Markov Equivalence in Cyclic Directed Graphs

[link]

authored by Tom Claassen, Joris Mooij

Arushi Arora, CS276

Outline

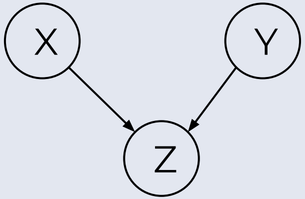
“[n]ew procedure to establish Markov equivalence between directed graphs that may or may not contain cycles under the d-separation criterion…based on the Cyclic Equivalence Theorem…rephrased from an ancestral perspective”

- **Claassen, Mooij**

Introduction

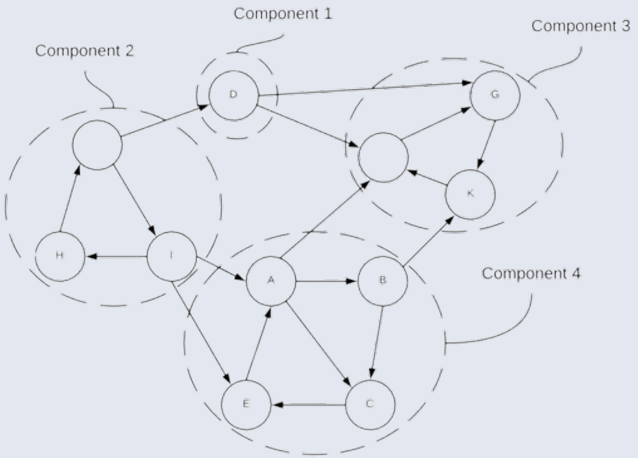
- need to handle feedback cycles in learning algorithms for real world causal discovery
- G1 and G2 are said to be d-separation (Markov) equivalent iff every d-separation in G1 also holds in G2 and vice versa
    - Polynomial time algorithm for deciding Markov equivalence of directed cyclic graphs (Richardson, 1997)
    - Linear complexity for sparse graphs (Claassen and Bucur, 2022)

Introduction



- A node Z is a *collider* on a path if the subpath is of the form X → Z ← Y
  - Otherwise, Z is a *noncollider*
- X is *d-connected* to Y given Z iff there is an x, y such that there is a path between X and Y on which every noncollider is not in Z and every collider is an ancestor of Z
  - Otherwise, X and Y are *d-separated* given Z

Introduction

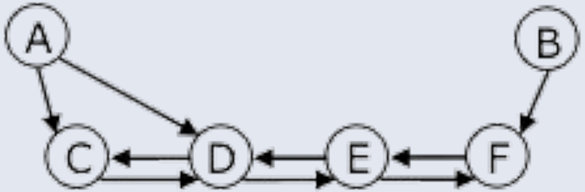


- *Strongly connected component* is a maximal set of vertices where every vertex is reachable via a directed path from every other vertex in the set
- Two nodes are *virtually adjacent* iff there is no edge between A and B but they have a common child C which is an ancestor of A or B
  - Two nodes connected by virtual edge cannot be d-separated by any set of nodes

Introduction

- Sequence of vertices where all neighboring nodes in the sequence are (virtually) adjacent in the graph is called an *itinerary*
  - Itinerary is *uncovered* if none of the nodes are (virtually) adjacent to each other except for the ones that occur consecutively

Introduction

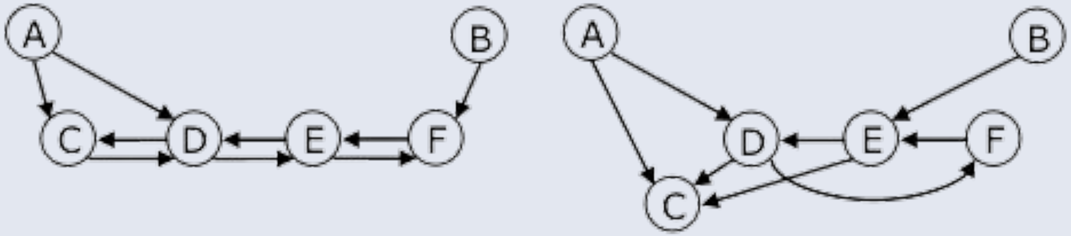


- Triple (A, B, C) forms *conductor* if (A, B, C) is an itinerary and B is an ancestor of A and/or C
    - A (non)conductor is *unshielded* if A and C are not (virtually) adjacent
- A nonconductor triple (A, B, C) is a *perfect nonconductor* if B is also a descendant of a common child of A and C

Introduction

- Triples (X_0, X_1, X_2) and (X_{n-1}, X_n, X_{n+1}) are *mutually exclusive conductors w.r.t an itinerary* if $(X_0 \ldots X_{n+1})$ is a sequence of vertices such that:
  - Each consecutive triple along the itinerary is a conductor
  - All nodes X_1 to X_n are ancestors of each other, but not ancestors of either X_0 or X_{n+1}

