POLITECNICO
MILANO 1863

Soft Computing – Probabilistic Reasoning

- Inference in Bayesian Networks-

Prof. Matteo Matteucci – *matteo.matteucci@polimi.it*

Course Syllabus (Tentative)

Probability basics (fast and furious)

- Frequentists vs Bayesians
- Joint and Naive Distributions

Probabilistic graphical models

- Directed graphical models (Bayesian Networks)
- Conditional independence and d-separation
- Inference in directed graphical models

Dynamical graphical models

- Markov chains
- Hidden Markov models

Learning directed graphical models …

Inference in Bayesian Networks

Bayesian Networks exist basically to make inference:

- A graphical model specifies a complete joint probability distribution
- Inference made via marginalization (i.e., summing out irrelevant variables)
- Brute force enumeration has $O(|X_i|^N)$ complexity

General inference in Bayesian Networks takes exponential time

Lots of work have been done to overcome this issue:

- Variable Elimination [Exact]
- Belief Propagation (message passing / sum product algo) [Exact]
- Junction Trees (will see only how to build) [Exact]
- Loopy Belief Propagation [Approximate]
- Sampling based methods [Approximate]

Soft Computing – Probabilistic Reasoning

- Variable Elimination-

Prof. Matteo Matteucci – *matteo.matteucci@polimi.it*

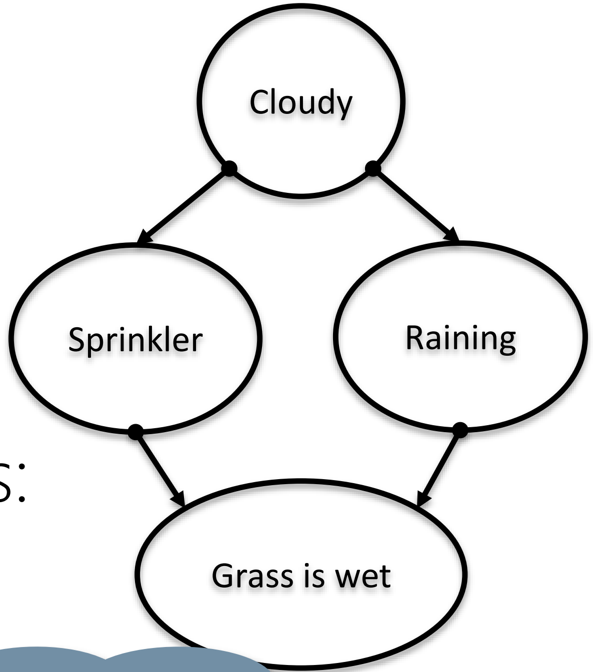
Variable Elimination

We can use the factored representation of Joint Probability to do marginalisation efficiently by "pushing sums in" as far as possible:

$$P(W) = \sum_C \sum_S \sum_R P(C,S,R,W)$$

$$= \sum_C \sum_S \sum_R P(W|S,R)P(S|C)P(R|C)P(C)$$

$$= \sum_C P(C) \sum_S P(S|C) \sum_R P(W|S,R)P(R|C)$$



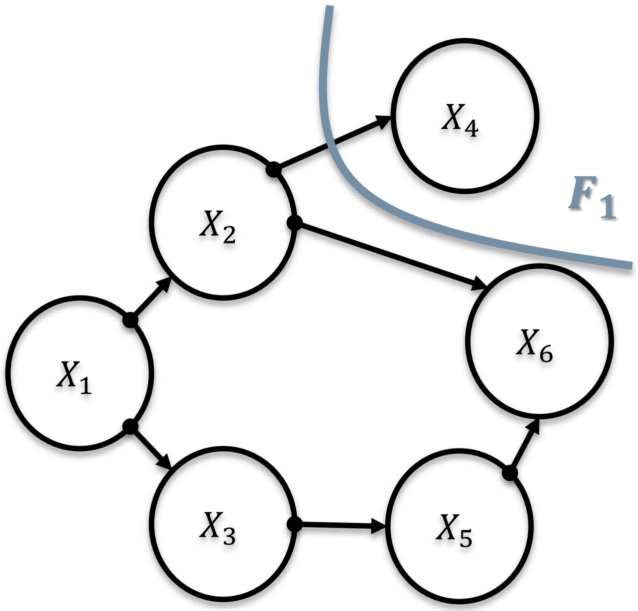
As we perform innermost sums we create new terms:

- $\mu_1(C,W,S) = \sum_R P(W|S,R)P(R|C)$
- $\mu_2(C,W) = \sum_S P(S|C)\mu_1(C,W,S)$
- $P(W) = \sum_C P(C)\mu_2(C,W)$

Bounded by the size of the largest term

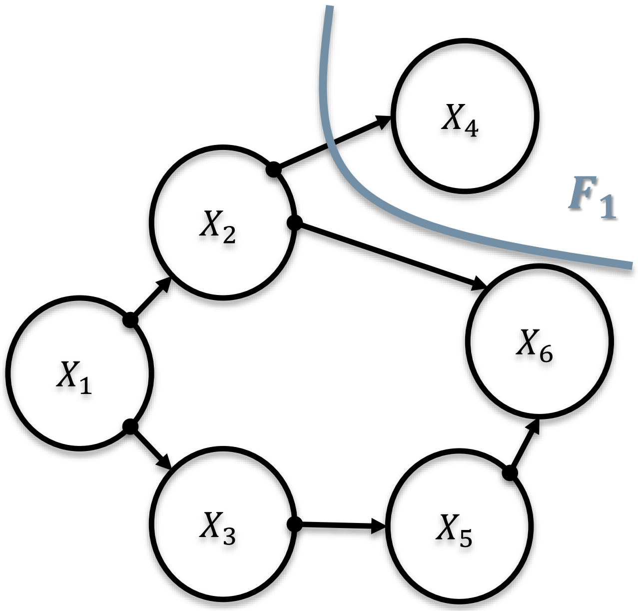
Variable Elimination Example

$$P(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3}\sum_{X_4}\sum_{X_6} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_4|X_2)P(X_5|X_3)P(X_6|X_5,X_2)$$



Variable Elimination Example

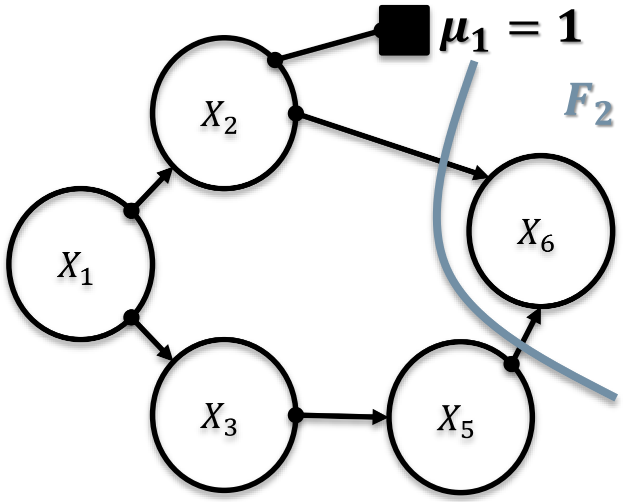
$$P(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3}\sum_{X_6} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_5|X_3)P(X_6|X_5,X_2)\sum_{X_4}\underbrace{P(X_4|X_2)}_{\boldsymbol{F_1(X_2,X_4)}}$$



Variable Elimination Example

$$P(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3}\sum_{X_6} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_5|X_3)P(X_6|X_5,X_2)\sum_{X_4}\underbrace{P(X_4|X_2)}_{\boldsymbol{F_1(X_2,X_4)}}$$

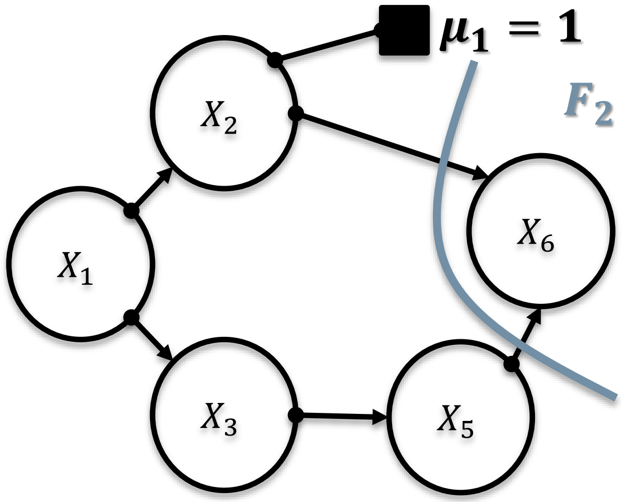
$$P(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3}\sum_{X_6} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_5|X_3)P(X_6|X_5,X_2)\mu_1(X_2)$$



Variable Elimination Example

$$P(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3}\sum_{X_6} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_5|X_3)P(X_6|X_5,X_2)\sum_{X_4}\underbrace{P(X_4|X_2)}_{\boldsymbol{F_1(X_2,X_4)}}$$

$$PP(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_5|X_3)\mu_1(X_2)\sum_{X_6}\underbrace{P(X_6|X_5,X_2)}_{\boldsymbol{F_2(X_6,X_2,X_5)}}$$

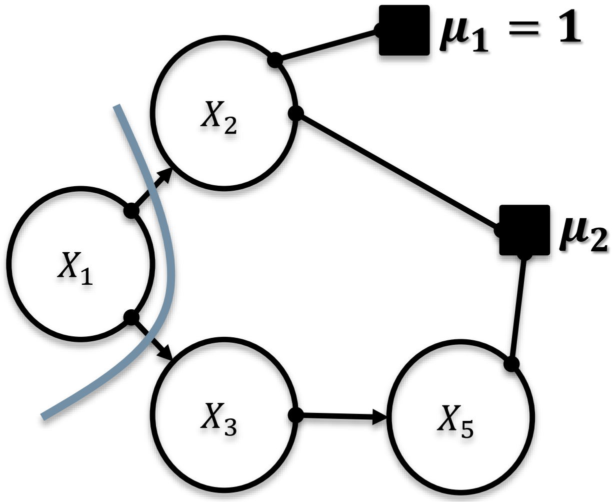


Variable Elimination Example

$$P(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3}\sum_{X_6} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_5|X_3)P(X_6|X_5,X_2)\sum_{X_4}\underbrace{P(X_4|X_2)}_{F_1(X_2,X_4)}$$

$$P(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_5|X_3)\mu_1(X_2)\sum_{X_6}\underbrace{P(X_6|X_5,X_2)}_{F_2(X_6,X_2,X_5)}$$

$$P(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_5|X_3)\mu_1(X_2)\mu_2(X_2,X_5)$$

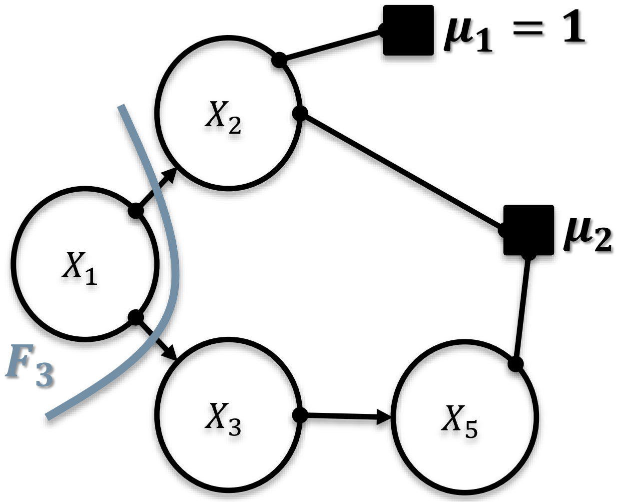


Variable Elimination Example

$$P(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3}\sum_{X_6} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_5|X_3)P(X_6|X_5,X_2)\sum_{X_4}\underbrace{P(X_4|X_2)}_{\boldsymbol{F_1(X_2,X_4)}}$$

$$P(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_5|X_3)\mu_1(X_2)\sum_{X_6}\underbrace{P(X_6|X_5,X_2)}_{\boldsymbol{F_2(X_6,X_2,X_5)}}$$

$$P(X_5) = \sum_{X_2}\sum_{X_3} P(X_5|X_3)\mu_1(X_2)\mu_2(X_2,X_5)\sum_{X_1}\underbrace{P(X_1)P(X_2|X_1)\,P(X_3|X_1)}_{\boldsymbol{F_3(X_1,X_2,X_3)}}$$



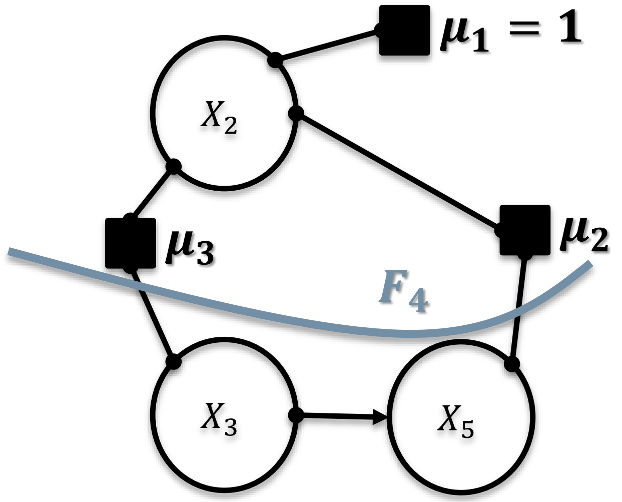
Variable Elimination Example

$$P(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3}\sum_{X_6} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_5|X_3)P(X_6|X_5,X_2)\sum_{X_4}\underbrace{P(X_4|X_2)}_{\boldsymbol{F_1(X_2,X_4)}}$$

$$P(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_5|X_3)\mu_1(X_2)\sum_{X_6}\underbrace{P(X_6|X_5,X_2)}_{\boldsymbol{F_2(X_6,X_2,X_5)}}$$

$$P(X_5) = \sum_{X_2}\sum_{X_3} P(X_5|X_3)\mu_1(X_2)\mu_2(X_2,X_5)\sum_{X_1}\underbrace{P(X_1)P(X_2|X_1)\,P(X_3|X_1)}_{\boldsymbol{F_3(X_1,X_2,X_3)}}$$

$$P(X_5) = \sum_{X_2}\sum_{X_3} P(X_5|X_3)\mu_1(X_2)\mu_2(X_2,X_5)\mu_3(X_2,X_3)$$



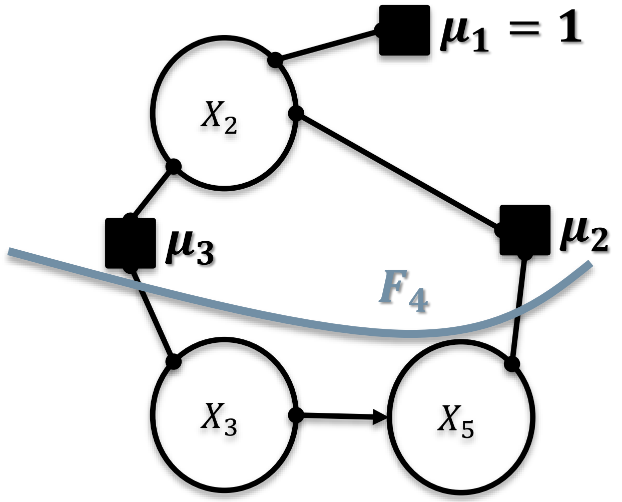
Variable Elimination Example

$$P(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3}\sum_{X_6} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_5|X_3)P(X_6|X_5,X_2)\sum_{X_4}\underbrace{P(X_4|X_2)}_{\boldsymbol{F_1(X_2,X_4)}}$$

$$P(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_5|X_3)\mu_1(X_2)\sum_{X_6}\underbrace{P(X_6|X_5,X_2)}_{\boldsymbol{F_2(X_6,X_2,X_5)}}$$

$$P(X_5) = \sum_{X_2}\sum_{X_3} P(X_5|X_3)\mu_1(X_2)\mu_2(X_2,X_5)\sum_{X_1}\underbrace{P(X_1)P(X_2|X_1)\,P(X_3|X_1)}_{\boldsymbol{F_3(X_1,X_2,X_3)}}$$

$$P(X_5) = \sum_{X_3} P(X_5|X_3)\sum_{X_2}\underbrace{\mu_1(X_2)\mu_2(X_2,X_5)\mu_3(X_2,X_3)}_{\boldsymbol{F_4(X_5,X_3)}}$$



Variable Elimination Example

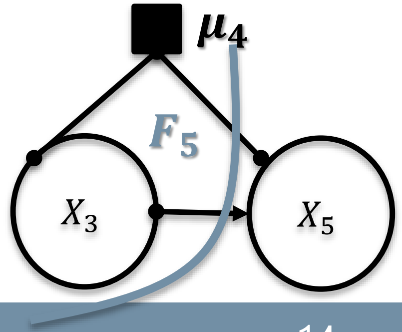
$$P(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3}\sum_{X_6} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_5|X_3)P(X_6|X_5,X_2)\sum_{X_4}\underbrace{P(X_4|X_2)}_{F_1(X_2,X_4)}$$

$$P(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_5|X_3)\mu_1(X_2)\sum_{X_6}\underbrace{P(X_6|X_5,X_2)}_{F_2(X_6,X_2,X_5)}$$

$$P(X_5) = \sum_{X_2}\sum_{X_3} P(X_5|X_3)\mu_1(X_2)\mu_2(X_2,X_5)\sum_{X_1}\underbrace{P(X_1)P(X_2|X_1)\,P(X_3|X_1)}_{F_3(X_1,X_2,X_3)}$$

$$P(X_5) = \sum_{X_3} P(X_5|X_3)\sum_{X_2}\underbrace{\mu_1(X_2)\mu_2(X_2,X_5)\mu_3(X_2,X_3)}_{F_4(X_5,X_3)}$$

$$P(X_5) = \sum_{X_3}\underbrace{P(X_5|X_3)\mu_4(X_3,X_5)}_{F_5(X_5,X_3)}$$



Variable Elimination Example

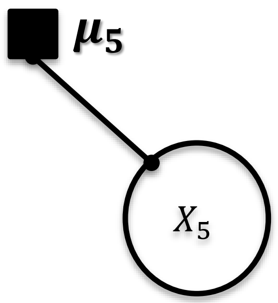
$$P(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3}\sum_{X_6} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_5|X_3)P(X_6|X_5,X_2)\sum_{X_4}\underbrace{P(X_4|X_2)}_{\boldsymbol{F_1(X_2,X_4)}}$$

$$P(X_5) = \sum_{X_1}\sum_{X_2}\sum_{X_3} P(X_1)P(X_2|X_1)\,P(X_3|X_1)P(X_5|X_3)\mu_1(X_2)\sum_{X_6}\underbrace{P(X_6|X_5,X_2)}_{\boldsymbol{F_2(X_6,X_2,X_5)}}$$

$$P(X_5) = \sum_{X_2}\sum_{X_3} P(X_5|X_3)\mu_1(X_2)\mu_2(X_2,X_5)\sum_{X_1}\underbrace{P(X_1)P(X_2|X_1)\,P(X_3|X_1)}_{\boldsymbol{F_3(X_1,X_2,X_3)}}$$

$$P(X_5) = \sum_{X_3} P(X_5|X_3)\sum_{X_2}\underbrace{\mu_1(X_2)\mu_2(X_2,X_5)\mu_3(X_2,X_3)}_{\boldsymbol{F_4(X_5,X_3)}}$$

$$P(X_5) = \sum_{X_3}\underbrace{P(X_5|X_3)\mu_4(X_3,X_5)}_{\boldsymbol{F_5(X_5,X_3)}} = \mu_5(X_5)$$



Variable Elimination and Dynamic Programming

Variable Elimination is based on Dynamic Programming:

- For eliminating $X_{5,4,6}$ we use the solution of eliminating $X_{4,6}$
- "Sub-problems" represented by F terms, their solutions by remaining μ terms

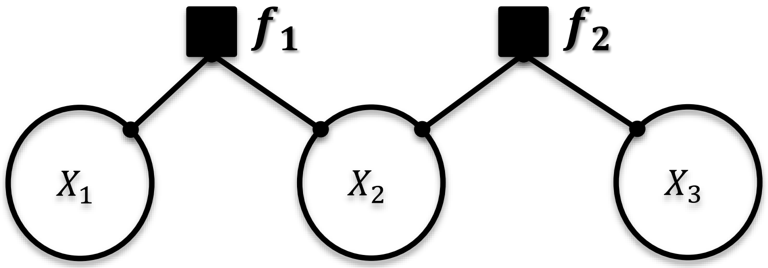
The factorization of the Joint Distribution

- Determines in which order Variable Elimination is efficient
- Determines what the terms $F\,(\ldots)$ and $\mu(\ldots)$ depend on

Can automate Variable Elimination, but need to introduce Factor Graphs.

Factor Graphs

Form of Graphical Model where the box notation indicates terms that depend on some variables.



A Factor graph is a

- Bipartite graph
- Each circle node represents a random variable X_i
- Each box node represents a **factor** f_k, which is a function $f_k(X_{C_k})$
- The joint probability distribution is given as

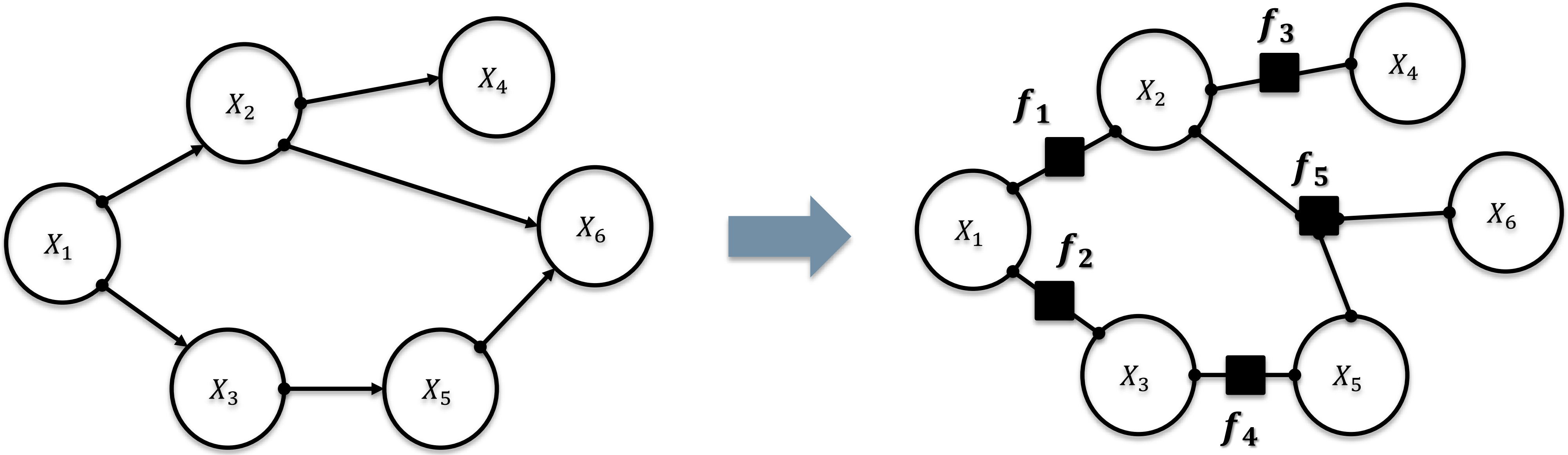
$$P(X_1, X_2, \dots, X_N) = \prod_{k=1}^{K} f_k(X_{C_k})$$

C_k is the set of neighbors of k also named Clique

Note: Factors can be more general than conditional probabilities

From Bayesian Network to Factor Graphs

We can transform a Bayesian Network into a Factor Graph, via ***moralization***, i.e., joining unmarried parents into a single factor.



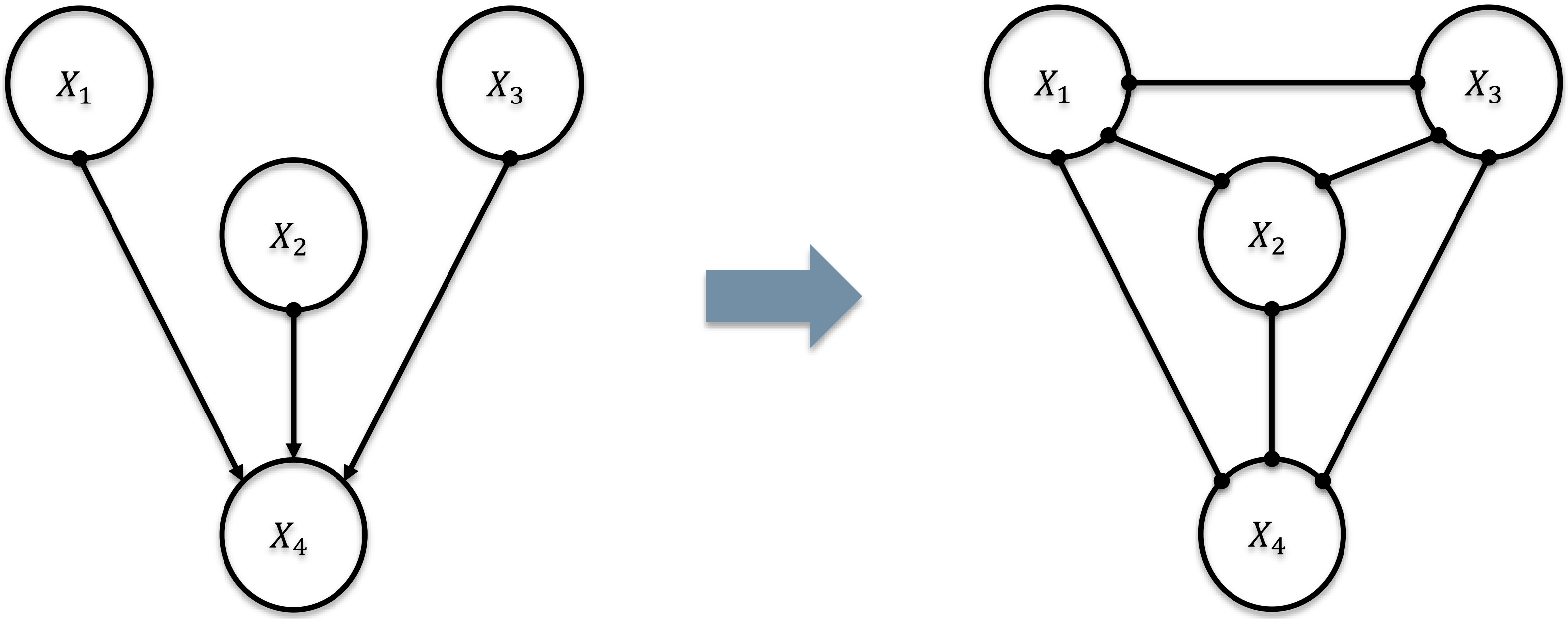
$P(X_1)P(X_2|X_1)P(X_3|X_1)P(X_4|X_2)P(X_5|X_3)P(X_6|X_5,X_2)$

$f_1(X_1,X_2)f_2(X_1,X_3)f_3(X_2,X_4)f_4(X_3,X_5)f_5(X_4,X_5,X_6)$

That's a different factorization

Directed to Undirected Graph: Moralization

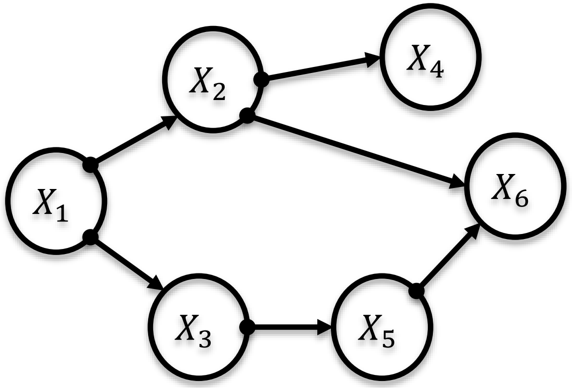
<u>Moralization</u>, i.e., marrying the parents, adds the fewest extra links but maintains the maximum number of independence properties.



Factorization in Directed and Undirected Graphs

For directed graph:

$$P(X_1, X_2, \dots, X_N) = \prod_{n=1}^{N} P\big(X_n | parents(X_n)\big)$$



For undirected graph:

Potential

$$P(X_1, X_2, \dots, X_N) = \frac{1}{Z} \prod_{k=1}^{K} f_k\big(X_{C_k}\big)$$

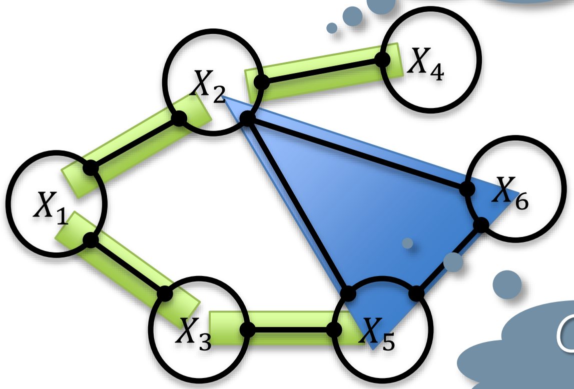
$$f_k\big(X_{C_k}\big) = \exp\{-E\big(X_{C_k}\big)\}$$

Energy

Partition Function

$$Z = \sum \prod_{k=1}^{K} f_k\big(X_{C_k}\big)$$

Order Two Cliques

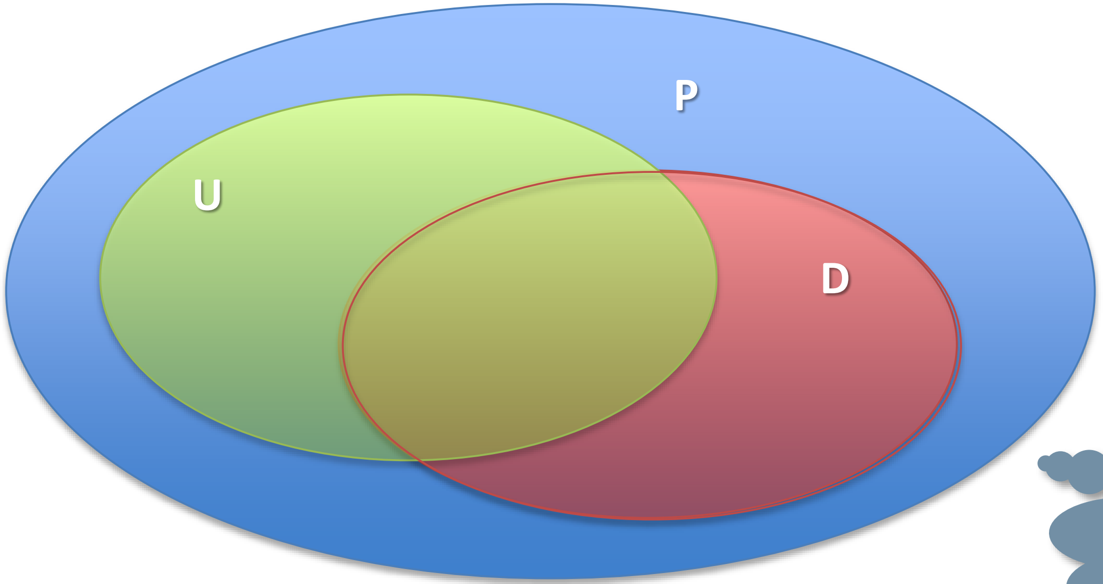


Order Three Clique

Perfect Map

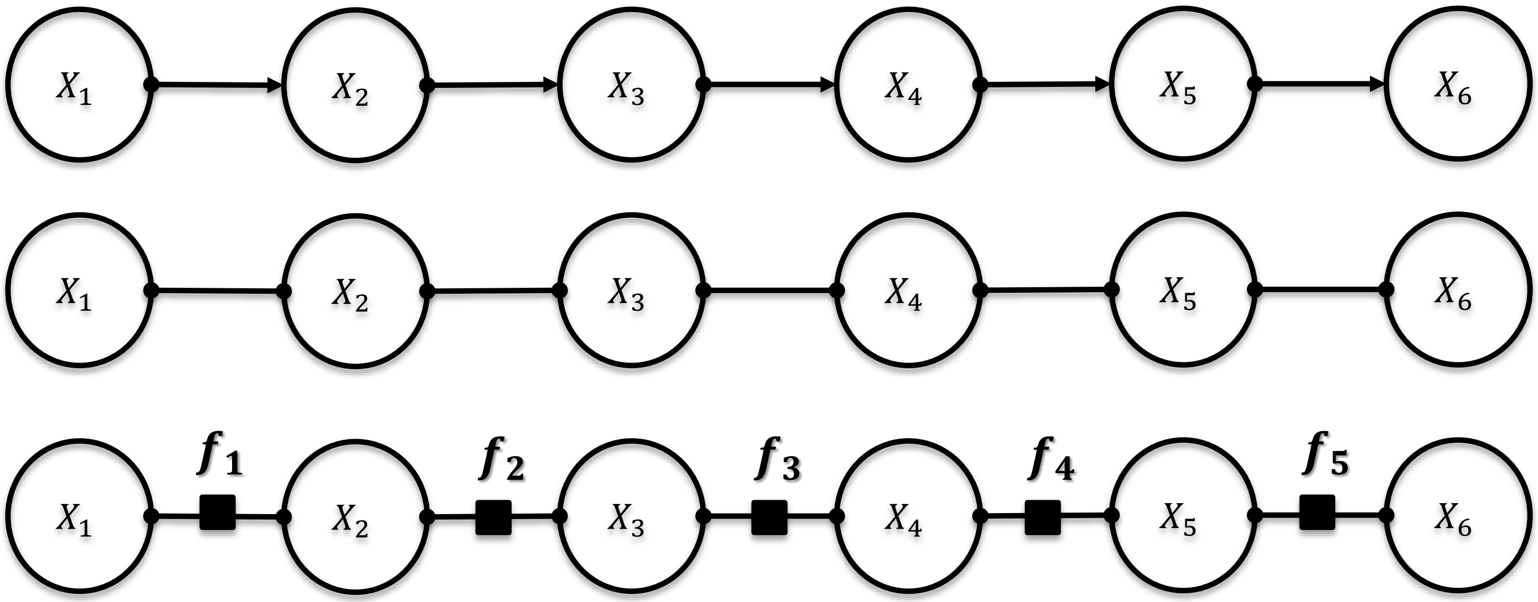
Every independence property of a distribution is reflected in the graph and vice versa, then the graph is a perfect map:

- Not all distributions can be represented as a directed/undirected graph
- Not all directed graph can be represented as undirected graph
- Not all undirected graph can be represented as directed graph.



