Class Query 1

Based on one of the questions raised in class, ma'am clarified that:

$$X \perp\!\!\!\perp \{Y_1, \ldots, Y_k\} \mid Z \implies (X \perp\!\!\!\perp Y_1|Z) \ldots (X \perp\!\!\!\perp Y_k|Z)$$

However, the reverse might not be true. So, the following does not hold:

$$(X \perp\!\!\!\perp Y_1|Z) \ldots (X \perp\!\!\!\perp Y_k|Z) \implies X \perp\!\!\!\perp \{Y_1, \ldots, Y_k\} \mid Z$$

Undirected Graphical Models 2

In this model, the underlying graph which represents the probability distribution is undirected. Such a model is useful when the variables interact symmetrically, and there is no natural parent-child relationship. Some of the natural examples, which represent such a scenario include friends on a social media platform, atoms in a crystal, labelling pixels in an image, etc.
In an undirected graphical model, we define potentials over arbitrary cliques of the graph G. (Cliques are subsets of nodes of a graph which are fully connected, i.e they are complete subgraphs of a graph.) The potentials are denoted by $\psi_C(y_C)$. The first subscript C denotes the clique in consideration, and y_C denotes the assignment to the variables in the clique.
Potentials can take arbitrary non-negative values, however they cannot be considered equivalent to probabilities.
Here is how we define the joint distribution in an undirected graphical model:

$$Pr(y_1, \ldots, y_n) = \frac{1}{Z} \prod_{C \in \mathcal{C}} \psi_C(y_C)$$

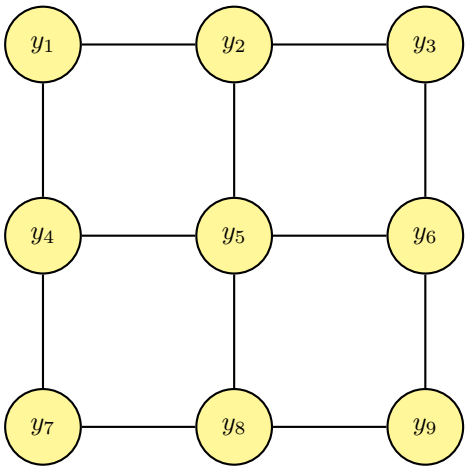
where $\mathcal{C}$ is the set of all cliques in the graph, and Z is the normalizing constant, which ensures that the sum of all probabilities is equal to 1. For the numerator, what we essentially do is for that particular assignment of variables, we take a product over all the cliques in the graph, and multiply the potentials for that assignment. Here is how we define Z:

$$Z = \sum_{y_1} \cdots \sum_{y_n} \prod_{C \in \mathcal{C}} \psi_C(y_C)$$

This expression of Z is also called the partition function. For calculating Z, what we essentially do is to sum over all possible assignments of variables, and take a product over all the cliques in the graph, and multiply the potentials for that assignment. This ensures that the sum of all probabilities is equal to 1.

2.a Example

Consider the graph G shown on the right. There are 9 binary variables $y_1, y_2, \ldots, y_9$ (each can take only two values 0 or 1).
There are two types of cliques in this graph, one is the set of all edges, and the other is the set of all nodes. The potentials for the edges are denoted by $\psi_{ij}(y_i, y_j)$, and the potentials for the nodes are denoted by $\psi_i(y_i)$.
The joint distribution for this graph can be written as:



$$Pr(y_1, \ldots, y_9) = \frac{1}{Z} \prod_{i=1}^{9} \psi_i(y_i) \prod_{(i,j) \in E} \psi_{ij}(y_i, y_j)$$

where E is the set of all edges in the graph.

