint, MB^V)$
{Learn as H is a spouse of T }
14: **if** $\exists$ a T's child $X \in MB_T^V$ s.t. X has another parent than T **then**
15: $Constraint \leftarrow zeros(N, N)$;
16: $Constraint(H, T) \leftarrow -1$; $Constraint(T, H) \leftarrow -1$; //strictly prohibit
17: $Constraint(T, C) \leftarrow 1$; $Constraint(H, C) \leftarrow 1$; //strictly enforced
18: $[MB^S, score] \leftarrow CSEM(\mathcal{D}, T, Constraint, MB^V)$
19: **end if**
{Step 3: Find the best MB satisfied the constraints}
20: **if** none of MB^P, MB^C, and MB^S satisfied Constraint 1 and 2 **then**
21: $MB \leftarrow MB^V$
22: **else**
23: $MB \leftarrow \max_{i \in \{P,C,S,V\}} score(MB^i)$ s.t. MB^i satisfies constraints
24: **end if**
{To find more latent variables}
25: $MB^V \leftarrow MB$;
26: Repeat Step 2 $\sim$ 3 to discover one more latent variable H.
27: **Return:** MB;

spouse node of T if $X \perp\!\!\!\perp T | \mathbf{MB}_T^V \cup H$. Therefore, X is not in $\mathbf{MB}_T$, contradicting the assumption in the beginning. □

Step 2: Learning with Constrained SEM

We use the observed MB set to help learn the latent MB set via a divide and conquer approach. Depending on the MB subspaces as shown in Figure 2, we impose different constraints on top of a structure EM-like procedure to learn one optimal MB with latent variables.

Constraint Set A: When there exists a latent MB variable H and H is a parent of T, we enforce the existence of link $H \rightarrow T$ and use Algorithm 2, the constrained structure EM (CSEM) algorithm, to discover one latent MB. CSEM first estimates the expectation of latent variable H given observed variables, and estimates the values of H according to the observed data samples. The expectation steps of CSEM are the same as SEM, except that the initial graph structure must satisfy the constraint set for each respective subspace (Line 5 of Algorithm 2). The structure initialization has a big impact due to many local optima during SEM learning procedure. One can randomly initialize the graph, subject to the constraint set. One can also use a similar approach from clique-based SEM algorithms [Elidan and Friedman, 2005] to initialize the graph structure: When H is a parent of T, H is initialized as the sole parent of all observed PC set of T, the edges among the observed PC set are all removed, and then H is set as either a parent of T. The main intuition of this initialization procedure is to use H to reduce the structure complexity. After updating the parameter of H using EM, one can sample H to complete the data D (Line 6 of Algorithm 2). To robustly reflect the $P(H|A)$, one can obtain multiple samples of H for each observed samples in D. Since this additional sampling procedure is done for every sample, the probabilistic distributions of observed variables remain unchanged.

Then in the maximization step, CSEM maximizes the now completed data likelihood with respect to the graph structure and parameters. Different from SEM, CSEM also imposes constraints on H's possible locations based on the MB topology. We formulate constraints on the possible latent MB variable relationships with the observed variables as edge constraints (i.e., the enforcement or prohibition of edge existences) in a DAG, and then solve the MB learning with constraints as a DAG structure learning problem with these edge constraints. Some existing DAG learning algorithms can enforce these edge constraints, such as the constrained Branch-and-Bound (B&B) DAG structure learning algorithm [De Campos and Ji, 2011], and we can use them with few to no modifications. B&B first finds all the possible parent sets for each node, by using edge constraints to eliminate those parent sets that violates the constraints, and hence reduce the overall search space for the valid DAGs. Then B&B uses an efficient learning method to find the best scoring DAG from the valid parent sets of each node. For more details, we refer readers back to the original paper [De Campos and Ji, 2011].

Constraint Set B: When there exists a latent variable H and H is a child of T, we can use the same CSEM algorithm to learn a latent MB with Constraint Set B. Specifically, we enforce the existence of the link $T \rightarrow H$. In addition, if H is a child of T, an observed PC set variable $X \in \mathbf{PC}_T^V \subseteq \mathbf{MB}_T^V$ can only connect to H as a child of H. If X were a parent of H, X would form a V-structure with T and their common child H. This V-structure cannot happen because a V-structure would indicate independence between X and T[3].