We won't bother too much about this …

Directed to Undirected Graph: The Chain



$$f_1(X_1, X_2) = P(X_1)P(X_2|X_1)$$

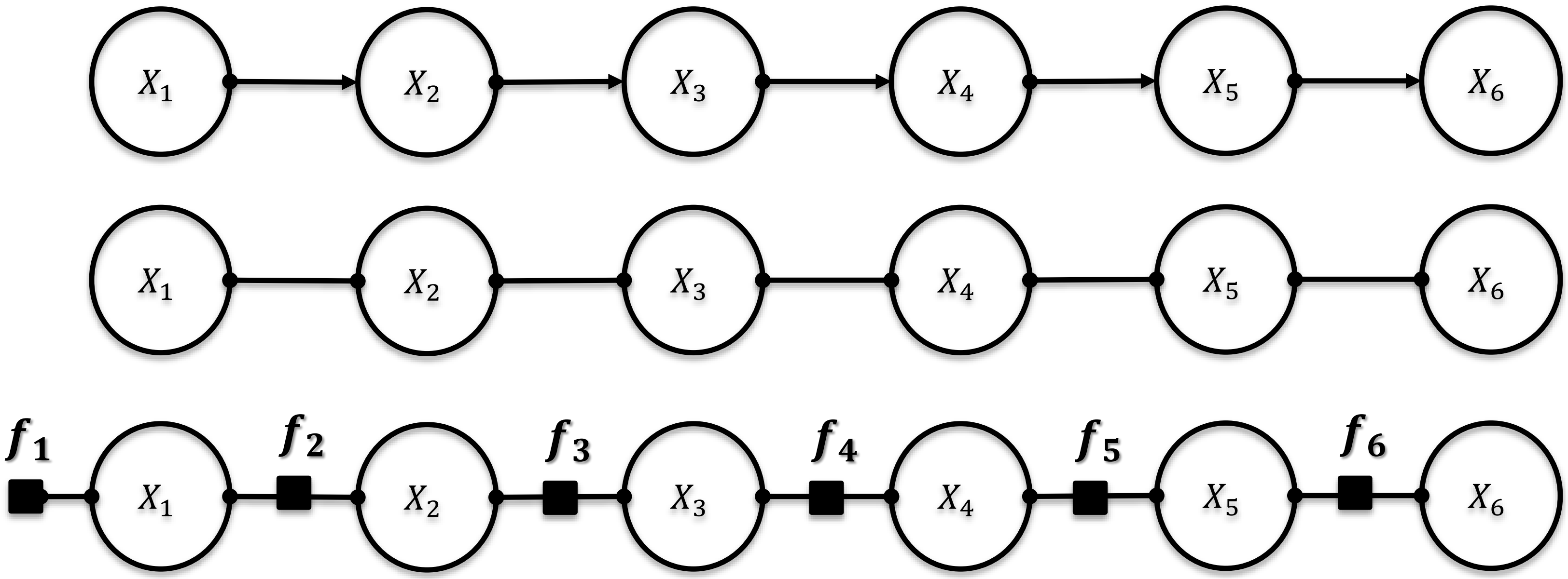
$$f_2(X_2, X_3) = P(X_3|X_2)$$

$$f_3(X_3, X_4) = P(X_4|X_3)$$

$$f_4(X_4, X_5) = P(X_5|X_4)$$

$$f_5(X_5, X_6) = P(X_6|X_5)$$

Directed to Undirected Graph: The Chain (Alternative)



$f_1(X_1) = P(X_1)$

$f_2(X_1, X_2) = P(X_1|X_2)$

$f_3(X_2, X_3) = P(X_3|X_2)$

$f_4(X_3, X_4) = P(X_4|X_3)$

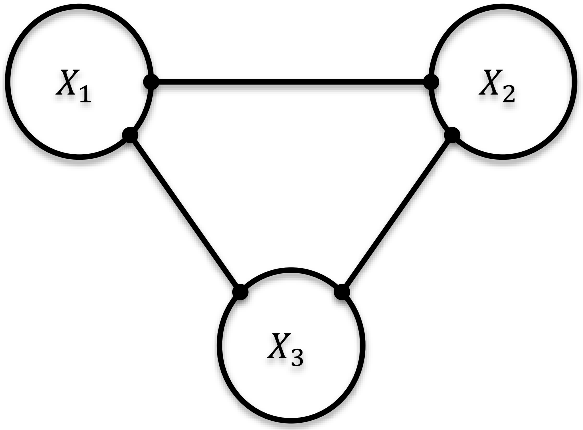
$f_5(X_4, X_5) = P(X_5|X_4)$

$f_5(X_5, X_6) = P(X_6|X_5)$

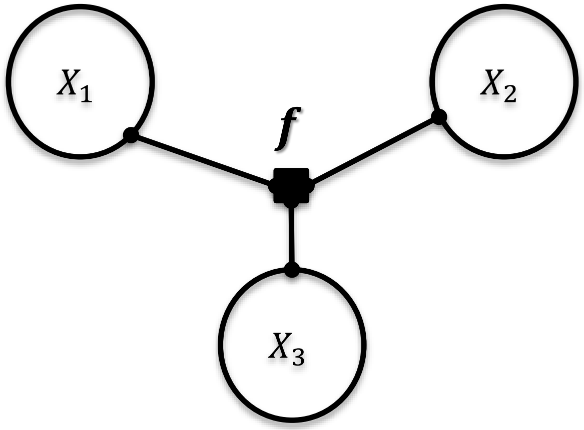
Direct mapping with the Bayesian Network

The undirected graph is the same, but with a different factorization

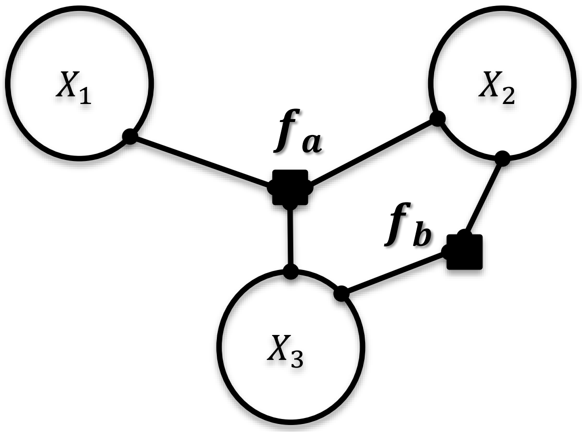
Fractor Graphs are not Unique



$P(X_1, X_2, X_3)$

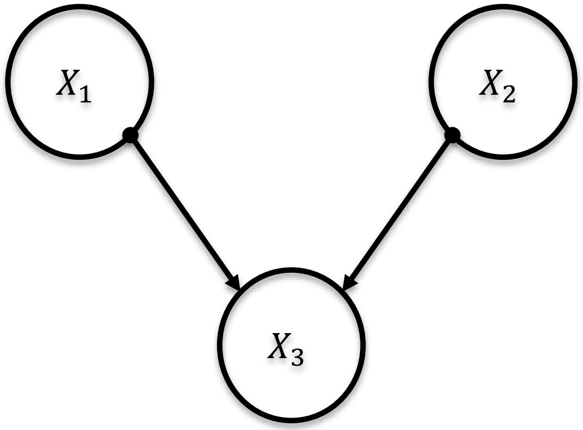


$f(X_1, X_2, X_3) = P(X_1, X_2, X_3)$

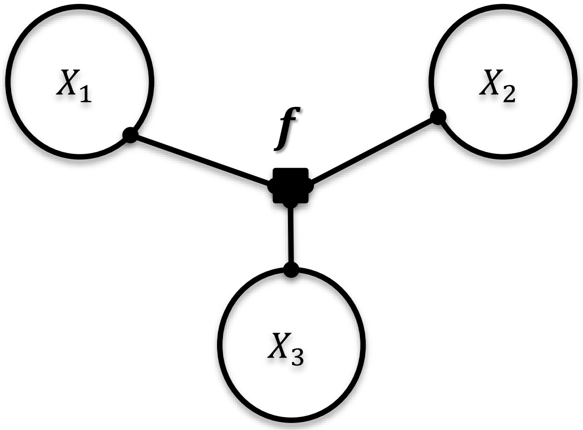


$f_a(X_1, X_2, X_3) f_b(X_2, X_3) = P(X_1, X_2, X_3)$

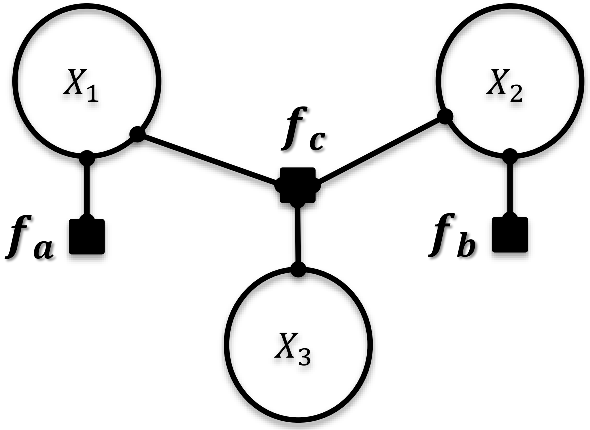
Fractor Graphs are not Unique



$$P(X_1, X_2, X_3) = P(X_3|X_1, X_2)P(X_1)P(X_2)$$



$$f(X_1, X_2, X_3) = P(X_3|X_1, X_2)P(X_1)P(X_2)$$



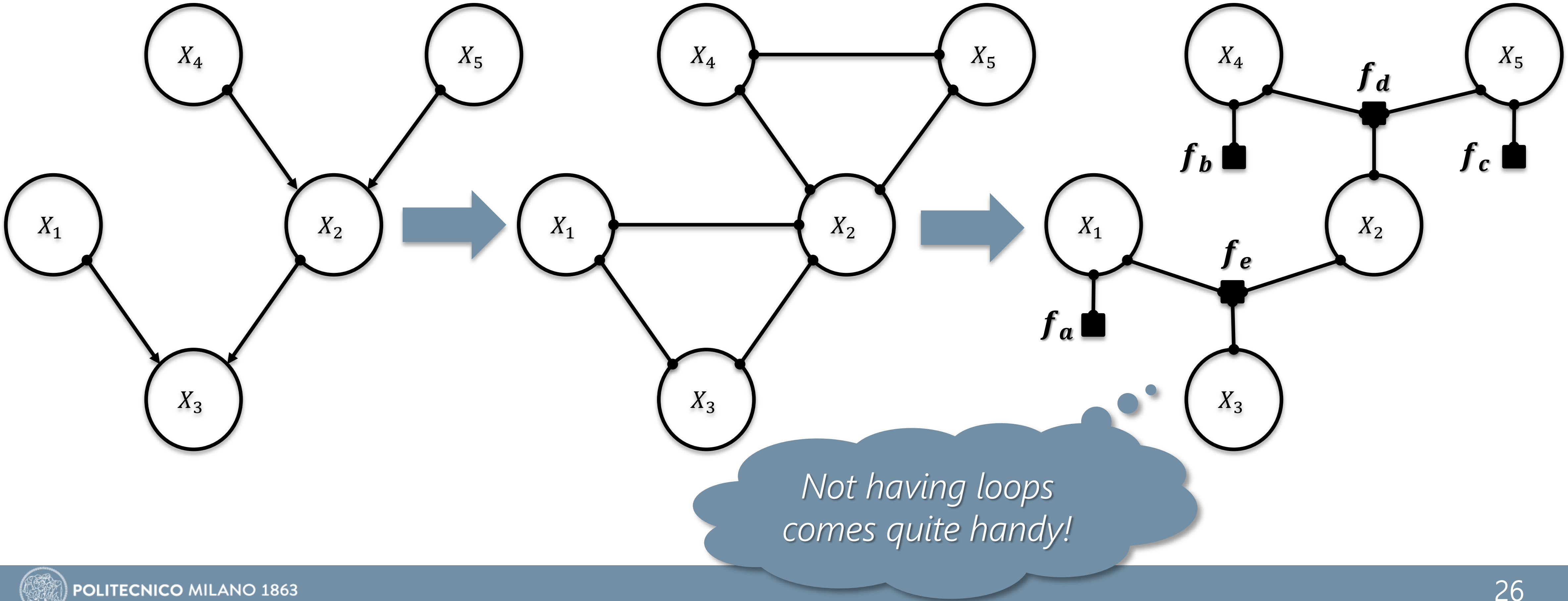
$$f_a(X_1) = P(X_1)$$

$$f_b(X_2) = P(X_2)$$

$$f_c(X_1, X_2, X_3) = P(X_3|X_1, X_2)$$

Polytree Example

Polytree can be converted in a tree shaped factor graph -> no loops



Variable Elimination Algorithm

Variables in the output factor

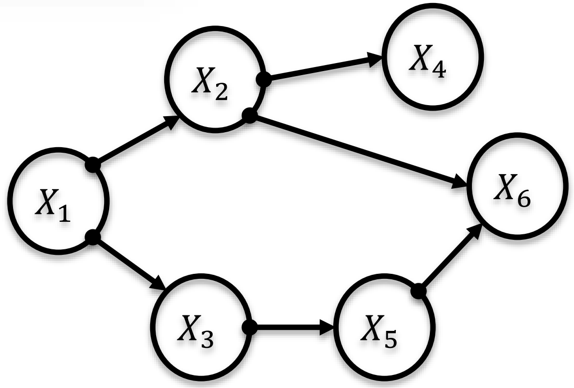
`elimination_algorithm`(m, F, C_o)

Input: list F of factors, tuple C_o of output variables ids
Output: single factor m over variables X_{C_o}
define all variables present in F: $V = \text{vars}(F)$
define variables to be eliminated: $E = V \setminus C_o$
for all $i \in E$: `eliminate_single_variable`(F, i)
for all remaining factors, compute the product $m = \prod_{f \in F} f$
return m

The output factor

<u>*Example*</u>: Compute the following marginal

$$P(X_1, X_6) = \mu(X_1, X_6)$$



Variable Elimination Algorithm

`elimination_algorithm`(m, F, C_o)

Input: list F of factors, tuple C_o of output variables ids

Output: single factor m over variables X_{C_o}

define all variables present in F: $V = \text{vars}(F)$

define variables to be eliminated: $E = V \setminus C_o$

for all $i \in E$: `eliminate_single_variable`(F, i)

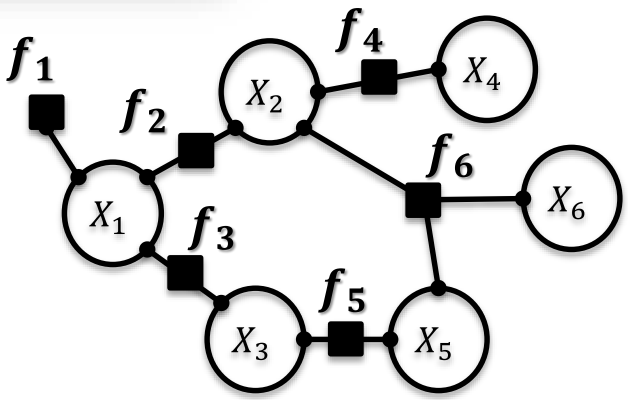
for all remaining factors, compute the product $m = \prod_{f \in F} f$

return m

<u>*Example*</u>: Compute the following marginal

$P(X_1, X_6) = \mu(X_1, X_6) \quad F = \{f_1, f_2, f_3, f_4, f_5, f_6\} \quad C_0 = \{X_1, X_6\}$

$V = \{X_1, X_2, X_3, X_4, X_5, X_6\} \quad E = \{X_2, X_3, X_4, X_5\}$



Variable Elimination Algorithm

`eliminate_single_variable`(F, i)

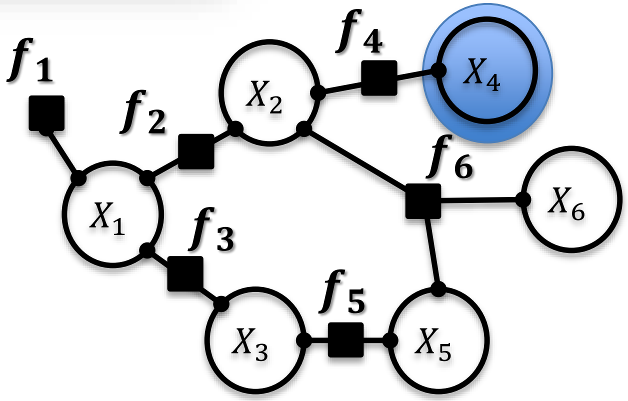
Input: list F of factors, variable id i
Output: list F of factors
find relevant subset $f \subset F$ of factors over i: $f = \{C : i \in C\}$
define remaining clique C_t = all variables in f except i C_t = vars$(f) \setminus \{i\}$
compute temporary factor $\mu(X_{C_t}) = \sum_{X_i} \prod_{f \in F} f$
remove old factors f and append new temporary factor t to F
return F

$$E = \{X_2, X_3, X_4, X_5\}$$

<u>*Example*</u>: Compute the following marginal

$P(X_1, X_6) = \mu(X_1, X_6)$ $\quad F = \{f_1, f_2, f_3, f_4, f_5, f_6\}$ $\quad i = X_4$

$f = \{f_4\}$ $\quad C_t = \{X_2\}$ $\quad \mu_1(X_2) = \sum_{X_4} P(X_4 | X_2)$



Variable Elimination Algorithm

`eliminate_single_variable`(F, i)

Input: list F of factors, variable id i

Output: list F of factors

find relevant subset $f \subset F$ of factors over i: $f = \{C : i \in C\}$

define remaining clique C_t = all variables in f except i $C_t = \text{vars}(f) \setminus \{i\}$

compute temporary factor $\mu(X_{C_t}) = \sum_{X_i} \prod_{f \in F} f$

remove old factors f and append new temporary factor t to F

return F

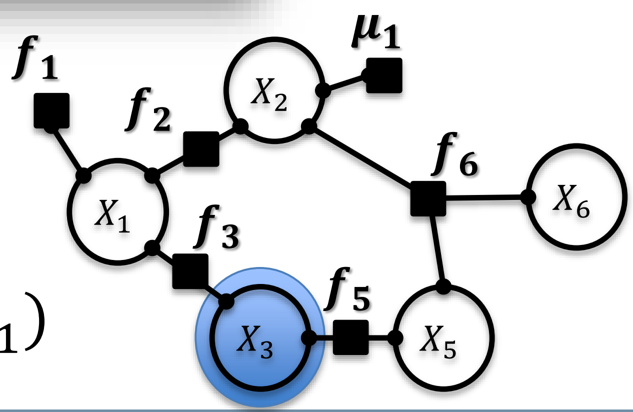
return m

$E = \{X_2, X_3, \cancel{X_4}, X_5\}$

<ins>*Example*</ins>: Compute the following marginal

$P(X_1, X_6) = \mu(X_1, X_6) \qquad F = \{f_1, f_2, f_3, f_5, f_6, \mu_1\} \quad i = X_3$

$f = \{f_3, f_5\} \quad C_t = \{X_1, X_5\} \quad \mu_2(X_1, X_5) = \sum_{X_3} P(X_5|X_3)P(X_3|X_1)$



Variable Elimination Algorithm

`eliminate_single_variable`(F, i)

Input: list F of factors, variable id i

Output: list F of factors

find relevant subset $f \subset F$ of factors over i: $f = \{C : i \in C\}$

define remaining clique C_t = all variables in f except i C_t = vars$(f) \setminus \{i\}$

compute temporary factor $\mu(X_{C_t}) = \sum_{X_i} \prod_{f \in F} f$

remove old factors f and append new temporary factor t to F

return F

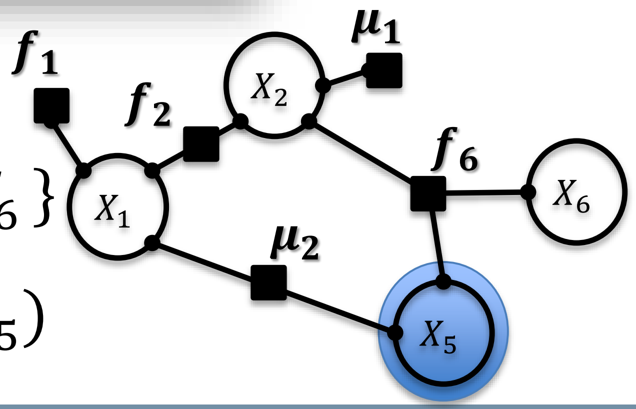
return m

$$E = \{X_2, \cancel{X_3}, \cancel{X_4}, X_5\}$$

<u>*Example*</u>: Compute the following marginal

$P(X_1, X_6) = \mu(X_1, X_6) \quad F = \{f_1, f_2, f_6, \mu_1, \mu_2\} \;\; i = X_5 \;\; f = \{\mu_2, f_6\}$

$C_t = \{X_1, X_2, X_6\} \quad \mu_3(X_1, X_2, X_6) = \sum_{X_5} \mu_2(X_1, X_5) P(X_6 | X_2, X_5)$



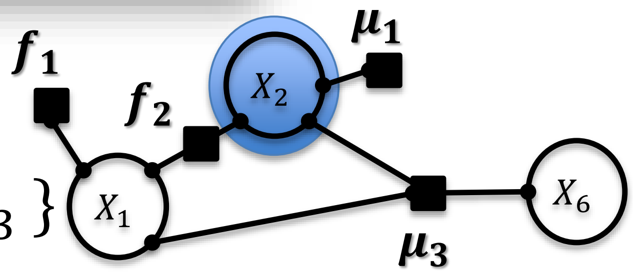
Variable Elimination Algorithm

```
eliminate_single_variable(F, i)
Input: list F of factors, variable id i
Output: list F of factors
find relevant subset f ⊂ F of factors over i:  f = {C : i ∈ C}
define remaining clique C_t = all variables in f except i  C_t = vars(f) \ {i}
compute temporary factor  μ(X_{C_t}) = Σ_{X_i} Π_{f∈F} f
remove old factors f and append new temporary factor t to F
return F
```

$E = \{X_2, \cancel{X_3}, \cancel{X_4}, \cancel{X_5}\}$

return m

<u>*Example*</u>: Compute the following marginal



$P(X_1, X_6) = \mu(X_1, X_6) \quad F = \{f_1, f_2, \mu_1, \mu_3\} \quad i = X_2 \quad f = \{\mu_1, f_2, \mu_3\}$

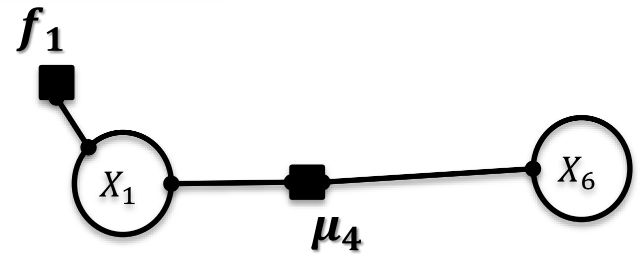
$C_t = \{X_1, X_6\} \quad \mu_4(X_1, X_6) = \sum_{X_2} \mu_1(X_2) P(X_2|X_1) \mu_3(X_1, X_2, X_6)$

Variable Elimination Algorithm

elimination_algorithm(m, F, C_o)

Input: list F of factors, tuple C_o of output variables ids
Output: single factor m over variables X_{C_o}
define all variables present in F: $V = \text{vars}(F)$
define variables to be eliminated: $E = V \setminus C_o$
for all $i \in E$: eliminate_single_variable(F, i)
for all remaining factors, compute the product $m = \prod_{f \in F} f$
return m

Example: Compute the following marginal



$P(X_1, X_6) = \mu(X_1, X_6) \quad F = \{f_1, \mu_4\} \quad C_0 = \{X_1, X_6\}$

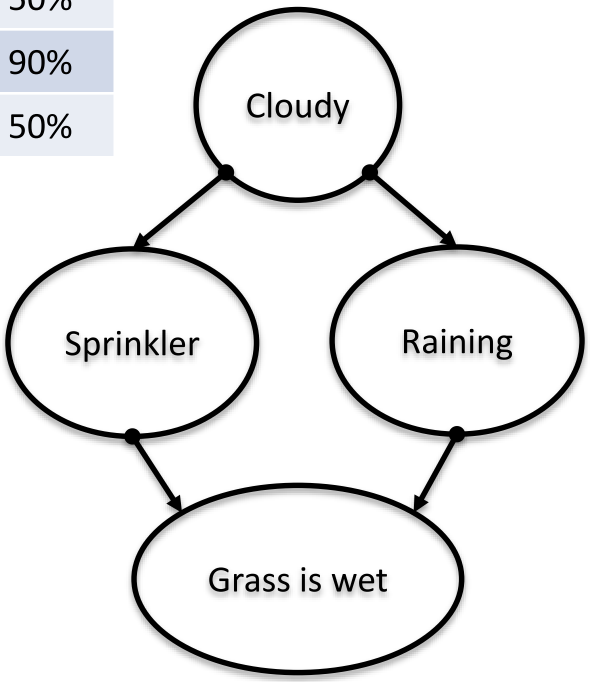
$E = \{\cancel{X_2}, \cancel{X_3}, \cancel{X_4}, \cancel{X_5}\} \quad P(X_1, X_6) = \mu(X_1, X_6) = P(X_1)\mu_4(X_1, X_6)$

The Sprinkler Example

Compute $P(W)$ using Variable Elimination

$Cloudy$	$P(C)$
0	50%
1	50%

$Sprinkler$	$Cloudy$	$P(S\|C)$
0	0	10%
0	1	50%
1	0	90%
1	1	50%



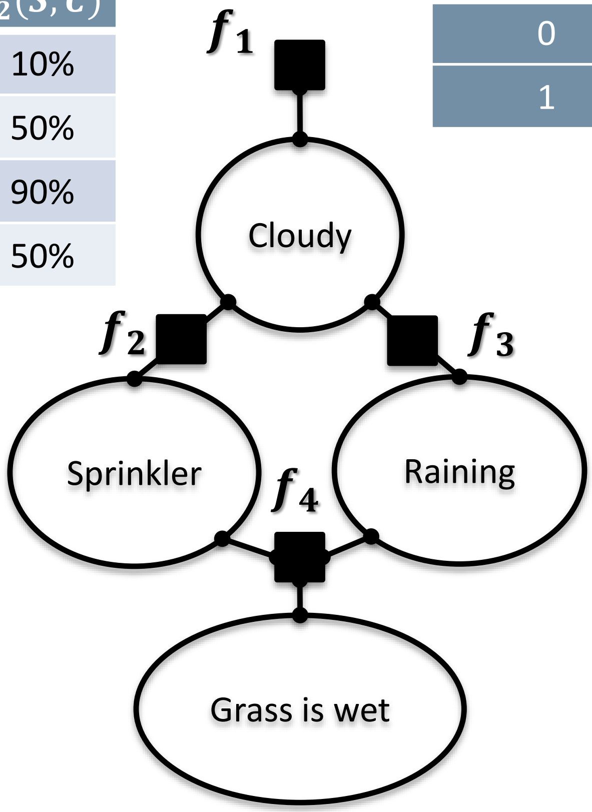
Wet	$Sprinkler$	$Raining$	$P(W\|S,R)$
0	0	0	1
0	0	1	10%
0	1	0	10%
0	1	1	1%
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%

$Raining$	$Cloudy$	$P(R\|C)$
0	0	80%
0	1	50%
1	0	20%
1	1	50%

The Sprinkler Example

Compute $P(W)$ using Variable Elimination

$Sprinkler$	$Cloudy$	$f_2(S,C)$
0	0	10%
0	1	50%
1	0	90%
1	1	50%



$Cloudy$	$f_1(C)$
0	50%
1	50%

Wet	$Sprinkler$	$Raining$	$f_4(W,S,R)$
0	0	0	1
0	0	1	10%
0	1	0	10%
0	1	1	1%
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%

