Appendix for Bounding the Family-Wise Error Rate in Local Causal Discovery using Rademacher Averages

Dario Simionato[0000−0001−5533−425X] and Fabio Vandin[0000−0003−2244−2320]

Department of Information Engineering, University of Padova, Italy
dario.simionato@phd.unipd.it,fabio.vandin@unipd.it

1 PCMB, IAMB and other useful pseudocode

Algorithm 1: Pseudocode for IAMB [3]

Input: target variable T, set of variables $\mathbf{V}$
Output: $MB(T)$

```
/* Add true positives to MB */
MB ← ∅;
repeat
    Y ← arg max_{X ∈ V \ MB \ {T}} dep(T, X, MB)
    if T ⊥̸⊥ Y | MB then
        MB ← MB ∪ {Y}
until MB does not change;
/* Remove false positives from MB */
foreach X ∈ MB do
    if T ⊥⊥ X | MB \ {X} then
        MB ← MB \ {X}
return MB;
```

2 Proofs for our algorithms

Theorem 1. *`RAveL-PC(`T`,`$\mathbf{V}$`,`δ`)` outputs a set of elements in $PC(T)$ with FWER $\leq \delta$.*

Proof (sketch). Note that the number of false positives of `RAveL-PC(`T`,`$\mathbf{V}$`,`δ`)` is > 0 if and only if there is at least one variable X of $\mathbf{V} \setminus \{T\}$ that is not in $PC(T)$ and is in the set PC reported by `RAveL-PC(`T`,`$\mathbf{V}$`,`δ`)`. A variable X is returned in PC if and only if all independence tests between T and X (conditioning on the various sets $\mathbf{Z} \subseteq \mathbf{V} \setminus \{X, T\}$) reject the null hypothesis. Therefore `RAveL-PC(`T`,`$\mathbf{V}$`,`δ`)` reports a false positive only if at least one independence test returns a false positive, which happens with probability at most δ by definition of `test_indep(`T`,`X`,`$\mathbf{Z}$`,`δ`)`. □

Algorithm 2: Pseudocode for GetPCD [2]

Input: target variable T
Output: $GetPCD(T)$

$PCD \leftarrow \emptyset$;
$CanPCD \leftarrow \mathbf{V} \setminus \{T\}$;
repeat
  /* *Remove false positives from CanPCD* */
  foreach $X \in CanPCD$ **do**
    $Sep[X] \leftarrow \arg\min_{\mathbf{Z} \subseteq PCD} dep(T, X|\mathbf{Z})$
  foreach $X \in CanPCD$ **do**
    if $T \perp\!\!\!\perp X|Sep[X]$ **then**
      $CanPCD \leftarrow CanPCD \setminus \{X\}$
  /* *Add the best candidate to PCD* */
  $Y \leftarrow \arg\max_{X \in CanPCD} dep(T, X|Sep[X])$
  $PCD \leftarrow PCD \cup \{Y\}$;
  $CanPCD \leftarrow CanPCD \setminus \{Y\}$;
  /* *Remove false positives from PCD* */
  foreach $X \in PCD$ **do**
    $Sep[X] \leftarrow \arg\min_{\mathbf{Z} \subseteq PCD \setminus \{X\}} dep(T, X|\mathbf{Z})$
  foreach $X \in PCD$ **do**
    if $T \perp\!\!\!\perp X|Sep[X]$ **then**
      $PCD \leftarrow PCD \setminus \{X\}$
until *PCD does not change*;
return *PCD;*

Algorithm 3: Pseudocode for GetPC [2]

Input: target variable T
Output: $GetPC(T)$

$PC \leftarrow \emptyset$;
foreach $X \in GetPCD(T)$ **do**
  if $T \in GetPCD(X)$ **then**
    $PC \leftarrow PC \cup \{X\}$
return *PC;*

Theorem 2. *`RAveL-MB` outputs a set of elements in $MB(T)$ with $FWER \leq \delta$.*

Proof (sketch). The set of `RAveL-MB`'s output elements is the union of the set O_1 of variables returned by `RAveL-PC(`T`,`$\mathbf{V}$`,`δ`)`, and the set O_2 of candidate spouses Y for which `test_indep(`T`,`Y`,`$\mathbf{V} \setminus \{Y, T\}$`,`$\delta$`)` rejects the null hypothesis. Then, a necessary condition to return a false positive is that at least one between sets O_1 and O_2 contains a false positive. The last event happens if and only if all calls to `test_indep(`T`,`X`,`$\mathbf{Z}$`)` returns at least a false positive, which happens with probability at most δ. □

Algorithm 4: Pseudocode for PCMB [2]

