Exact Inference in Bayesian Networks

Outline

I. Efficient representation of conditional distributions

II. Probabilistic query using a BN

III. Variable Elimination

IV. Variable ordering and relevance

* Figures are from the textbook site.

I. Efficient Representations

- The size 2^k of the conditional probability table (CPT) for a node with k parents is the worst-case scenario in which the relationships with the parents are arbitrary.

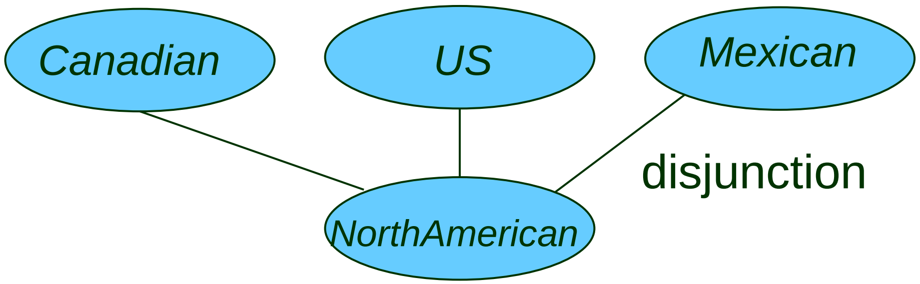
I. Efficient Representations

- ♣ The size 2^k of the conditional probability table (CPT) for a node with k parents is the worst-case scenario in which the relationships with the parents are arbitrary.
- ♦ Deterministic nodes. Such a node has its value specified exactly by those of its parents, with no uncertainty. Specify deterministic functions in a BN.

I. Efficient Representations

- The size 2^k of the conditional probability table (CPT) for a node with k parents is the worst-case scenario in which the relationships with the parents are arbitrary.

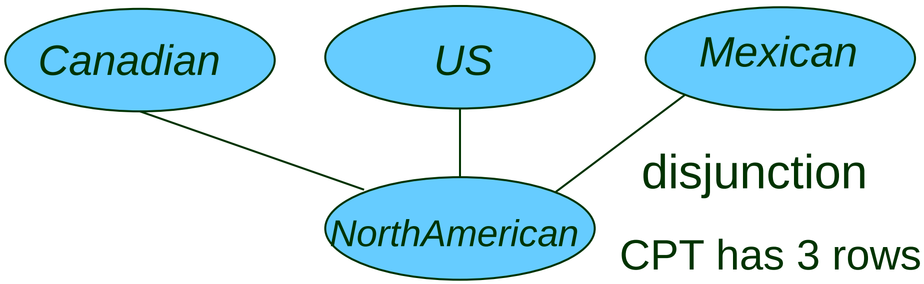
- Deterministic nodes. Such a node has its value specified exactly by those of its parents, with no uncertainty. Specify deterministic functions in a BN.



I. Efficient Representations

- The size 2^k of the conditional probability table (CPT) for a node with k parents is the worst-case scenario in which the relationships with the parents are arbitrary.

- Deterministic nodes. Such a node has its value specified exactly by those of its parents, with no uncertainty. Specify deterministic functions in a BN.

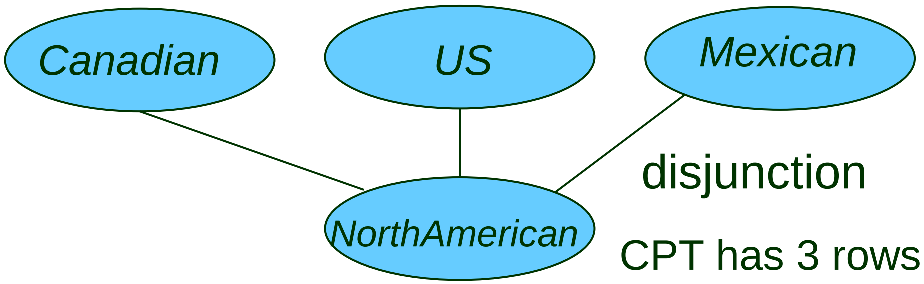


CPT has 3 rows instead of 8.

I. Efficient Representations

- The size 2^k of the conditional probability table (CPT) for a node with k parents is the worst-case scenario in which the relationships with the parents are arbitrary.

- Deterministic nodes. Such a node has its value specified exactly by those of its parents, with no uncertainty. Specify deterministic functions in a BN.



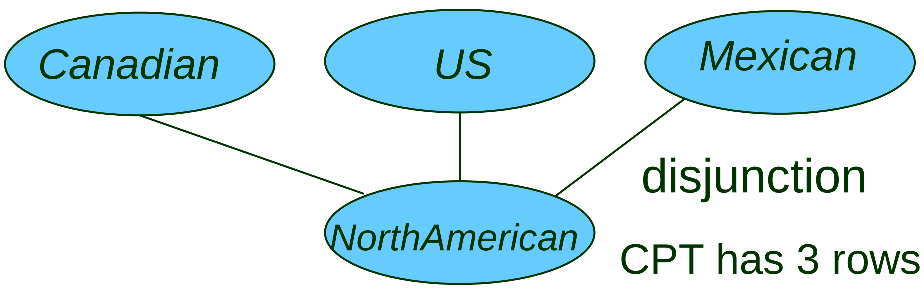
CPT has 3 rows instead of 8.

- Context-specific independence. A variable is conditionally independent of some of its parents given certain values of others.

I. Efficient Representations

- The size 2^k of the conditional probability table (CPT) for a node with k parents is the worst-case scenario in which the relationships with the parents are arbitrary.

- Deterministic nodes. Such a node has its value specified exactly by those of its parents, with no uncertainty. Specify deterministic functions in a BN.



disjunction

CPT has 3 rows instead of 8.

- Context-specific independence. A variable is conditionally independent of some of its parents given certain values of others.

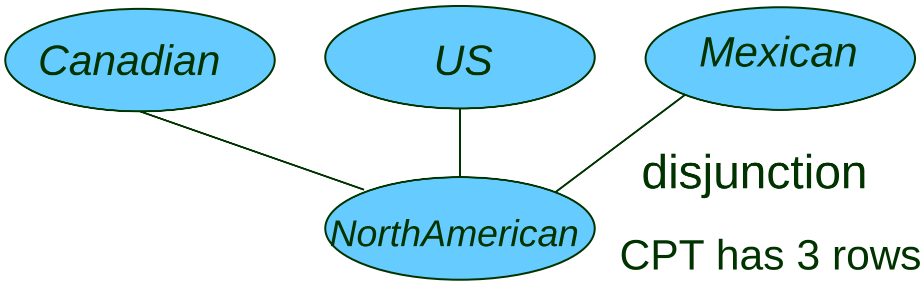
$$\mathbf{P}(\mathit{Damage} \mid \mathit{Ruggedness}, \mathit{Accident}) =$$
$$\text{if } (\mathit{Accident} = \text{false}) \text{ then } d_1 \text{ else } d_2(\mathit{Ruggedness})$$

// if no accident, damage to
// your car does not depend
// on ruggedness.

