# Lecture Notes #6

## d-seperation:

→ for sets of nodes $X, Y, E$, when is

$$P(Y|E, X) = P(Y|E)$$
$$P(X|E, Y) = P(X|E)$$
$$P(X, Y|E) = P(X|E)\,P(Y|E)$$
?

→ True if all paths from $X$ to $Y$ are "blocked"

→ A path is blocked if it has a node $Z$ that satisfies either

i) $Z \in E$ $\quad \rightarrow Z \rightarrow$ $\quad$ intervening cause

ii) $Z \in E$ $\quad \leftarrow Z \rightarrow$ $\quad$ Common Cause

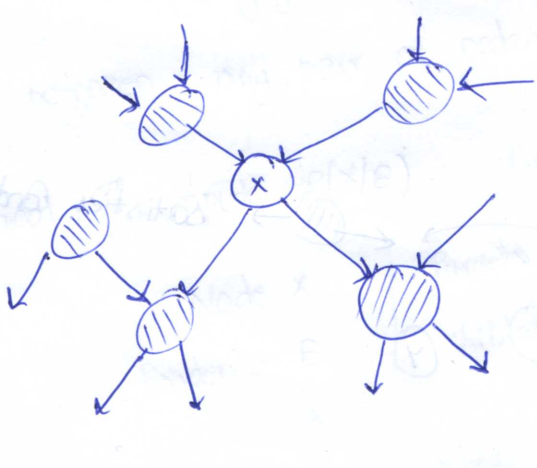
OR) iii) $Z \notin E$ $\quad \rightarrow Z \leftarrow$ $\quad$ no observed common effect.

decendents$(Z) \notin E$

## Markov Blanket:

Markov Blanket $B_X$ of node $X$ Consists of parents, children and "spouses" (other parents of children).

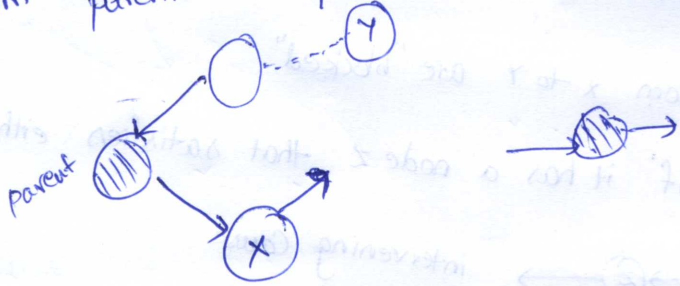
**THEOREM**: $P(X|B_X, Y) = P(X|B_X)$ where $Y \notin \{X, B_X\}$



All the other nodes, no matter wherever they appear, ~~it~~ will be independent of $x$, if conditioned on $B_x$

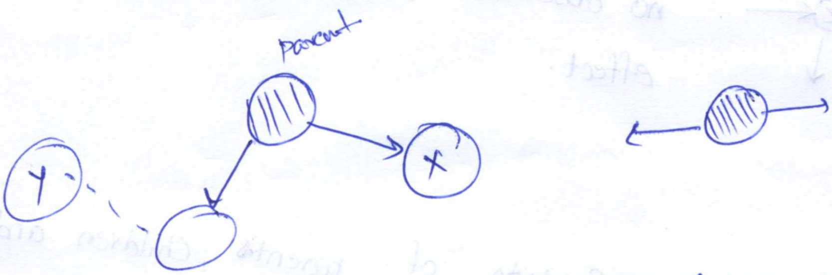
There are 5 different cases to consider. The path from $x$ to $Y$ may pass