$Raining$	$Cloudy$	$f_3(R,C)$
0	0	80%
0	1	50%
1	0	20%
1	1	50%

The Sprinkler Example

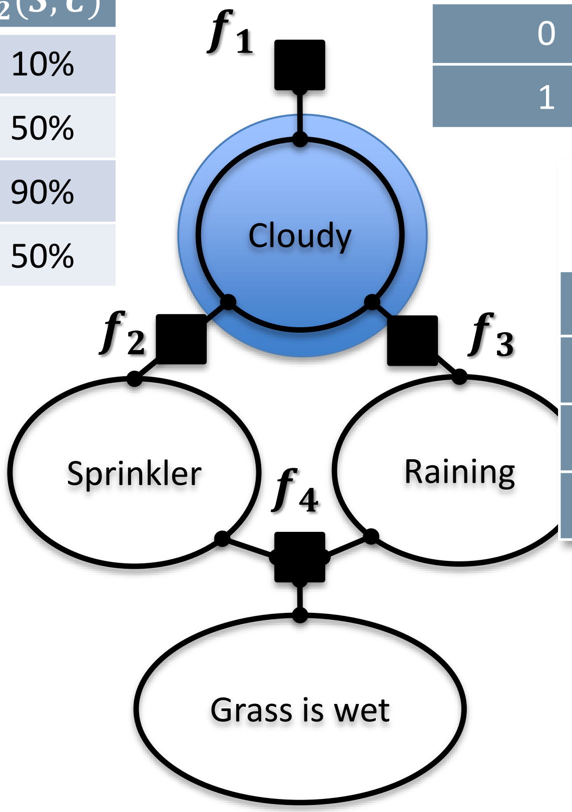
$$C_t = \{S, R\} \qquad f = \{f_1, f_2, f_3\}$$

$$\mu_1(S,R) = \sum_C f_1 \cdot f_2 \cdot f_3$$

Compute $P(W)$ using Variable Elimination

Sprinkler	*Cloudy*	$f_2(S,C)$
0	0	10%
0	1	50%
1	0	90%
1	1	50%

Cloudy	$f_1(C)$
0	50%
1	50%



Sprinkler	*Raining*	$\sum_C f_1 \cdot f_2 \cdot f_3$	$\mu_1(S,R)$
0	0	0.5*0.1*0.8+0.5*0.5*0.5	0.165
0	1	0.5*0.1*0.2+0.5*0.5*0.5	0.135
1	0	0.5*0.9*0.8+0.5*0.5*0.5	0.485
1	1	0.5*0.9*0.2+0.5*0.5*0.5	0.215

Wet	*Sprinkler*	*Raining*	$f_4(W,S,R)$
0	0	0	1
0	0	1	10%
0	1	0	10%
0	1	1	1%
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%

Raining	*Cloudy*	$f_4(R,C)$
0	0	80%
0	1	50%
1	0	20%
1	1	50%

The Sprinkler Example

Compute $P(W)$ using Variable Elimination

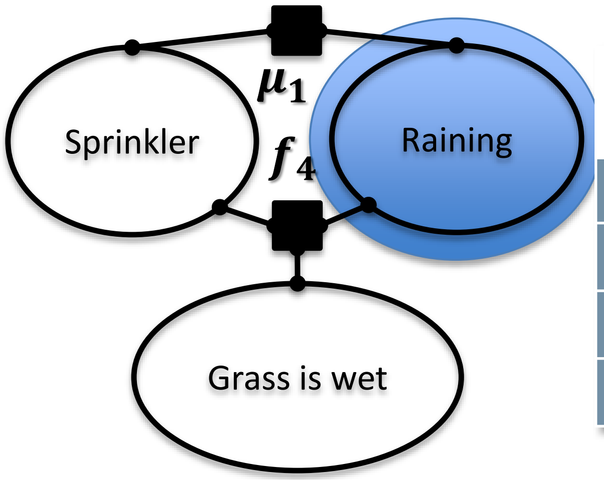
$$C_t = \{S, W\}$$

$$f = \{\mu_1, f_4\}$$

$$\mu_2(S, W) = \sum_R \mu_1 \cdot f_4$$

$Sprinkler$	$Raining$	$\sum_C f_1 \cdot f_2 \cdot f_3$	$\mu_1(S,R)$
0	0	0.5*0.1*0.8+0.5*0.5*0.5	0.165
0	1	0.5*0.1*0.2+0.5*0.5*0.5	0.135
1	0	0.5*0.9*0.8+0.5*0.5*0.5	0.485
1	1	0.5*0.9*0.2+0.5*0.5*0.5	0.215

Wet	$Sprinkler$	$Raining$	$f_4(W,S,R)$
0	0	0	1
0	0	1	10%
0	1	0	10%
0	1	1	1%
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%



$Sprinkler$	Wet	$\sum_R \mu_1 \cdot f_4$	$\mu_2(S,W)$
0	0	0.165*1+0.135*0.1	0.1785
0	1	0.165*0+0.135*0.9	0.1215
1	0	0.485*0.1+0.215*0.01	0.05065
1	1	0.485*0.9+0.215*.99	0.64935

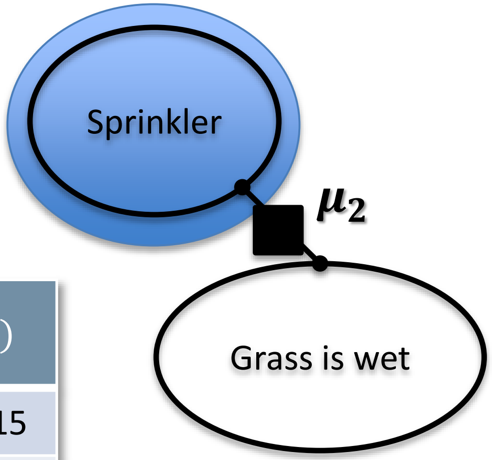
The Sprinkler Example

Compute $P(W)$ using Variable Elimination

$$C_t = \{W\}$$

$$f = \{\mu_2\}$$

$$\mu_3(W) = \sum_S \mu_2$$



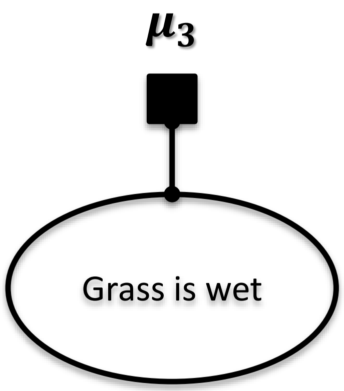
$Sprinkler$	Wet	$\sum_R \mu_1 \cdot f_4$	$\mu_2(\boldsymbol{S},\boldsymbol{W})$
0	0	0.165*1+0.135*0.1	0.1785
0	1	0.165*0+0.135*0.9	0.1215
1	0	0.485*0.1+0.215*0.01	0.05065
1	1	0.485*0.9+0.215*.99	0.64935

Wet	$\sum_S \mu_2$	$\mu_3(\boldsymbol{W})$
0	0.1785+0.05065	0.22915
1	0.1215+0.64935	0.77085

The Sprinkler Example

Compute $P(W)$ using Variable Elimination

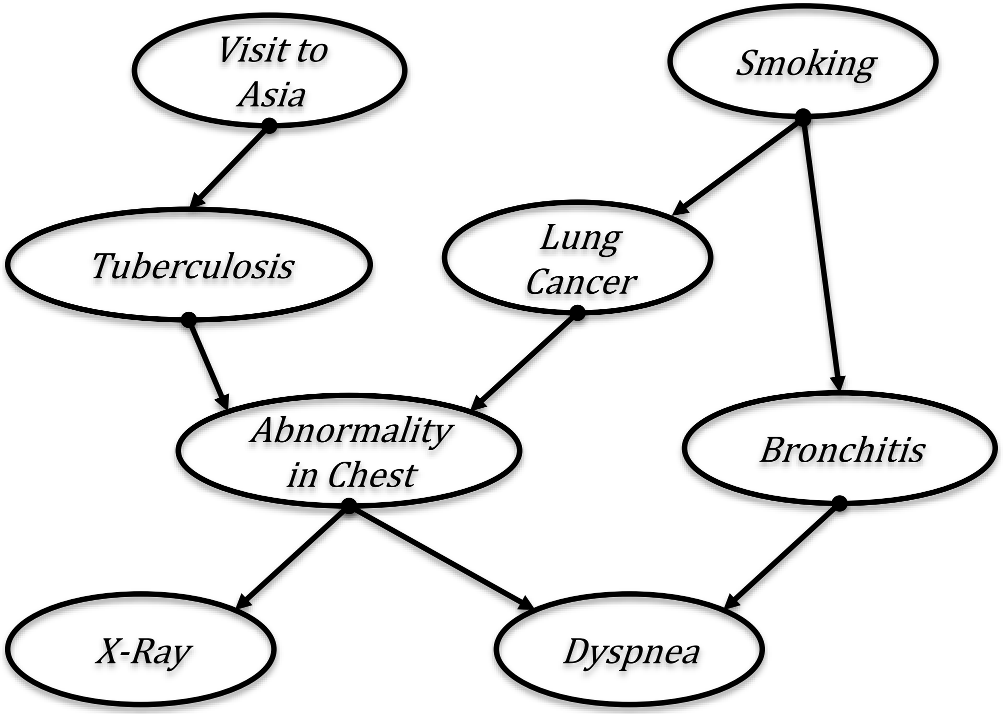
Wet	$\sum_S \mu_2$	$\mu_3(\boldsymbol{W})$
0	0.1785+0.05065	0.22915
1	0.1215+0.64935	0.77085



The Asia Network Example

Suppose we are interested in $P(d)$

We need to eliminate: v, s, x, t, l, a, b



If we apply the Brute force approach we know is $O(2^N)$

$$P(d) = \sum_v \sum_s \sum_x \sum_t \sum_l \sum_a \sum_b P(v)\, P(s)\, P(t\,|\,v)\, P(l\,|\,s)\, P(b\,|\,s)\, P(a\,|\,t,l)\, P(x\,|\,a)\, P(d\,|\,a,b)$$

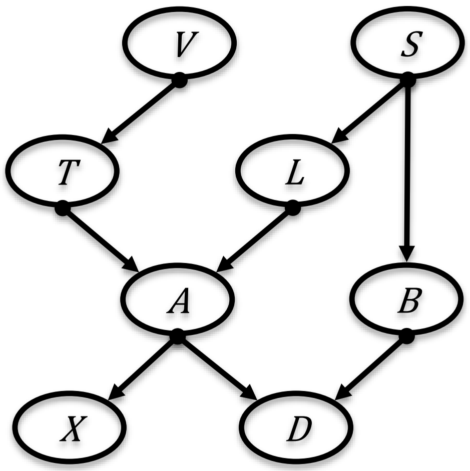
The Asia Network Example

Eliminate variables in order

$$v \to s \to x \to t \to l \to a \to b$$

Initial factors

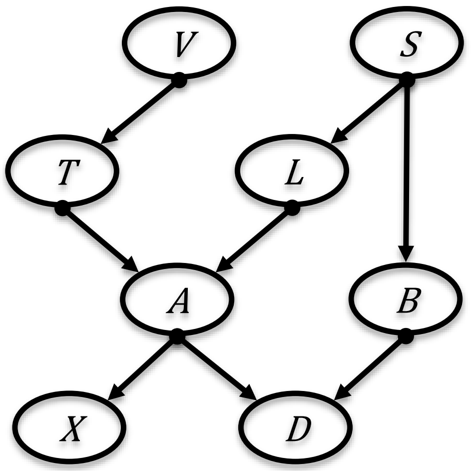
$$\boxed{P(v)}\, P(s)\, \boxed{P(t \mid v)}\, P(l \mid s)\, P(b \mid s)\, P(a \mid t, l)\, P(x \mid a)\, P(d \mid a, b)$$



$$f_v(t) = \sum_v P(v) P(t \mid v) \qquad \Rightarrow f_v(t) P(s) P(l \mid s) P(b \mid s) P(a \mid t, l) P(x \mid a) P(d \mid a, b)$$

Note: $f_v(t) = P(t)$, in general, result is not necessarily a probability term

The Asia Network Example



Eliminate variables in order

$$v \to s \to x \to t \to l \to a \to b$$

Initial factors

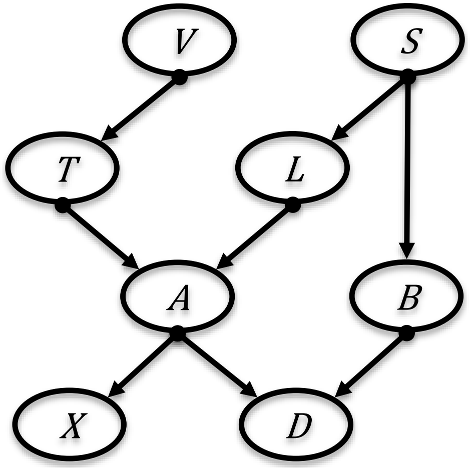
$$P(v)\,P(s)\,P(t|v)\,P(l|s)\,P(b|s)\,P(a|t,l)\,P(x|a)\,P(d|a,b)$$

$$\Rightarrow f_v(t)\boxed{P(s)}\boxed{P(l|s)}\boxed{P(b|s)}P(a|t,l)P(x|a)P(d|a,b)$$

$$f_s(b,l) = \sum_s P(s)P(b|s)P(l|s) \qquad \Rightarrow f_v(t)\,f_s(b,l)P(a|t,l)P(x|a)P(d|a,b)$$

Note: result of elimination may be a function of several variables

The Asia Network Example



Eliminate variables in order

$$v \to s \to x \to t \to l \to a \to b$$

Initial factors

$$P(v)\, P(s)\, P(t \mid v)\, P(l \mid s)\, P(b \mid s)\, P(a \mid t,l)\, P(x \mid a)\, P(d \mid a,b)$$

$$\Rightarrow f_v(t) P(s) P(l \mid s) P(b \mid s) P(a \mid t,l) P(x \mid a) P(d \mid a,b)$$

$$\Rightarrow f_v(t)\, f_s(b,l) P(a \mid t,l) \boxed{P(x \mid a)} P(d \mid a,b)$$

$$f_x(a) = \sum_x P(x \mid a) \quad \Rightarrow f_v(t)\, f_s(b,l)\, f_x(a) P(a \mid t,l) P(d \mid a,b)$$

Note: $f_x(a) = 1$ for all values of a

The Asia Network Example

Eliminate variables in order

$$v \rightarrow s \rightarrow x \rightarrow t \rightarrow l \rightarrow a \rightarrow b$$

Initial factors

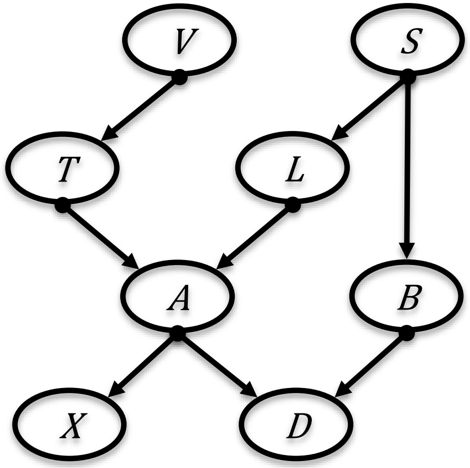
$$P(v)\,P(s)\,P(t\,|\,v)\,P(l\,|\,s)\,P(b\,|\,s)\,P(a\,|\,t,l)\,P(x\,|\,a)\,P(d\,|\,a,b)$$

$$\Rightarrow f_v(t)P(s)P(l\,|\,s)P(b\,|\,s)P(a\,|\,t,l)P(x\,|\,a)P(d\,|\,a,b)$$

$$\Rightarrow f_v(t)\,f_s(b,l)P(a\,|\,t,l)P(x\,|\,a)P(d\,|\,a,b)$$

$$\Rightarrow \boxed{f_v(t)}\,f_s(b,l)\,f_x(a)\boxed{P(a\,|\,t,l)}P(d\,|\,a,b)$$

$$f_t(a,l) = \sum_t f_v(t)P(a\,|\,t,l) \qquad \Rightarrow f_s(b,l)\,f_x(a)\,f_t(a,l)P(d\,|\,a,b)$$



The Asia Network Example

Eliminate variables in order

$$v \to s \to x \to t \to l \to a \to b$$

Initial factors

$$P(v)\,P(s)\,P(t|v)\,P(l|s)\,P(b|s)\,P(a|t,l)\,P(x|a)\,P(d|a,b)$$

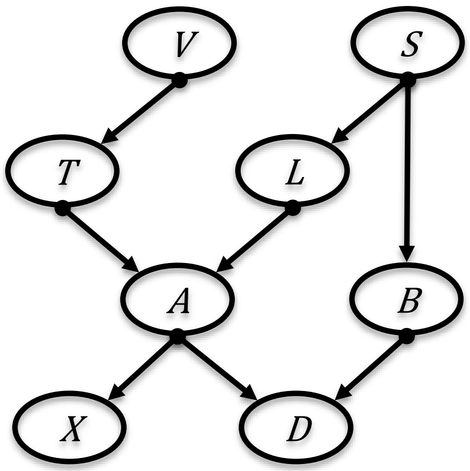
$$\Rightarrow f_v(t)P(s)P(l|s)P(b|s)P(a|t,l)P(x|a)P(d|a,b)$$

$$\Rightarrow f_v(t)\,f_s(b,l)P(a|t,l)P(x|a)P(d|a,b)$$

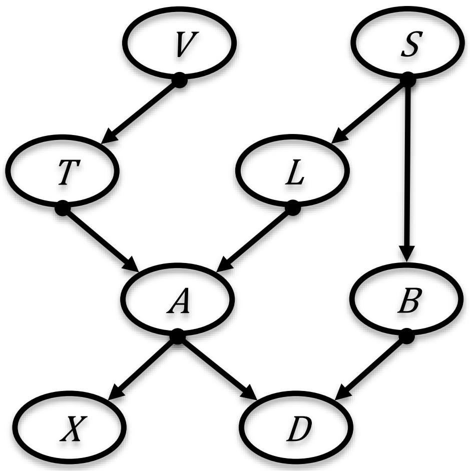
$$\Rightarrow f_v(t)\,f_s(b,l)\,f_x(a)P(a|t,l)P(d|a,b)$$

$$\Rightarrow \boxed{f_s(b,l)}\,f_x(a)\,\boxed{f_t(a,l)}P(d|a,b)$$

$$f_l(a,b) = \sum_l f_s(b,l)\,f_t(a,l) \qquad \Rightarrow f_l(a,b)\,f_x(a)P(d|a,b)$$



The Asia Network Example



Eliminate variables in order

$$v \to s \to x \to t \to l \to a \to b$$

Initial factors

$$P(v)\, P(s)\, P(t|v)\, P(l|s)\, P(b|s)\, P(a|t,l)\, P(x|a)\, P(d|a,b)$$

$$\Rightarrow f_v(t)P(s)P(l|s)P(b|s)P(a|t,l)P(x|a)P(d|a,b)$$

$$\Rightarrow f_v(t)\, f_s(b,l)P(a|t,l)P(x|a)P(d|a,b)$$

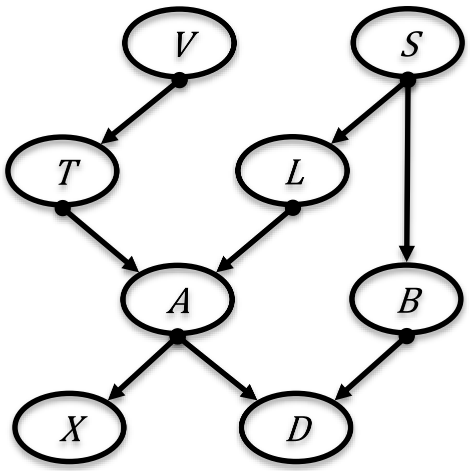
$$\Rightarrow f_v(t)\, f_s(b,l)\, f_x(a)P(a|t,l)P(d|a,b)$$

$$\Rightarrow f_s(b,l)\, f_x(a)\, f_t(a,l)P(d|a,b)$$

$$\Rightarrow \boxed{f_l(a,b)}\,\boxed{f_x(a)}\,\boxed{P(d|a,b)}$$

$$f_a(b,d) = \sum_a f_l(a,b)\, f_x(a)\, p(d|a,b) \qquad \Rightarrow f_a(b,d)$$

The Asia Network Example



Eliminate variables in order

$$v \to s \to x \to t \to l \to a \to b$$

Initial factors

$$P(v)\, P(s)\, P(t|v)\, P(l|s)\, P(b|s)\, P(a|t,l)\, P(x|a)\, P(d|a,b)$$

$$\Rightarrow f_v(t) P(s) P(l|s) P(b|s) P(a|t,l) P(x|a) P(d|a,b)$$

$$\Rightarrow f_v(t)\, f_s(b,l) P(a|t,l) P(x|a) P(d|a,b)$$

$$\Rightarrow f_v(t)\, f_s(b,l)\, f_x(a) P(a|t,l) P(d|a,b)$$

$$\Rightarrow f_s(b,l)\, f_x(a)\, f_t(a,l) P(d|a,b)$$

$$\Rightarrow f_l(a,b)\, f_x(a) P(d|a,b)$$

$$\Rightarrow \boxed{f_a(b,d)}$$

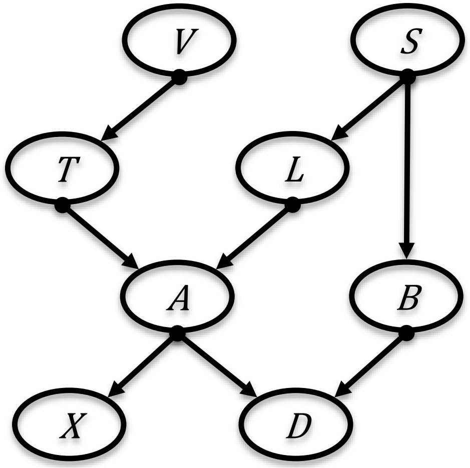
$$f_b(d) = \sum_b f_a(b,d) \Rightarrow f_b(d)$$

The Asia Networks Example

Selected ordering

$$v \rightarrow s \rightarrow x \rightarrow t \rightarrow l \rightarrow a \rightarrow b$$

$f_v(t)$
$f_s(b,l)$
$f_x(a)$
$f_t(a,l)$
$f_l(a,b)$
$f_a(b,d)$
$f_b(d)$



With a different ordering

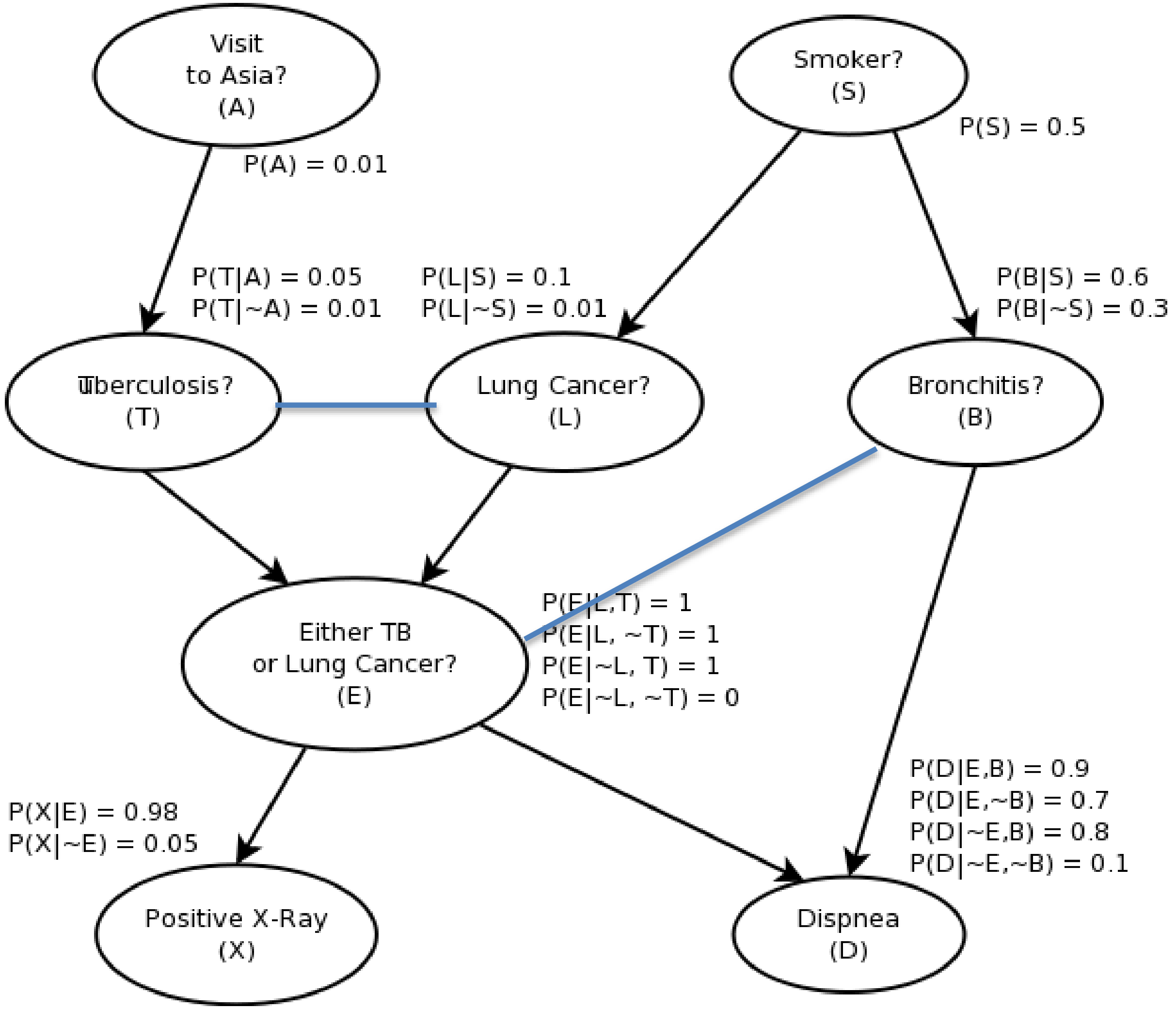
$$a \rightarrow b \rightarrow x \rightarrow t \rightarrow v \rightarrow s \rightarrow l$$

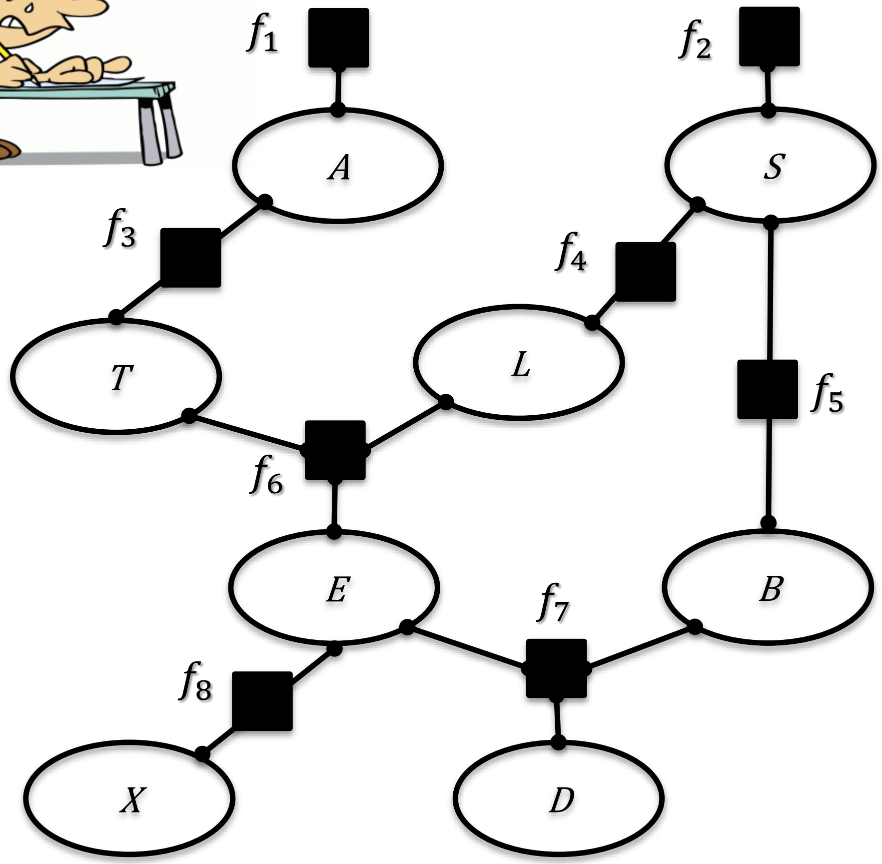
$g_a(l,t,d,b,x)$
$g_b(l,t,d,x,s)$
$g_x(l,t,d,s)$
$g_t(l,t,s,v)$
$g_v(l,d,s)$
$g_s(l,d)$
$g_l(d)$

Complexity is exponential in the size of these factors!

We should find the right ordering!