For example, the value of $Pr(1, 0, \ldots, 0)$ can be calculated as:

$$Pr(1,0,\ldots,0) = \frac{1}{Z}\psi_1(1)\psi_2(0)\psi_3(0)\ldots\psi_9(0)\psi_{12}(1,0)\psi_{23}(0,0)\psi_{14}(1,0)\psi_{25}(0,0)$$
$$\psi_{36}(0,0)\psi_{45}(0,0)\psi_{56}(0,0)\psi_{47}(0,0)\psi_{58}(0,0)$$
$$\psi_{69}(0,0)\psi_{78}(0,0)\psi_{89}(0,0)$$

Z can be calculated as follows:

$$Z = \sum_{y_1}\cdots\sum_{y_9}\psi_1(y_1)\psi_2(y_2)\psi_3(y_3)\ldots\psi_9(y_9)\psi_{12}(y_1,y_2)\psi_{23}(y_2,y_3)\psi_{14}(y_1,y_4)\psi_{25}(y_2,y_5)$$
$$\psi_{36}(y_3,y_6)\psi_{45}(y_4,y_5)\psi_{56}(y_5,y_6)\psi_{47}(y_4,y_7)\psi_{58}(y_5,y_8)$$
$$\psi_{69}(y_6,y_9)\psi_{78}(y_7,y_8)\psi_{89}(y_8,y_9)$$

In this case, we need to sum over 512 possible assignments of variables, which is computationally expensive.
Another example was given. It was a K3 (complete graph with 3 nodes). If we consider only one clique (the complete graph), then the joint distribution can be written as (an example):

$$P(y_1=0,y_2=1,y_3=1) = \frac{\psi_{123}(0,1,1)}{\psi_{123}(0,0,0)+\psi_{123}(0,0,1)+\cdots+\psi_{123}(1,1,1)}$$

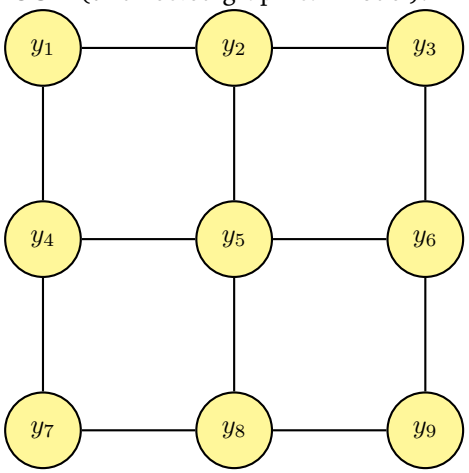
Similarly, if we consider only the three edges as the cliques, then the joint distribution can be written as:

$$P(y_1=0,y_2=1,y_3=1) = \frac{\psi_{12}(0,1)\psi_{23}(1,1)\psi_{13}(0,1)}{\psi_{12}(0,0)\psi_{23}(0,0)\psi_{13}(0,0)+\cdots+\psi_{12}(1,1)\psi_{23}(1,1)\psi_{13}(1,1)}$$

The types of cliques, on which we define potential depends on the real world clue which we have.
Also, the number of parameters needed to define potential is exponential in the number of nodes in the clique.
Therefore, computing Z is tough and for certain graphs we exploit factorization to simplify.
In general, if $|C|=k$, and $y_i \in \{1,\ldots,m\}$, then the number of potential scores we need to report will be m^k.

Conditional Independencies in an Undirected Graphical Model 3

Let $V=\{y_1,\ldots,y_n\}$.
Let distribution P be represented by an undirected graphical model G. If Z separates X and Y in G, then $X \perp\!\!\!\perp Y|Z$ in P. The set of all such CIs are called Global-CI of the UGM (undirected graphical model).



Example:

1. $y_1 \perp\!\!\!\perp y_3, y_5, y_6, y_7, y_8, y_9 | y_2, y_4$
2. $y_1 \perp\!\!\!\perp y_3 | y_2, y_4, y_5, y_6, y_7, y_8, y_9$ (the separator Z need not be minimal)
3. $y_1, y_2, y_3 \perp\!\!\!\perp y_7, y_8, y_9 | y_4, y_5, y_6$
4. $y_1 \perp\!\!\!\perp y_3 | y_2, y_4$

(Basically when we remove Z, X and Y should be disconnected in the graph.)