Introduction



(A, D, F) and (D, F, B) → M.E. conductors on an uncovered itinerary (A, D, F, B)

Introduction

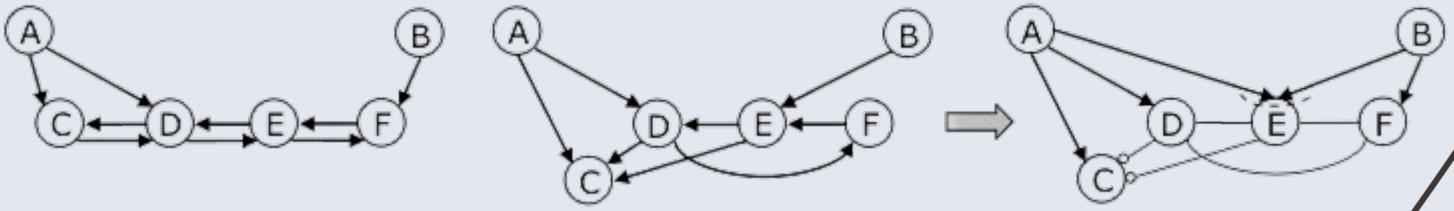
- Cyclic Equivalence Theorem (CET) conditions:
  - Same (virtual) adjacencies
  - Same unshielded conductors
  - Same unshielded perfect nonconductors
  - M.E. conductors (A, B, C) and (X, Y, Z) on some uncovered itinerary in G1 iff they are also M.E. conductors on some uncovered itinerary in G2

Introduction

- Cyclic Equivalence Theorem (CET) conditions (cont'd):
    - Unshielded imperfect nonconductors (A, X, B) and (A, Y, B) of G1 and G2, then X is an ancestor of Y in G1 iff X is an ancestor of Y in G2
    - M.E. conductors on uncovered itinerary (A, B, C) and (X, Y, Z) and unshielded imperfect nonconductor (A, M, Z), then M is a descendant of B in G1 iff M is a descendant of B in G2

Introduction

- Cyclic Partial Ancestral Graph (CPAG)
  - G1 and G2 are Markov equivalent iff the algorithm (Richardson, 1996) outputs the same CPAG for both graphs
    - Algorithm is $O(n^7)$ and d-separation complete



Ancestral Perspective

- Creating a simpler set of rules that directly correspond to features in the CPAG
- Cyclic Maximal Ancestral Graph (CMAG)
    - Edge between every distinct pair of vertices iff they can't be d-separated by any subset of vertices
    - Tail mark X –* Y iff there is a directed path from X to Y in G; otherwise, arrowhead mark X ←* Y
    - Every v-structure X → Z ← Y where Z is not a descendant of a common child of X and Y in G is represented by a (dashed) underline X <u>→ Z ←</u> Y (virtual v-structures)

Ancestral Perspective

- Mapping of elements in CET to ancestral counterpart
  - Virtual adjacencies → edges
  - Itineraries → paths
  - Unshielded conductors → standard unshielded noncolliders
  - Unshielded nonconductors → v-structures
  - Unshielded imperfect nonconductors → virtual v-structures

Ancestral Perspective

- CMAG M has a *u-structure,* quadruple of distinct nodes (X, Z, Z', Y) if X → Z and Z' ← Y are in M, Z' is part of SCC(Z) and there is an uncovered path (X, Z...Z', Y) in M where all intermediate nodes are also in SCC(Z)
    - Similar to a v-structure but central collider replaced by uncovered path through SCC
    - Not explicitly recorded in CMAG
- Triple of distinct nodes (X, Z, Y) is *virtual collider triple* iff (X, Z, Y) is virtual v-structure or there is some Z' in SCC(Z) such that either (X, Z, Z', Y) or (X, Z', Z, Y) is u-structure

Ancestral Perspective

- Restating CET in terms of CMAGs
  - CMAGs M1 and M2 are Markov equivalent iff
    - Same skeleton
    - Same v-structures
    - Same virtual collider triples
    - If (A, B, C) is a virtual collider triple and (A, D, C) is virtual v-structure,, B is an ancestor of D in M1 iff B is an ancestor of D in M2