I. Efficient Representations

- The size 2^k of the conditional probability table (CPT) for a node with k parents is the worst-case scenario in which the relationships with the parents are arbitrary.

- Deterministic nodes. Such a node has its value specified exactly by those of its parents, with no uncertainty. Specify deterministic functions in a BN.



CPT has 3 rows instead of 8.

- Context-specific independence. A variable is conditionally independent of some of its parents given certain values of others.

$\mathbf{P}(Damage \mid Ruggedness, Accident) =$
if $(Accident = \text{false})$ then d_1 else $d_2(Ruggedness)$

some distributions

// if no accident, damage to
// your car does not depend
// on ruggedness.

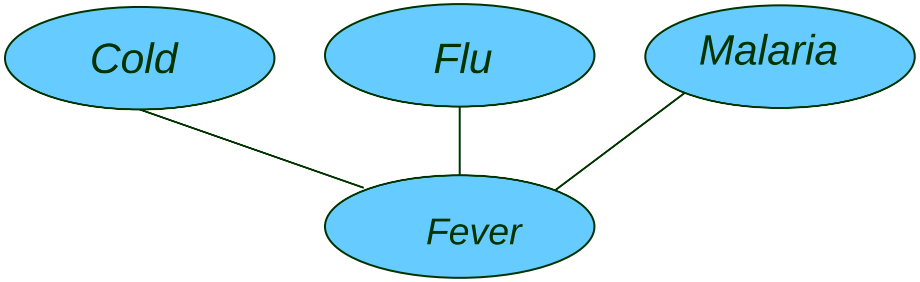
Noisy-OR Relation

- Characterizes uncertain relationships.
- Generalizes logical OR.

Noisy-OR Relation

- Characterizes uncertain relationships.
- Generalizes logical OR.

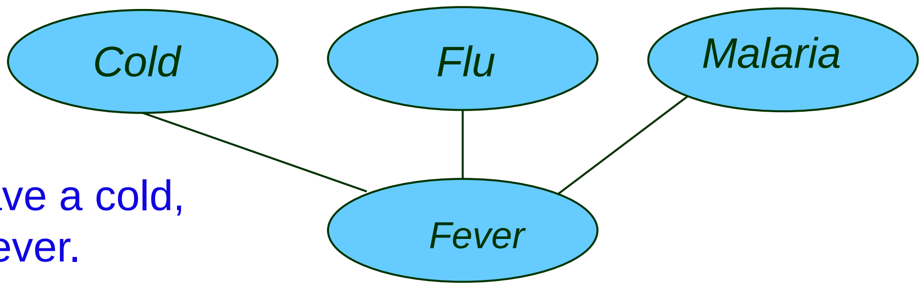
$$Fever \Leftrightarrow Cold \vee Flu \vee Malaria$$ (Propositional logic)



Noisy-OR Relation

- Characterizes uncertain relationships.
- Generalizes logical OR.

$Fever \Leftrightarrow Cold \vee Flu \vee Malaria$ (Propositional logic)

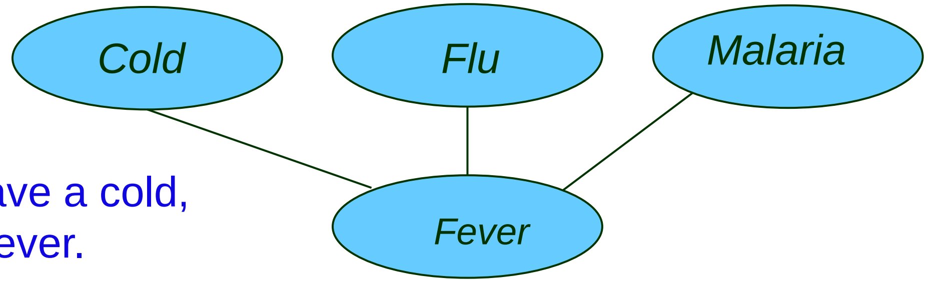


A patient could have a cold,
but not exhibit a fever.

Noisy-OR Relation

- Characterizes uncertain relationships.
- Generalizes logical OR.

$Fever \Leftrightarrow Cold \vee Flu \vee Malaria$ (Propositional logic)



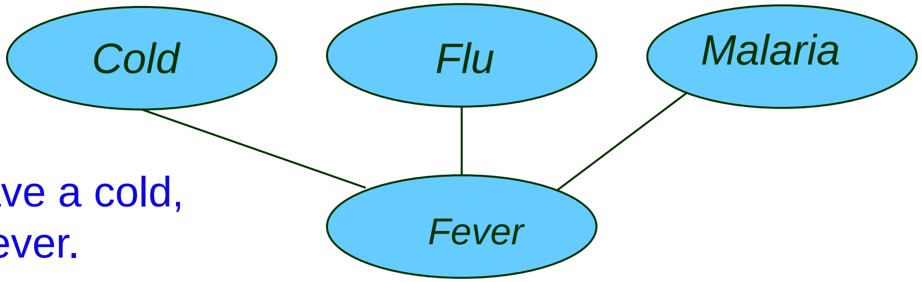
A patient could have a cold,
but not exhibit a fever.

Noisy-OR allows for uncertainty about each of *Cold*, *Flu*, *Malaria* to cause Fever.

Noisy-OR Relation

- Characterizes uncertain relationships.
- Generalizes logical OR.

$Fever \Leftrightarrow Cold \vee Flu \vee Malaria$ (Propositional logic)



A patient could have a cold,
but not exhibit a fever.

Noisy-OR allows for uncertainty about each of *Cold*, *Flu*, *Malaria* to cause Fever.

- The causal relationship between a node and its parents may be *inhibited*.
  - List all the possible causes.
  - Inhibition of each parent is independent of inhibition of any other parents.

Causes of Fever

$Cold$	Flu	$Malaria$	$P(fever \mid \cdot)$	$P(\neg fever \mid \cdot)$
f	f	f	0.0	1.0
f	f	t	0.9	**0.1**
f	t	f	0.8	**0.2**
f	t	t	0.98	$0.02 = 0.2 \times 0.1$
t	f	f	0.4	**0.6**
t	f	t	0.94	$0.06 = 0.6 \times 0.1$
t	t	f	0.88	$0.12 = 0.6 \times 0.2$
t	t	t	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

- Whatever inhibits cold from causing a fever is independent of whatever inhibits flu from causing a fever.
- *Fever* is false if and only if all its parents are inhibited.