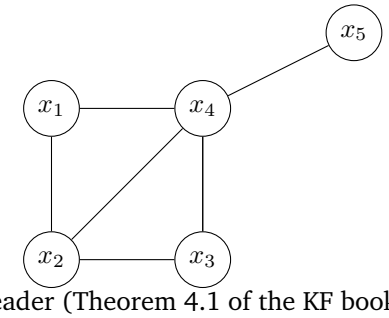
Factorization implies Global CI 4

Here is the theorem:
Let G be the undirected graph over $V = x_1,\ldots,x_n$ nodes and $P(x_1,\ldots,x_n)$ be a distribution. If P is represented by G that is, if it can be factorized as per the cliques of G, then P will also satisfy the global-CIs of G.
Factorize$(P,G) \implies$ Global-CI(P,G)

4.a Example

One of the valid options to express P is as follows:

$$P(x_1, \ldots, x_5) \propto \psi_{124}(x_1, x_2, x_4)\psi_{234}(x_2, x_3, x_4)\psi_{45}(x_4, x_5)$$

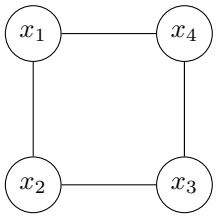


Since, P can be factorized as per the cliques of G, it will also satisfy the global-CIs of G. For example, $x_1 \perp\!\!\!\perp x_5 | x_2, x_3, x_4$ is a valid CI.

The proof of this theorem has been left as an exercise for the reader (Theorem 4.1 of the KF book).

4.b Global CI does not imply Factorization

The counter example wasn't discussed in class, but was left for the students to read through the slides. It has been taken from example 4.4 of the KF book.
Consider a distribution over 4 binary variables $P(x_1, x_2, x_3, x_4)$.
The graph G is shown on the right.
Let $P(x_1, x_2, x_3, x_4)$ be $\frac{1}{8}$ when x_1, x_2, x_3, x_4 takes values from this set: $\{0000, 1000, 1100, 1110, 1111, 0111, 0011, 0001\}$. In all other cases it is 0. It is left as an exercise for the reader to check that all four global CIs hold in the graph: $x_1 \perp\!\!\!\perp x_3 | x_2, x_4$ etc.
Now, we will look at factorization. The factors correspond to the edges in $\psi(x_1, x_2)$. Each of the four possible assignments of each factor is non-zero (as per the set of values mentioned above). But, this cannot represent the zero probability for cases like $x_1, x_2, x_3, x_4 = 0101$.

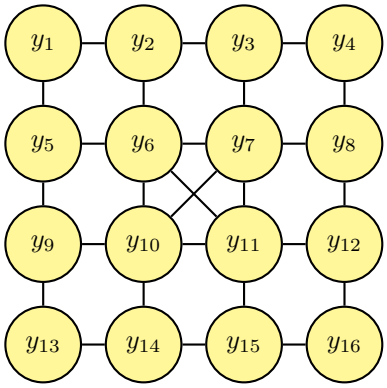


Drawing an undirected graphical model 5

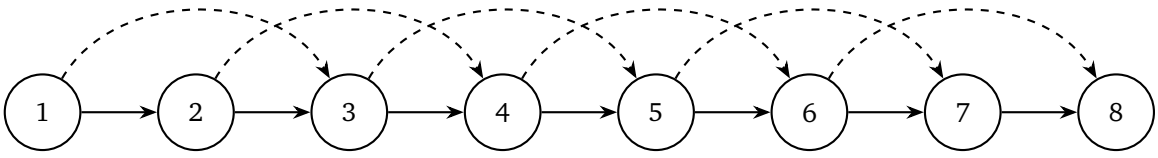
There are majorly two ways to draw an undirected graphical model:

1. **Starting from factors:** We simply connect together all variables that appear together in a factor. Here are some real-life examples:

   - Image pixels



   - Language Models from n-gram scores



     The figure above illustrates a 3-gram model, where nodes separated by a distance of 2 are connected. This ensures that contiguous groups of three words are linked, allowing the model to capture the context and meaning within each phrase.