The Asia Network Example





Variable Elimination Algorithm

Variable Elimination has good properties

- Very simple to implement
- Does exactly what you would do on paper
- With optimal ordering complexity is $O(N2^K)$

But also some drawbacks

- Finding the optimal ordering is an NP-Hard problem
- It computes only one marginal at the time
- Requires N executions to compute all marginals

A.k.a. Sum-Product Algorithms

To improve on this we can use *Belief Propagation* which uses «reusable» local factors and message passing …

Soft Computing – Probabilistic Reasoning

- Belief Propagation -

Prof. Matteo Matteucci – *matteo.matteucci@polimi.it*

The Sum-Product Algorithm (1)

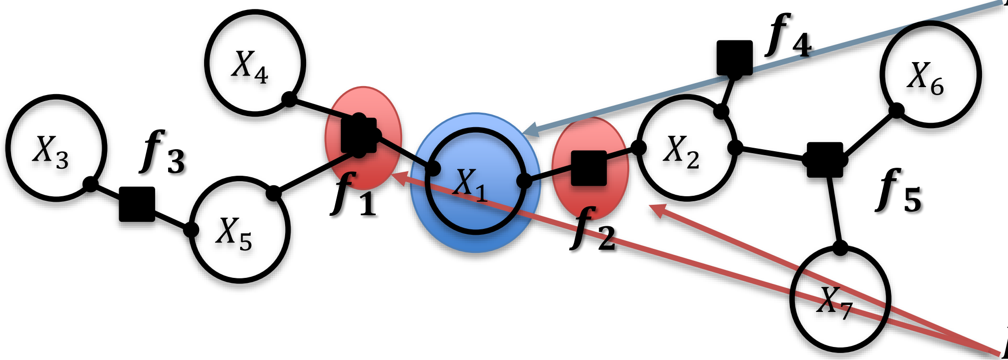
Objectives:

- To obtain an efficient, exact inference algorithm for finding marginals
- To share efficiently computations when several marginals are required

Key ideas:

- Let's focus on undirected trees and polytrees
- Let's exploit $ab + ac = a(b + c)$

We know how to built them.



$$P(X_i) = \sum_{X_1,X_2,\ldots,X_{i-1},X_{i+1},\ldots X_N} P(X_1, X_2, \ldots, X_N) = \sum_{X \setminus X_i} P(X_1, X_2, \ldots, X_N)$$

$$P(X_1, X_2, \ldots, X_N) = \prod_{s'} f_{s'}(X_{s'}) = \prod_{s \in ne(X_i)} F_s(X_i, X_s)$$

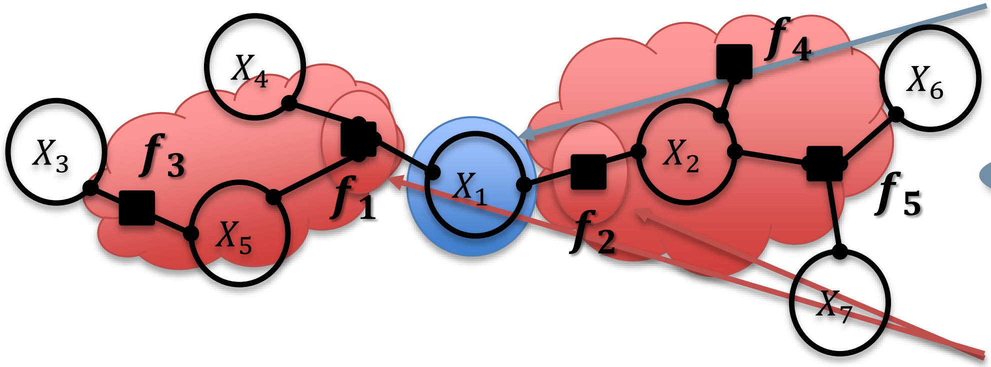
The Sum-Product Algorithm (1)

Objectives:

- To obtain an efficient, exact inference algorithm for finding marginals
- To share efficiently computations when several marginals are required

Key ideas:

- Let's focus on undirected trees and polytrees
- Let's exploit $ab + ac = a(b + c)$



$$P(X_i) = \sum_{X_1, X_2, \ldots, X_{i-1}, X_{i+1}, \ldots X_N} P(X_1, X_2, \ldots, X_N) = \sum_{X \setminus X_i} P(X_1, X_2, \ldots, X_N)$$

F_s product of factors in the group associated to f_s

$$P(X_1, X_2, \ldots, X_N) = \prod_{s'} f_{s'}(X_{s'}) = \prod_{s \in ne(X_i)} F_s(X_i, X_s)$$

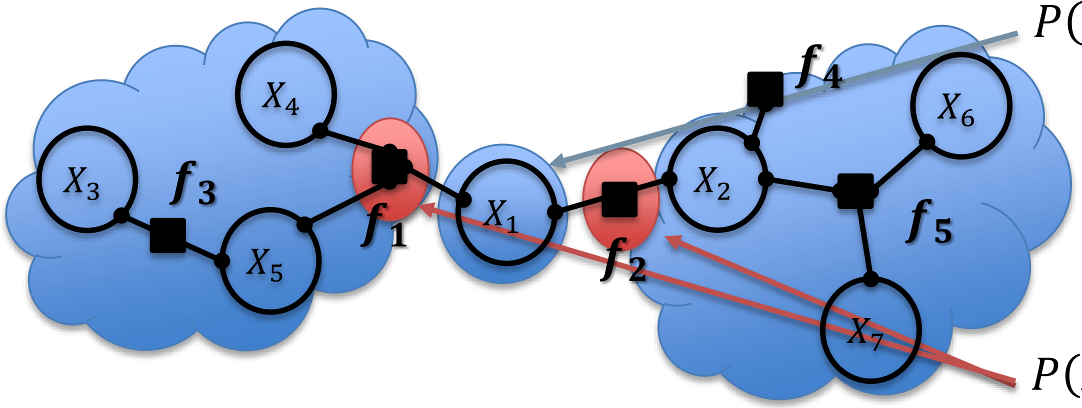
The Sum-Product Algorithm (1)

Objectives:

- To obtain an efficient, exact inference algorithm for finding marginals
- To share efficiently computations when several marginals are required

Key ideas:

- Let's focus on undirected trees and polytrees
- Let's exploit $ab + ac = a(b + c)$



$$P(X_i) = \sum_{X_1, X_2, \ldots, X_{i-1}, X_{i+1}, \ldots X_N} P(X_1, X_2, \ldots, X_N) = \sum_{X \setminus X_i} P(X_1, X_2, \ldots, X_N)$$

X_s set of variables in the subtree connected to X_i

$$P(X_1, X_2, \ldots, X_N) = \prod_{s'} f_{s'}(X_{s'}) = \prod_{s \in ne(X_i)} F_s(X_i, X_s)$$

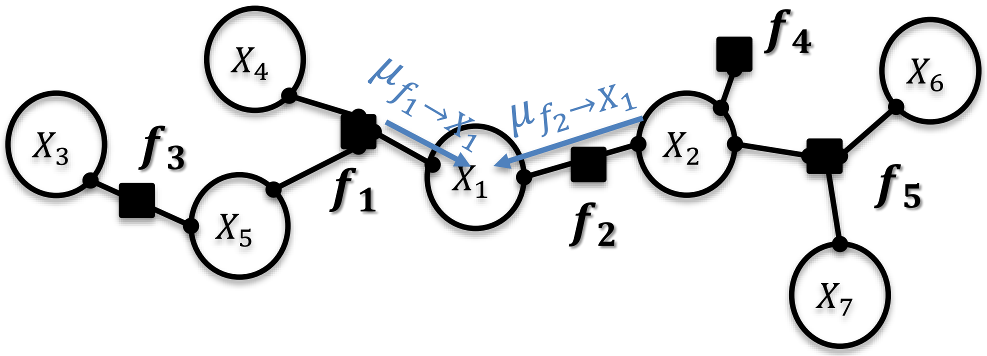
The Sum-Product Algorithm (1)

Objectives:

- To obtain an efficient, exact inference algorithm for finding marginals
- To share efficiently computations when several marginals are required

Key ideas:

- Let's focus on undirected trees and polytrees
- Let's exploit $ab + ac = a(b + c)$



$$P(X_i) = \sum_{X_1,X_2,\ldots,X_{i-1},X_{i+1},\ldots X_N} P(X_1, X_2, \ldots, X_N) = \sum_{X \setminus X_i} P(X_1, X_2, \ldots, X_N)$$

$$P(X_i) = \sum_{X \setminus X_i} \prod_{s \in ne(X_i)} F_s(X_i, X_s) = \prod_{s \in ne(X_i)} \sum_{X_s} F_s(X_i, X_s) = \prod_{s \in ne(X_i)} \boxed{\mu_{f_s \to X_i}(X_i)}$$

$$P(X_1, X_2, \ldots, X_N) = \prod_{s'} f_{s'}(X_{s'}) = \prod_{s \in ne(X_i)} F_s(X_i, X_s)$$

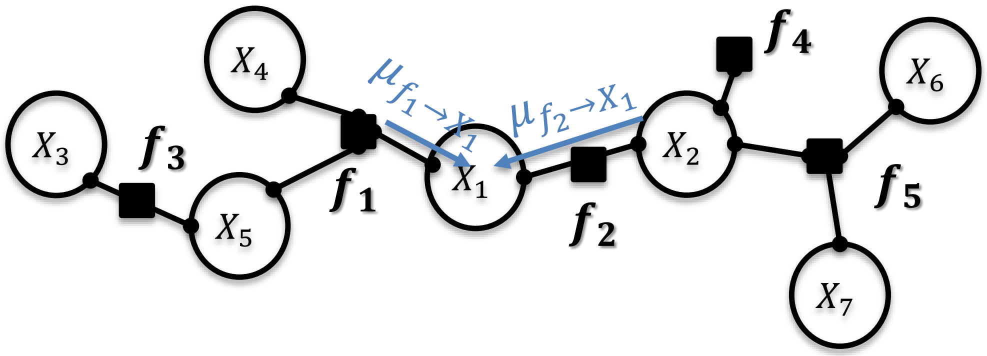
The Sum-Product Algorithm (1)

Objectives:

- To obtain an efficient, exact inference algorithm for finding marginals
- To share efficiently computations when several marginals are required

Key ideas:

- Let's focus on undirected trees and polytrees
- Let's exploit $ab + ac = a(b + c)$



$$P(X_i) = \sum_{X_1,X_2,\dots,X_{i-1},X_{i+1},\dots X_N} P(X_1,X_2,\dots,X_N) = \sum_{X\setminus X_i} P(X_1,X_2,\dots,X_N)$$

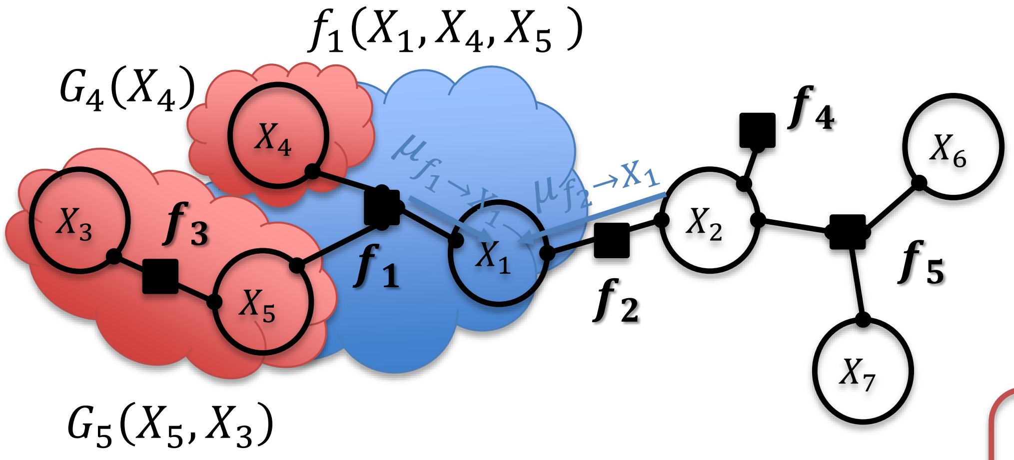
$$P(X_i) = \sum_{X\setminus X_i} \prod_{s\in ne(X_i)} F_s(X_i,X_s) = \prod_{s\in ne(X_i)} \sum_{X_s} F_s(X_i,X_s) = \prod_{s\in ne(X_i)} \mu_{f_s\to X_i}(X_i)$$

$$P(X_1,X_2,\dots,X_N) = \prod_{s'} f_{s'}(X_{s'}) =$$

Messages from factors to variables

The Sum-Product Algorithm (2)

Marginals are computed multiplying messages from factors to variables



$$P(X_i) = \sum_{X_1,X_2,\dots,X_{i-1},X_{i+1},\dots X_N} P(X_1,X_2,\dots,X_N) = \sum_{X\backslash X_i} P(X_1,X_2,\dots,X_N)$$

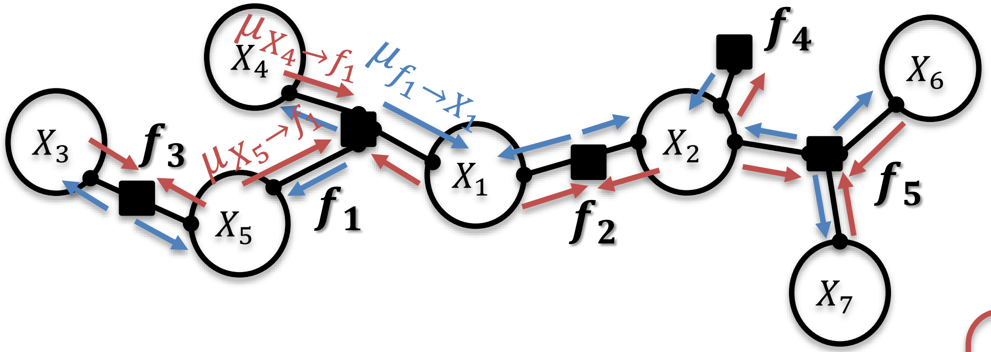
$$P(X_i) = \sum_{X\backslash X_i} \prod_{s\in ne(X_i)} F_s(X_i,X_s) = \prod_{s\in ne(X_i)} \sum_{X_s} F_s(X_i,X_s) = \prod_{s\in ne(X_i)} \boxed{\mu_{f_s\to X_i}(X_i)}$$

$$X_s = \{x_1, x_2, \dots, x_m, \dots, x_M\}$$

$$F_s(X_i,X_s) = \boxed{f_s(X_i,x_1,\dots,x_M)}\,\boxed{G_1\left(x_1, X_{s(x_1)}\right)} \dots \boxed{G_M\left(x_M, X_{s(x_M)}\right)}$$

The Sum-Product Algorithm (2)

Marginals are computed multiplying messages from factors to variables



$$P(X_i) = \sum_{X_1, X_2, \dots, X_{i-1}, X_{i+1}, \dots X_N} P(X_1, X_2, \dots, X_N) = \sum_{X \setminus X_i} P(X_1, X_2, \dots, X_N)$$

$$P(X_i) = \sum_{X \setminus X_i} \prod_{s \in ne(X_i)} F_s(X_i, X_s) = \prod_{s \in ne(X_i)} \sum_{X_s} F_s(X_i, X_s) = \prod_{s \in ne(X_i)} \boxed{\mu_{f_s \to X_i}(X_i)}$$

$$X_s = \{x_1, x_2, \dots, x_m, \dots, x_M\}$$

$$F_s(X_i, X_s) = \boxed{f_s(X_i, x_1, \dots, x_M)} \boxed{G_1(x_1, X_{s(x_1)})} \dots \boxed{G_M(x_M, X_{s(x_M)})}$$

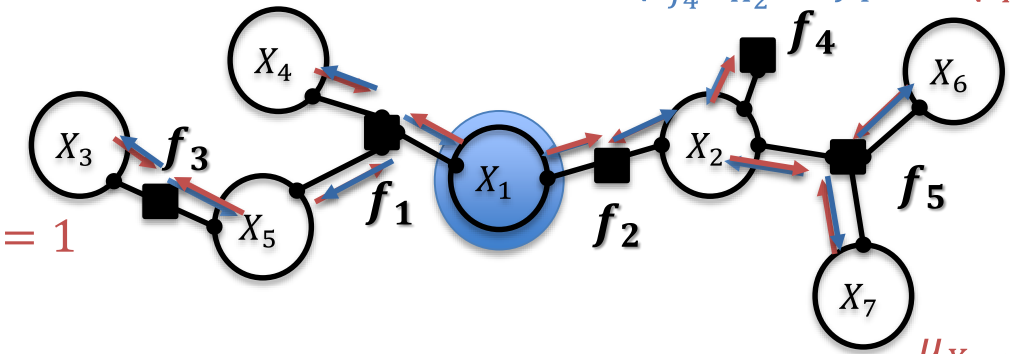
$$\mu_{f_{s \to X_i}}(X_i) = \sum_{x_1} \dots \sum_{x_M} f(X_i, x_1, \dots, x_M) \prod_{m \in ne(f_s) \setminus X_i} \sum_{X_{s_m}} G_m(x_m, X_{s_m}) = \sum_{x_1} \dots \sum_{x_M} f(X_i, x_1, \dots, x_M) \prod_{m \in ne(f_s) \setminus X_i} \boxed{\mu_{x_m \to f_s}(x_m)}$$

$$\mu_{x_m \to f_s}(x_m) = \sum_{X_{s_m}} G_m(x_m, X_{s_m}) = \sum_{s_m} \prod_{l \in ne(x_m) \setminus f_s} F_l(x_m, X_{m_l}) = \prod_{l \in ne(x_m) \setminus f_s} \boxed{\mu_{f_{l \to x_m}}(x_m)}$$

The Sum-Product Algorithm (3)

- Pick an arbitrary node as root, and initialize leaves
- Compute and propagate messages from the leaf nodes to the root, (storing received messages at every node)
- Compute and propagate messages from the root to the leaf nodes, (storing received messages at every node)
- Compute the product of received messages at each node for which the marginal is required, and normalize if necessary.

$\mu_{X_4 \to f_1} = 1$ $\mu_{f_4 \to X_2} = f_4$ $\mu_{X_6 \to f_5} = 1$



$\mu_{X_3 \to f_3} = 1$ $\mu_{X_7 \to f_5} = 1$

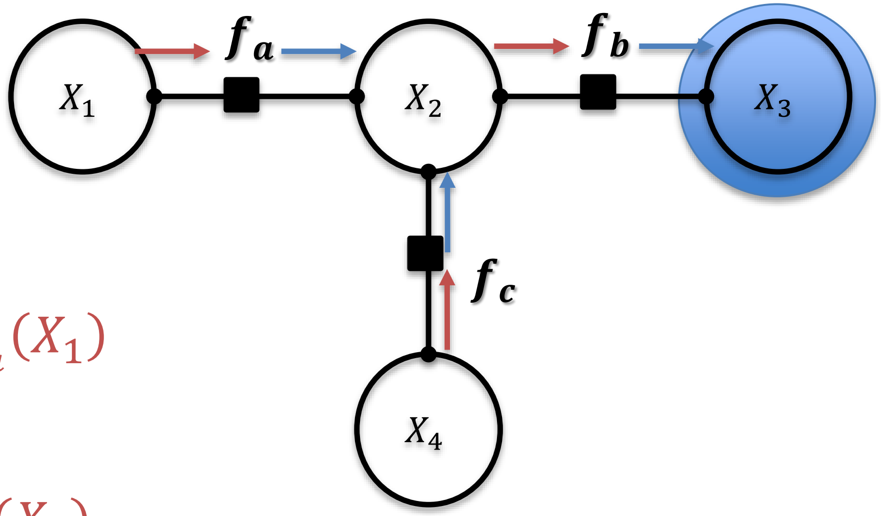
$$\mu_{f_s \to X_i}(X_i) = \sum_{X_s \setminus X_i} f_s(X_s) \prod_{j \in s, j \neq i} \mu_{X_j \to f_s}(X_j)$$

$$\mu_{X_i \to f_s}(X_i) = \prod_{f_l \in ne(X_i), f_l \neq f_s} \mu_{f_l \to X_i}(X_i)$$

$$P(X_i) = \prod_{f_s \in ne(X_i)} \mu_{f_s \to X_i}(X_s)$$

Sum-Product: Example (1)

$$P(X_1, X_2, X_3, X_4) = f_a(X_1, X_2) f_b(X_2, X_3) f_c(X_2, X_4)$$



$$\mu_{X_1 \to f_a}(X_1) = 1$$

$$\mu_{X_4 \to f_c}(X_4) = 1$$

$$\mu_{f_a \to X_2}(X_2) = \sum_{X_s \setminus X_2} f_a\,(X_s) \prod_{j \in s, j \neq 2} \mu_{X_j \to f_a}(X_j) = \sum_{X_1} f_a(X_1, X_2)\, \mu_{X_1 \to f_a}(X_1)$$

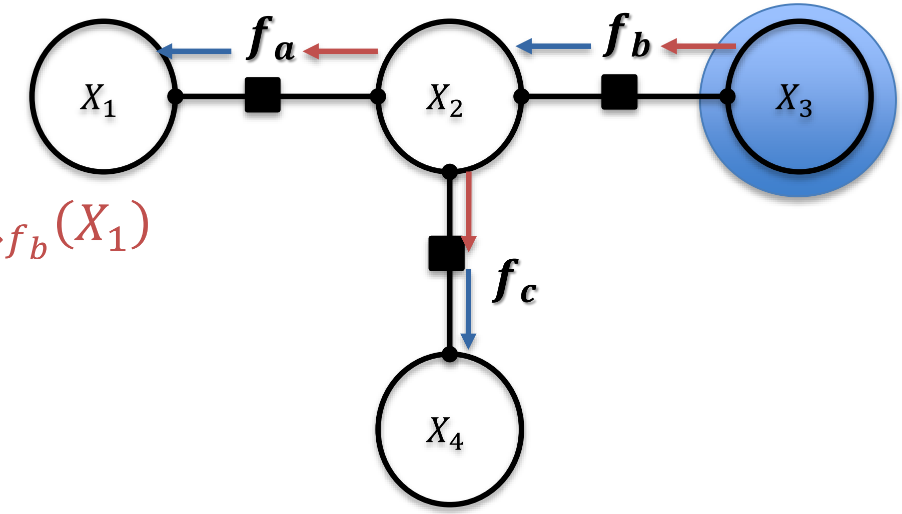
$$\mu_{f_c \to X_2}(X_2) = \sum_{X_s \setminus X_2} f_c\,(X_s) \prod_{j \in s, j \neq 2} \mu_{X_j \to f_c}(X_j) = \sum_{X_4} f_c(X_4, X_2)\, \mu_{X_4 \to f_c}(X_4)$$

$$\mu_{X_2 \to f_b}(X_2) = \prod_{\substack{f_l \in ne(X_2) \\ f_l \neq f_b}} \mu_{f_l \to X_2}(X_2) = \mu_{f_a \to X_2}(X_2) \mu_{f_c \to X_2}(X_2)$$

$$\mu_{f_b \to X_3}(X_3) = \sum_{X_s \setminus X_3} f_b\,(X_s) \prod_{j \in s, j \neq 3} \mu_{X_j \to f_b}(X_j) = \sum_{X_2} f_b(X_2, X_3)\, \mu_{X_2 \to f_b}(X_2)$$

Sum-Product: Example (1)

$$P(X_1, X_2, X_3, X_4) = f_a(X_1, X_2) f_b(X_2, X_3) f_c(X_2, X_4)$$



$$\mu_{X_3 \to f_b}(X_3) = 1$$

$$\mu_{f_b \to X_2}(X_2) = \sum_{X_s \setminus X_2} f_b\,(X_s) \prod_{j \in s, j \neq 2} \mu_{X_j \to f_b}(X_j) = \sum_{X_3} f_b(X_2, X_3)\, \mu_{X_3 \to f_b}(X_1)$$

$$\mu_{X_2 \to f_a}(X_2) = \prod_{\substack{f_l \in ne(X_2) \\ f_l \neq f_a}} \mu_{f_l \to X_2}(X_2) = \mu_{f_b \to X_2}(X_2) \mu_{f_c \to X_2}(X_2)$$

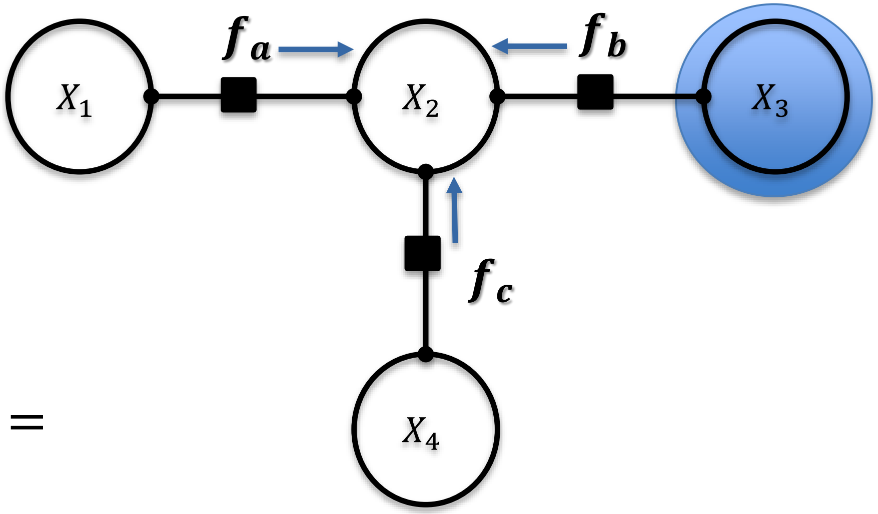
$$\mu_{X_2 \to f_c}(X_2) = \prod_{\substack{f_l \in ne(X_2) \\ f_l \neq f_c}} \mu_{f_l \to X_2}(X_2) = \mu_{f_a \to X_2}(X_2) \mu_{f_b \to X_2}(X_2)$$

$$\mu_{f_a \to X_1}(X_1) = \sum_{X_s \setminus X_1} f_a\,(X_s) \prod_{j \in s, j \neq 1} \mu_{X_j \to f_a}(X_j) = \sum_{X_2} f_a(X_1, X_2)\, \mu_{X_2 \to f_a}(X_2)$$

$$\mu_{f_c \to X_4}(X_4) = \sum_{X_s \setminus X_4} f_c\,(X_s) \prod_{j \in s, j \neq 4} \mu_{X_j \to f_c}(X_j) = \sum_{X_2} f_c(X_2, X_4)\, \mu_{X_2 \to f_c}(X_2)$$

Sum-Product: Example (1)

$$P(X_1, X_2, X_3, X_4) = f_a(X_1, X_2) f_b(X_2, X_3) f_c(X_2, X_4)$$



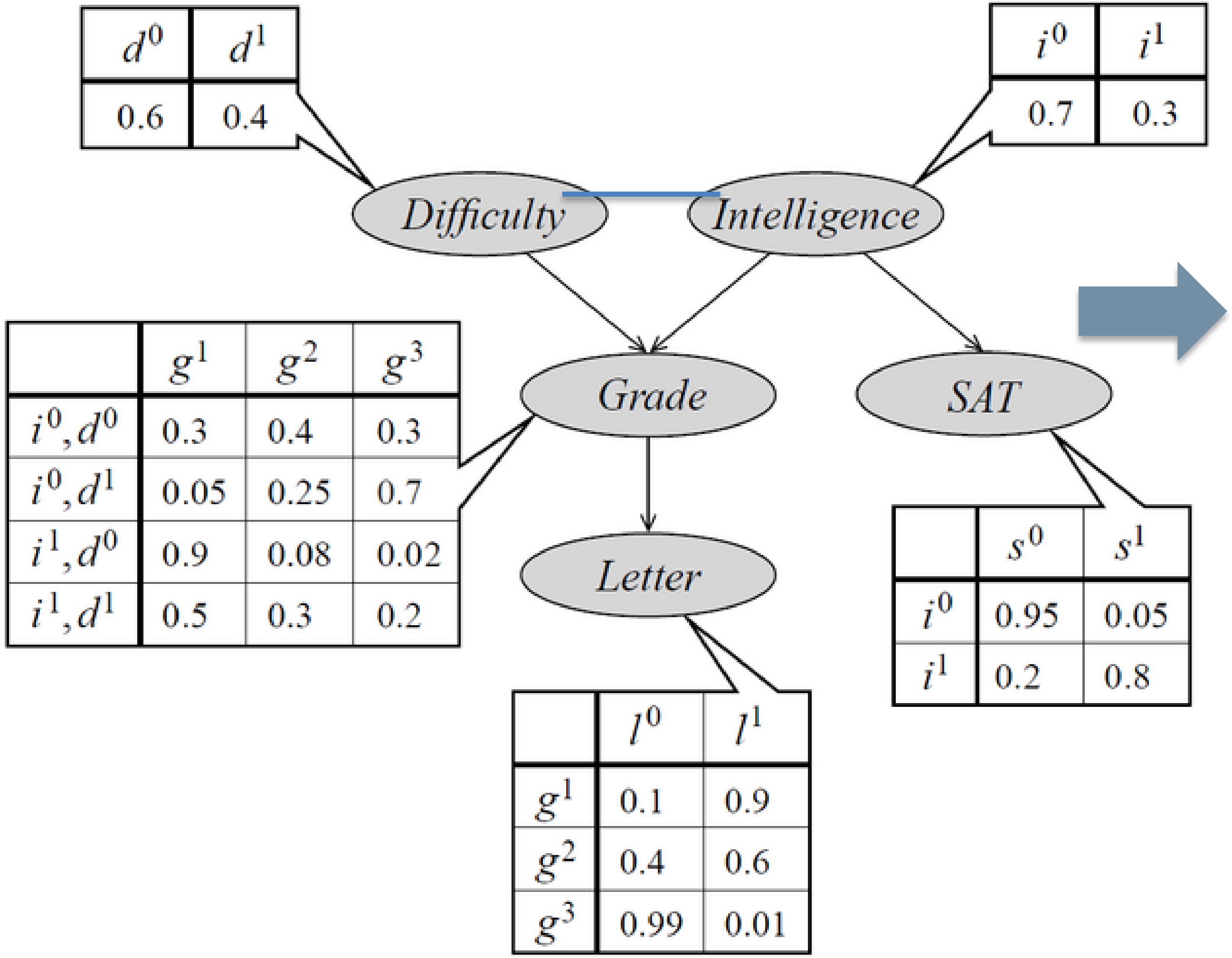
$$P(X_2) = \prod_{f_s \in ne(X_2)} \mu_{f_s \to X_2}(X_s) = \mu_{f_a \to X_2}(X_2)\mu_{f_b \to X_2}(X_2)\mu_{f_c \to X_2}(X_2) =$$

$$= \sum_{X_1} f_a(X_1, X_2)\, \mu_{X_1 \to f_a}(X_1) \sum_{X_3} f_b(X_2, X_3)\, \mu_{X_3 \to f_b}(X_1) \sum_{X_4} f_c(X_4, X_2)\, \mu_{X_4 \to f_c}(X_4) =$$

$$= \sum_{X_1} f_a(X_1, X_2) \sum_{X_3} f_b(X_2, X_3) \sum_{X_4} f_c(X_4, X_2) = \sum_{X_1} \sum_{X_3} \sum_{X_4} f_a(X_1, X_2) f_b(X_2, X_3) f_c(X_4, X_2) =$$

$$= \sum_{X_1} \sum_{X_3} \sum_{X_4} P(X_1, X_2, X_3, X_4)$$

Belief Propagation Example



d^0	d^1
0.6	0.4

i^0	i^1
0.7	0.3

	g^1	g^2	g^3
i^0, d^0	0.3	0.4	0.3
i^0, d^1	0.05	0.25	0.7
i^1, d^0	0.9	0.08	0.02
i^1, d^1	0.5	0.3	0.2

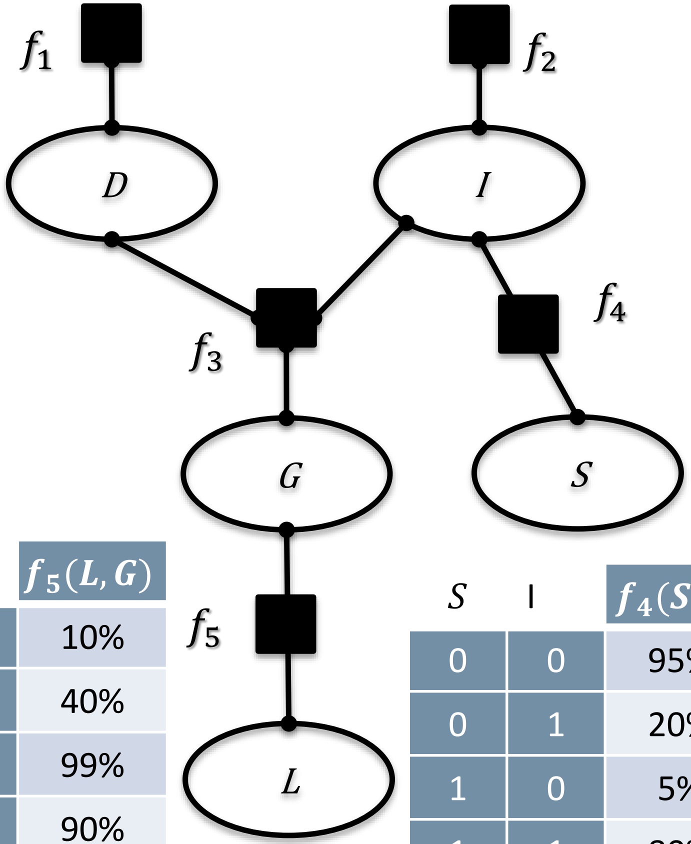
	s^0	s^1
i^0	0.95	0.05
i^1	0.2	0.8

	l^0	l^1
g^1	0.1	0.9
g^2	0.4	0.6
g^3	0.99	0.01

G	I	D	$f_3(G,I,D)$
1	0	0	30%
1	0	1	5%
1	1	0	90%
1	1	1	50%
2	0	0	40%
2	0	1	25%
2	1	0	8%
2	1	1	30%
3	0	0	30%
3	0	1	70%
3	1	0	2%
3	1	1	20%

D	$f_1(D)$
0	60%
1	40%

I	$f_2(I)$
0	70%
1	30%



L	G	$f_5(L,G)$
0	1	10%
0	2	40%
0	3	99%
1	1	90%
1	2	60%
1	3	1%

S	I	$f_4(S,I)$
0	0	95%
0	1	20%
1	0	5%
1	1	80%

Belief Propagation Example

$$\mu_{L\to 5}(L) = 1 \qquad \mu_{S\to 4}(S) = 1$$

$$\mu_{1\to D}(D) = f_1(D) \qquad \mu_{2\to I}(I) = f_2(I)$$

$$\mu_{5\to G}(G) = \sum_{X_s\setminus G} f_5\,(X_s) \prod_{j\in s, j\neq 5} \mu_{j\to 5}(X_j) =$$

$$= \sum_{L} f_5(L,G)\,\mu_{L\to 5}(L) = \sum_{L} f_5(L,G)$$

G	$\mu_{5\to G}\,(G)$	$\mu_{5\to G}\,(G)$
1	0.1+0.9	1
2	0.4+60	1
3	0.01+0.99	1

$$\mu_{4\to I}(I) = \sum_{X_s\setminus I} f_4\,(X_s) \prod_{j\in s, j\neq 4} \mu_{j\to 4}(X_j) =$$