Causes of Fever

Cold	*Flu*	*Malaria*	$P(fever \mid \cdot)$	$P(\neg fever \mid \cdot)$
f	f	f	0.0	1.0
f	f	t	0.9	**0.1**
f	t	f	0.8	**0.2**
f	t	t	0.98	$0.02 = 0.2 \times 0.1$
t	f	f	0.4	**0.6**
t	f	t	0.94	$0.06 = 0.6 \times 0.1$
t	t	f	0.88	$0.12 = 0.6 \times 0.2$
t	t	t	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

- Random variables:

  $X_1 \equiv Cold$, $X_2 \equiv Flu$, $X_3 \equiv Malaria$

- Whatever inhibits cold from causing a fever is independent of whatever inhibits flu from causing a fever.
- *Fever* is false if and only if all its parents are inhibited.

Causes of Fever

$Cold$	Flu	$Malaria$	$P(fever \mid \cdot)$	$P(\neg fever \mid \cdot)$
f	f	f	0.0	1.0
f	f	t	0.9	**0.1**
f	t	f	0.8	**0.2**
f	t	t	0.98	$0.02 = 0.2 \times 0.1$
t	f	f	0.4	**0.6**
t	f	t	0.94	$0.06 = 0.6 \times 0.1$
t	t	f	0.88	$0.12 = 0.6 \times 0.2$
t	t	t	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

- Random variables:

  $X_1 \equiv Cold$, $X_2 \equiv Flu$, $X_3 \equiv Malaria$

- Inhibition probabilities:

- Whatever inhibits cold from causing a fever is independent of whatever inhibits flu from causing a fever.
- *Fever* is false if and only if all its parents are inhibited.

Causes of Fever

Cold	*Flu*	*Malaria*	$P(fever \mid \cdot)$	$P(\neg fever \mid \cdot)$
f	f	f	0.0	1.0
f	f	t	0.9	**0.1**
f	t	f	0.8	**0.2**
f	t	t	0.98	$0.02 = 0.2 \times 0.1$
t	f	f	0.4	**0.6**
t	f	t	0.94	$0.06 = 0.6 \times 0.1$
t	t	f	0.88	$0.12 = 0.6 \times 0.2$
t	t	t	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

- Random variables:

  $X_1 \equiv Cold$, $X_2 \equiv Flu$, $X_3 \equiv Malaria$

- Inhibition probabilities:

- Whatever inhibits cold from causing a fever is independent of whatever inhibits flu from causing a fever.
- *Fever* is false if and only if all its parents are inhibited.

Causes of Fever

$Cold$	Flu	$Malaria$	$P(fever \mid \cdot)$	$P(\neg fever \mid \cdot)$
f	f	f	0.0	1.0
f	f	t	0.9	**0.1**
f	t	f	0.8	**0.2**
f	t	t	0.98	$0.02 = 0.2 \times 0.1$
t	f	f	0.4	**0.6**
t	f	t	0.94	$0.06 = 0.6 \times 0.1$
t	t	f	0.88	$0.12 = 0.6 \times 0.2$
t	t	t	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

- Random variables:

  $X_1 \equiv Cold$, $X_2 \equiv Flu$, $X_3 \equiv Malaria$

- Inhibition probabilities:

$$q_1 = q_{\text{cold}} \quad \text{// false alarm}$$
$$= P(\neg fever \mid cold, \neg flu, \neg malaria) = 0.6$$

- Whatever inhibits cold from causing a fever is independent of whatever inhibits flu from causing a fever.
- *Fever* is false if and only if all its parents are inhibited.

Causes of Fever

$Cold$	Flu	$Malaria$	$P(fever \mid \cdot)$	$P(\neg fever \mid \cdot)$
f	f	f	0.0	1.0
f	f	t	0.9	**0.1**
f	t	f	0.8	**0.2**
f	t	t	0.98	$0.02 = 0.2 \times 0.1$
t	f	f	0.4	**0.6**
t	f	t	0.94	$0.06 = 0.6 \times 0.1$
t	t	f	0.88	$0.12 = 0.6 \times 0.2$
t	t	t	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

- Random variables:

  $X_1 \equiv Cold$, $X_2 \equiv Flu$, $X_3 \equiv Malaria$

- Inhibition probabilities:

$q_1 = q_{\text{cold}}$ // false alarm

$= P(\neg fever \mid cold, \neg flu, \neg malaria) = 0.6$

- Whatever inhibits cold from causing a fever is independent of whatever inhibits flu from causing a fever.
- *Fever* is false if and only if all its parents are inhibited.

Causes of Fever

Cold	*Flu*	*Malaria*	$P(fever \mid \cdot)$	$P(\neg fever \mid \cdot)$
f	f	f	0.0	1.0
f	f	t	0.9	**0.1**
f	t	f	0.8	**0.2**
f	t	t	0.98	$0.02 = 0.2 \times 0.1$
t	f	f	0.4	**0.6**
t	f	t	0.94	$0.06 = 0.6 \times 0.1$
t	t	f	0.88	$0.12 = 0.6 \times 0.2$
t	t	t	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

- Random variables:

  $X_1 \equiv Cold$, $X_2 \equiv Flu$, $X_3 \equiv Malaria$

- Inhibition probabilities:

$q_1 = q_{\text{cold}}$ // false alarm

$= P(\neg fever \mid cold, \neg flu, \neg malaria) = 0.6$

$q_2 = q_{\text{flu}}$

$= P(\neg fever \mid \neg cold, flu, \neg malaria) = 0.2$

- Whatever inhibits cold from causing a fever is independent of whatever inhibits flu from causing a fever.
- *Fever* is false if and only if all its parents are inhibited.

Causes of Fever

$Cold$	Flu	$Malaria$	$P(fever \mid \cdot)$	$P(\neg fever \mid \cdot)$
f	f	f	0.0	1.0
f	f	t	0.9	**0.1**
f	t	f	0.8	**0.2**
f	t	t	0.98	$0.02 = 0.2 \times 0.1$
t	f	f	0.4	**0.6**
t	f	t	0.94	$0.06 = 0.6 \times 0.1$
t	t	f	0.88	$0.12 = 0.6 \times 0.2$
t	t	t	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

- Random variables:

  $X_1 \equiv \textit{Cold}$, $X_2 \equiv \textit{Flu}$, $X_3 \equiv \textit{Malaria}$

- Inhibition probabilities:

$$q_1 = q_{\text{cold}} \quad \text{// false alarm}$$
$$= P(\neg \textit{fever} \mid \textit{cold}, \neg \textit{flu}, \neg \textit{malaria}) = 0.6$$
$$q_2 = q_{\text{flu}}$$
$$= P(\neg \textit{fever} \mid \neg \textit{cold}, \textit{flu}, \neg \textit{malaria}) = 0.2$$

- Whatever inhibits cold from causing a fever is independent of whatever inhibits flu from causing a fever.
- *Fever* is false if and only if all its parents are inhibited.