2. **Starting from CIs**

Constructing an UGM from a positive distribution 6

Positive distribution is a distribution where all the probabilities are non-negative. We are given $P(x_1, \ldots, x_n)$ to which we can ask any CI of the form $X \perp\!\!\!\perp Y|Z$ and get a yes, no answer.
Our goal is to draw a minimal, correct UGM G to represent P. Here are the two options which we have (V denotes the set of all n variables):

- **Using pairwise CI:** For each pair of vertices x_i, x_j, if $x_i \not\perp\!\!\!\perp x_j|V - \{x_i, x_j\}$ in P, we add an edge between x_i and x_j in G. This is because for a UGM, the following is true:

  $$X \perp\!\!\!\perp Y|Z \implies X \perp\!\!\!\perp Y|Z, W$$

  The above might not be true for a bayesian network. It is true for a UGM because even after adding nodes to Z, X and Y will still be disconnected in the graph. Hence considering the entire set $V - \{x_i, x_j\}$ will also work.

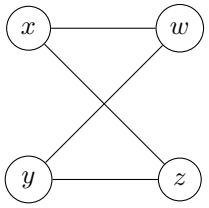
- **Using local CI:** For each node x_i, we need to find the smallest subset U such that $x_i \perp\!\!\!\perp V - U - \{x_i\}|U$ in P. After this, we make the nodes in U, the neighbours of x_i in G.

6.a Example

We are given the positive distribution $P(x, y, z, w)$ for which the following CIs hold:

- $x \perp\!\!\!\perp y|z, w$
- $z \perp\!\!\!\perp w|x, y$

Let us first apply pairwise-CI algorithm to draw the graph G. We need to iterate over all the pairs of variables and check if the CI holds. If it does not hold, we add an edge between the two variables. So, for the edge x, y, they are independent given the other two variables ($x \perp\!\!\!\perp y|z, w$), so we do not add an edge. Similarly, for the edge z, w, they are independent given the other two variables ($z \perp\!\!\!\perp w|x, y$), so we do not add an edge. For all other edges, we add an edge (because the CI does not hold). So, the graph G will look like this:



Now, let us try the same problem with the local-CI algorithm. We need to find U for each of the four nodes. For x, U is $\{z, w\}$ because $x \perp\!\!\!\perp V - U - \{x\}|U$ holds. For y, U is $\{z, w\}$, for z, U is $\{x, y\}$, for w, U is $\{x, y\}$. We need to connect each node to the nodes in U. So, we will get the same graph as above.

6.b Markov Blanket

UGMs are also called Markov Random Fields.
The Markov Blanket of a variable x_i, $MB(x_i)$ is the smallest subset of variables V that makes x_i CI of others given the Markov Blanket. This is essentially the U, which we used above.

$$x_i \perp\!\!\!\perp V - MB(x_i) - \{x_i\}|MB(x_i)$$

Also, one of the theorems says that $MB(x_i)$ is always unique for a positive distribution. The proof of this, has been left as a self-reading exercise (given in the slides).

6.c Hammersly Clifford Theorem

If a positive distribution $P(x_1, \ldots, x_n)$ conforms to the pairwise CIs of a UDGM G, then it can be factorized as per the cliques of G. This is the Hammersly Clifford Theorem.

$$P(x_1, \ldots, x_n) \propto \prod_{C \in \mathcal{C}} \psi_C(y_C)$$

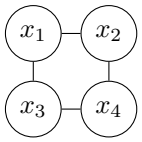
The proof of this theorem has been left as an exercise for the reader (Theorem 4.8 of the KF book).