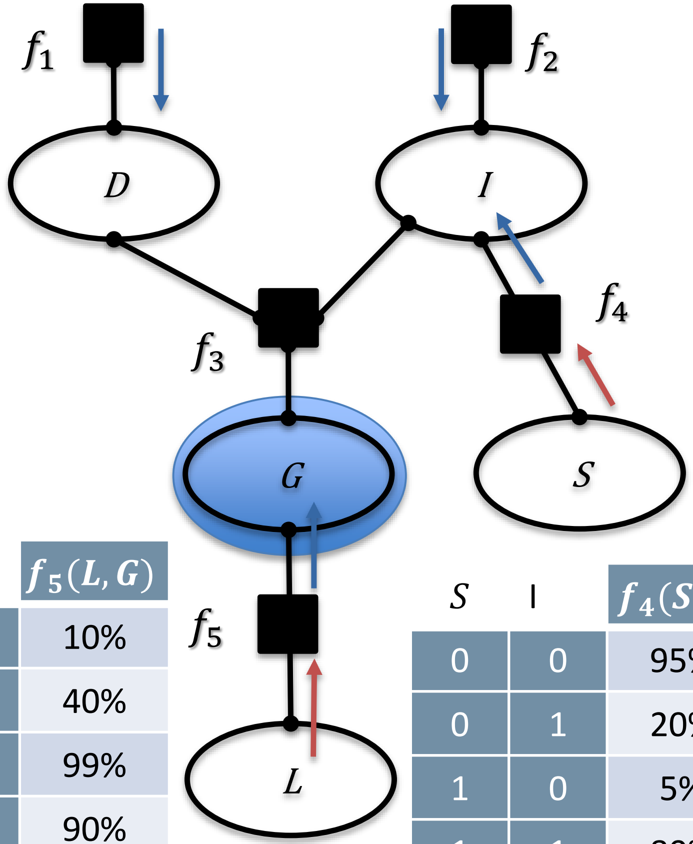
$$= \sum_{S} f_4(S,I)\,\mu_{S\to 4}(S) = \sum_{S} f_4(S,I)$$

I	$\mu_{4\to I}\,(I)$	$\mu_{4\to I}\,(I)$
1	0.05+0.95	1
2	0.2+0.8	1

G	I	D	$f_3(G,I,D)$
1	0	0	30%
1	0	1	5%
1	1	0	90%
1	1	1	50%
2	0	0	40%
2	0	1	25%
2	1	0	8%
2	1	1	30%
3	0	0	30%
3	0	1	70%
3	1	0	2%
3	1	1	20%

D	$f_1(D)$
0	60%
1	40%

I	$f_2(I)$
0	70%
1	30%



L	G	$f_5(L,G)$
0	1	10%
0	2	40%
0	3	99%
1	1	90%
1	2	60%
1	3	1%

S	I	$f_4(S,I)$
0	0	95%
0	1	20%
1	0	5%
1	1	80%

Belief Propagation Example

$\mu_{L\to 5}(L) = 1$ $\mu_{S\to 4}(S) = 1$

$\mu_{1\to D}(D) = f_1(D)$ $\mu_{2\to I}(I) = f_2(I)$

$\mu_{5\to G}(G) =$

G	$\mu_{5\to G}\,(G)$	$\mu_{5\to G}\,(G)$
1	0.1+0.9	1
2	0.4+60	1
3	0.01+0.99	1

$\mu_{4\to I}(I) =$

I	$\mu_{4\to I}\,(I)$	$\mu_{4\to I}\,(I)$
1	0.05+0.95	1
2	0.2+0.8	1

$$\mu_{D\to 3}(D) = \prod_{\substack{f_l \in ne(D) \\ f_l \neq f_3}} \mu_{f_l\to D}(D) = \mu_{1\to D}(D)$$

D	$\mu_{D\to 3}(D)$
0	0.6
1	0.4

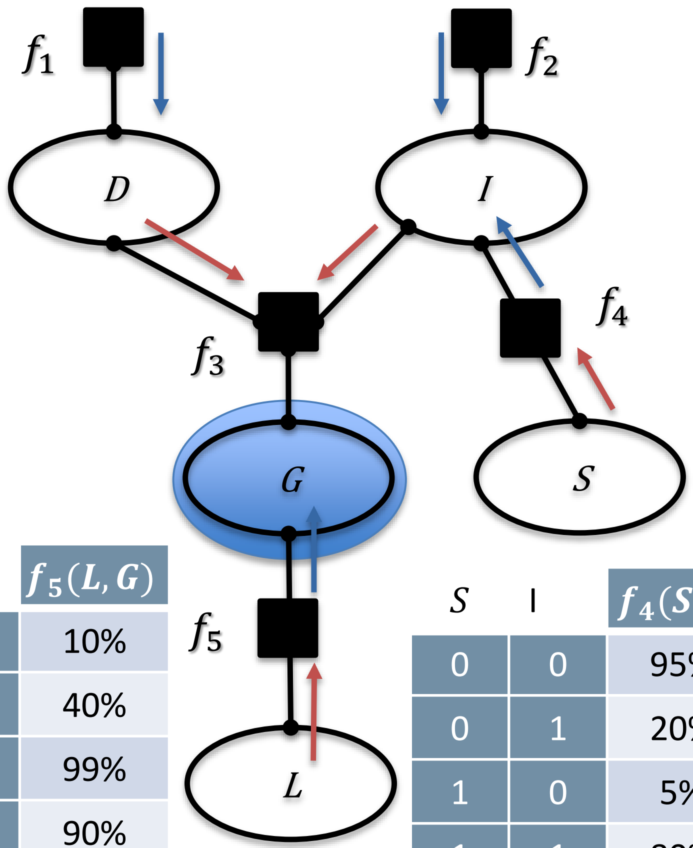
$$\mu_{I\to 3}(I) = \prod_{\substack{f_l \in ne(I) \\ f_l \neq f_3}} \mu_{f_l\to I}(I) = \mu_{2\to I}(I)\mu_{4\to I}(I)$$

I	$\mu_{I\to 3}(I)$
0	0.7
1	0.3

G	I	D	$f_3(G,I,D)$
1	0	0	30%
1	0	1	5%
1	1	0	90%
1	1	1	50%
2	0	0	40%
2	0	1	25%
2	1	0	8%
2	1	1	30%
3	0	0	30%
3	0	1	70%
3	1	0	2%
3	1	1	20%

D	$f_1(D)$
0	60%
1	40%

I	$f_2(I)$
0	70%
1	30%



L	G	$f_5(L,G)$
0	1	10%
0	2	40%
0	3	99%
1	1	90%
1	2	60%
1	3	1%

S	I	$f_4(S,I)$
0	0	95%
0	1	20%
1	0	5%
1	1	80%

Belief Propagation Example

$\mu_{L\to 5}(L) = 1$ $\quad$ $\mu_{S\to 4}(S) = 1$

$\mu_{1\to D}(D) = f_1(D)$ $\quad$ $\mu_{2\to I}(I) = f_2(I)$

$\mu_{5\to G}(G) =$

G	$\mu_{5\to G}\ (G)$	$\mu_{5\to G}\ (G)$
1	0.1+0.9	1
2	0.4+60	1
3	0.01+0.99	1

$\mu_{4\to I}(I) =$

I	$\mu_{4\to I}\ (I)$	$\mu_{4\to I}\ (I)$
1	0.05+0.95	1
2	0.2+0.8	1

$\mu_{D\to 3}(D) =$

D	$\mu_{D\to 3}(D)$
0	0.6
1	0.4

$\mu_{I\to 3}(I) =$

I	$\mu_{I\to 3}(I)$
0	0.7
1	0.3

$$\mu_{3\to G}(G) = \sum_{X_S\setminus G} f_3\,(X_s) \prod_{j\in s, j\neq 3} \mu_{j\to 3}(X_j) =$$

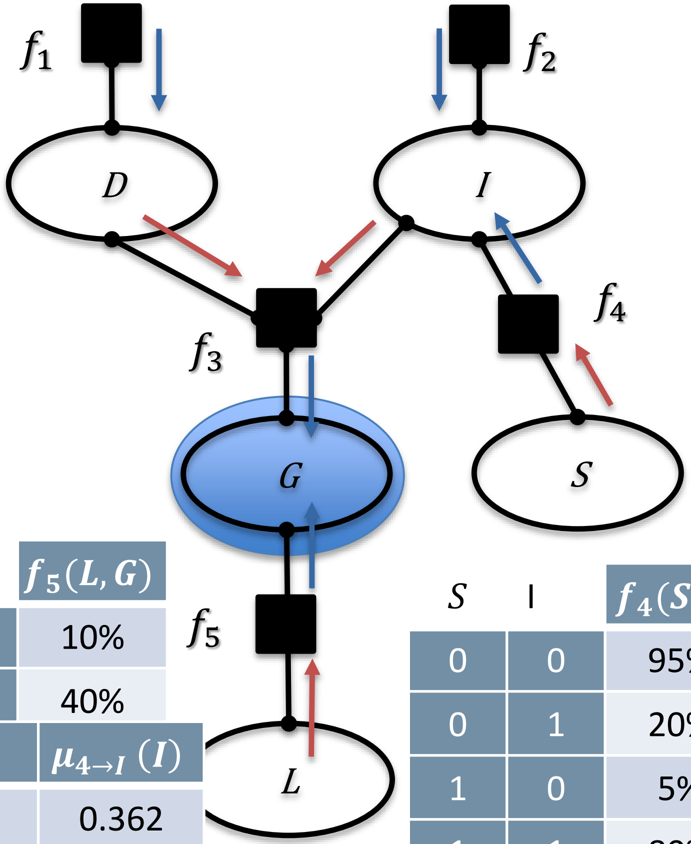
$$= \sum_{D,I} f_3(G,I,D)\,\mu_{D\to 3}(D)\mu_{I\to 3}(I)$$

G	$\mu_{3\to G}\ (G)$	$\mu_{4\to I}\ (I)$
1	.3*.7*.6+.05*.7*.4+.9*.3*.6+.5*.3*.4	0.362
2	.4*.7*.6+.25*.7*.4+.08*.3*.6+.3*.3*.4	0.2884
3	.3*.7*.6+.7*.7*.4+.02*.3*.6+.2*.3*.4	0.3496

G	I	D	$f_3(G,I,D)$
1	0	0	30%
1	0	1	5%
1	1	0	90%
1	1	1	50%
2	0	0	40%
2	0	1	25%
2	1	0	8%
2	1	1	30%
3	0	0	30%
3	0	1	70%
3	1	0	2%
3	1	1	20%

D	$f_1(D)$
0	60%
1	40%

I	$f_2(I)$
0	70%
1	30%



L	G	$f_5(L,G)$
0	1	10%
0	2	40%

S	I	$f_4(S,I)$
0	0	95%
0	1	20%
1	0	5%
1	1	80%

Belief Propagation Example

$\mu_{L\to 5}(L) = 1$ $\mu_{S\to 4}(S) = 1$

$\mu_{1\to D}(D) = f_1(D)$ $\mu_{2\to I}(I) = f_2(I)$

$\mu_{5\to G}(G) =$

G	$\mu_{5\to G}(G)$	$\mu_{5\to G}(G)$
1	0.1+0.9	1
2	0.4+60	1
3	0.01+0.99	1

$\mu_{4\to I}(I) =$

I	$\mu_{4\to I}(I)$	$\mu_{4\to I}(I)$
1	0.05+0.95	100%
2	0.2+0.8	100%

$\mu_{D\to 3}(D) =$

D	$\mu_{D\to 3}(D)$
0	60%
1	40%

$\mu_{I\to 3}(I) =$

I	$\mu_{I\to 3}(I)$
0	70%
1	30%

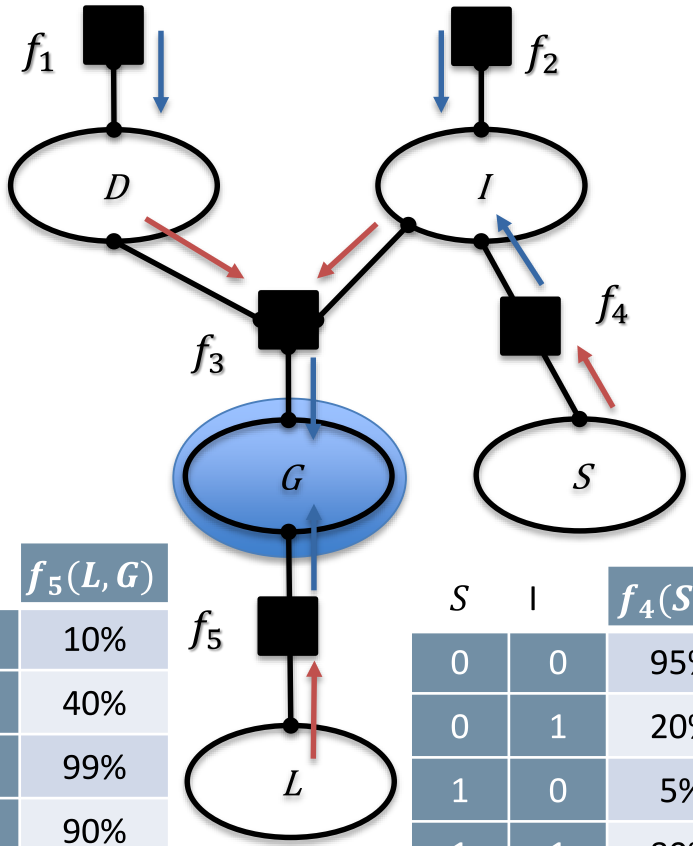
$\mu_{3\to G}(G) =$

G	$\mu_{3\to G}(G)$	$\mu_{3\to G}(G)$
1	.3*.7*.6+.05*.7*.4+.9*.3*.6+.5*.3*.4	0.362
2	.4*.7*.6+.25*.7*.4+.08*.3*.6+.3*.3*.4	0.2884
3	.3*.7*.6+.7*.7*.4+.02*.3*.6+.2*.3*.4	0.3496

G	I	D	$f_3(G,I,D)$
1	0	0	30%
1	0	1	5%
1	1	0	90%
1	1	1	50%
2	0	0	40%
2	0	1	25%
2	1	0	8%
2	1	1	30%
3	0	0	30%
3	0	1	70%
3	1	0	2%
3	1	1	20%

D	$f_1(D)$
0	60%
1	40%

I	$f_2(I)$
0	70%
1	30%



L	G	$f_5(L,G)$
0	1	10%
0	2	40%
0	3	99%
1	1	90%
1	2	60%
1	3	1%

S	I	$f_4(S,I)$
0	0	95%
0	1	20%
1	0	5%
1	1	80%

Belief Propagation Example

$\mu_{G\to 3}(G) = 1 \qquad \mu_{G\to 5}(G) = 1$

$$\mu_{3\to D}(D) = \sum_{X_s\setminus D} f_3\,(X_s) \prod_{j\in s, j\neq 3} \mu_{j\to 3}(X_j) =$$

$$= \sum_{I,G} f_3(G,I,D)\,\mu_{G\to 3}(G)\mu_{I\to 3}(I) =$$

I	$\mu_{I\to 3}(I)$
0	0.7
1	0.3

D	$\mu_{3\to D}\,(D)$	$\mu_{3\to D}\,(D)$
0	.3*1*.7+.9*1*.3+.4*1*.7+.08 *1*.3+.3*1*.7+.02*1*.3	1
1	.05*1*.7+.5*1*.3+.25*1*.7+. 3*1*.3+.7*1*.7+.2*1*.3	1

$$\mu_{3\to I}(I) = \sum_{X_s\setminus I} f_3\,(X_s) \prod_{j\in s, j\neq 3} \mu_{j\to 3}(X_j) =$$

$$= \sum_{D,G} f_3(G,I,D)\,\mu_{G\to 3}(G)\mu_{D\to 3}(D) =$$

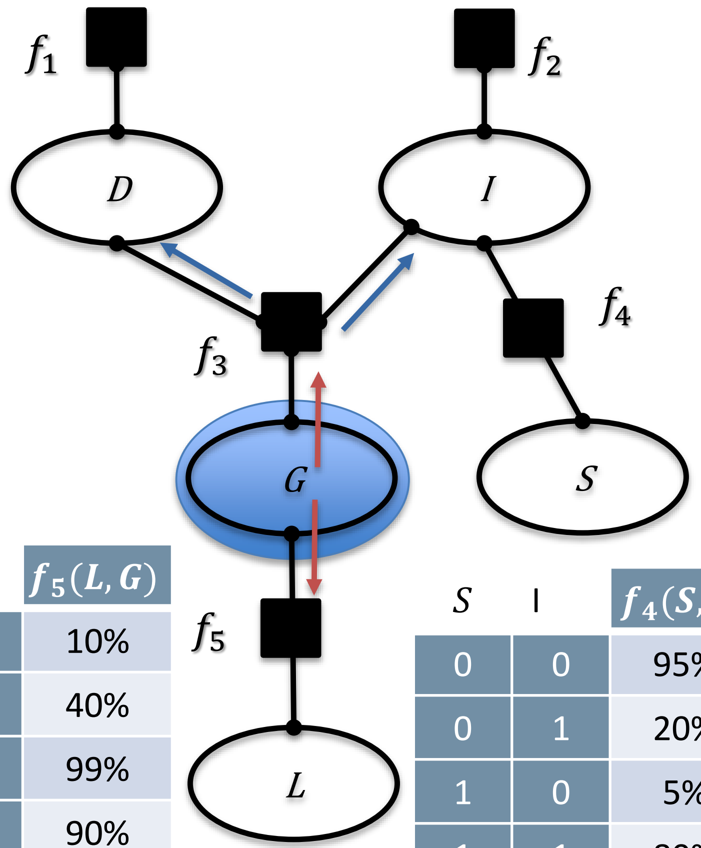
D	$\mu_{D\to 3}(D)$
0	0.6
1	0.4

I	$\mu_{3\to I}\,(I)$	$\mu_{3\to I}\,(I)$
0	.3*1*.6+.05*1*.4+.4*1*.6+.2 5*1*.4+.3*1*.6+.7*1*.4	1
1	.9*1*.6+.5*1*.4+.08*1*.6+.3 *1*.4+.02*1*.6+.2*1*.4	1

G	I	D	$f_3(G,I,D)$
1	0	0	30%
1	0	1	5%
1	1	0	90%
1	1	1	50%
2	0	0	40%
2	0	1	25%
2	1	0	8%
2	1	1	30%
3	0	0	30%
3	0	1	70%
3	1	0	2%
3	1	1	20%

D	$f_1(D)$
0	60%
1	40%

I	$f_2(I)$
0	70%
1	30%



L	G	$f_5(L,G)$
0	1	10%
0	2	40%
0	3	99%
1	1	90%
1	2	60%
1	3	1%

S	I	$f_4(S,I)$
0	0	95%
0	1	20%
1	0	5%
1	1	80%

Belief Propagation Example

$\mu_{G\to3}(G)=1 \qquad \mu_{G\to5}(G)=1$

$\mu_{3\to D}(D)=$

D	$\mu_{3\to D}\ (D)$	$\mu_{3\to D}\ (D)$
0	.3*1*.7+.9*1*.3+.4*1*.7+.08 *1*.3+.3*1*.7+.02*1*.3	1
1	.05*1*.7+.5*1*.3+.25*1*.7+. 3*1*.3+.7*1*.7+.2*1*.3	1

$\mu_{3\to I}(I)=$

I	$\mu_{3\to I}\ (I)$	$\mu_{3\to I}\ (I)$
0	.3*1*.6+.05*1*.4+.4*1*.6+.2 5*1*.4+.3*1*.6+.7*1*.4	1
1	.9*1*.6+.5*1*.4+.08*1*.6+.3 *1*.4+.02*1*.6+.2*1*.4	1

$$\mu_{5\to L}(L)=\sum_{X_s\setminus L} f_5\,(X_s)\prod_{j\in s, j\neq 5}\mu_{j\to5}(X_j)=$$

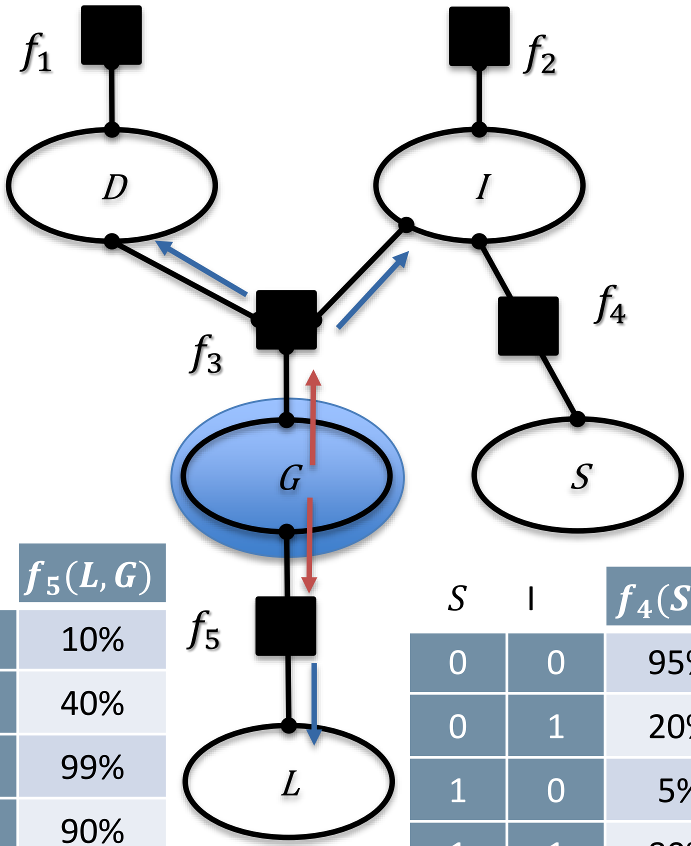
$$=\sum_{G} f_5(L,G)\,\mu_{G\to5}(G)=$$

L	$\mu_{5\to L}\ (L)$	$\mu_{5\to L}\ (L)$
0	.1*1+.4*1+.99*1	1.49
1	.9*1+.6*1+.01*1	1.51

G	I	D	$f_3(G,I,D)$
1	0	0	30%
1	0	1	5%
1	1	0	90%
1	1	1	50%
2	0	0	40%
2	0	1	25%
2	1	0	8%
2	1	1	30%
3	0	0	30%
3	0	1	70%
3	1	0	2%
3	1	1	20%

D	$f_1(D)$
0	60%
1	40%

I	$f_2(I)$
0	70%
1	30%



L	G	$f_5(L,G)$
0	1	10%
0	2	40%
0	3	99%
1	1	90%
1	2	60%
1	3	1%

S	I	$f_4(S,I)$
0	0	95%
0	1	20%
1	0	5%
1	1	80%

Belief Propagation Example

$\mu_{G\to 3}(G) = 1 \qquad \mu_{G\to 5}(G) = 1$

$\mu_{3\to D}(D) = 1$

$\mu_{3\to I}(I) = 1$

$\mu_{5\to L}(L) =$

L	$\mu_{5\to L}\ (L)$	$\mu_{5\to L}\ (L)$
0	.1*1+.4*1+.99*1	1.49
1	.9*1+.6*1+.01*1	1.51

$$\mu_{D\to 1}(D) = \prod_{\substack{f_l \in ne(D) \\ f_l \neq f_1}} \mu_{f_l\to D}(D) = \mu_{3\to D}(D) = 1$$

$$\mu_{I\to 2}(I) = \prod_{\substack{f_l \in ne(I) \\ f_l \neq f_2}} \mu_{f_l\to I}(I) = \mu_{3\to I}(I)\mu_{4\to I}(I) =$$

I	$\mu_{I\to 2}\ (I)$	$\mu_{I\to 2}\ (2)$
0	1*1	1
1	1*1	1

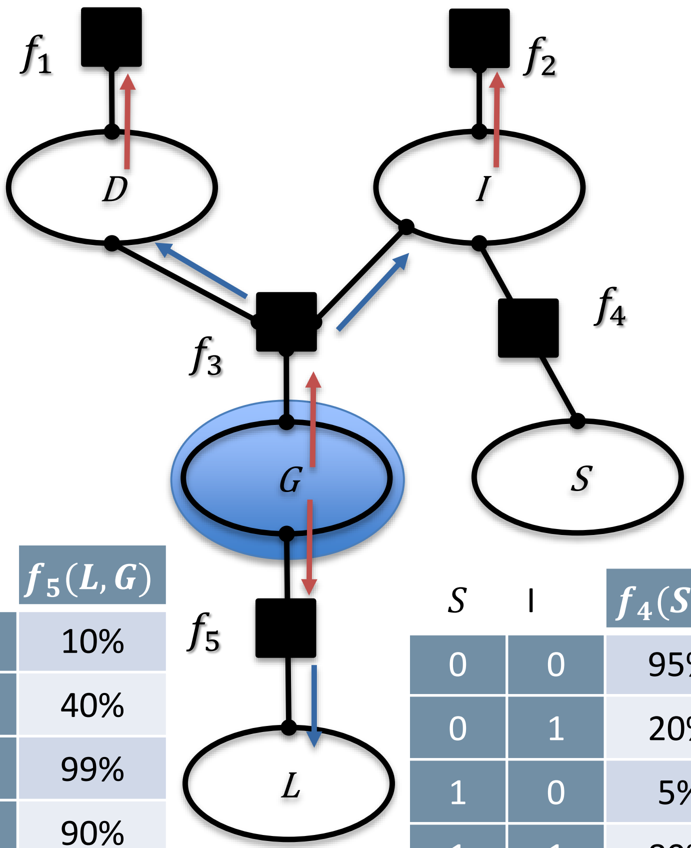
$\mu_{4\to I}(I) =$

I	$\mu_{4\to I}\ (I)$	$\mu_{4\to I}\ (I)$
1	0.05+0.95	1
2	0.2+0.8	1

G	I	D	$f_3(G,I,D)$
1	0	0	30%
1	0	1	5%
1	1	0	90%
1	1	1	50%
2	0	0	40%
2	0	1	25%
2	1	0	8%
2	1	1	30%
3	0	0	30%
3	0	1	70%
3	1	0	2%
3	1	1	20%

D	$f_1(D)$
0	60%
1	40%

I	$f_2(I)$
0	70%
1	30%



L	G	$f_5(L,G)$
0	1	10%
0	2	40%
0	3	99%
1	1	90%
1	2	60%
1	3	1%

S	I	$f_4(S,I)$
0	0	95%
0	1	20%
1	0	5%
1	1	80%

Belief Propagation Example

$\mu_{G\to 3}(G) = 1 \qquad \mu_{G\to 5}(G) = 1$

$\mu_{3\to D}(D) = 1$

$\mu_{3\to I}(I) = 1$

$\mu_{5\to L}(L) =$

L	$\mu_{5\to L}(L)$	$\mu_{5\to L}(L)$
0	.1*1+.4*1+.99*1	1.49
1	.9*1+.6*1+.01*1	1.51

$\mu_{D\to 1}(D) = 1$

$\mu_{I\to 2}(I) =$

I	$\mu_{I\to 2}(I)$	$\mu_{I\to 2}(I)$
0	1*1	1
1	1*1	1

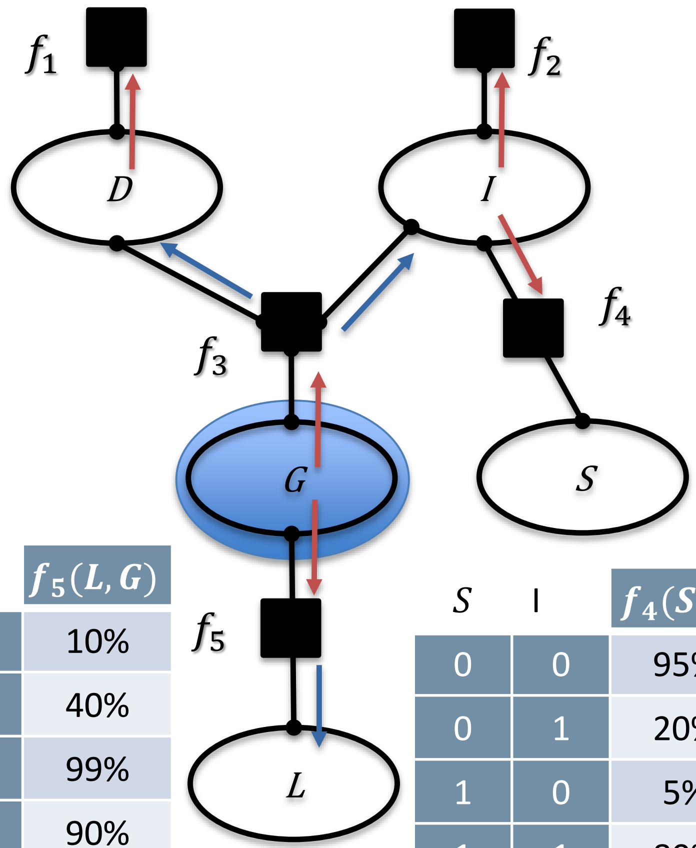
$$\mu_{I\to 4}(I) = \prod_{\substack{f_l \in ne(I)\\ f_l \neq f_4}} \mu_{f_l\to I}(I) = \mu_{3\to I}(I)\mu_{2\to I}(I) =$$

I	$\mu_{I\to 4}(I)$	$\mu_{I\to 4}(I)$
0	1*.7	0.7
1	1*.3	0.3

G	I	D	$f_3(G,I,D)$
1	0	0	30%
1	0	1	5%
1	1	0	90%
1	1	1	50%
2	0	0	40%
2	0	1	25%
2	1	0	8%
2	1	1	30%
3	0	0	30%
3	0	1	70%
3	1	0	2%
3	1	1	20%

D	$f_1(D)$
0	60%
1	40%

I	$f_2(I)$
0	70%
1	30%



L	G	$f_5(L,G)$
0	1	10%
0	2	40%
0	3	99%
1	1	90%
1	2	60%
1	3	1%

S	I	$f_4(S,I)$
0	0	95%
0	1	20%
1	0	5%
1	1	80%

Belief Propagation Example

$\mu_{G\to 3}(G) = 1 \qquad \mu_{G\to 5}(G) = 1$

$\mu_{3\to D}(D) = 1$

$\mu_{3\to I}(I) = 1$

$\mu_{5\to L}(L) =$

L	$\mu_{5\to L}(L)$	$\mu_{5\to L}(L)$
0	.1*1+.4*1+.99*1	1.49
1	.9*1+.6*1+.01*1	1.51

$\mu_{D\to 1}(D) = 1$

$\mu_{I\to 2}(I) = 1$

$\mu_{I\to 4}(I) =$

I	$\mu_{I\to 4}(I)$	$\mu_{I\to 4}(I)$
0	1*.7	0.7
1	1*.3	0.3

$$\mu_{4\to S}(S) = \sum_{X_S \setminus S} f_4(X_s) \prod_{j\in s, j\neq 4} \mu_{j\to 4}(X_j) =$$