Causes of Fever

Cold	*Flu*	*Malaria*	$P(fever \mid \cdot)$	$P(\neg fever \mid \cdot)$
f	f	f	0.0	1.0
f	f	t	0.9	**0.1**
f	t	f	0.8	**0.2**
f	t	t	0.98	$0.02 = 0.2 \times 0.1$
t	f	f	0.4	**0.6**
t	f	t	0.94	$0.06 = 0.6 \times 0.1$
t	t	f	0.88	$0.12 = 0.6 \times 0.2$
t	t	t	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

- Random variables:

  $X_1 \equiv Cold$, $X_2 \equiv Flu$, $X_3 \equiv Malaria$

- Inhibition probabilities:

$q_1 = q_{\text{cold}}$ // false alarm

$= P(\neg fever \mid cold, \neg flu, \neg malaria) = 0.6$

$q_2 = q_{\text{flu}}$

$= P(\neg fever \mid \neg cold, flu, \neg malaria) = 0.2$

$q_3 = q_{\text{malaria}}$

$= P(\neg fever \mid \neg cold, \neg flu, malaria) = 0.1$

- Whatever inhibits cold from causing a fever is independent of whatever inhibits flu from causing a fever.
- *Fever* is false if and only if all its parents are inhibited.

Causes of Fever

$Cold$	Flu	$Malaria$	$P(fever \mid \cdot)$	$P(\neg fever \mid \cdot)$
f	f	f	0.0	1.0
f	f	t	0.9	**0.1**
f	t	f	0.8	**0.2**
f	t	t	0.98	$0.02 = 0.2 \times 0.1$
t	f	f	0.4	**0.6**
t	f	t	0.94	$0.06 = 0.6 \times 0.1$
t	t	f	0.88	$0.12 = 0.6 \times 0.2$
t	t	t	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

- Random variables:

  $X_1 \equiv Cold,\ X_2 \equiv Flu,\ X_3 \equiv Malaria$

- Inhibition probabilities:

$q_1 = q_{\text{cold}}$ // false alarm

$= P(\neg fever \mid cold, \neg flu, \neg malaria) = 0.6$

$q_2 = q_{\text{flu}}$

$= P(\neg fever \mid \neg cold, flu, \neg malaria) = 0.2$

$q_3 = q_{\text{malaria}}$

$= P(\neg fever \mid \neg cold, \neg flu, malaria) = 0.1$

- Whatever inhibits cold from causing a fever is independent of whatever inhibits flu from causing a fever.
- *Fever* is false if and only if all its parents are inhibited.

Causes of Fever

Cold	*Flu*	*Malaria*	$P(fever \mid \cdot)$	$P(\neg fever \mid \cdot)$
f	f	f	0.0	1.0
f	f	t	0.9	**0.1**
f	t	f	0.8	**0.2**
f	t	t	0.98	$0.02 = 0.2 \times 0.1$
t	f	f	0.4	**0.6**
t	f	t	0.94	$0.06 = 0.6 \times 0.1$
t	t	f	0.88	$0.12 = 0.6 \times 0.2$
t	t	t	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

- Random variables:

  $X_1 \equiv Cold$, $X_2 \equiv Flu$, $X_3 \equiv Malaria$

- Inhibition probabilities:

$q_1 = q_{\text{cold}}$ // false alarm

$= P(\neg fever \mid cold, \neg flu, \neg malaria) = 0.6$

$q_2 = q_{\text{flu}}$

$= P(\neg fever \mid \neg cold, flu, \neg malaria) = 0.2$

$q_3 = q_{\text{malaria}}$

$= P(\neg fever \mid \neg cold, \neg flu, malaria) = 0.1$

- Whatever inhibits cold from causing a fever is independent of whatever inhibits flu from causing a fever.

- *Fever* is false if and only if all its parents are inhibited.

Causes of Fever

Cold	Flu	Malaria	$P(fever \mid \cdot)$	$P(\neg fever \mid \cdot)$
f	f	f	0.0	1.0
f	f	t	0.9	**0.1**
f	t	f	0.8	**0.2**
f	t	t	0.98	$0.02 = 0.2 \times 0.1$
t	f	f	0.4	**0.6**
t	f	t	0.94	$0.06 = 0.6 \times 0.1$
t	t	f	0.88	$0.12 = 0.6 \times 0.2$
t	t	t	0.988	$0.012 = 0.6 \times 0.2 \times 0.1$

- Random variables:

  $X_1 \equiv Cold,\ X_2 \equiv Flu,\ X_3 \equiv Malaria$

- Inhibition probabilities:

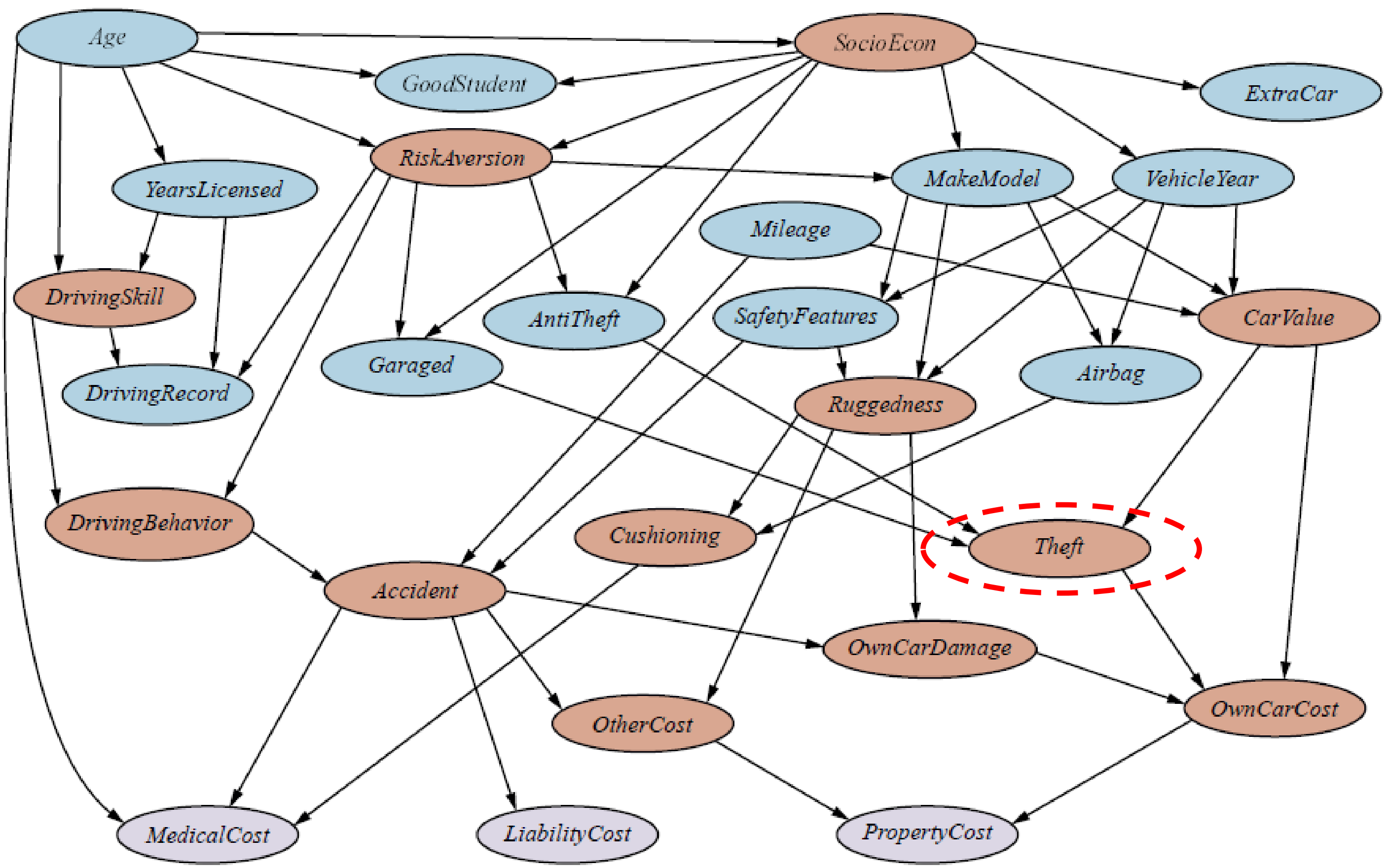
$$q_1 = q_{\text{cold}} \quad \text{// false alarm}$$
$$= P(\neg fever \mid cold, \neg flu, \neg malaria) = 0.6$$
$$q_2 = q_{\text{flu}}$$
$$= P(\neg fever \mid \neg cold, flu, \neg malaria) = 0.2$$
$$q_3 = q_{\text{malaria}}$$
$$= P(\neg fever \mid \neg cold, \neg flu, malaria) = 0.1$$