Summary 7

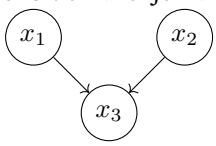
Let P be a distribution and H be an undirected graph of the same set of nodes.
$Factorize(P,H) \implies Global-CI(P,H) \implies Local-CI(P,H) \implies Pairwise-CI(P,H)$
But only for positive distributions, we have the following:
$Pairwise-CI(P,H) \implies Factorize(P,H)$

Lessons learned so far 8

- d-separation test identifies the complete set of global CIs. These can also be derived using the CIs enlisted using the local CI rule and various conditional independency axioms.
- Potentials in undirected graphs cannot be interpreted as conditional probabilities unlike in directed graphs.
- To get the complete set of CIs we use the d-separation algorithm in a Bayesian Network whereas in an undirected model, we use graph-separability.
- Some probability distributions can be expressed using only one of the directed and undirected graphical models.
  Example 1: Consider the joint distribution $P(x_1,x_2,x_3,x_4)$ for which the following CIs hold:
  - $x_1 \perp\!\!\!\perp x_4 | \{x_2,x_3\}$
  - $x_2 \perp\!\!\!\perp x_3 | \{x_1,x_4\}$



  This distribution cannot be represented as a BN
  Example 2: Consider the joint distribution $P(x_1,x_2,x_3)$ which can be expressed only using BN:



Converting BN to MRFs 9

9.a Markov Blanket of a node in a BN

The Markov Blanket of a variable x_i, $MB(x_i)$ in a BN is such that

$$x_i \perp\!\!\!\perp V - MB(x_i) - \{x_i\} | MB(x_i)$$

$$MB(x_i) = \{Parents(x_i) \cup Children(x_i) \cup Spouses(x_i)\}$$

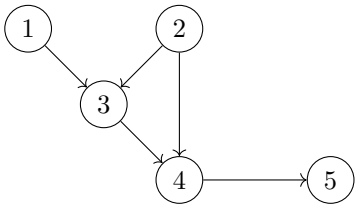
Using the d-separation algorithm, we can show that x_i will be independent of the rest of the network given its Markov blanket.

9.b Markov Blanket Algorithm

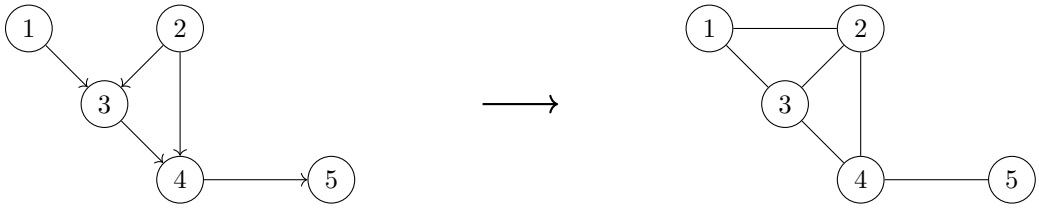
Thus in the MRF corresponding to the BN, every node must be directly connected to each of the nodes in its Markov blanket through an edge. The MRF is obtained by moralizing a BN (connecting spouses in immorality) and converting all directed edges to undirected edges forming a UGM.

9.c Example

Consider the following BN:



- $MB(1) = \{2, 3\}$
- $MB(2) = \{1, 3, 4\}$
- $MB(3) = \{1, 2, 4\}$
- $MB(4) = \{2, 3, 5\}$
- $MB(5) = \{4\}$



BN MRF

9.d Remarks

- Conversion of a minimal Bayesian network doesn't always give a minimal UGM.
- The MB algorithm is correct even if the BN represents a non-positive distribution.
- **Intersection property:**

  $$X \perp\!\!\!\perp Y|Z \text{ and } X \perp\!\!\!\perp Z|Y \implies X \perp\!\!\!\perp \{Y, Z\}$$

  This property holds in a BN.
  MB algorithm is correct for all distributions that satisfy the intersection property.

Which BNs have perfect MRFs 10

A BN which has no immoralities, will not require the addition of any new edges when converting into an MRF. Thus all CIs enumerated in such a BN using d-separation can be implied using graph separation on the corresponding MRF. We refer to the MRFs of such networks as Perfect MRFs.

Converting MRFs to BN 11

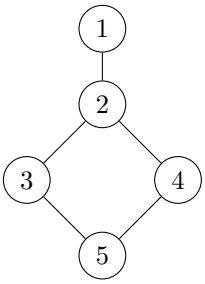
Start with a random order of variables. Use Global-CI on MRF to answer the conditional independence queries, utilising the BN construction algorithm for the conversion.