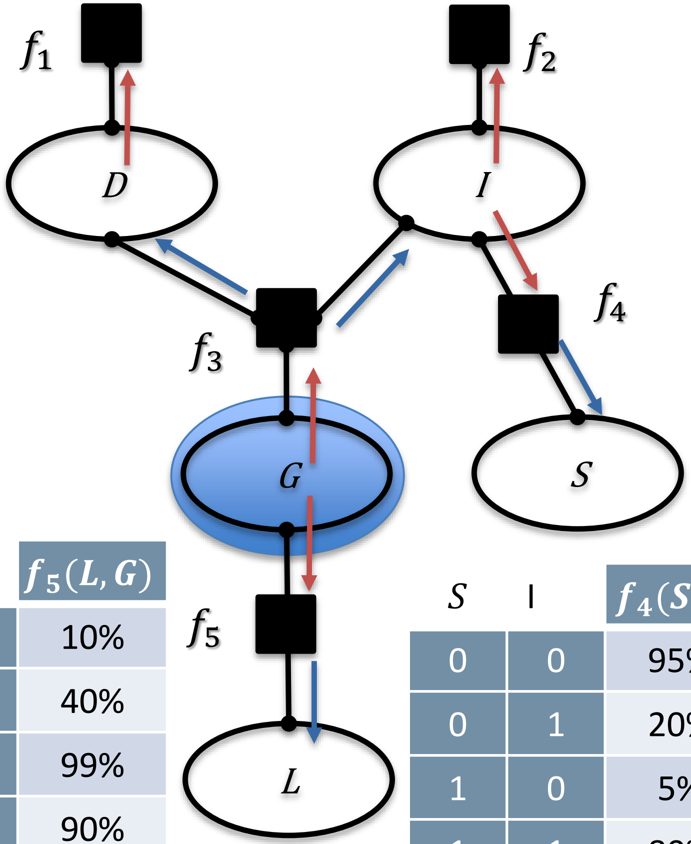
$$= \sum_{I} f_4(S,I)\, \mu_{I\to 4}(I) =$$

S	$\mu_{4\to S}(S)$	$\mu_{4\to S}(S)$
0	.95*.7+.2*.3	0.725
1	.05*.7+.8*.3	0.275

G	I	D	$f_3(G,I,D)$
1	0	0	30%
1	0	1	5%
1	1	0	90%
1	1	1	50%
2	0	0	40%
2	0	1	25%
2	1	0	8%
2	1	1	30%
3	0	0	30%
3	0	1	70%
3	1	0	2%
3	1	1	20%

D	$f_1(D)$
0	60%
1	40%

I	$f_2(I)$
0	70%
1	30%



L	G	$f_5(L,G)$
0	1	10%
0	2	40%
0	3	99%
1	1	90%
1	2	60%
1	3	1%

S	I	$f_4(S,I)$
0	0	95%
0	1	20%
1	0	5%
1	1	80%

Belief Propagation Example

$\mu_{G\to 3}(G) = 1 \qquad \mu_{G\to 5}(G) = 1$

$\mu_{3\to D}(D) = 1$

$\mu_{3\to I}(I) = 1$

$\mu_{5\to L}(L) =$

L	$\mu_{5\to L}(L)$	$\mu_{5\to L}(L)$
0	.1*1+.4*1+.99*1	1.49
1	.9*1+.6*1+.01*1	1.51

$\mu_{D\to 1}(D) = 1$

$\mu_{I\to 2}(I) = 1$

$\mu_{I\to 4}(I) =$

I	$\mu_{I\to 4}(I)$	$\mu_{I\to 4}(I)$
0	1*.7	0.7
1	1*.3	0.3

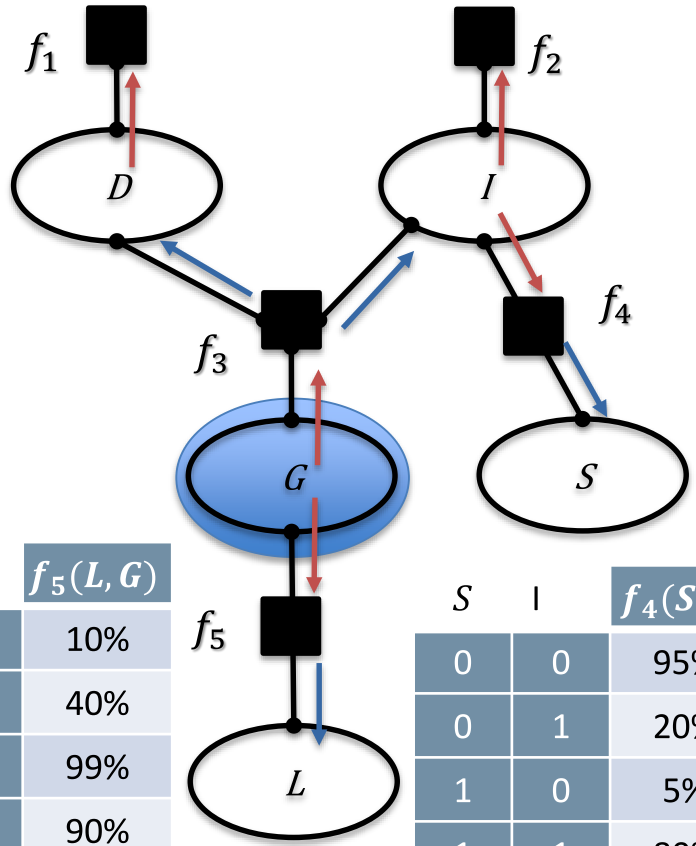
$\mu_{4\to S}(S) =$

S	$\mu_{4\to S}(S)$	$\mu_{4\to S}(S)$
0	.95*.7+.2*.3	0.725
1	.05*.7+.8*.3	0.275

G	I	D	$f_3(G,I,D)$
1	0	0	30%
1	0	1	5%
1	1	0	90%
1	1	1	50%
2	0	0	40%
2	0	1	25%
2	1	0	8%
2	1	1	30%
3	0	0	30%
3	0	1	70%
3	1	0	2%
3	1	1	20%

D	$f_1(D)$
0	60%
1	40%

I	$f_2(I)$
0	70%
1	30%



L	G	$f_5(L,G)$
0	1	10%
0	2	40%
0	3	99%
1	1	90%
1	2	60%
1	3	1%

S	I	$f_4(S,I)$
0	0	95%
0	1	20%
1	0	5%
1	1	80%

Belief Propagation Example

$$P(D) = \prod_{j \in ne(D)} \mu_{j\to D}(D) = \mu_{1\to D}(D)\mu_{3\to D}(D) = 1 \cdot f_1\,(D)$$

$$P(I) = \prod_{j \in ne(I)} \mu_{j\to I}(I) = \mu_{3\to I}(I)\mu_{4\to I}(I)\mu_{2\to I}(I) = 1 \cdot 1 \cdot f_2(I)$$

$$P(S) = \prod_{j \in ne(S)} \mu_{j\to S}(S) = \mu_{4\to S}(S) =$$

S	$\mu_{4\to S}\,(S)$	$\mu_{4\to S}\,(S)$
0	.95*.7+.2*.3	0.725
1	.05*.7+.8*.3	0.275

$$P(G) = \prod_{j \in ne(G)} \mu_{j\to G}(G) = \mu_{3\to G}(G)\mu_{5\to G}(G) = \mu_{3\to G}(G) \cdot 1$$

G	$\mu_{3\to G}\,(G)$	$\mu_{3\to G}\,(G)$
1	.3*.7*.6+.05*.7*.4+.9*.3*.6+.5*.3*.4	0.362
2	.4*.7*.6+.25*.7*.4+.08*.3*.6+.3*.3*.4	0.2884
3	.3*.7*.6+.7*.7*.4+.02*.3*.6+.2*.3*.4	0.3496

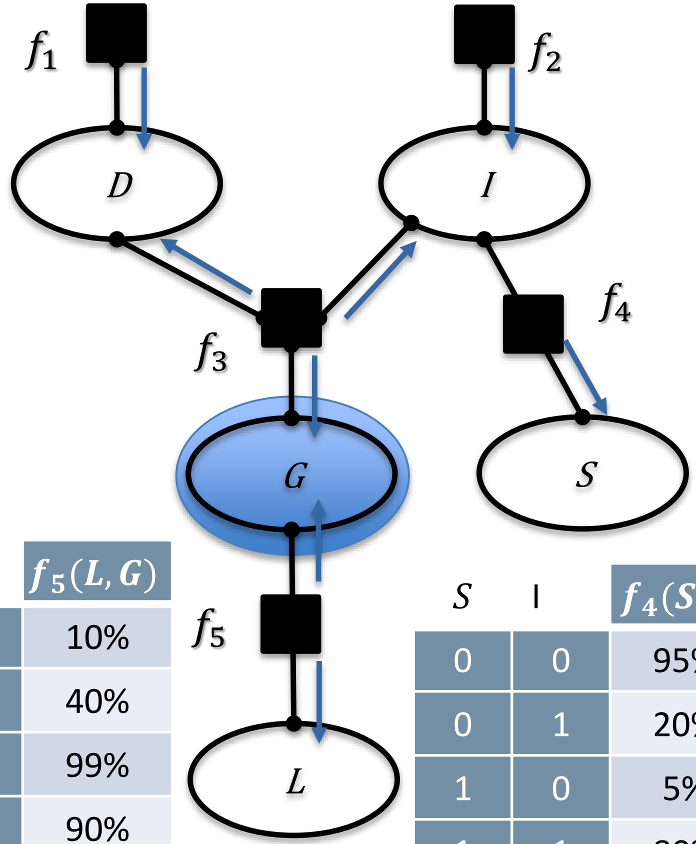
$$P(L) = \prod_{j \in ne(L)} \mu_{j\to L}(L) = \mu_{5\to L}(L) =$$

L	$\mu_{5\to L}\,(L)$	$\mu_{5\to L}\,(L)$	$\mu_{5\to L}\,(L)$
0	.1*1+.4*1+.99*1	1.49	0.496
1	.9*1+.6*1+.01*1	1.51	0.503

G	I	D	$f_3(G,I,D)$
1	0	0	30%
1	0	1	5%
1	1	0	90%
1	1	1	50%
2	0	0	40%
2	0	1	25%
2	1	0	8%
2	1	1	30%
3	0	0	30%
3	0	1	70%
3	1	0	2%
3	1	1	20%

D	$f_1(D)$
0	60%
1	40%

I	$f_2(I)$
0	70%
1	30%



L	G	$f_5(L,G)$
0	1	10%
0	2	40%
0	3	99%
1	1	90%
1	2	60%
1	3	1%

S	I	$f_4(S,I)$
0	0	95%
0	1	20%
1	0	5%
1	1	80%

Adding Evidence

Let's consider $P(X) = P(X_1)P(X_2|X_1)P(X_3|X_2)P(X_4|X_3)$ the marginal probability of X_2 and observations $X_1 = x_{e_1}$ and $X_3 = x_{e_3}$ is:

$$P\left(X_2|X_1 = x_{e_1}, X_3 = x_{e_3}\right) = P\left(X_1 = x_{e_1}\right)P\left(X_2|X_1 = x_{e_1}\right)P\left(X_3 = x_{e_3}|X_2\right)\sum_{X_4} P(X_4|X_3)$$

while the marginal probability of X_2 having no observation is:

$$P(X_2) = \sum_{X_1} P(X_1)\, P(X_2|X_1) \sum_{X_3} P(X_3|X_2) \sum_{X_4} P(X_4|X_3)$$

We can compute it on a tree easily via Belief propagation

- If some nodes X_e are observed, we simply use their observed values instead of summing over all possible values when computing their messages
- After normalization, this gives the conditional probability given the evidence

Belief Propagation Update Equations

We can write Belief Propagation as an Parallel Procedure

- Initialize all messages as one $\mu_{f_S \to X_i} = 1$, $\mu_{X_i \to f_S} = 1$
- Update messages

$$\mu^{new}_{f_s \to X_i}(X_i) = \sum_{X_S \setminus X_i} f_S(X_i, X_S) \prod_{X_j \in ne(X_i), j \neq i} \mu^{old}_{X_j \to f_S}(X_j)$$

$$\mu^{new}_{X_i \to f_s}(X_i) = \prod_{\substack{f_l \in ne(X_i) \\ f_l \neq f_s}} \mu^{old}_{f_l \to X_i}(X_i)$$

Allows for parallel update of messages!

Iterate until convergence

- Update believes

$$b_{f_s}(X_S) = f_S(X_S) \prod_{j \in f_s} \mu_{X_j \to f_S}(X_j)$$

$$b_{X_i}(X_i) = \prod_{f_S \in ne(i)} \mu_{f_S \to X_i}(X_i)$$

Belief Propagation Update Equations (Alternative)

We can rewrite Belief Propagation updates in an alternative way

- Initialize all messages as one $\mu_{f_s \to X_i} = 1$, $\mu_{X_i \to f_S} = 1$
- Update believes

$$b_{f_S}^{new}(X_S) = f_S(X_S) \prod_{X_j \in f_S} \mu_{X_j \to f_S}^{new}(X_j)$$

$$b_{X_i}^{new}(X_i) = \prod_{f_S \in ne(X_i)} \mu_{f_s \to X_i}^{new}(X_i)$$

- Update messages

$$\mu_{f_s \to X_i}^{new}(X_i) = \frac{1}{\mu_{X_i \to f_s}^{old}(X_j)} \sum_{X \setminus X_i} b_{f_s}^{old}(X_S)$$

$$\mu_{X_i \to f_s}^{new}(X_i) = \frac{1}{\mu_{f_s \to X_i}^{old}(X_i)} b_i^{old}(X_i)$$

Iterate until convergence

Converges to Marginal Consistency!

Belief Propagation and Marginal Consistency

Given two cliques C and D sharing a variable X_i, then their marginal beliefs should coincide:

$$b(X_i) = \sum_{X_C \setminus \mathrm{X}_i} b(X_C) = \sum_{X_D \setminus \mathrm{X}_i} b(X_D)$$

Note, marginal consistency also implies $b(X_i) = \mu_{C\to i}(X_i)\mu_{i\to C}(X_i)$

Marginal consistency is a *__fixed point of the Belief Propagation updates__*, i.e., Belief Update does not change the messages.

Easy to see when using the alternative updates

$$\mu_{C\to i}^{new}(X_i) = \frac{1}{\mu_{j\to C}^{old}(X_j)} \sum_{X_C \setminus \mathrm{X}_i} b_C^{old}(X_C) \qquad \mu_{i\to C}^{new}(X_i) = \frac{1}{\mu_{C\to i}^{old}(X_i)} b_i^{old}(X_i)$$

Belief Propagation Wrap-Up!

Belief Propagation, a.k.a, Sum-Product Algorithm with recursive computation of messages, leads to:

- Exact inference on tree (equivalent to Variable Elimination Algorithm)
- Marginal consistency on any graph (including non-trees)

Marginal consistency means:

- On trees, the parallel update of messages will converge to the true messages
- On polytrees, it will converge to the true messages as well
- On non-trees, it just reaches a state of marginal consistency

We use it for non-tree graphs, but …

Loopy Belief Propagation

We can apply Belief Update equation to loopy graphs, but:

- Having loops, branches of nodes to not represent independent information!
- Belief Propagation is multiplying (=fusion) messages from dependent sources

Approximate solution of an otherwise NP-hard problem

What can happen in practice?

- There might be some echo effects and could diverge
- Typically converges to a perturbed results (overconfident)
- …

Some theory exist on loopy Belief Propagation converging to Bethe approximation, (Yedidia, Freeman, & Weiss, 2001)

Can we do exact efficient inference in non-tree graphs?

POLITECNICO
MILANO 1863

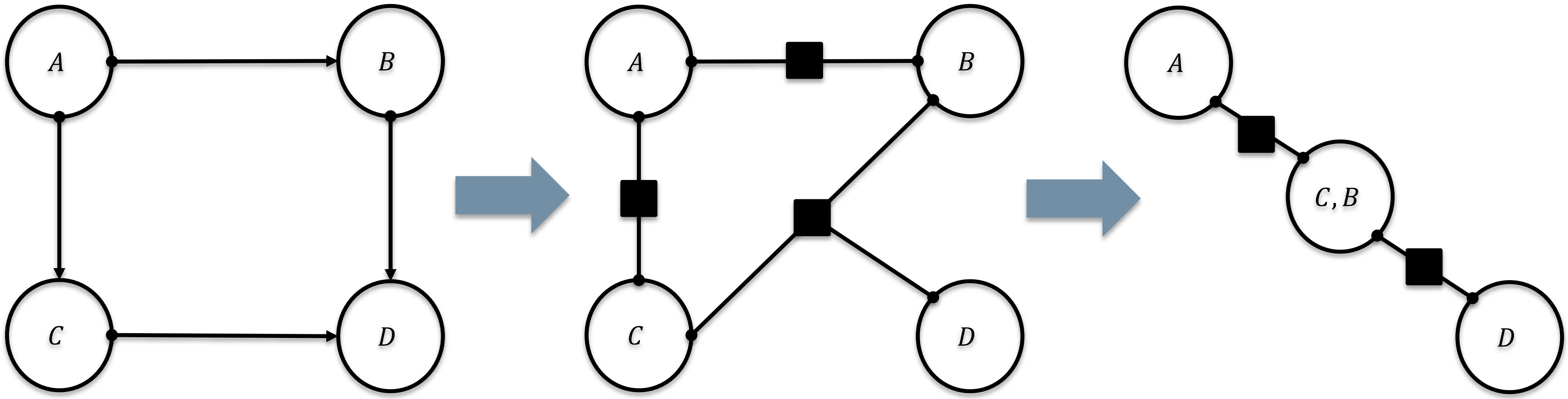
Soft Computing – Probabilistic Reasoning

- Junction Trees -

Prof. Matteo Matteucci – *matteo.matteucci@polimi.it*

Junction Tree Algorithm

Many models have loops, it is possible to convert them into a tree by defining variable groups (separators) on which messages are defined



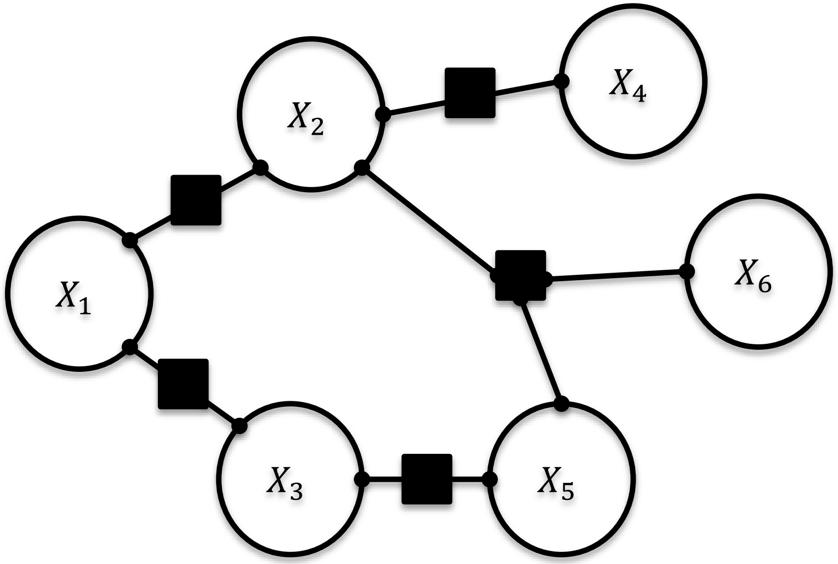
A variable substitution, where we rename $(B, C) = E$. A random variable may be part of multiple separators, but only along a running intersection

Junction Tree via Variable Elimination

Recall a clique is a fully connected subset of nodes in a graph:

- Start with a factor graph
- Choose an order of variable elimination
- Keep track of the μ terms, variables they depend on identify the separators

Example: elimination order 4, 6, 5, 1, 2, 3



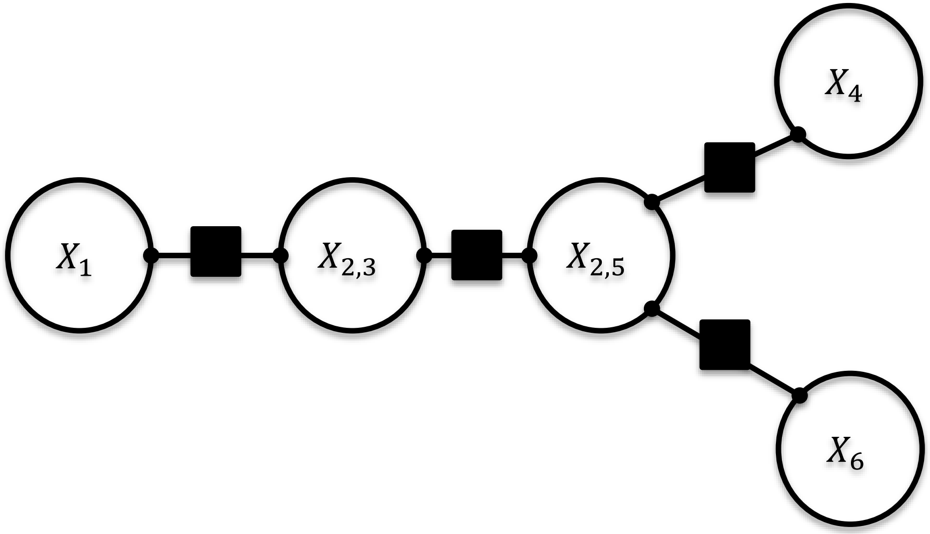
$\mu_1(X_2)$

$\mu_2(X_2, X_5)$

$\mu_3(X_2, X_3)$

$\mu_4(X_2, X_3)$

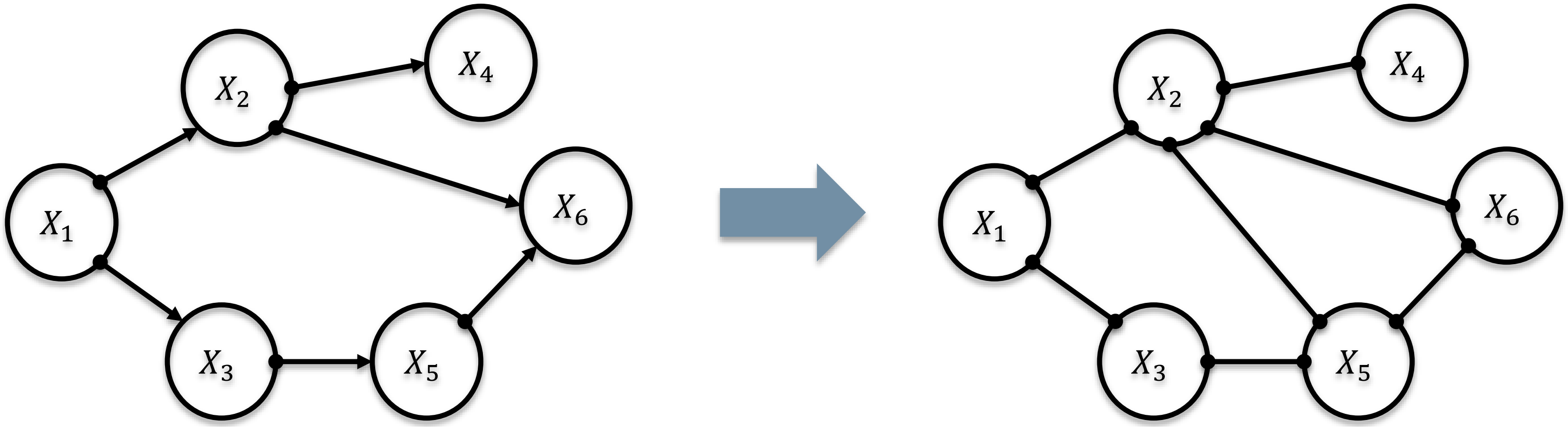
$\mu_5(X_3)$



Junction Tree via Moralization & Triangulation

Recall a clique is a fully connected subset of nodes in a graph:

- Moralize the graph (if directed)
- Choose a node ordering and find the cliques generated by variable elimination. This gives a triangulation of the graph
- Build a junction graph from the eliminated cliques
- Find an appropriate spanning tree



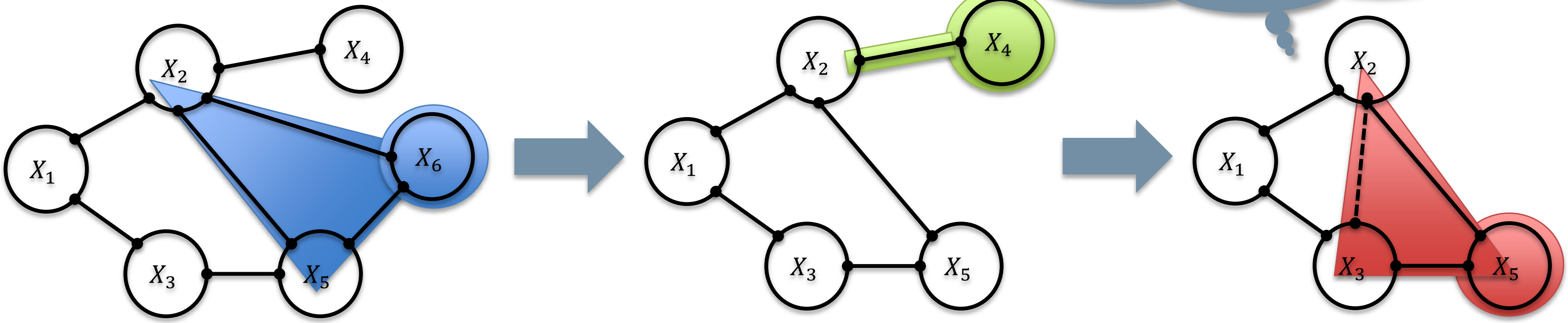
Junction Tree via Moralization & Triangulation

Recall a clique is a fully connected subset of nodes in a graph:

- Moralize the graph (if directed)
- Choose a node ordering and find the cliques generated by variable elimination. This gives a graph triangulation
- Build a junction graph from the eliminated cliques
- Find an appropriate spanning tree

Removed	Clique	Added
X_6	X_6, X_2, X_5	-
X_4	X_2, X_4	-
X_5	X_2, X_3, X_5	$X_3 - X_2$

Add connections to build new cliques.

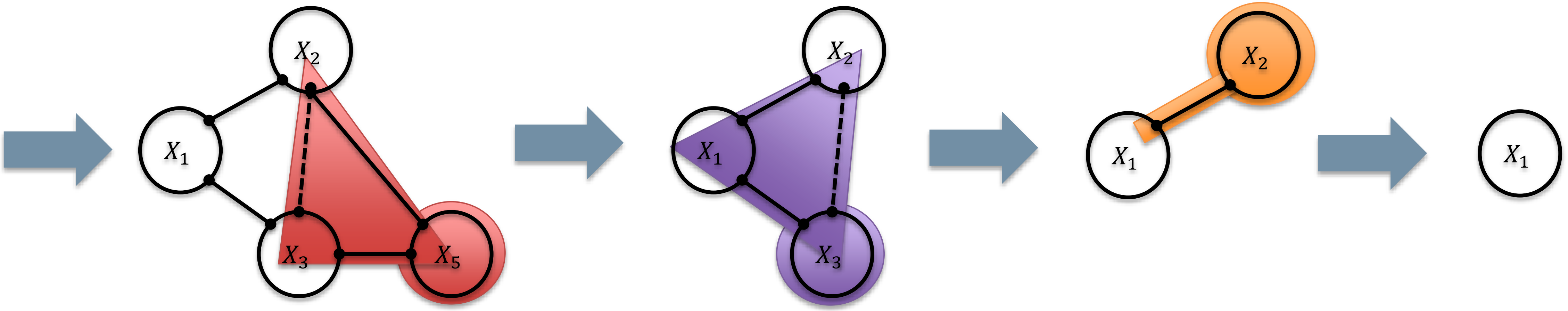


Junction Tree via Moralization & Triangulation

Recall a clique is a fully connected subset of nodes in a graph:

- Moralize the graph (if directed)
- Choose a node ordering and find the cliques generated by variable elimination. This gives a graph triangulation
- Build a junction graph from the eliminated cliques
- Find an appropriate spanning tree

Removed	Clique	Added
X_6	X_6, X_2, X_5	-
X_4	X_2, X_4	-
X_5	X_2, X_3, X_5	$X_3 - X_2$
X_3	X_2, X_3, X_1	-
X_2	X_2, X_1	-

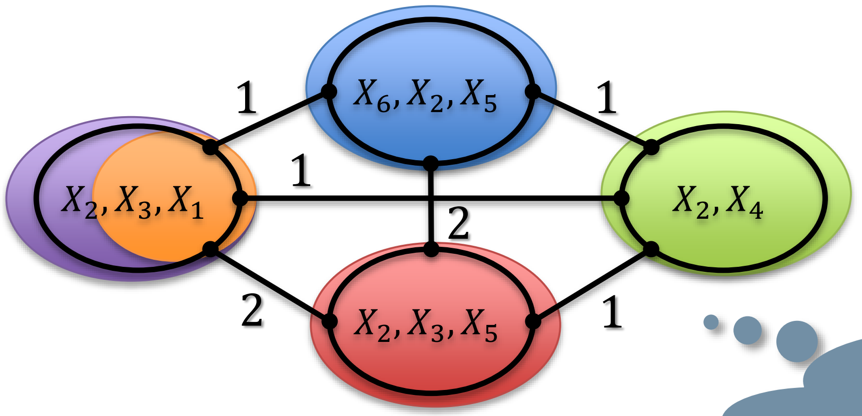


Junction Tree via Moralization & Triangulation

Recall a clique is a fully connected subset of nodes in a graph:

- Moralize the graph (if directed)
- Choose a node ordering and find the cliques generated by variable elimination. This gives a graph triangulation
- Build a junction graph from the eliminated cliques
- Find an appropriate spanning tree

Removed	Clique	Added
X_6	X_6, X_2, X_5	-
X_4	X_2, X_4	-
X_5	X_2, X_3, X_5	$X_3 - X_2$
X_3	X_2, X_3, X_1	-
X_2	X_2, X_1	-



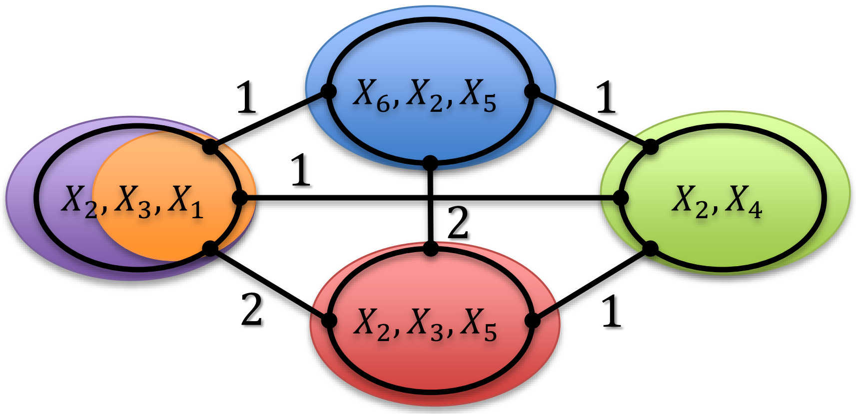
Weights are given by number of intersections

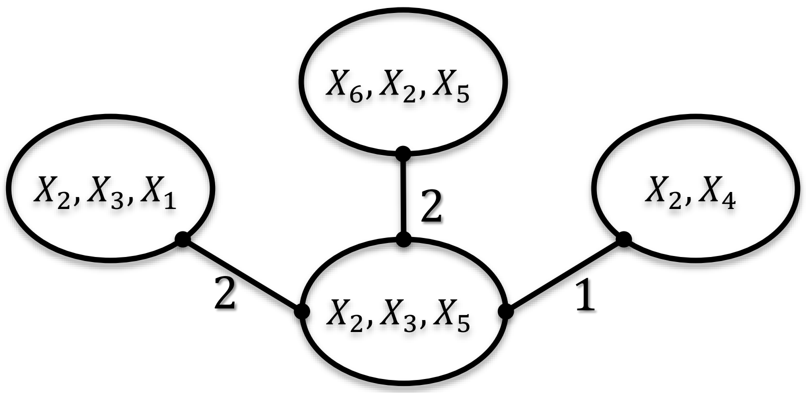
Junction Tree via Moralization & Triangulation

Recall a clique is a fully connected subset of nodes in a graph:

- Moralize the graph (if directed)
- Choose a node ordering and find the cliques generated by variable elimination. This gives a graph triangulation
- Build a junction graph from the eliminated cliques
- Find an appropriate spanning tree