- Whatever inhibits cold from causing a fever is independent of whatever inhibits flu from causing a fever.
- *Fever* is false if and only if all its parents are inhibited.

$$P(x_i \mid parents(X_i)) = 1 - \prod_{\{j : X_j = true\}} q_j$$

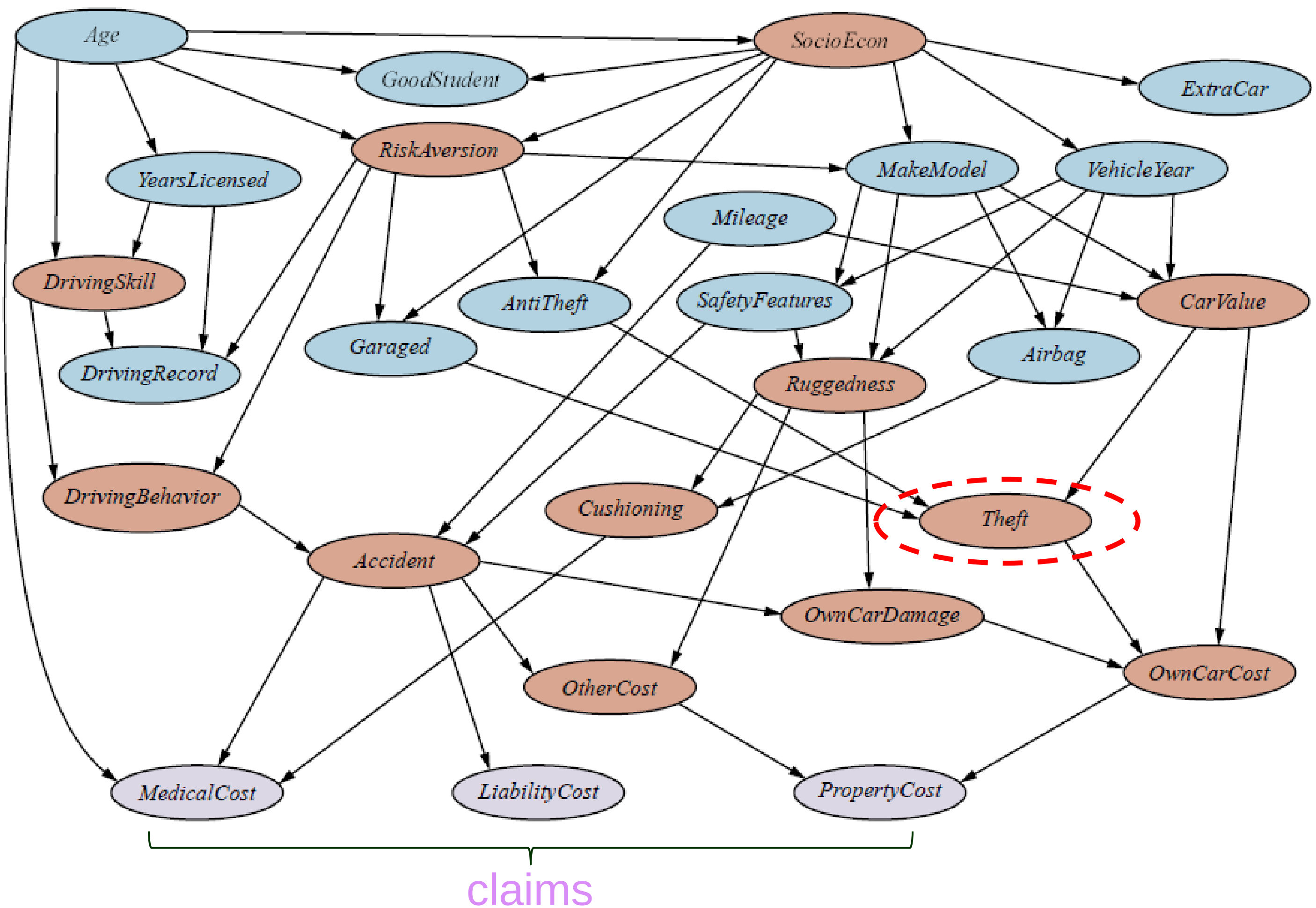
Case Study: Car Insurance

A car insurance company processes a car insurance application to decide on the annual premium based on the anticipated claims it will pay out for the applicant.