[3]If X is also directly connected to T forming a fully connected

Algorithm 2 CSEM, the Constrained Structure EM(CSEM) Subroutine for Latent MB Learning

1: **Input:** $\mathcal{D}$: Data; T: the target variable; C: the constraint matrix; MB_T^V: the MB from Observed Variables
2: **Output:** MB_T: MB of the target with latent variables
3: $A \leftarrow union(MB_T^V, T)$;
{Step 2: Learning with LVs and Constraints}
4: **for** each iteration i until I iterations are reached **do**
5: Initialize MB_i for A and H, s.t. Constraint C;
6: **repeat**
7: $(MB_i, P(H|A)) \leftarrow EM(\mathcal{D}^A, MB_i)$;
8: $D_i^H \leftarrow SampleData(\mathcal{D}^A, P(H|A))$;
9: $\mathcal{D}_i^J \leftarrow Combine(\mathcal{D}^A, \mathcal{D}_i^H)$;
10: $[MB_i, score(MB_i)] \leftarrow \text{B\&B}(D_i^J, T, C)$
11: **until** MB_i does not change ;
12: **end for**
{Step 3: Find the best DAG}
13: Compute $score(MB_i)$ for each MB_i with $\mathcal{D}_i^J$;
14: $MB \leftarrow \max_{i \in n} score(MB_i)$
15: **Return:** MB

Therefore the edge constraint $H \rightarrow X$ must hold if H and X are adjacent. An efficient way to enforce it is to prohibit the existence of $X \rightarrow H, \forall X \in \mathbf{PC}_T^V$. For this subspace, H is initialized as the sole parent of all observed PC set of T, the edges among the observed PC set are all removed, and then H is set as a child node of T.

Constraint Set C: When there exists a latent MB variable H and H is a spouse of T, H and T should share at least one common child. We enforce the latent MB variable H as a spouse of T with the same common child X (i.e., with the edge constraints $H \rightarrow X$ and $T \rightarrow X$). For the initial graph structure, T is initialized as the parent of candidate child nodes X, H is initialized to be the parent of X and T's observed spouses with X, and all other edges among X and these observed spouses are removed. We again use the same CSEM to find one best latent MB set.

Step 3: Finding the Best MB from All Subspaces
In the last step of LMB-CSEM, learned MBs from different subspaces are compared to each other to find the most optimal one. We directly enforce the constraint that the size of $\mathbf{MB}_T$ must be equal or smaller than that of $\mathbf{MB}_T^V$. If a learned $\mathbf{MB}_T$ violates the constraint, we disregard it; otherwise, we choose the highest scoring latent MB set from different subspaces. The $score$ used for comparing different MB subspaces in Line 23 of LMB-CSEM cannot be the traditional Bayesian or BiC scores because of the existence of latent variables. To be consistent with the MB property and discovery selection criterion, we use the mutual information of the latent MB set as the score criterion.

LMB-CSEM Algorithm with Multiple Latent Variables
When multiple variables in a target MB are not observed, we can use a greedy method to iteratively learn one latent variable at a time. Beginning with one latent variable, we can compare scores of $\mathbf{MB}_T^V$ with $\mathbf{MB}_T$. If $\mathbf{MB}_T$ has a better score, then there exists one latent variable. Then we proceed to assume there exist two latent variables. Using the previously learned MB set with one latent variable as the new observed MB set, we introduce one more latent variable and repeat the same procedure. We keep adding more latent variables until the new latent MB score is lower or violates the size constraint. This procedure is reflected by Line 25 $\sim$ 26 of Algorithm 1 and converges to local optima.

clique with T and H, we can always change some arc directions between these three nodes to make X a child of H and preserve the same independence relationships.

3 Experiments

We first demonstrate the performance of the proposed methods on synthetic datasets, and then apply our method to benchmark feature selection and discovery datasets. We fix the cardinality of latent variables to be 2 in all experiments.

3.1 Synthetic Datasets for Latent MB Discovery

We test the proposed LMB-CSEM algorithm on synthetic datasets if it can discover the latent variables and correctly learn the graph structure. We make a 4BN, consisting of node T, C, B, and F from Figure 1 with the same edges among them. 7BN is also constructed similarly with We sample 10k data from the structure, and remove all samples for C. We run the SEM [Friedman and others, 1997], semi-clique SEM (SCSEM) [Elidan *et al.*, 2000], and the proposed LMB-CSEM algorithm to see whether three methods can recover the ground truth MB sets of the target variable T in each network. We randomly initialize the structures with one latent variable and parameters for SEM. For SCSEM, we use the semi-clique rules to initialize the structures with one latent variable, and allow SCSEM with the complete flexibility in adapting the network structures. From the latent structures of each algorithm, we extract the latent MB set. We use the same MB discovery error metric as previous works, namely the distance between true MB set, including the latent variable H, and the learned MB set of T in each network: $d = \sqrt{(1 - Precision)^2 + (1 - Recall)^2}$. *Precision* is the number of true positives in the detected MB divided by the total size of the detected MB. *Recall* is the number of true positives in the detected MB divided by the size of the ground truth MB. Thus, the lower d is better. We repeat the experiment 100 times using different samples, and compare the errors of the learned latent MBs for different algorithms. The learned results are shown in Table 1. LMB-CSEM outperforms SEM and SCSEM algorithms in finding the correct latent MB set, with lower MB discovery errors. SEM fails to find H in the learned MB set 6 of 10 times, and SCSEM fails to find H in the learned MB set 5 times. Both methods never find H alone as the MB set. In comparison, the proposed LMB-CSEM is able to find H as the entire MB set 8 out of 10 times. While previous latent BN structure learning algorithms often find the highest scoring structures fitted to the data, these structures are not necessarily the data-generating structures [Elidan *et al.*, 2000]. LMB-CSEM could alleviate such a problem using constraints.