Removed	Clique	Added
X_6	X_6, X_2, X_5	-
X_4	X_2, X_4	-
X_5	X_2, X_3, X_5	$X_3 - X_2$
X_3	X_2, X_3, X_1	-
X_2	X_2, X_1	-



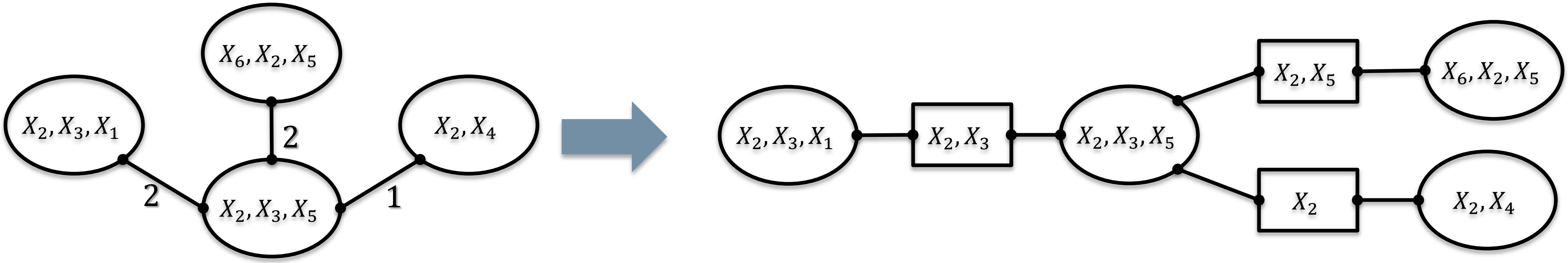


Junction Tree via Moralization & Triangulation

Recall a clique is a fully connected subset of nodes in a graph:

- Moralize the graph (if directed)
- Choose a node ordering and find the cliques generated by variable elimination. This gives a graph triangulation
- Build a junction graph from the eliminated cliques
- Find an appropriate spanning tree

Removed	Clique	Added
X_6	X_6, X_2, X_5	-
X_4	X_2, X_4	-
X_5	X_2, X_3, X_5	$X_3 - X_2$
X_3	X_2, X_3, X_1	-
X_2	X_2, X_1	-



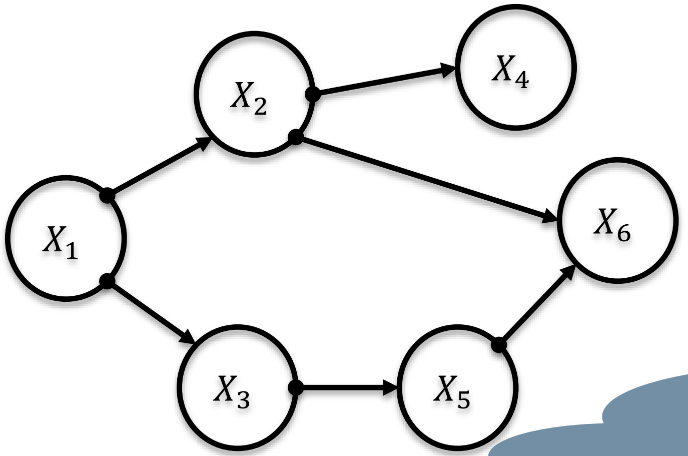
Junction Tree via Moralization & Triangulation

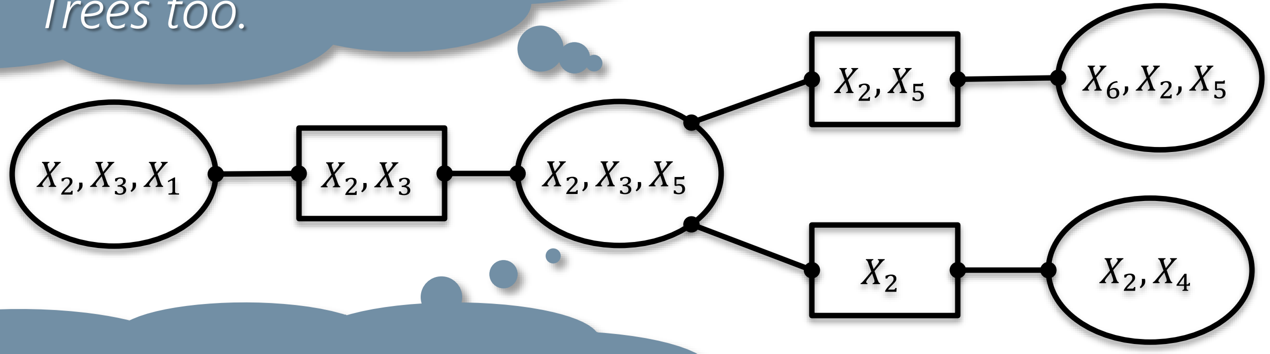
Recall a clique is a fully connected subset of nodes in a graph:

- Moralize the graph (if directed)
- Choose a node ordering and find the cliques generated by variable elimination. This gives a graph triangulation
- Build a junction graph from the eliminated cliques
- Find an appropriate span

Removed	Clique	Added
X_6	X_6, X_2, X_5	-
X_4	X_2, X_4	-
X_5	X_2, X_3, X_5	$X_3 - X_2$
X_3	X_2, X_3, X_1	-
X_2	X_2, X_1	-

You can perform Belief Propagation on Junction Trees too.

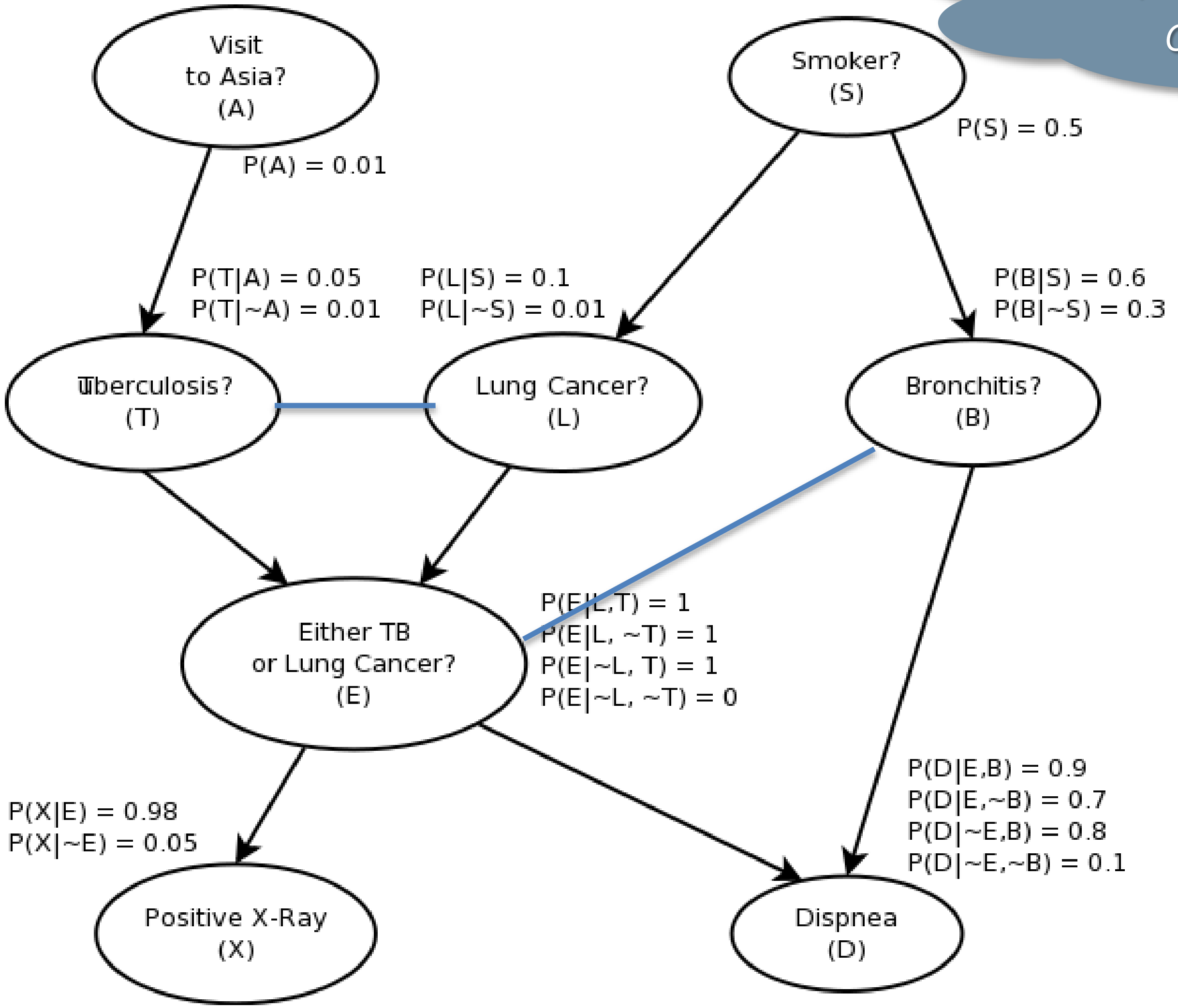


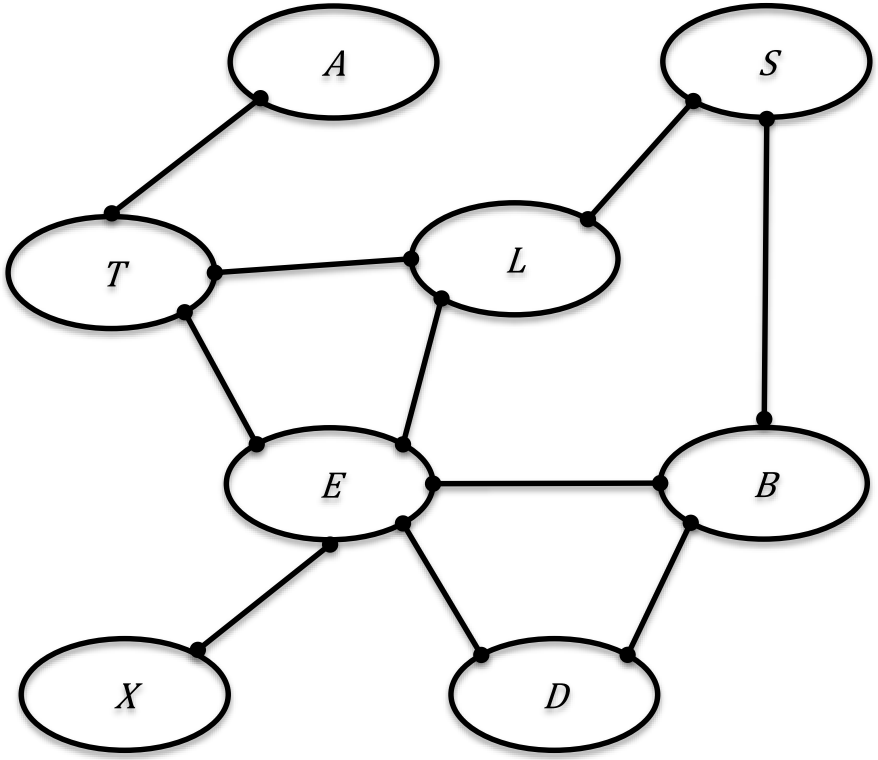


Complexity is exponential in the number of variables in the nodes

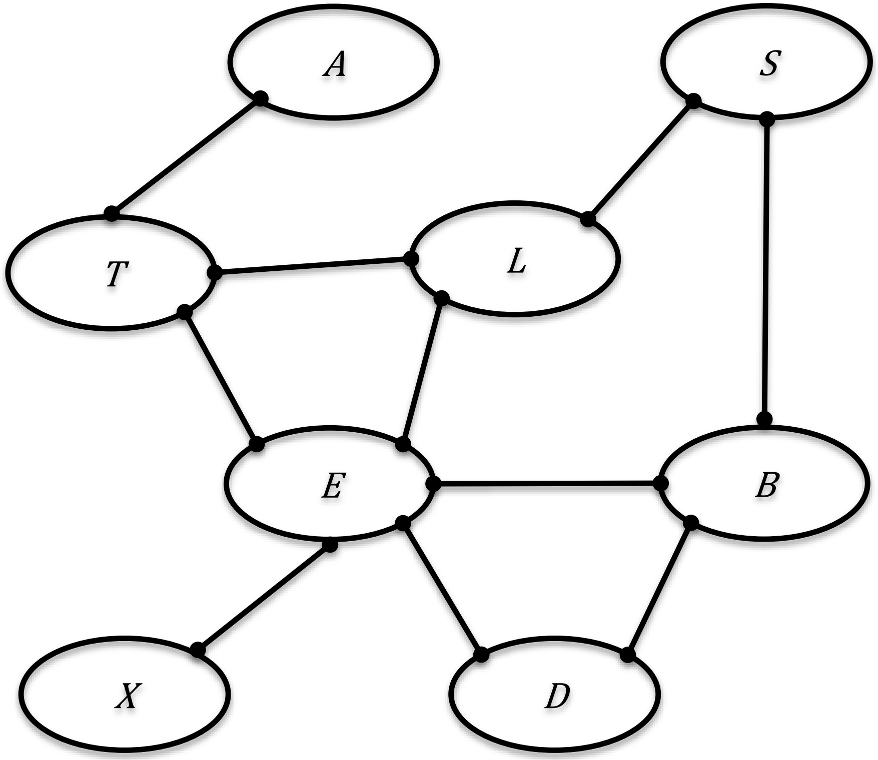
The Asia Network Example

The trick (heuristic) is to start from the biggest (existing) cliques then triangulate



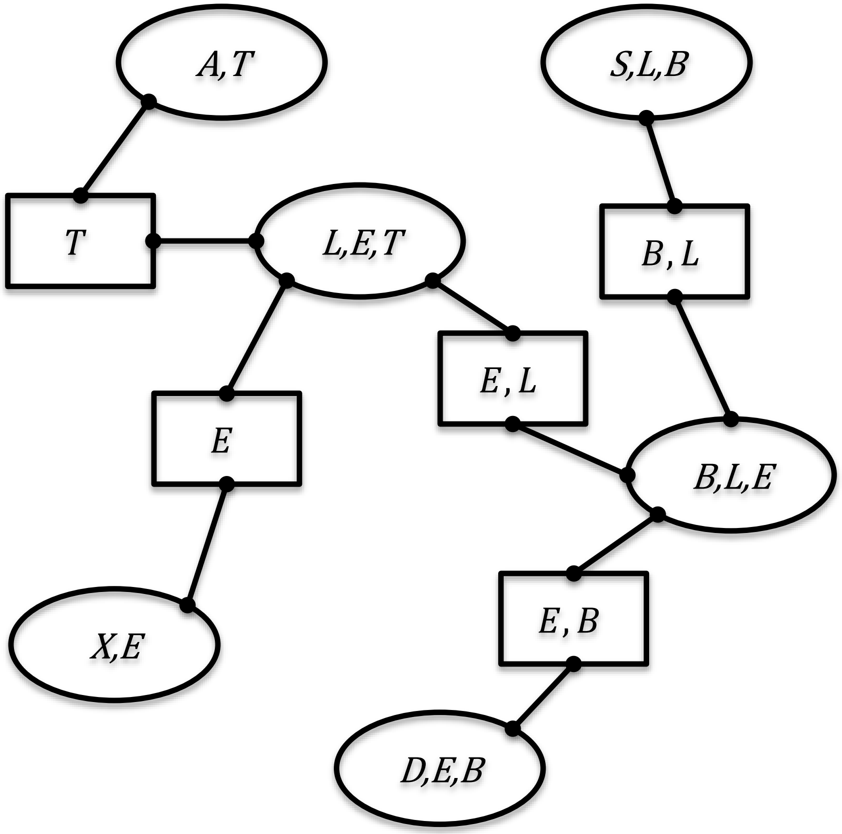


The Asia Network



Removed	Clique	Added
D	D, E, B	-
X	X, E	-
A	A, T	-
S	S, L, B	$L - B$
B	B, L, E	-
L	L,E,T	-
T	T, E	-
E	E	-

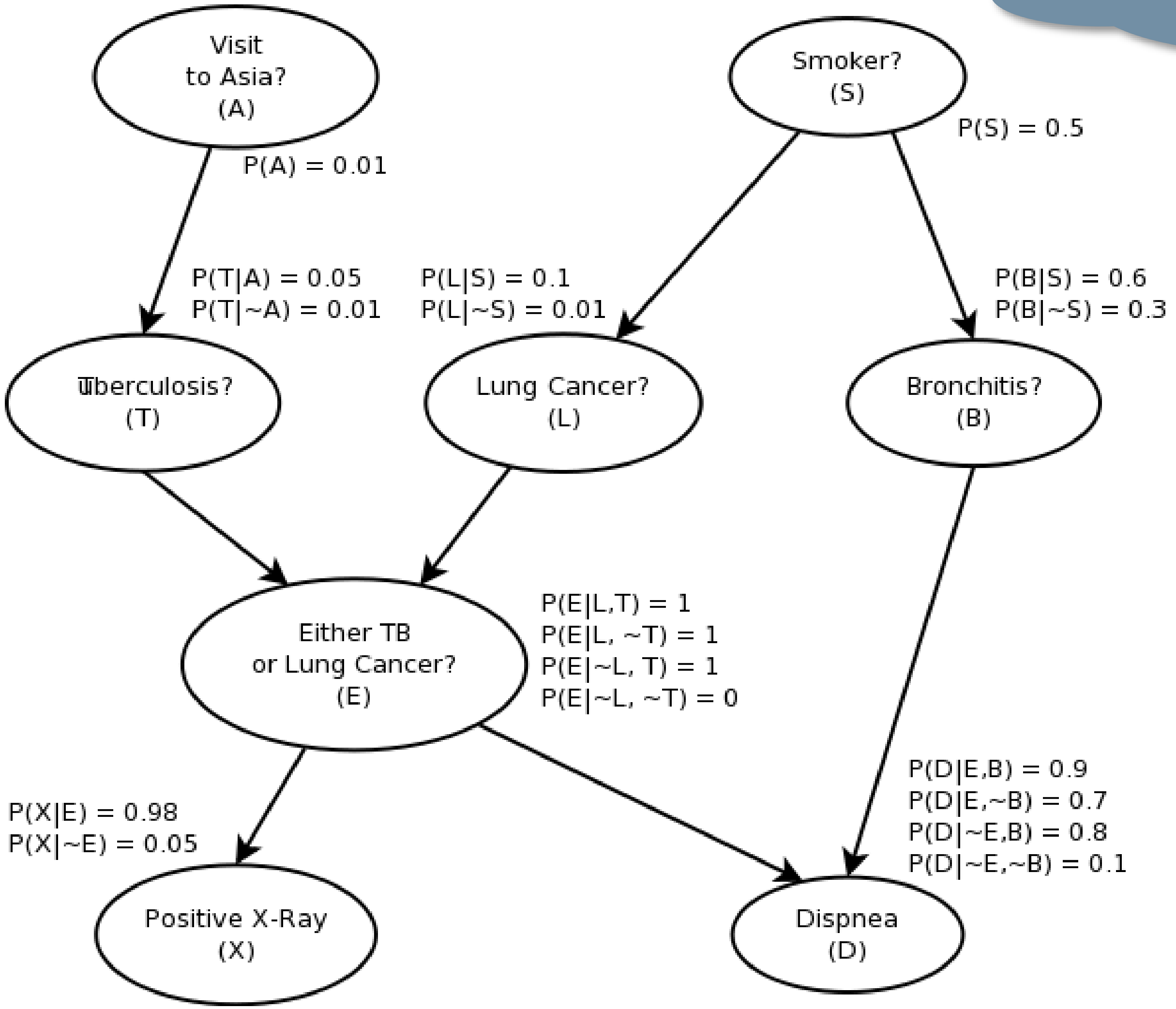
The Asia Network Example

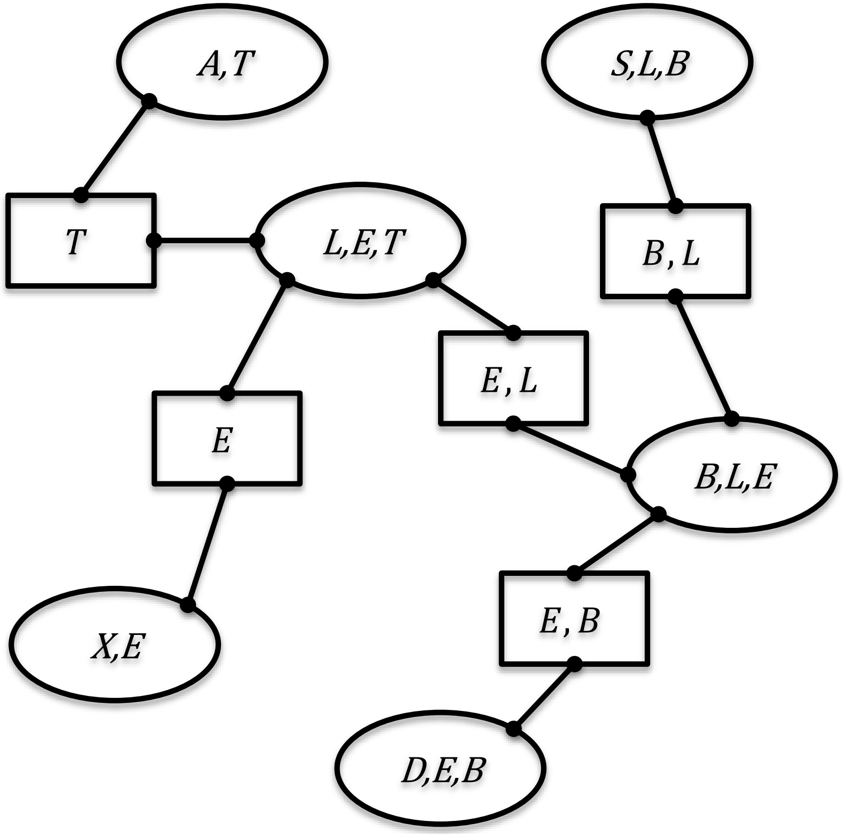


Removed	Clique	Added
D	D, E, B	-
X	X, E	-
A	A, T	-
S	S, L, B	$L - B$
B	B, L, E	-
L	L,E,T	-
T	T, E	-
E	E	-

The Asia Network Example

Now you can apply exact belief propagation on the Junction Tree





Soft Computing – Probabilistic Reasoning

- Sampling based methods-

Prof. Matteo Matteucci – *matteo.matteucci@polimi.it*

Sampling from a Bayesian Network

Let a Bayesian Network with random variables $X_1, \dots, X_N$, some of which are observed: $X_{obs} = y_{obs},\ \ obs \subset \{1,2,\dots,N\}$; our goal is to compute marginal posteriors $P(X_i | X_{obs} = y_{obs})$ conditioned on the observations.

We can generate a set of K (joint) samples $S = \{(x_1, x_2, \dots, x_N)\}_{k=1}^{K}$ being each sample $(x_1, x_2, \dots, x_N)$ a list of instantiations of all $X_1, \dots, X_N$.

Having these samples we can compute

$$P(X_i = x | X_{obs} = y_{obs}) \approx \frac{count_S\left(x_i^k = x \wedge x_{obs}^k = y_{obs}\right)}{count_S\left(x_{obs}^k = y_{obs}\right)}$$

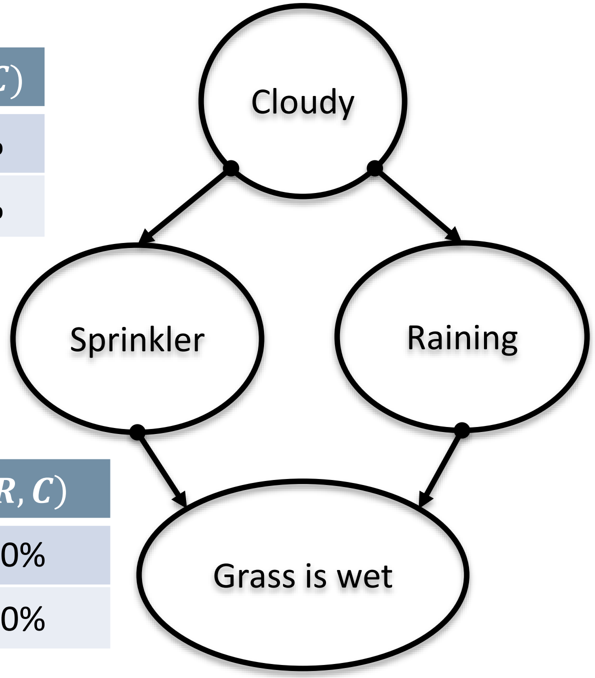
The Sprinkler Example: Sampling (1)

$Cloudy$	$P(C)$
1	50%

Let's start sampling x^k based on $U(0,1)$

- Generate a number $s \sim U(0,1)$
  - If $s \leq P(C)\ \rhd\ x_C^k = 1$ else $x_C^k = 0$
- Generate a number $s \sim U(0,1)$
  - If $s \leq P\left(S|x_C^k\right)\ \rhd\ x_S^k = 1$ else $x_S^k = 0$
- Generate a number $s \sim U(0,1)$
  - If $s \leq P\left(R|x_C^k\right)\ \rhd\ x_R^k = 1$ else $x_R^k = 0$
- Generate a number $s \sim U(0,1)$
  - If $s \leq P\left(W|x_S^k, x_R^k\right)\ \rhd\ x_W^k = 1$ else $x_W^k = 0$

$Sprinkler$	$Cloudy$	$P(S, C)$
1	0	90%
1	1	50%



$Raining$	$Cloudy$	$P(R, C)$
1	0	20%
1	1	50%

Wet	$Sprinkler$	$Raining$	$P(W, S, R)$
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%

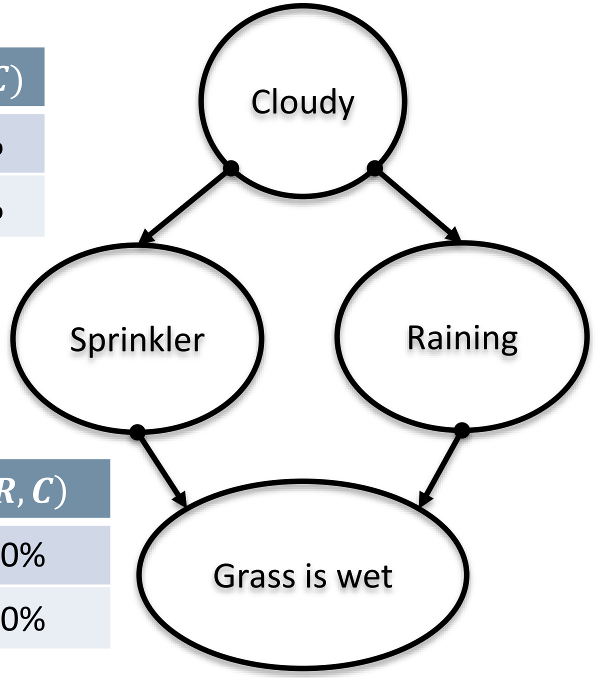
The Sprinkler Example: Sampling (2)

$Cloudy$	$P(C)$
1	50%

Let's start sampling x^k based on $U(0,1)$

- $s \sim U(0,1) \triangleright s = 0.814 \triangleright x_C^k = 0$
- $s \sim U(0,1) \triangleright s = 0.631 \triangleright x_S^k = 1$
- $s \sim U(0,1) \triangleright s = 0.467 \triangleright x_R^k = 0$
- $s \sim U(0,1) \triangleright s = 0.683 \triangleright x_W^k = 1$

$Sprinkler$	$Cloudy$	$P(S,C)$
1	0	90%
1	1	50%



$Raining$	$Cloudy$	$P(R,C)$
1	0	20%
1	1	50%

C	S	R	W
0	1	0	1

Wet	$Sprinkler$	$Raining$	$P(W,S,R)$
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%

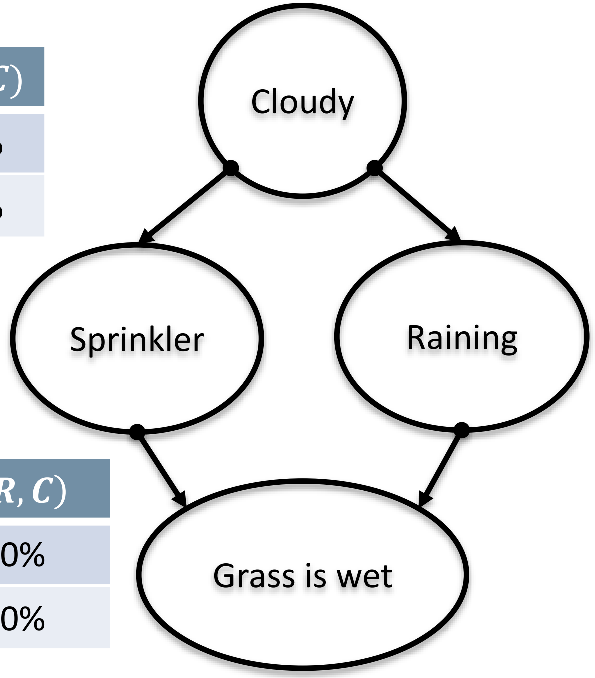
The Sprinkler Example: Sampling (3)

Cloudy	$P(C)$
1	50%

Let's start sampling x^k based on $U(0,1)$

- $s \sim U(0,1) \triangleright s = 0.586 \triangleright x_C^k = 0$
- $s \sim U(0,1) \triangleright s = 0.951 \triangleright x_S^k = 0$
- $s \sim U(0,1) \triangleright s = 0.883 \triangleright x_R^k = 0$
- $s \sim U(0,1) \triangleright s = 0.549 \triangleright x_W^k = 0$

Sprinkler	*Cloudy*	$P(S, C)$
1	0	90%
1	1	50%



Raining	*Cloudy*	$P(R, C)$
1	0	20%
1	1	50%

C	S	R	W
0	1	0	1
0	0	0	0

Wet	*Sprinkler*	*Raining*	$P(W, S, R)$
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%

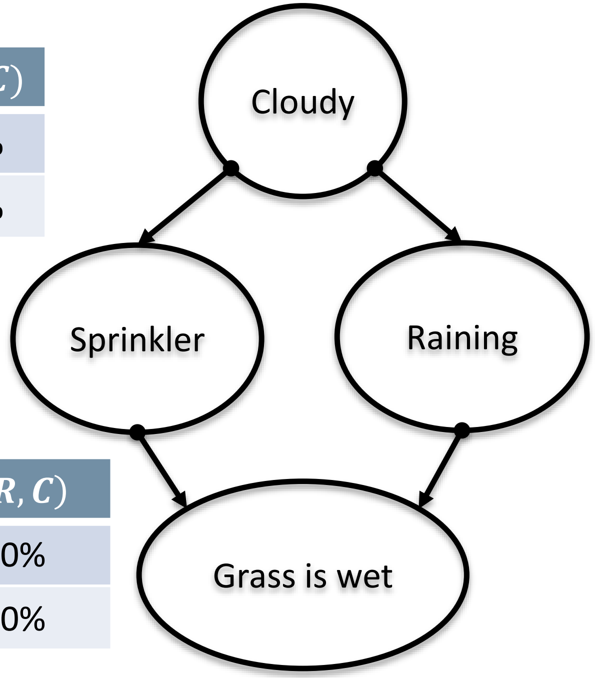
The Sprinkler Example: Sampling (4)

$Cloudy$	$P(C)$
1	50%

Let's start sampling x^k based on $U(0,1)$

- $s \sim U(0,1) \rhd s = 0.999 \rhd x_C^k = 0$
- $s \sim U(0,1) \rhd s = 0.855 \rhd x_S^k = 1$
- $s \sim U(0,1) \rhd s = 0.179 \rhd x_R^k = 1$
- $s \sim U(0,1) \rhd s = 0.142 \rhd x_W^k = 1$