Markov Equivalence

Algorithm 1 Cyclic-Graph-to-CMAG

Input: directed cyclic graph $\mathcal{G}$ over nodes $\mathbf{V}$
Output: CMAG $\mathcal{M}$, SCCs,
$SCC \leftarrow Get\_StronglyConnComps(\mathcal{G})$
part 1: CMAG rules (i) + (ii)
for all $X \in \mathbf{V}$ **do**
    $\mathbf{Z} \leftarrow pa_{\mathcal{G}}(X)$
    $\mathbf{Z}_{cyc} \leftarrow \mathbf{Z} \cap SCC(X)$
    $\mathbf{Z}_{acy} \leftarrow \mathbf{Z} \setminus \mathbf{Z}_{cyc}$
    add all arcs $\mathbf{Z}_{acy} \longrightarrow \mathbf{Z}_{cyc} \cup \{X\}$ to $\mathcal{M}$
    add all undirected edges $\mathbf{Z}_{cyc} — \mathbf{Z}_{cyc} \cup \{X\}$ to $\mathcal{M}$
end for
part 2: CMAG rule (iii)
for all $X \in \mathbf{V} : |SCC(X)| \geq 2$ **do**
    $\mathbf{Z} \leftarrow pa_{\mathcal{M}}(X)$
    for all non-adjacent pairs $\{Z_i, Z_j\} \subseteq \mathbf{Z}$ **do**
        if $\{Z_i, Z_j\} \nsubseteq adj_{\mathcal{G}}(X)$ **then**
            if $X \notin de_{\mathcal{G}}(ch_{\mathcal{G}}(Z_i) \cap ch_{\mathcal{G}}(Z_j))$ **then**
                mark virtual v-structure $\langle Z_i, X, Z_j \rangle$ in $\mathcal{M}$
    end for
end for

- Derive consistent CPAG that uniquely defines equivalence class of cyclic directed graph without d-separation tests via *intermediate CMAG representation*

Markov Equivalence

Algorithm 2 Graph-to-CPAG

Input: directed cyclic graph $\mathcal{G}$ over nodes $\mathbf{V}$,
Output: CPAG $\mathcal{P}$,
$(\mathcal{M}, SCC) \leftarrow$ Cyclic-Graph-to-CMAG($\mathcal{G}$)
part 1: new-CET rules (i)-(iii)
$\mathcal{P} \leftarrow$ skeleton of $\mathcal{M}$ with all $\circ\!-\!\circ$ edges
$\mathcal{P} \leftarrow$ copy all (virtual) v-structures from $\mathcal{M}$
for all $X \circ\!-\!\circ Z$ in $\mathcal{P}$, $X \longrightarrow Z$ in $\mathcal{M}$, $|SCC(Z)| \geq 2$ **do**
  if $\exists \langle X, Z, Z', Y \rangle$ as u-structure in $\mathcal{M}$ **then**
    orient $X \longrightarrow Z$ in $\mathcal{P}$ {*Lemma* 2}
end for
part 2: new-CET rule (iv)
for all virtual v-structures $\langle X, Z, Y \rangle$ in $\mathcal{P}$ **do**
  for all not fully oriented edges $Z \ast\!-\!\ast W$ in $\mathcal{P}$ **do**
    if $\langle X, W, Y \rangle$ is virtual collider triple **then**
      copy edge $Z \ast\!-\!\ast W$ from $\mathcal{M}$ to $\mathcal{P}$
    end if
  end for
end for

- *Derive consistent CPAG that uniquely defines equivalence class of cyclic directed graph* without d-separation tests via intermediate CMAG representation
- G1 is Markov equivalent to G2 iff CPAG(G1) = CPAG(G2)