Table 1: Latent MB Discovery Errors for Different Algorithms in Synthetic Datasets

DATASET	SEM	SCSEM	LMB-CSEM
4BN	1.04 ± 0.43	0.84 ± 0.38	0.26 ± 0.58

3.2 Feature Selection and Discovery Datasets

We apply the proposed LMB-CSEM and current feature selection algorithms to standard feature selection and discovery datasets. We use five feature selection datasets from the UCI machine learning repository and related works [Brown *et al.*, 2012] to show the effectiveness of LMB-CSEM on feature selection applications. Although the feature sizes of these datasets are not large, ranging from 13 to 30, they present challenges, as training sample sizes are relatively low compared to cardinalities of each variable in the datasets. We use half the data size for training and half for testing.

Scaling up: In practice, the number of features can be large to be directly used with the exhaustive nature of LMB-CSEM. We employs a similar idea of random subspace projection as the popular random forest classifier [Breiman, 2001]. We randomly select a subset of variables with size d_L to find one set of MB with latent variables (usually $d_L = \sqrt{N}$ where N is the total number of the observed variables). For each set of MB, we use LMB-CSEM to find one latent MB set. By repeating this procedure L times, we learn a total of L sets of MBs. During testing, we first infer latent values of each set of MBs, given the values of observed variable values. Then we can combine all the latent MB sets to infer the class label. From inferred class labels of each individual classifier, we can use either simple majority voting or the average probability of all classifiers to obtain the final predicted class label.

Parameters: We run the proposed Algorithm 1 LMB-CSEM to find $L = 20$ different latent MB sets. To show that the learned latent features can improve classification tasks, we learn only one latent feature per latent MB set in LMB-CSEM. Experiments show that even one latent MB feature per MB set can lead to strong performance improvement with a good trade-off on efficiency. We also set the number of different initializations I in Algorithm 1 to be 80. In our observation, a higher number of I can lead to more performance gain, as different initializations for the EM algorithms can avoid local maxima better. Furthermore, we choose to use the BiC score to learn different MB sets within each subspace, and the training errors as $score(MB)$ to compare different MB sets across subspaces and from different initializations, which performs best compared to the mutual information and conditional likelihood of the target label. Lastly, we use both the predicted latent MB values and inferred probabilities $P(H|T, \mathbf{MB}_T)$ as features to train a SVM to obtain the final classification results, since the latent values alone lose some information about the confidence of the latent variables.

We compare the performance of features of LMB-CSEM to the MB learned without latent variables. Since MB-based feature selection methods are a filter approach [Koller and Sahami, 1996], we make a direct comparison to the state-of-the-art filtered feature selection method, the minimum Redundancy Maximum Relevance (mRMR) [Peng *et al.*, 2005]. We follow the standard experiment setups [Brown *et al.*, 2012] and use linear SVM classifiers to test the classification accuracy from the selected/learned features. We choose the top N mRMR features, where N is the total feature size from LMB-CSEM. Table 2 shows the error rates of different feature selection and LMB-CSEM algorithms with bold numbers representing the best results. Learned latent MB variables can improve the classification accuracy compared to the MB learned just from observed features in all five datasets, with the biggest absolute improvement of 14.7% observed in the HEART dataset. It confirms that latent variables can complement the observed feature set. Compared to mRMR, LMB-CSEM also performs better, with a 7% improvement on average.