$Sprinkler$	$Cloudy$	$P(S,C)$
1	0	90%
1	1	50%



$Raining$	$Cloudy$	$P(R,C)$
1	0	20%
1	1	50%

C	S	R	W
0	1	0	1
0	0	0	0
0	1	1	1

Wet	$Sprinkler$	$Raining$	$P(W,S,R)$
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%

The Sprinkler Example: Sampling (5)

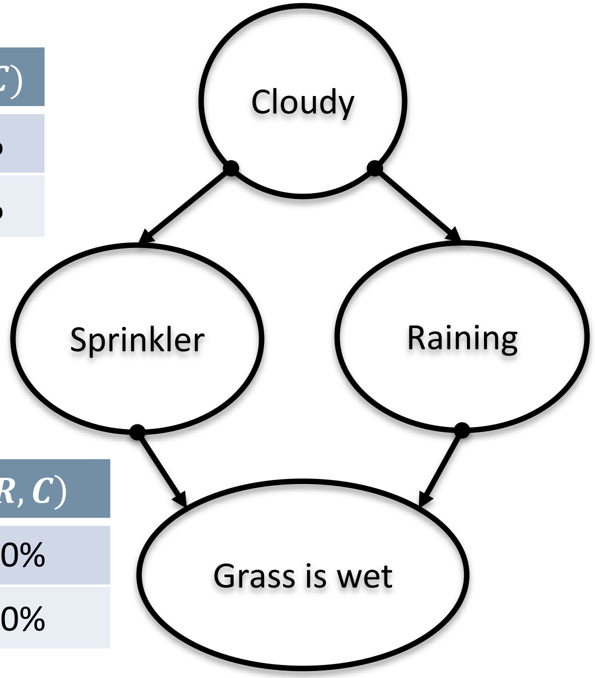
Let's start sampling x^k based on $U(0,1)$

- $s \sim U(0,1) \rhd s = 0.630 \rhd x_C^k = 0$
- $s \sim U(0,1) \rhd s = 0.455 \rhd x_S^k = 1$
- $s \sim U(0,1) \rhd s = 0.046 \rhd x_R^k = 1$
- $s \sim U(0,1) \rhd s = 0.745 \rhd x_W^k = 1$

C	S	R	W
0	1	0	1
0	0	0	0
0	1	1	1
0	1	1	1

$Cloudy$	$P(C)$
1	50%

$Sprinkler$	$Cloudy$	$P(S,C)$
1	0	90%
1	1	50%



$Raining$	$Cloudy$	$P(R,C)$
1	0	20%
1	1	50%

Wet	$Sprinkler$	$Raining$	$P(W,S,R)$
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%

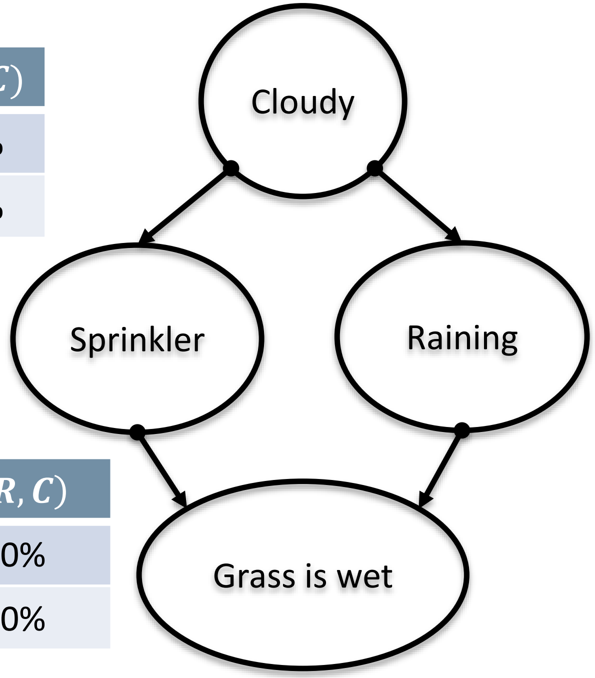
The Sprinkler Example: Sampling (6)

$Cloudy$	$P(C)$
1	50%

Let's start sampling x^k based on $U(0,1)$

- $s \sim U(0,1) \rhd s = 0.522 \rhd x_C^k = 0$
- $s \sim U(0,1) \rhd s = 0.193 \rhd x_S^k = 1$
- $s \sim U(0,1) \rhd s = 0.159 \rhd x_R^k = 1$
- $s \sim U(0,1) \rhd s = 0.702 \rhd x_W^k = 1$

$Sprinkler$	$Cloudy$	$P(S,C)$
1	0	90%
1	1	50%



$Raining$	$Cloudy$	$P(R,C)$
1	0	20%
1	1	50%

C	S	R	W
0	1	0	1
0	0	0	0
0	1	1	1
0	1	1	1
0	1	1	1

Wet	$Sprinkler$	$Raining$	$P(W,S,R)$
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%

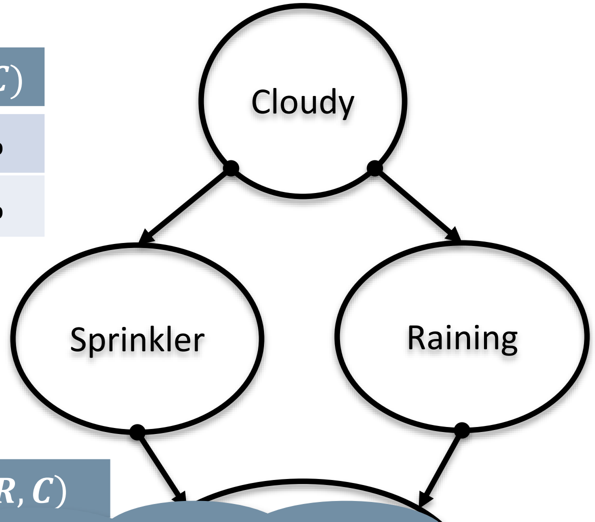
The Sprinkler Example: Sampling (7)

Cloudy	$P(C)$
1	50%

Let's start sampling x^k based on $U(0,1)$

- $s \sim U(0,1) \rhd s = 0.288 \rhd x_C^k = 1$
- $s \sim U(0,1) \rhd s = 0.794 \rhd x_S^k = 0$
- $s \sim U(0,1) \rhd s = 0.483 \rhd x_R^k = 1$
- $s \sim U(0,1) \rhd s = 0.216 \rhd x_W^k = 1$

Sprinkler	*Cloudy*	$P(S,C)$
1	0	90%
1	1	50%



Raining	*Cloudy*	$P(R,C)$
1		
1		

Several strategies exists for Sampling

C	S	R	W
0	1	0	1
0	0	0	0
0	1	1	1
0	1	1	1
0	1	1	1
1	0	1	1

$$P(C|W) \approx \frac{\#(C \wedge W)}{\#(W)} = \frac{1}{5}$$

Wet	*Sprinkler*	*Raining*	$P(W,S,R)$
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%

Rejection Sampling

To generate a single sample $x_{1:N}^k$:

- Sort all random variables in topological order (i.e., from root to leaves)
- Start with $i = 1$
- Sample a value $x_i^k \sim P\left(X_i | x_{Parents(X_i)}^k\right)$ conditional on $x_{1:i-1}^k$
- If $i \in obs$ compare the sampled value x_i^k with the observation y_i; reject and repeat from the previous steps if the sample is not equal to the observation
- Repeat with $i = i + 1$

Computers are fast, but for rare events this might waste lot of samples

Having K samples we can compute

$$P(X_i = x | X_{obs} = y_{obs}) \approx \frac{count_S\left(x_i^k = x\right)}{K}$$

Importance Sampling (with likelihood weighting)

To generate a single weighted sample $(w^k, x^k_{1:N})$:

- Sort all random variables in topological order (i.e., from root to leaves)
- Start with $i = 1$ and $\mathrm{w}^{\mathrm{k}} = 1$
- If $i \notin obs$ sample a value $x_i^k \sim P\left(X_i | x^k_{Parents(X_i)}\right)$ conditional on $x^k_{1:i-1}$
- If $i \in obs$, set the value $\mathrm{x}_i^{\mathrm{k}} = y_i$ and update $\mathrm{w}^{\mathrm{k}} = w^k \cdot P\left(\mathrm{X_i} = \mathrm{y}_i | x^k_{1:i-1}\right)$
- Repeat with $i = i + 1$

Having K samples we can compute

$I_{x_i^k = x}$ *equals* 1 *if* $x_i^k = x$ *otherwise it equals 0*

$$P(X_i = x | X_{obs} = y_{obs}) \approx \frac{\sum_{k=1}^{K} w^k \cdot I_{x_i^k = x}}{\sum_{k=1}^{K} w^k}$$

Gibbs Sampling

In Gibbs sampling the samples are not independent from each other anymore. The next sample "modifies" the previous one:

- First, all observed variables are clamped to their fixed value $\boldsymbol{x_i^k = y_i}$ for any $\boldsymbol{k}$
- To generate the $(k+1)^{th}$ sample, iterate latent variables $\boldsymbol{i \notin obs}$ updating:

$$
\begin{aligned}
x_i^{k+1} &\sim P\left(X_i | x_{1:N\setminus\mathrm{i}}^k\right) \\
&\sim P\left(X_i | x_1^k, x_2^k, \dots, x_{i-1}^k, x_{i+1}^k, \dots, x_N^k\right) \\
&\sim P\left(X_i | x_{parents(i)}^k\right) \prod_{j:i\in parents(j)} P\left(X_j = x_j^k | X_i, x_{parents(j)\setminus\mathrm{i}}^k\right)
\end{aligned}
$$

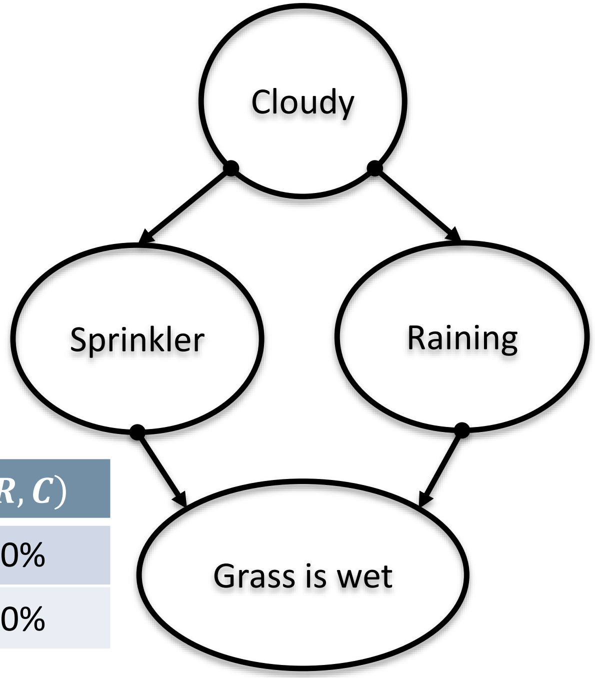
Each $\boldsymbol{x_i^{k+1}}$ is resampled conditional to the other (neighboring) current sample values. After an initial set (burn-in), samples can directly be used.

The Sprinkler Example: with other samplers!

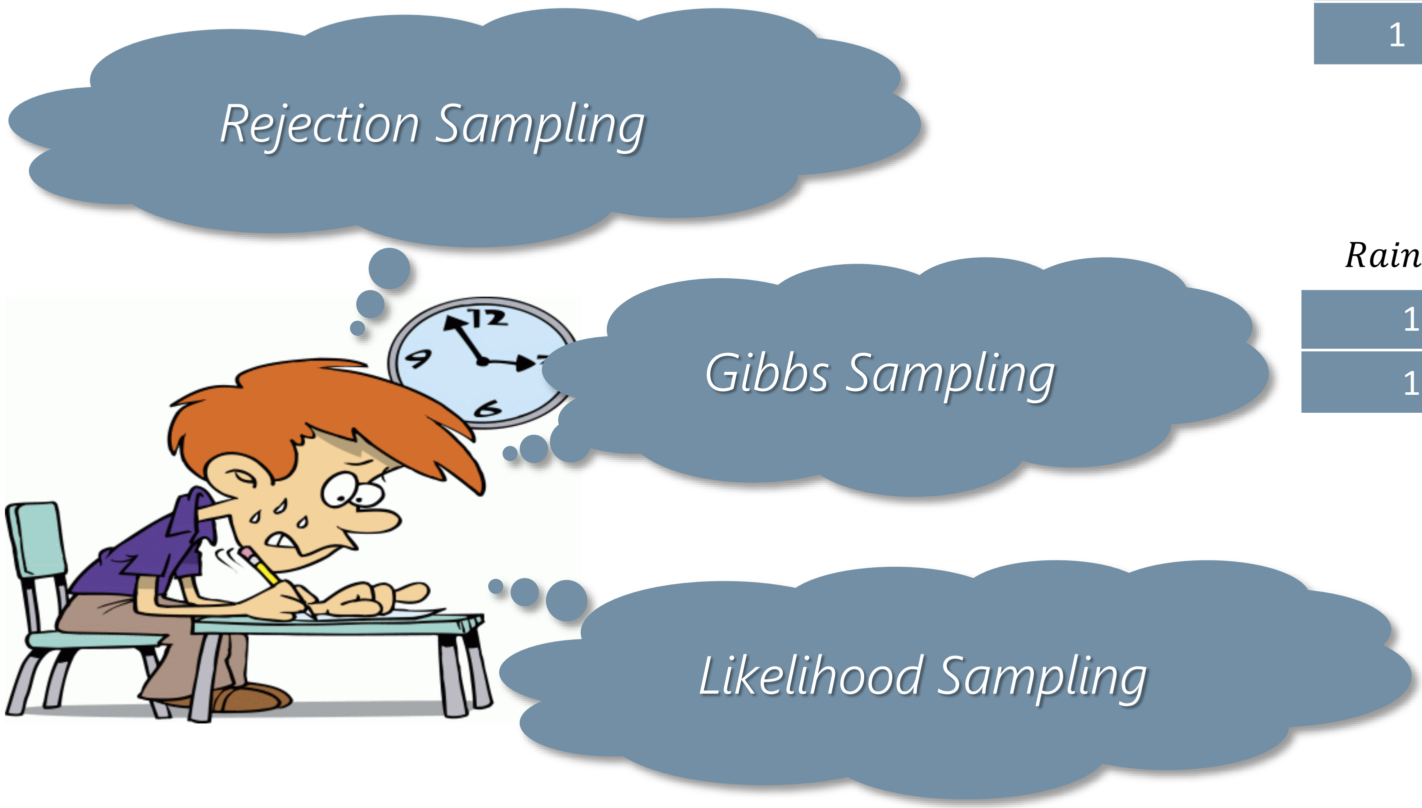
Cloudy	$P(C)$
1	50%

Let's start sampling to compute $P(C|W = true)$

Sprinkler	*Cloudy*	$P(S, C)$
1	0	90%
1	1	50%



Raining	*Cloudy*	$P(R, C)$
1	0	20%
1	1	50%



Wet	*Sprinkler*	*Raining*	$P(W, S, R)$
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%

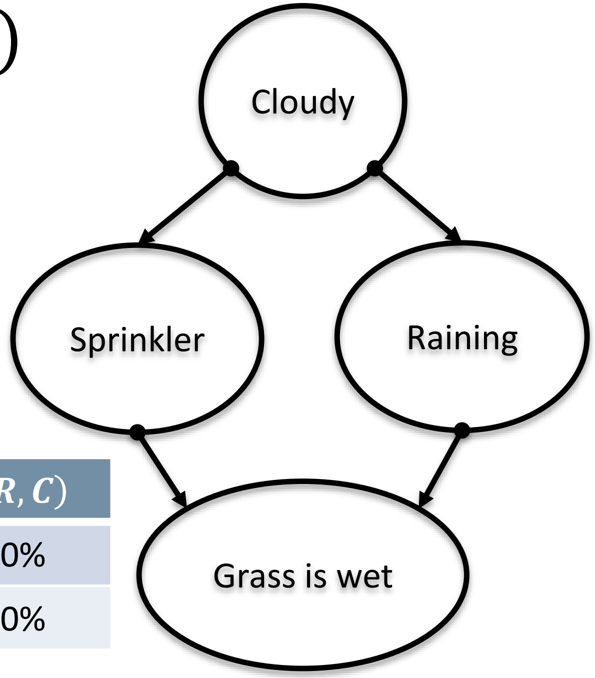
The Sprinkler Example: Likelihood Weighting (1)

$Cloudy$	$P(C)$
1	50%

To compute $P(C|W=1)$ by sampling x^k based on $U(0,1)$

- Set $w^k = 1$
- Generate a number $s \sim U(0,1)$
  - If $s \leq P(C)$ $\triangleright$ $x_C^k = 1$ else $x_C^k = 0$
- Generate a number $s \sim U(0,1)$
  - If $s \leq P(S|x_C^k)$ $\triangleright$ $x_S^k = 1$ else $x_S^k = 0$
- Generate a number $s \sim U(0,1)$
  - If $s \leq P(R|x_C^k)$ $\triangleright$ $x_R^k = 1$ else $x_R^k = 0$
- Force $x_W^k = 1$
  - $w^k = w^k \cdot P(x_W^k = 1|x_S^k, x_R^k)$

$Sprinkler$	$Cloudy$	$P(S,C)$
1	0	90%
1	1	50%



$Raining$	$Cloudy$	$P(R,C)$
1	0	20%
1	1	50%

Wet	$Sprinkler$	$Raining$	$P(W,S,R)$
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%

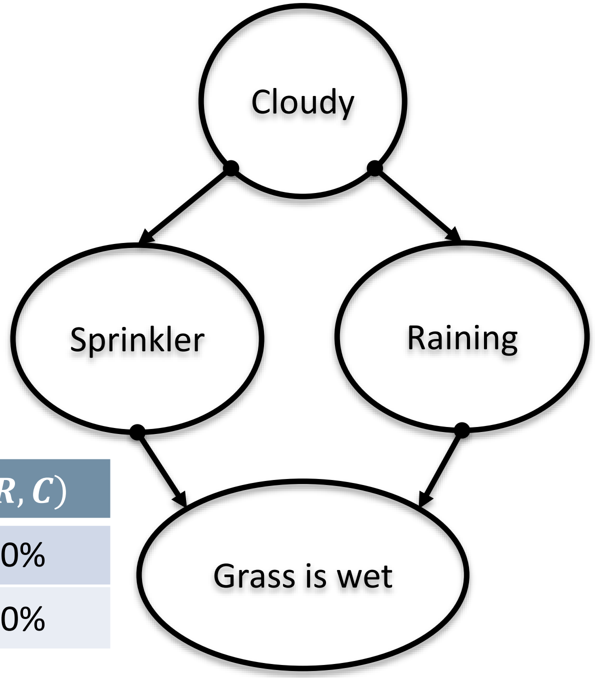
The Sprinkler Example: Sampling (2)

$Cloudy$	$P(C)$
1	50%

Let's start sampling x^k based on $U(0,1)$

- $s \sim U(0,1) \triangleright s = 0.814 \triangleright x_C^k = 0$
- $s \sim U(0,1) \triangleright s = 0.631 \triangleright x_S^k = 1$
- $s \sim U(0,1) \triangleright s = 0.467 \triangleright x_R^k = 0$
- $w^k = w^k \cdot P\left(x_W^k = 1 \middle| x_S^k, x_R^k\right) = 0.9$

$Sprinkler$	$Cloudy$	$P(S,C)$
1	0	90%
1	1	50%



$Raining$	$Cloudy$	$P(R,C)$
1	0	20%
1	1	50%

C	S	R	W	
0	1	0	1	$w = 0.9$

Wet	$Sprinkler$	$Raining$	$P(W,S,R)$
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%

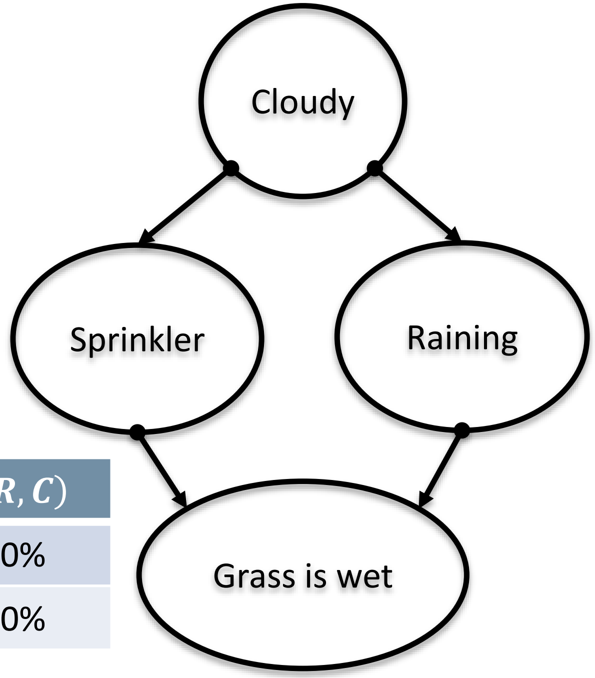
The Sprinkler Example: Sampling (3)

$Cloudy$	$P(C)$
1	50%

Let's start sampling x^k based on $U(0,1)$

- $s \sim U(0,1) \rhd s = 0.586 \rhd x_C^k = 0$
- $s \sim U(0,1) \rhd s = 0.951 \rhd x_S^k = 0$
- $s \sim U(0,1) \rhd s = 0.883 \rhd x_R^k = 0$
- $w^k = w^k \cdot P\left(x_W^k = 1 \middle| x_S^k, x_R^k\right) = 0$

$Sprinkler$	$Cloudy$	$P(S,C)$
1	0	90%
1	1	50%



$Raining$	$Cloudy$	$P(R,C)$
1	0	20%
1	1	50%

C	S	R	W	
0	1	0	1	$w = 0.9$
0	0	0	1	$w = 0$

Wet	$Sprinkler$	$Raining$	$P(W,S,R)$
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%

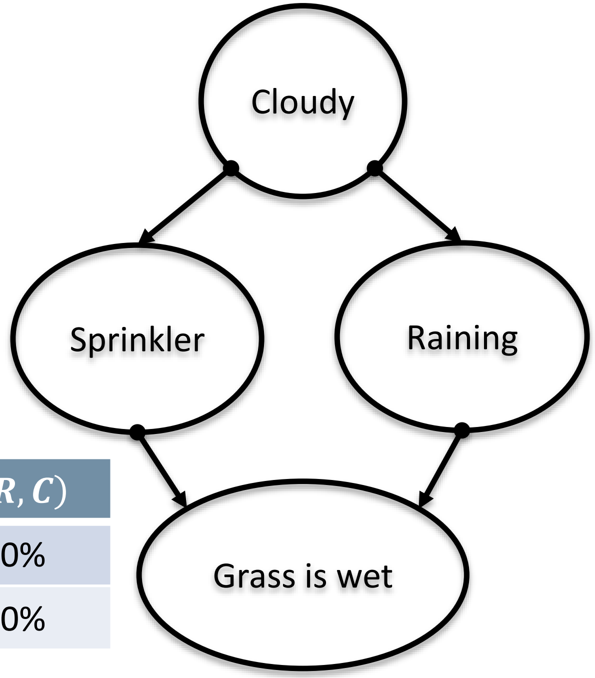
The Sprinkler Example: Sampling (4)

$Cloudy$	$P(C)$
1	50%

Let's start sampling x^k based on $U(0,1)$

- $s \sim U(0,1) \rhd s = 0.999 \rhd x_C^k = 0$
- $s \sim U(0,1) \rhd s = 0.855 \rhd x_S^k = 1$
- $s \sim U(0,1) \rhd s = 0.179 \rhd x_R^k = 1$
- $w^k = w^k \cdot P\left(x_W^k = 1 \middle| x_S^k, x_R^k\right) = 0.99$

$Sprinkler$	$Cloudy$	$P(S,C)$
1	0	90%
1	1	50%



$Raining$	$Cloudy$	$P(R,C)$
1	0	20%
1	1	50%

C	S	R	W	
0	1	0	1	$w = 0.9$
0	0	0	1	$w = 0$
0	1	1	1	$w = 0.99$

Wet	$Sprinkler$	$Raining$	$P(W,S,R)$
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%

The Sprinkler Example: Sampling (5)

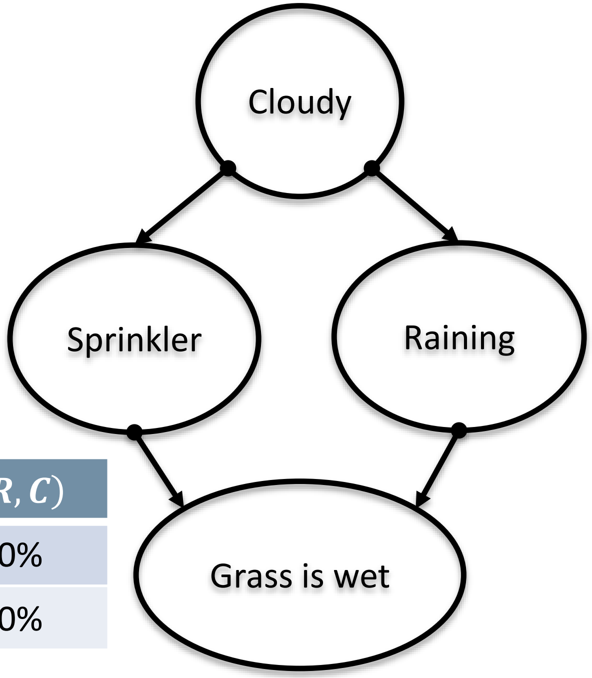
$Cloudy$	$P(C)$
1	50%

Let's start sampling x^k based on $U(0,1)$

- $s \sim U(0,1) \rhd s = 0.630 \rhd x_C^k = 0$
- $s \sim U(0,1) \rhd s = 0.455 \rhd x_S^k = 1$
- $s \sim U(0,1) \rhd s = 0.046 \rhd x_R^k = 1$
- $w^k = w^k \cdot P\left(x_W^k = 1 \middle| x_S^k, x_R^k\right) = 0.99$

C	S	R	W	
0	1	0	1	$w = 0.9$
0	0	0	1	$w = 0$
0	1	1	1	$w = 0.99$
0	1	1	1	$w = 0.99$

$Sprinkler$	$Cloudy$	$P(S,C)$
1	0	90%
1	1	50%



$Raining$	$Cloudy$	$P(R,C)$
1	0	20%
1	1	50%

Wet	$Sprinkler$	$Raining$	$P(W,S,R)$
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%

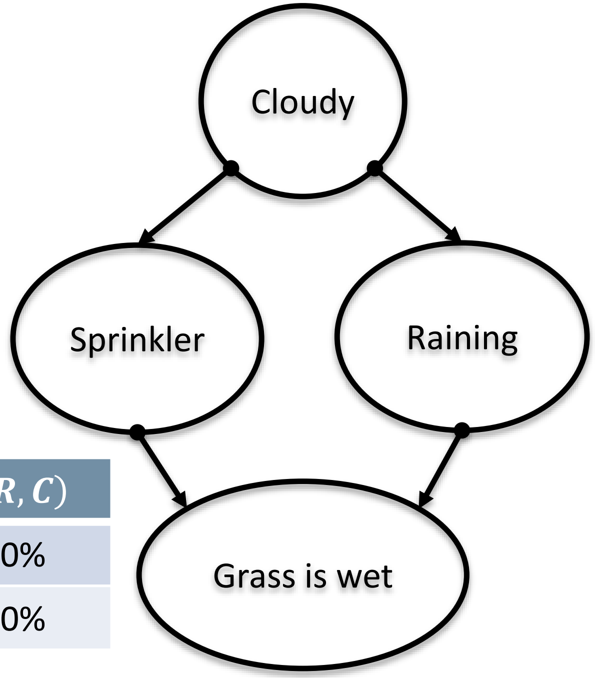
The Sprinkler Example: Sampling (6)

Cloudy	$P(C)$
1	50%

Let's start sampling x^k based on $U(0,1)$

- $s \sim U(0,1) \rhd s = 0.522 \rhd x_C^k = 0$
- $s \sim U(0,1) \rhd s = 0.193 \rhd x_S^k = 1$
- $s \sim U(0,1) \rhd s = 0.159 \rhd x_R^k = 1$
- $w^k = w^k \cdot P\left(x_W^k = 1 \middle| x_S^k, x_R^k\right) = 0.99$

Sprinkler	Cloudy	$P(S,C)$
1	0	90%
1	1	50%



Raining	Cloudy	$P(R,C)$
1	0	20%
1	1	50%

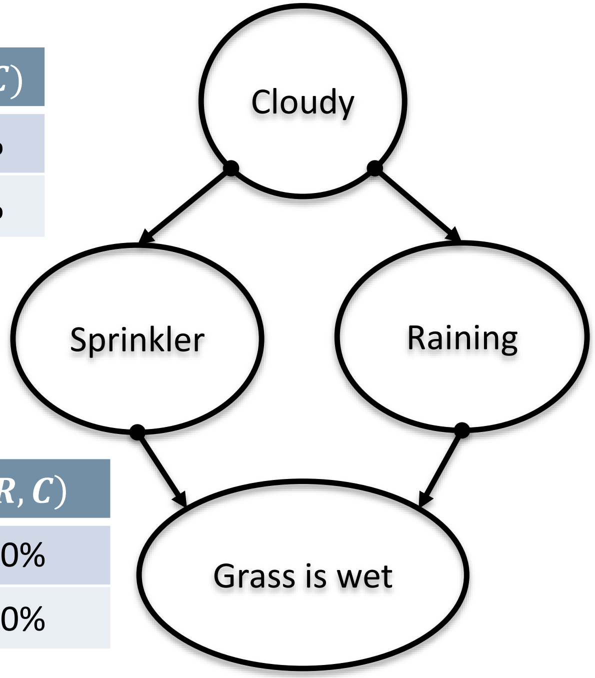
C	S	R	W	
0	1	0	1	$w = 0.9$
0	0	0	1	$w = 0$
0	1	1	1	$w = 0.99$
0	1	1	1	$w = 0.99$
0	1	1	1	$w = 0.99$

Wet	Sprinkler	Raining	$P(W,S,R)$
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%

The Sprinkler Example: Sampling (7)

Cloudy	$P(C)$
1	50%

Sprinkler	*Cloudy*	$P(S,C)$
1	0	90%
1	1	50%



Raining	*Cloudy*	$P(R,C)$
1	0	20%
1	1	50%

Wet	*Sprinkler*	*Raining*	$P(W,S,R)$
1	0	0	0%
1	0	1	90%
1	1	0	90%
1	1	1	99%

Let's start sampling x^k based on $U(0,1)$

- $s \sim U(0,1) \rhd s = 0.288 \rhd x_C^k = 1$
- $s \sim U(0,1) \rhd s = 0.794 \rhd x_S^k = 0$
- $s \sim U(0,1) \rhd s = 0.483 \rhd x_R^k = 1$
- $s \sim U(0,1) \rhd s = 0.216 \rhd x_W^k = 0.9$

C	S	R	W	
0	1	0	1	$w = 0.9$
0	0	0	1	$w = 0$
0	1	1	1	$w = 0.99$
0	1	1	1	$w = 0.99$
0	1	1	1	$w = 0.99$
1	0	1	1	$w = 0.9$

$$P(C|W) \approx \frac{\sum_{k=1}^{K} w^k \cdot I_{x_C^k=1}}{\sum_{k=1}^{K} w^k} = \frac{.9}{.9 + .99 + .99 + .99 + .9} = \frac{0.9}{4.77} = 0.1886$$