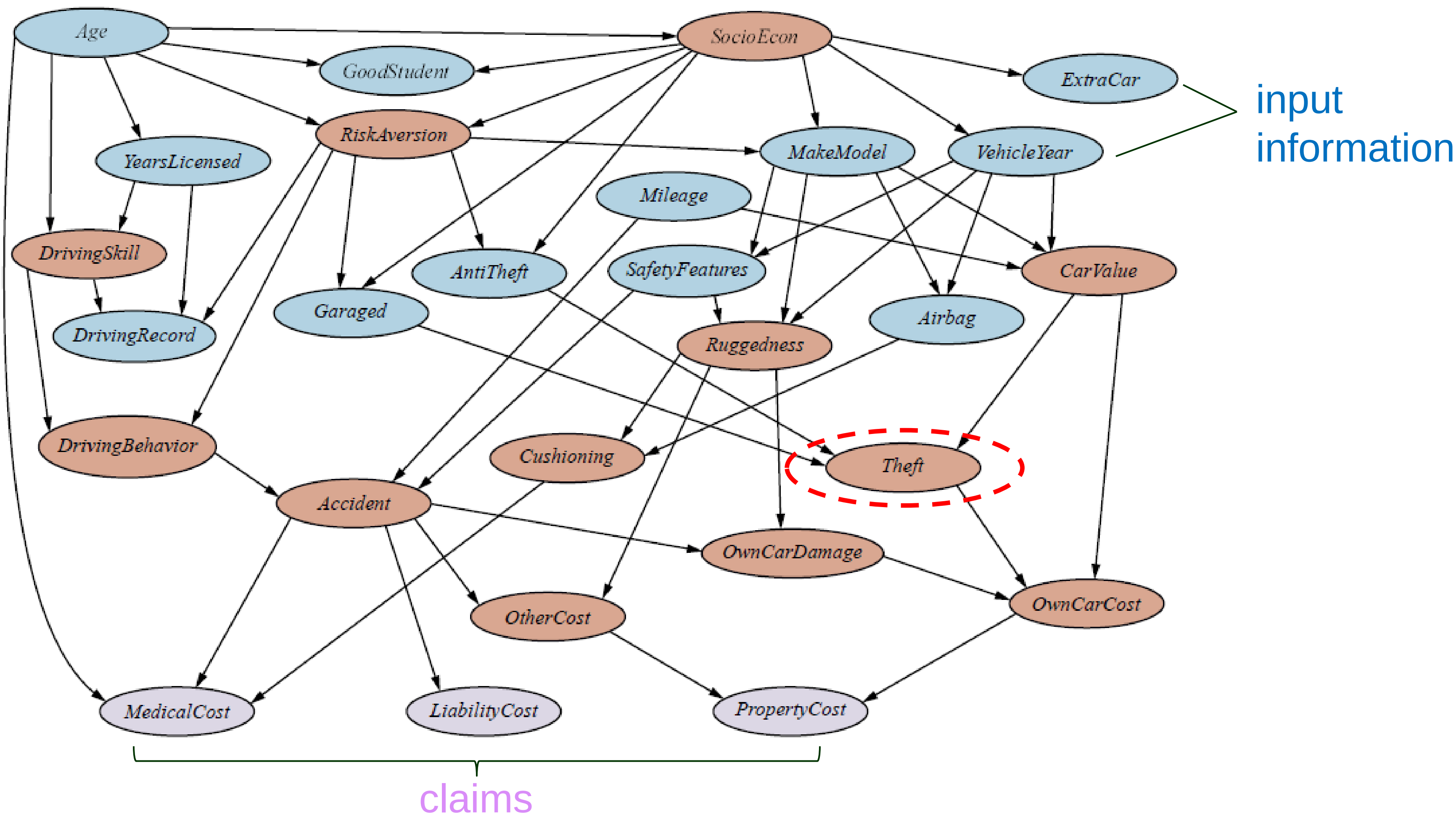
Case Study: Car Insurance

A car insurance company processes a car insurance application to decide on the annual premium based on the anticipated claims it will pay out for the applicant.



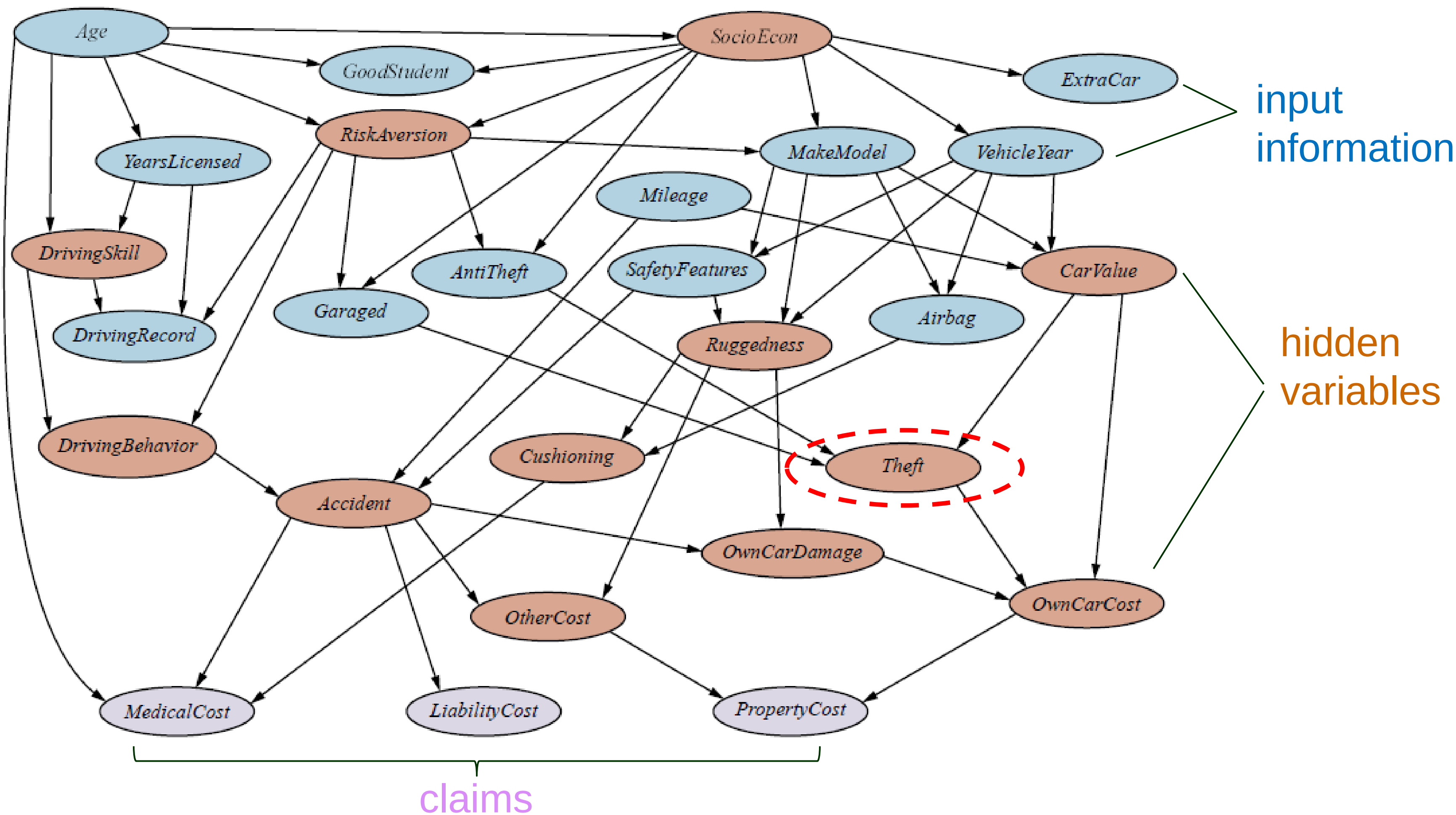
Case Study: Car Insurance

A car insurance company processes a car insurance application to decide on the annual premium based on the anticipated claims it will pay out for the applicant.