Input: target variable T
Output: $PCMB(T)$
1 /* Add true positives to MB */
2 $PC \leftarrow GetPC(T)$;
3 $MB \leftarrow PC$;
4 /* Add more true positives to MB */
5 **foreach** $Y \in PC$ **do**
6 **foreach** $X \in GetPC(Y)$ **do**
7 **if** $X \notin PC$ **then**
8 Find $\mathbf{Z}$ such that $T \perp\!\!\!\perp X|\mathbf{Z}$ and $T, X \notin \mathbf{Z}$
9 **if** $T \not\perp\!\!\!\perp X|\mathbf{Z} \cup Y$ **then**
10 $MB \leftarrow MB \cup \{X\}$
11 **return** MB;

3 Additional proofs for GetPC, PCMB and IAMB

Theorem 3 (Study of false positives in GetPCD). *An element $X \notin PCD(T)$ gets returned from GetPCD only if not all the parents of T are detected or the null hypotheses of some independence tests get wrongly rejected.*

Proof. Let us recall that an element X returned by GetPCD is a False Negative if and only if $X \notin Parents(T) \cup Descendants(T)$.

We see that an element is returned by $GetPCD(T)$ only if it is not removed at lines 8 and 16, which means that the null hypothesis of tests at lines 8 and 18 get always rejected[1]. The independence test determines dependence of T from X only if conditioning on $\mathbf{Z} = sep[X]$ there is a open path between X and T, or if the null hypothesis gets wrongly rejected.

Let us now study the two cases of X being disconnected to T and of T being connected to T.

Disconnected case. Let X be disconnected from T. Since there are no paths from X to T (therefore no open paths from X to T), X may be added to $GetPCD(T)$ only if independence tests at lines 8 and 18 gets wrongly rejected.

Connected case. Let $X \notin PCD(T)$ be connected to T. X gets added only if in any iteration of the cycle the null hypothesis on tests at lines 8 and 18 gets wrongly rejected or there is an open path conditioning on $Sep[X]$.

By supposing not to have wrong rejections of the null hypotheses, $\mathbf{Z} = Parents(T)$ d-separates X and T by definition of parents since X is not a descendant of T. This implies that if some parent of T gets undetected, then it may not be possible to d-separe X from T.

[1] The "if" clause does not hold since an element may be added and then subsequently removed leading to the end of the repeat cycle because PCD did not change, but there still are elements in canPCD i.e. unremoved elements.

Theorem 4. *$GetPCD(T)$ outputs a set of elements in $PCD(T)$ with FWER lower than δ if the FWER of every independence test performed by GetPCD is below δ and the infinite power assumption holds while testing independence of elements directly connected.*

Proof. By analyzing GetPCD structure as in Th. 3, an element is returned only if both independence tests at line 8 and 18 reject the null hypothesis therefore the algorithm outputs a false positive if under infinite power assumption for elements directly connected at least one independence test returns a false positive. Let us define the events $E =$ "*GetPCD(T) outputs a false positive*" and $E_i =$ "*the i-th independence test returns a false positive*". We then have

$$FWER = P(E) \leq P(\bigcup_i E_i) \leq \delta$$

by definition of FWER. □

Corollary 1. *Let us assume that the independence tests performed by GetPCD do not return any False Positive. $GetPCD(T)$ outputs a set of elements in $PCD(T)$ with FWER is lower than δ if only if the infinite power assumption while testing the independence of elements directly connected is satisfied.*

Proof. Let us prove this by proving the equivalent statement if the infinite power assumption while testing the independence of elements directly connected is NOT satisfied then $GetPCD(T)$ outputs a set of elements in $PCD(T)$

Theorem 5. *$GetPC(T)$ outputs a set of elements in $PC(T)$ whose FWER is lower than δ if the FWER of every independence test performed by GetPC is below δ and the infinite power assumption holds while testing independence of elements directly connected.*

Proof. *GetPC* outputs a false positive only if at least one call to *GetPCD* at lines 2-3 outputs a false positive and, under the infinite power assumption while testing independence of elements directly connected, this happens only if at least one independence test outputs a false positive. Let us define the events $E =$ "*GetPC(T) outputs a false positive*" and $E_i =$"the i-th independence test returns a false positive". We then have

$$FWER = P(E) \leq P(\bigcup_i E_i) \leq \delta$$

by definition of FWER. □

Theorem 6. *$PCMB(T)$ outputs a set of elements in $MB(T)$ with FWER lower than δ if the FWER of every independence test performed by PCMB is below δ and the infinite power assumption holds.*

Proof. *PCMB* outputs a false positive only if there is a false positive in any independence test performed by *GetPC* calls at lines 2 and 6, or if tests at lines

8 and 9 return a false negative or a false positive, respectively. Given the infinite power assumption and Corollary 5, $PCMB$ outputs a false postive only if at least one independence test outputs a false positive and by defining the events E ="PCMB(T) outputs a false positive" and E_i = "the i-th independence test returns a false positive" we have

$$FWER = P(E) \leq P(\bigcup_i E_i) \leq \delta$$

by definition of FWER. □

Theorem 7. *$IAMB(T)$ outputs a set of elements in $MB(T)$ with FWER lower than δ if the FWER of every independence test performed by IAMB is below δ and the infinite power assumption holds.*