Table 2: Testing Error Rates on Real Feature Selection Datasets using linear SVM

DATASET	MRMR	SLL	LMB-CSEM
CONGRESS	6.4%	7.3%	**5.9%**
HEART	16.5%	18.1%	**14.3%**
KRVSKP	24.1%	11.3%	**9.4%**
BREAST	5.3%	6.0%	**4.2%**
PARKINSONS	19.4%	25.5%	**18.4%**
MEAN	15.3%	11.6%	**7.7%**

4 Conclusion

We propose to learn local latent variables of a target variable, in the form of Markov Blankets, to complement the observed variables. The local latent variables can increase the mutual information to the target and possibly help regulate the conditional probability of the target. We formulate the local latent variable learning as the MB learning in the presence of latent variables, and propose a divide-and-conquer approach to automatically determine the existence of latent variables and learn the relationships they have with the target variable and observed variables. The learned MB set of a target from both the observed and latent variables is more compact, can provide more information for the target and improve the classification accuracy. We demonstrate the superior performance of proposed methods to state-of-the-art methods on synthetic networks and benchmark feature selection datasets. Future work could study the cardinality of latent variables of LMB-CSEM.

References

[Aliferis *et al.*, 2003] Constantin Aliferis, Ioannis Tsamardinos, Alexander Statnikov, C. F. Aliferis, I. Tsamardinos, and Er Statnikov. Hiton, a novel markov blanket algorithm for optimal variable selection, 2003.

[Anandkumar *et al.*, 2013] Animashree Anandkumar, Daniel Hsu, Adel Javanmard, and Sham Kakade. Learning linear bayesian networks with latent variables. In *Proceedings of the 30th International Conference on Machine Learning (ICML)*, pages 249–257, 2013.

[Blei *et al.*, 2003] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[Borchani *et al.*, 2008] Hanen Borchani, Nahla Ben Amor, and Fedia Khalfallah. Learning and evaluating bayesian network equivalence classes from incomplete data. *International Journal of Pattern Recognition and Artificial Intelligence*, 22(02):253–278, 2008.

[Breiman, 2001] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[Brown *et al.*, 2012] Gavin Brown, Adam Pocock, Ming-Jie Zhao, and Mikel Lujan. Conditional likelihood maximisation: A unifying framework for information theoretic feature selection. *Journal of Machine Learning Research*, page 27 66, January 2012.

[Chickering, 2002] David Maxwell Chickering. Optimal structure identification with greedy search. *Journal of Machine Learning Research*, 2002.

[Cussens, 2011] James Cussens. Bayesian network learning with cutting planes. In *Proceedings of the Twenty-Seventh Conference Annual Conference on Uncertainty in Artificial Intelligence (UAI-11)*, pages 153–160, Corvallis, Oregon, 2011. AUAI Press.

[De Campos and Ji, 2011] Cassio P De Campos and Qiang Ji. Efficient structure learning of bayesian networks using constraints. *The Journal of Machine Learning Research*, 12:663–689, 2011.

[Elidan and Friedman, 2005] Gal Elidan and Nir Friedman. Learning hidden variable networks: The information bottleneck approach. In *Journal of Machine Learning Research*, pages 81–127, 2005.

[Elidan *et al.*, 2000] Gal Elidan, Noam Lotner, Nir Friedman, Daphne Koller, et al. Discovering hidden variables: A structure-based approach. In *Advances in Neural Information Processing Systems (NIPS)*, volume 13, pages 479–485, 2000.

[Friedman and others, 1997] Nir Friedman et al. Learning belief networks in the presence of missing values and hidden variables. In *Internaltional Conference on Machine Learning (ICML)*, volume 97, pages 125–133, 1997.

[He *et al.*, 2014] Chao He, Kun Yue, Hao Wu, and Weiyi Liu. Structure learning of bayesian network with latent variables by weight-induced refinement. In *Proceedings of the 5th International Workshop on Web-scale Knowledge Representation Retrieval & Reasoning*, pages 37–44. ACM, 2014.

[Hinton *et al.*, 2006] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.

[Koller and Sahami, 1996] Daphne Koller and Mehran Sahami. Toward optimal feature selection. In *ICML 1996*, pages 284–292. Morgan Kaufmann, 1996.

[Niinimaki and Parviainen, 2012] Teppo Niinimaki and Pekka Parviainen. Local structure disocvery in bayesian network. In *Proceedings of Uncertainy in Artifical Intelligence, Workshop on Causal Structure Learning*, pages 634–643, 2012.

[Pearl, 1988] Judea Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann Publishers, Inc., 2 edition, 1988.

[Peng *et al.*, 2005] Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 27(8):1226–1238, 2005.

[Quattoni *et al.*, 2007] Ariadna Quattoni, Sybor Wang, Louis-Philippe Morency, Michael Collins, and Trevor Darrell. Hidden conditional random fields. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, pages 1848–1852, 2007.

[Tsamardinos *et al.*, 2003] Ioannis Tsamardinos, Constantin Aliferis, Alexander Statnikov, and Er Statnikov. Algorithms for large scale markov blanket discovery. In *In The 16th International FLAIRS Conference, St*, pages 376–380. AAAI Press, 2003.