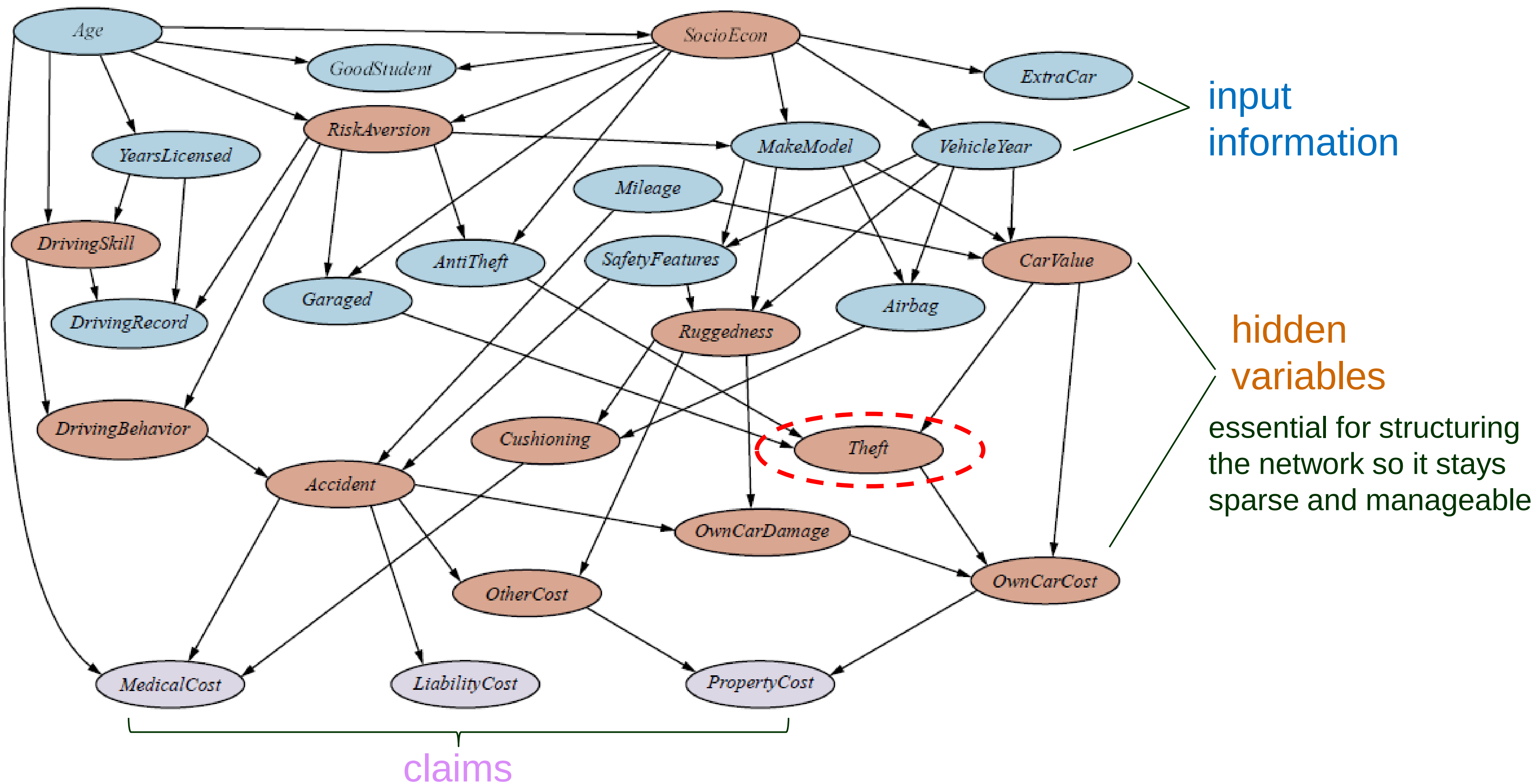
Case Study: Car Insurance

A car insurance company processes a car insurance application to decide on the annual premium based on the anticipated claims it will pay out for the applicant.