Proof. $IAMB$ outputs a false positive only if an element $X \notin MB(T)$ gets added to MB at lines 5-6, and it does not get removed from MB at lines 10-11. Under the infinite power assumption, all elements in $PC(T)$ gets added at lines 5-6 by definition of PC, therefore X gets returned by IAMB only if independence tests at lines 10-11 output a false positive. Then, by defining the events E = "GetPC(T) outputs a false positive" and E_i = "the i-th independence test returns a false positive", we have

$$FWER = P(E) \leq P(\bigcup_i E_i) \leq \delta$$

by definition of FWER. □

4 Relaxation or removal of infinite power assumpion scenarios

Scenario 1: Relaxing infinite power assumption to hold only for directly connected elements Consider as an example a set $\mathbf{Z}$ for which T and $X \notin PC(T)$ are dependent and for which adding Y does not change the dependencies. If there is a false negative when testing the conditional independence of T from X given $\mathbf{Z}$ but the test conditioning on $\mathbf{Z} \cup \{Y\}$ outputs the correct result, then X will be erroneously considered as a spouse in PCMB. Figure 1 shows one such example by considering $T = A, X = B$, $\mathbf{Z} = \emptyset$ and $Y = C$.

Scenario 2: Removing the infinite power assumption Consider as an example the calculus of $GetPC(D)$ in the scenario of Figure 1. Let us suppose that a False Negative occurs when testing the unconditional independencies between D and B and between A and B. Let us further suppose $\mathbf{Z} = \{B, C\}$ to be the only set for which the null hypothesis of independence between A and D is not rejected. Then $GetPCD(D)$ will contain A (because the independence conditioning on $\mathbf{Z} = \{B, C\}$ is never tested), and similarly $GetPCD(A)$ will contain D leading A to be returned by $GetPC(D)$.

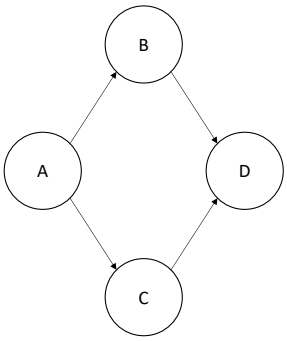
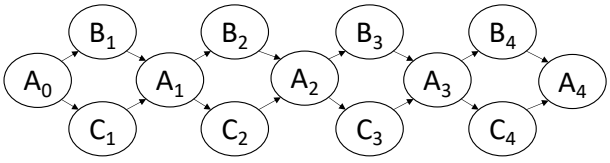


Fig. 1. Example of Bayesian Newtork for which algorithms GetPC, PCMB, and IAMB may fail if infinite power assumption is not met.

5 Structural model used in synthetic experiments

Here we report the structural model used in our synthetic experiments



$$A_0 \sim N(0,1)$$
$$\forall i \in \{1,2,3,4\}$$
$$B_i \sim 3 + 3A_{i-1} + N(0,1)$$
$$C_i \sim -1 - 2A_{i-1} + N(0,1)$$
$$A_i \sim 3 + 2B_{i-1} + 3C_{i-1} + N(0,1)$$

Fig. 2. Bayesian Network used in our synthetic data generation.

6 Variables in Boston housing dataset

Variables description follows from dataset introductory paper [1].

Variable name	Explanation
CRIM	per capita crime rate by town
ZN	proportion of residential land zoned for lots over 25,000 sq.ft.
INDUS	proportion of non-retail business acres per town.
CHAS	Charles River dummy variable (1 if tract bounds river; 0 otherwise)
NOX	nitric oxides concentration (parts per 10 million)
RM	average number of rooms per dwelling
AGE	proportion of owner-occupied units built prior to 1940
DIS	weighted distances to five Boston employment centres
RAD	index of accessibility to radial highways
TAX	full-value property-tax rate per $10,000
PTRATIO	pupil-teacher ratio by town
B	$1000(Bk - 0.63)^2$ where Bk is the proportion of blacks by town
LSTAT	% lower status of the population
MEDV	Median value of owner-occupied homes in $1000's

References

1. Harrison Jr, D., Rubinfeld, D.L.: Hedonic housing prices and the demand for clean air. Journal of environmental economics and management **5**(1), 81–102 (1978)
2. Peña, J.M., Nilsson, R., Björkegren, J., Tegnér, J.: Towards scalable and data efficient learning of Markov boundaries. International Journal of Approximate Reasoning **45**(2), 211–232 (2007). https://doi.org/10.1016/j.ijar.2006.06.008
3. Tsamardinos, I., Aliferis, C., Statnikov, A., Statnikov, E.: Algorithms for Large Scale Markov Blanket Discovery. FLAIRS Conference (i), 376–381 (2003), http://www.aaai.org/Papers/FLAIRS/2003/Flairs03-073.pdf