SGS algorithm

Throughout the exercise we will consider *only* the following conditional independences as true (and all other possible combinations as conditional depedences):

$X_1 \perp X_4$
$X_2 \perp X_4$
$X_1 \perp X_3 | X_2$
$X_1 \perp X_4 | X_2$
$X_2 \perp X_4 | X_1$
$X_1 \perp X_4 | X_2, X_3$

1 | 1 point

Skeleton learning

What is the skeleton of the graph over nodes {1,2,3,4} that one can learn using the above conditional independences (and no other independence)?

I'm copying the conditional independence here for convenience:

$$X_1 \perp X_4$$
$$X_2 \perp X_4$$
$$X_1 \perp X_3 | X_2$$
$$X_1 \perp X_4 | X_2$$
$$X_2 \perp X_4 | X_1$$
$$X_1 \perp X_4 | X_2, X_3$$

Here is the pseudo-code for the skeleton learning phase

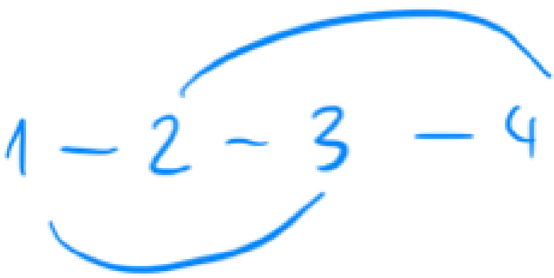
1. Start with **completely connected undirected graph** U
2. For each pair $i, j \in \mathbf{V}, i \neq j$, and for any subset $\mathbf{S} \subseteq \mathbf{V} \backslash \{i, j\}$
   - Check **if** $X_i \perp\!\!\!\perp X_j | X_{\mathbf{S}}$ for any $\mathbf{S}$ in data
     - If this is true, by faithfulness $i \perp_G j | \mathbf{S}$, so we can **remove** $i - j$ **in** U

- [ ] 

.

- [ ] 

.

- [ ] 

.

2 1 point

Given the correct skeleton from the question above, what are the unshielded triples in the skeleton (note that the definition is the same for undirected graphs):

- A triple of nodes (i, j, k) in a DAG G is a **an unshielded triple** if $i - j, j - k$ and **i is not adjacent to k**, i.e. $i \not- k$, in G

- ( ) (1,3,4)
- ( ) (1,2,3) and (2,3,4)
- ( ) none
- ( ) (1,4,3) and (4,3,2)

3 1 point

Which are the v-structures that we can determine using the following rules and the original conditional independences? Note that if a conditional independences is not written in the original list, then the variables are dependent.

$X_1 \perp X_4$
$X_2 \perp X_4$
$X_1 \perp X_3 | X_2$
$X_1 \perp X_4 | X_2$
$X_2 \perp X_4 | X_1$
$X_1 \perp X_4 | X_2, X_3$

Here are the steps to determine v-structures:

1. Start from the skeleton U from previous step
2. For each unshielded triple (i, j, k) in U, i.e. $i - j, j - k$ **and** $i \not- k$ **in** U
   - For all $\mathbf{S} \subseteq \mathbf{V} \setminus \{i, j, k\}$ check if $X_i \not\perp\!\!\!\perp X_k | X_j \cup X_\mathbf{S}$ in data
     - If this is true, $i \rightarrow j \leftarrow k$ **is a v-structure**

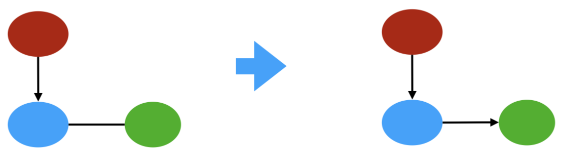
- ( ) (1,4,3)
- ( ) (1,2,3)
- ( ) (2,3,4)
- ( ) none

4 1 point

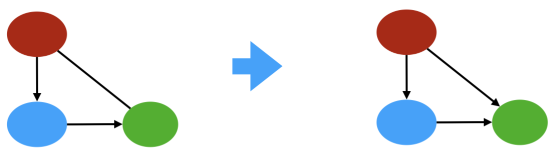
Consider the graph you have obtained up to now, after phase 1: skeleton learning, and phase 2: determining the v-structures.

Given this graph in phase 3 we can orient one or more edges by using the acyclicity or "no new v-structures" constraint, **true or false?**

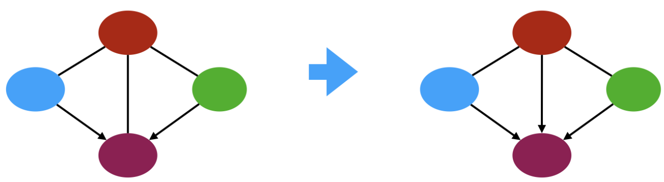
Hint: you can just reason about it directly, but if you want some extra help, here are the relevant Meek's rules:



Otherwise v-structure



Otherwise cycle



Otherwise cycle or v-structure

- True
- False

5 1 point

In the final CPDAG (after all phases) is fully oriented (there are no undirected edges, only directed ones)

- True
- False

6 1 point

In the true causal graph, what is the relationship between 1 and 2 (based on what you can learn from the CPDAG)?

We can represent the skeleton and the orientations (edge marks) all DAGs in a Markov equivalence class (MEC) have in common with a **Complete Partially Directed Acyclic Graph (CPDAG):**

- We have a directed edge $i \to j$ if all DAGs in the MEC have $i \to j$
- We have an undirected edge $i - j$ if some DAGs in the MEC have $i \to j$ and others have $j \to i$

- 1 -> 2 (and there are no other options)
- 1 <- 2 (and there are no other options)
- It could be either 1 -> 2 or 1 <- 2, we have no way of knowing just from these data.
- 1 is not adjacent to 2