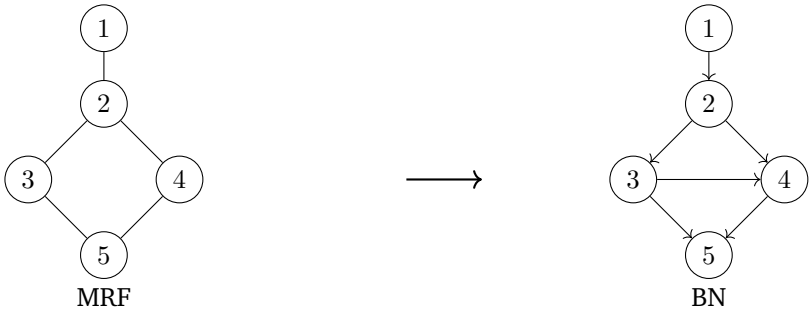
11.a Example

For the following MRF:
Consider the ordering of variables: x_1, x_2, x_3, x_4, x_5

11.b Remarks

- Different ordering of variables will lead to a different BN for a given MRF.
- We cannot use the PC algorithm because there is no guarantee that a given UGM can be perfectly represented by a BN.
- Every time we add an extra edge in the conversion, we lose some CI and thus obtain an imperfect BN.





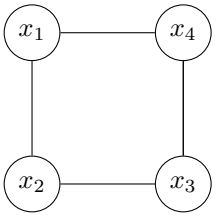
Which MRFs have perfect BNs 12

12.a Chordal or Triangulated Graphs

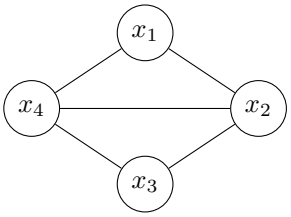
A graph is chordal if it has no minimal cycle of length ≥ 4.

Here a minimal cycle is a cycle without a shortcut i.e, it is a cycle $x_1, x_2 \dots x_n, x_1$ such that there is no edge connecting any of the cycle's vertices apart from the edges that make up the cycle.

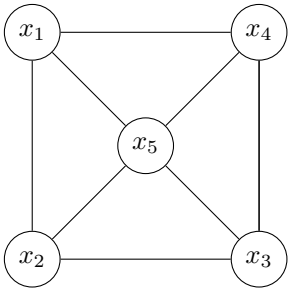
Consider the following examples



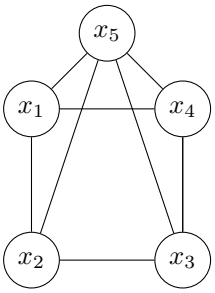
Not a chordal graph, since it contains a minimal cycle of length 4.



Chordal graph, since it contains minimal cycles of only length 3. $x_1 - x_2 - x_3 - x_4 - x_1$ is not a minimal cycle because there is a shortcut between $x_2 - x_4$.



Not a chordal graph, since it contains a minimal cycle of length 4 ($x_1 - x_2 - x_3 - x_4 - x_1$). The paths via x_5 are not shortcuts because they consist of more than a single edge.



The same graph as the one on the left, drawn for better visualization. Now we clearly see that the 4-cycle is minimal and doesn't contain a shortcut.

12.b Perfect conversion of MRF to BN

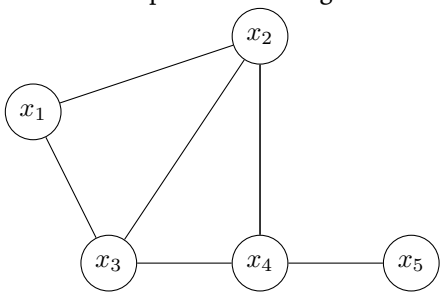
Theorem: An MRF can be converted perfectly into a BN iff it is chordal.

(The proof of this theorem is left as an exercise using theorems 4.11 and 4.13 of the KF book).

This conversion can be done using the PC algorithm since we know the BN is perfectly constructed.

Chordal Graphs

Theorem: Every triangulated graph is either complete or has atleast two non-adjacent simplicial vertices. A vertex is simplicial if its neighbours form a complete graph.