i) from parent of parent of $x$



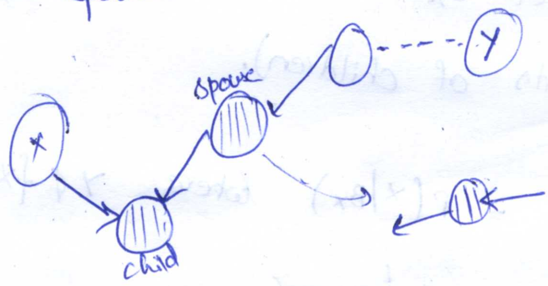

satisfies condition ①

ii) from child of parent of $x$



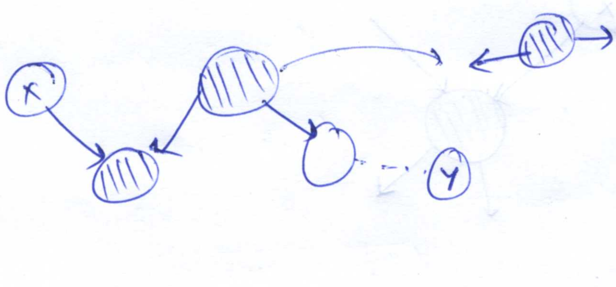

satisfies condition ②

iii) from parent of spouse of $x$



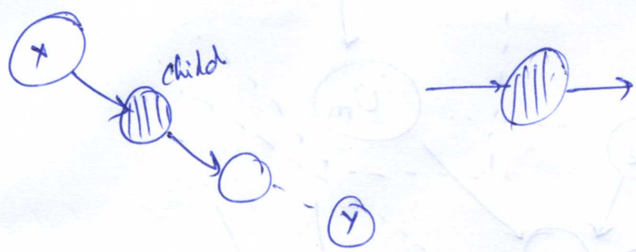

satisfies condition ①

iv) from child of spouse of $x$




satisfies condition ②

v) from child of Child of x




satisfies Condition ①.

Thus all paths from x to y are blocked.

$$\Rightarrow \quad P(x|B_x, y) = P(x|B_x)$$

## INFERENCE

→ Problem

E = set of Evidence Nodes.

Q = set of query nodes.

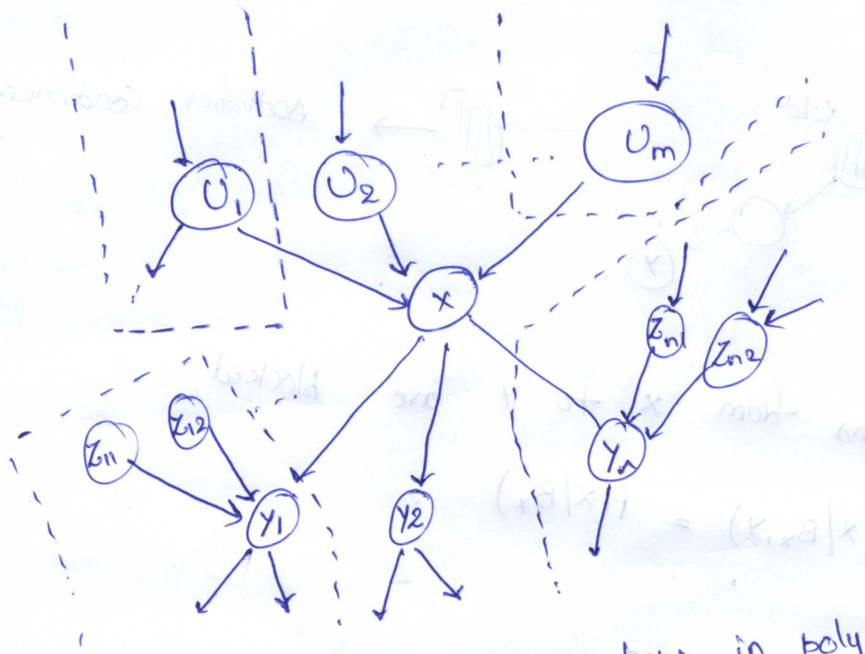
How to Compute posterior probabilities $P(Q|E)$

Q: When can we perform inference efficiently? (polynomial time in size of DAG and CPTs)

A: Polytrees
↳ Singly Connected network; at most one undirected path between any pair of nodes; no loops.

→ Goal: To Compute $P(X/E)$

Node X | Parents $U_i$ spouses $Z_{jk}$
Evidence E | children $Y_j$ ↳ kth parent of jth Child.




claim: Boxes don't overlap — no loops in polytree!