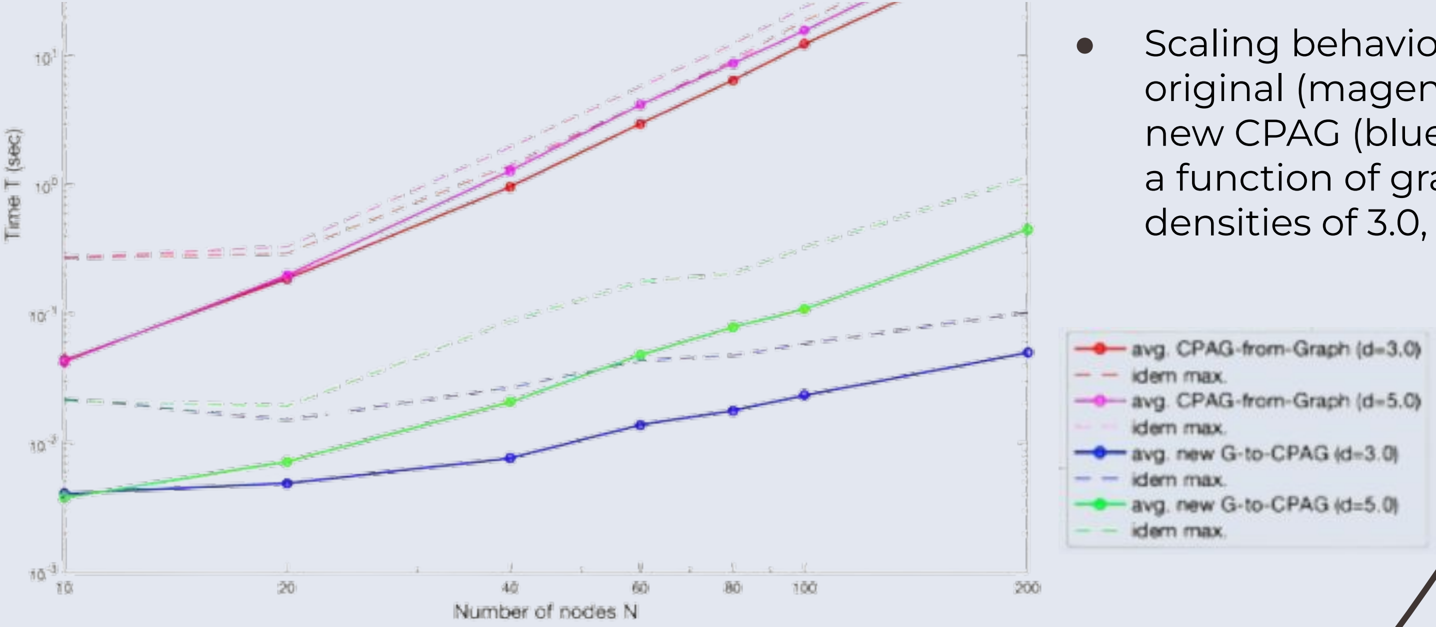
Markov Equivalence

- O(N+Nd) to find SCC, loop over N vertices comparing d^2 parents
  - $O(Nd^2)$
- Initializing skeleton and virtual v-structures
  - $O(Nd^2)$
- Loop over O(Nd) edges and establish connectedness in O(Nd) steps for u-structures
  - $O(N^2d^2)$
- Loop over v-structures considering links to d other edges while testing for connectedness
  - $O(N^2d^3)$

Markov Equivalence

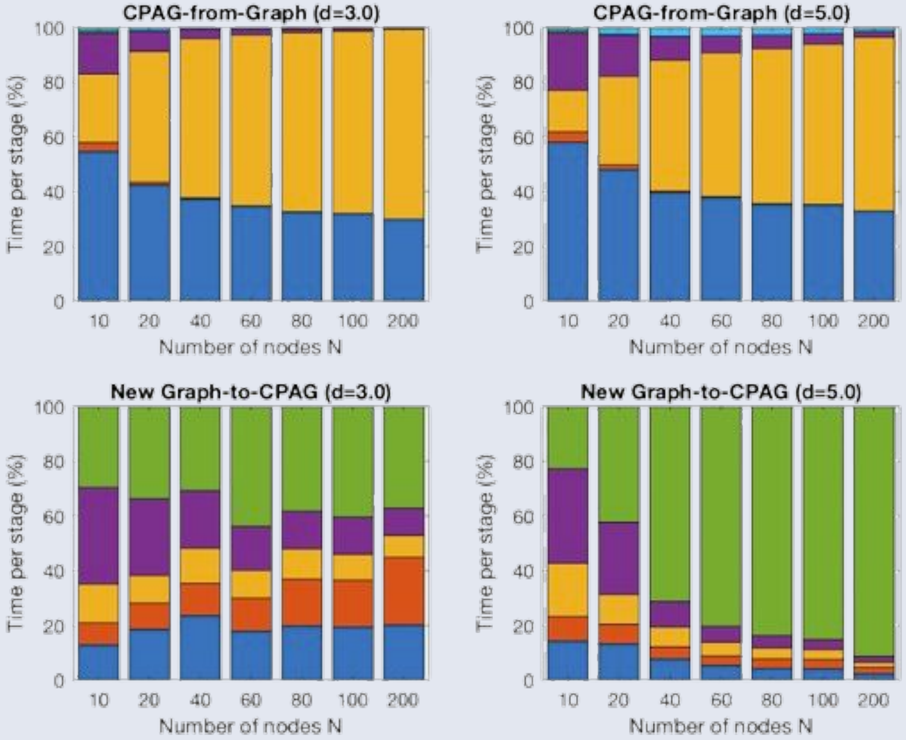
- Overall time complexity is $O(N^2d^3)$ or $O(N^5)$ worst case arbitrary density
    - Significant improvement over the previous $O(N^7)$

Evaluation



- Scaling behavior of original (magenta) and new CPAG (blue/green) as a function of graph size for densities of 3.0, 5.0

Evaluation



- Time per stage for original vs new CPAG from graph algorithms

Conclusion

- Faster and more efficient procedure to obtain CPAG from arbitrary directed graph
  - Used to establish Markov equivalence

Thanks!