In the graph on the left, x_1 and x_5 are simplicial vertices. To see this, consider the neighbours of x_1 i.e., $\{x_2, x_3\}$. These vertices form a complete $K2$ graph, so x_1 is simplicial. x_5 has only one neighbour, x_4, which trivially forms a complete graph, making x_5 also simplicial. The other 3 vertices are not simplicial; this can be checked using their neighbour sets.

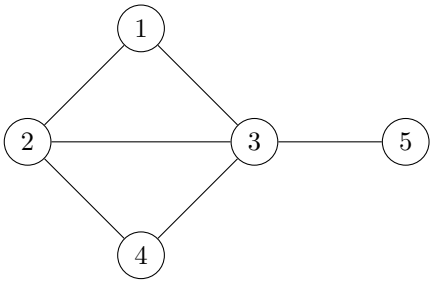
The proof of this theorem is out of the scope of the course and can be found online.

12.c Algorithm to convert UGM to BN

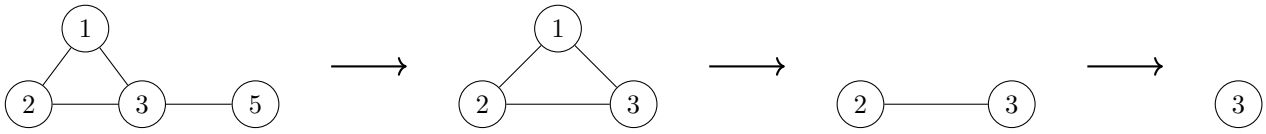
Algorithm 1 Conversion of UGM to BN

Input: Undirected graph H with n nodes
Output: Directed graph G over the same n nodes
for $i : 1 \rightarrow n$ **do**
    $x_i \leftarrow$ a simplicial node in H
    Remove x_i and its associated edges from H. H is now a reduced triangulated graph.
end for
We now have the ordering $x_1, x_2 \dots x_n$
Initialize empty graph $G \leftarrow \phi$
for $i : n \rightarrow 1$ **do**
    Add node x_i to G
    Draw an edge from x_j to x_i if x_j is connected to x_i in H and $j > i$
end for
return G

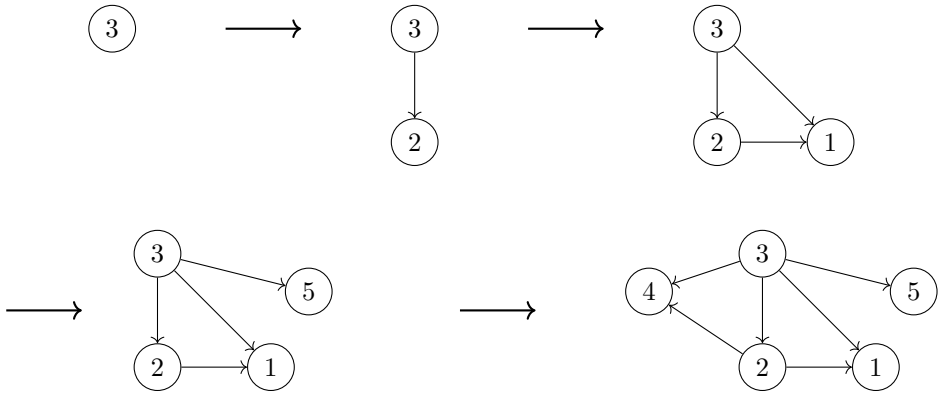
Consider the following example to illustrate how the algorithm works



Following the steps of the algorithm, we remove the simplicial vertices one by one, in the order $4, 5, 1, 2, 3$ and H is transformed as shown below.



Now starting from an empty graph G, we add the nodes in the order $3, 2, 1, 5, 4$ as per the algorithm.



The final graph G we obtain above is a perfect Bayesian Network representation of the undirected graph H. It has no immoralities (if it did, it could not be converted back to UGM without loss of information). Since the construction is perfect, we can go back and forth between BN and UGM while maintaining the set of CIs.