Type of evidence:

$E_x^+$ — upstream evidence "above" x

Connected to x through its parents.

$E_x^-$ — downstream evidence "below" x,

Connected to x through children

$E = E_x^+ \cup E_x^-$ (Assume $x \notin E$, otherwise trivial)

Inference in polytree:

$$P(x|E) = P(x|E_x^-, E_x^+) \quad [\because E = E_x^- \cup E_x^+]$$

$$= \cancel{\frac{P(E_x^- | x, E_x^+)\, P(E_x^-|x)}{P(E_x^+|x)}}$$

Conditional Bays Rule $P(x|y,z) = \frac{P(y|x,z)P(x|z)}{P(y|z)}$

$$= \frac{P(x, E_x^- | E_x^+)}{P(E_x^- | E_x^+)}$$ Conditionalized product rule

$$= \frac{P(x, E_x^- | E_x^+)}{\sum_x P(X=x, E_x^- | E_x^+)}$$ Marginalisation

↳ Denominator is same Computation as numerator, but summed over $x$

→ so focus on numerator.

$$P(x, E_x^- / E_x^+) = \underline{P(x/E_x^+) \quad P(E_x^- / x, E_x^+)}$$ Conditionalised product rule

$$= \underbrace{P(x/E_x^+)}_{\substack{\downarrow \\ \text{upstream recursion}}} \quad \underbrace{P(E_x^- / x)}_{\substack{\downarrow \\ \text{downstream recursion}}}$$ ( d-seperation Case 1 $E_x^-$, $E_x^+$ are Conditionally ind, given $x$ )

goal!

"upstream" Recursion!

$$P(x/E_x^+) = \sum P(x, \vec{U} = \vec{a} \mid E_x^+)$$ { marginalisation over parents

$$= \sum_{\vec{u}} P(\vec{U} = \vec{u} \mid E_x^+) \; P(x / \vec{U} = \vec{u}, E_x^+)$$ { Conditionalised product rule

$$= \sum_{\vec{u}} P(\vec{U} = \vec{u} \mid E_x^+) \; P(x / \vec{U} = \vec{u})$$ d-separation Case ① or ②

$$= \sum_{\vec{u}} \left[ P(x/\vec{U} = \vec{a}) \prod_{i=1}^{m} P(U_i / E_x^+) \right]$$ d-separation Case③ { x is the unobserved common effect }

Let $E_{U_i \setminus x}$ denote evidence inside ith parent's box Connected to $U_i$ via path that doesn't ~~Common effec~~ Contain $x$ )

$$P(x \mid E_x^+) = \sum_{\vec{u}} P(x \mid \vec{U} = \vec{u}) \prod_{i=1}^{m} P(U_i \mid E_{U_i \setminus x})$$

→ Downstream recursion

How to Compute $P(E_x^- | x)$ ?

possible but slightly more Complicated

→ Termination Conditions.

- root node (no parents)
- leaf node (no children)
- evidence node (trivial inference

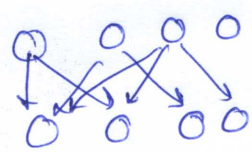
→ Running Time:

- Linear n # nodes of BN
- Linear in size of CPTs because we must sum over parents

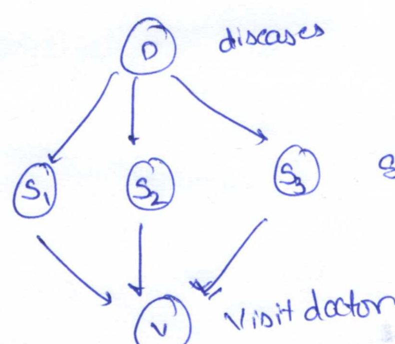
$$\Sigma P(\vec{U} = \vec{u}, \ldots)$$

Loopy BNs, how to perform inference ?

Ex: medical diagnosis.




2-layer network.




Ex: Simpler example

Turn loopy-BN into polytree ←


→ one approach node clustering

→ merge node $S_1, S_2, S_3$ to form polytree

→ How to merge ? apply polytree algorithm