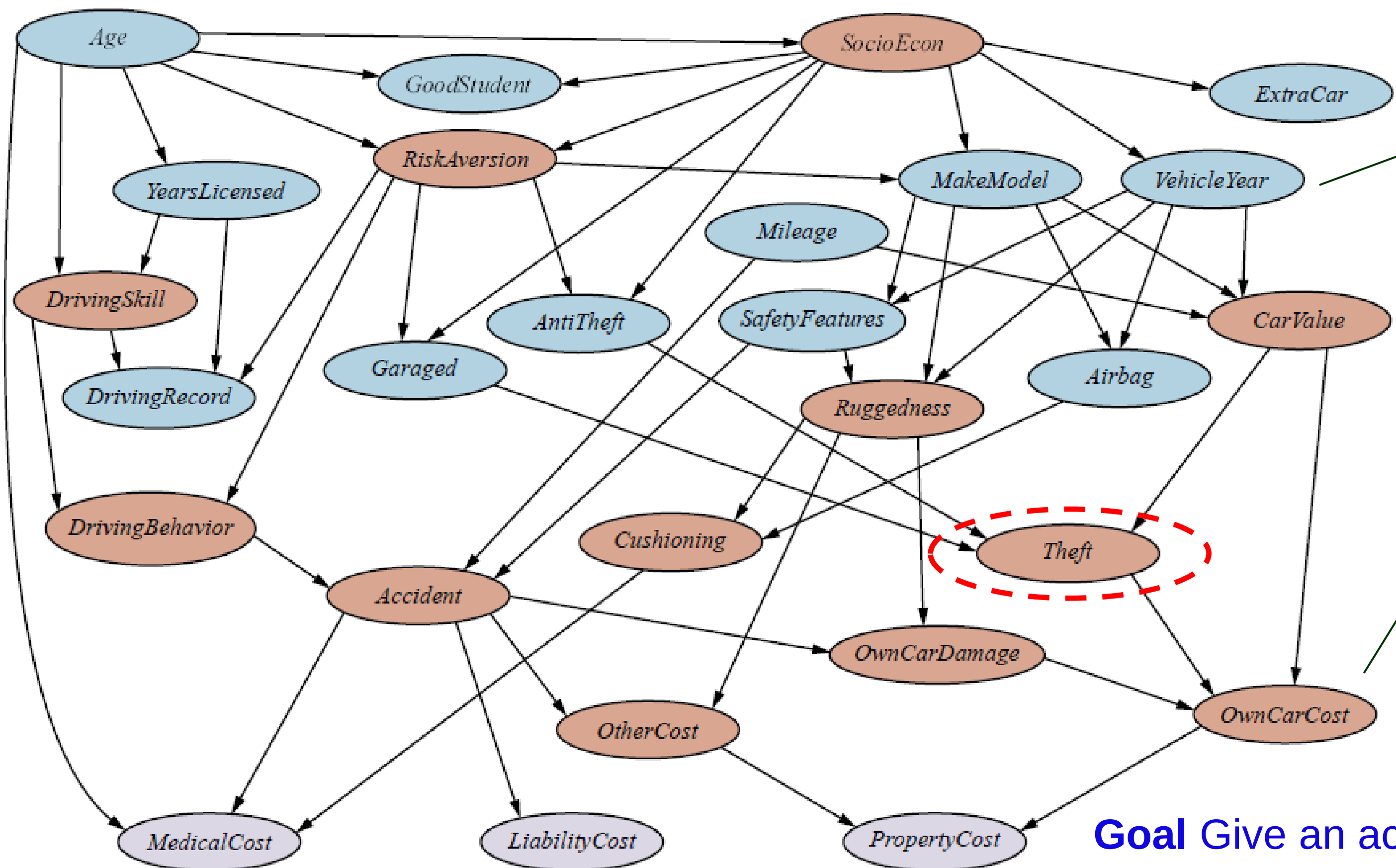
Case Study: Car Insurance

A car insurance company processes a car insurance application to decide on the annual premium based on the anticipated claims it will pay out for the applicant.



Case Study: Car Insurance

A car insurance company processes a car insurance application to decide on the annual premium based on the anticipated claims it will pay out for the applicant.



input information

hidden variables

essential for structuring the network so it stays sparse and manageable

claims

Goal Give an accurate, well-calibrated distribution over the output variables given the evidence available form the input.

II. Probabilistic Query

X: query variable

$\boldsymbol{E} = \{E_1, \ldots. E_m\}$: evidence variables

$\boldsymbol{e} = \{e_1, \ldots. e_m\}$: an observed event

$\boldsymbol{Y} = \{Y_1, \ldots. Y_l\}$: hidden variables

II. Probabilistic Query

X: query variable

$\boldsymbol{E} = \{E_1, \dots . E_m\}$: evidence variables

$\boldsymbol{e} = \{e_1, \dots . e_m\}$: an observed event

$\boldsymbol{Y} = \{Y_1, \dots . Y_l\}$: hidden variables

Complete variable set: $\{X\} \cup \boldsymbol{E} \cup \boldsymbol{Y}$

II. Probabilistic Query

X: query variable

$\boldsymbol{E} = \{E_1, \ldots . E_m\}$: evidence variables

$\boldsymbol{e} = \{e_1, \ldots . e_m\}$: an observed event

$\boldsymbol{Y} = \{Y_1, \ldots . Y_l\}$: hidden variables

Complete variable set: $\{X\} \cup \boldsymbol{E} \cup \boldsymbol{Y}$

Query $\boldsymbol{P}(X \mid \boldsymbol{e})$?

II. Probabilistic Query

X: query variable

$\boldsymbol{E} = \{E_1, \ldots. E_m\}$: evidence variables

$\boldsymbol{e} = \{e_1, \ldots. e_m\}$: an observed event

$\boldsymbol{Y} = \{Y_1, \ldots. Y_l\}$: hidden variables

Complete variable set: $\{X\} \cup \boldsymbol{E} \cup \boldsymbol{Y}$

Query $\boldsymbol{P}(X \mid \boldsymbol{e})$?

// $\boldsymbol{P}(Burglary \mid j, m)$

$\boldsymbol{P}(Burglary \mid JohnCalls = true, MaryCalls = true) = \langle 0.284, 0.716 \rangle$

II. Probabilistic Query

X: query variable

$\boldsymbol{E} = \{E_1, \ldots . E_m\}$: evidence variables

$\boldsymbol{e} = \{e_1, \ldots . e_m\}$: an observed event

$\boldsymbol{Y} = \{Y_1, \ldots . Y_l\}$: hidden variables

Complete variable set: $\{X\} \cup \boldsymbol{E} \cup \boldsymbol{Y}$

Query $\boldsymbol{P}(X \mid \boldsymbol{e})$?

// $\boldsymbol{P}(Burglary \mid j, m)$

$\boldsymbol{P}(Burglary \mid JohnCalls = true, MaryCalls = true) = \langle 0.284, 0.716 \rangle$

We will discuss exact algorithms for posterior probability computation.

Inference by Enumeration

$$\boldsymbol{P}(X, \boldsymbol{e}) = \boldsymbol{P}(X \mid \boldsymbol{e}) P(\boldsymbol{e})$$

vector of evidences

Inference by Enumeration

$$\boldsymbol{P}(X, \boldsymbol{e}) = \boldsymbol{P}(X \mid \boldsymbol{e}) P(\boldsymbol{e})$$

vector of evidences ⇓

$$\boldsymbol{P}(X \mid \boldsymbol{e}) = \alpha \boldsymbol{P}(X, \boldsymbol{e}) = \alpha \sum_{\boldsymbol{y}} \boldsymbol{P}(X, \boldsymbol{e}, \boldsymbol{y})$$

Inference by Enumeration

$$\mathbf{P}(X, \mathbf{e}) = \mathbf{P}(X \mid \mathbf{e})P(\mathbf{e})$$

vector of evidences

$$\mathbf{P}(X \mid \mathbf{e}) = \alpha \mathbf{P}(X, \mathbf{e}) = \alpha \sum_{\mathbf{y}} \mathbf{P}(X, \mathbf{e}, \mathbf{y})$$

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(X_i))$$

Inference by Enumeration

$$\mathbf{P}(X, \mathbf{e}) = \mathbf{P}(X \mid \mathbf{e})P(\mathbf{e})$$

vector of evidences

$$\mathbf{P}(X \mid \mathbf{e}) = \alpha \mathbf{P}(X, \mathbf{e}) = \alpha \sum_{\mathbf{y}} \mathbf{P}(X, \mathbf{e}, \mathbf{y})$$

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(X_i))$$

Inference by Enumeration

$$\mathbf{P}(X, \mathbf{e}) = \mathbf{P}(X \mid \mathbf{e}) P(\mathbf{e})$$

vector of evidences

$$\mathbf{P}(X \mid \mathbf{e}) = \alpha \mathbf{P}(X, \mathbf{e}) = \alpha \sum_{\mathbf{y}} \mathbf{P}(X, \mathbf{e}, \mathbf{y})$$

$$P(x_1, \dots, x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(X_i))$$

computable as products of
conditional probabilities from the BN.

Inference by Enumeration

$$\mathbf{P}(X, \mathbf{e}) = \mathbf{P}(X \mid \mathbf{e}) P(\mathbf{e})$$

vector of evidences

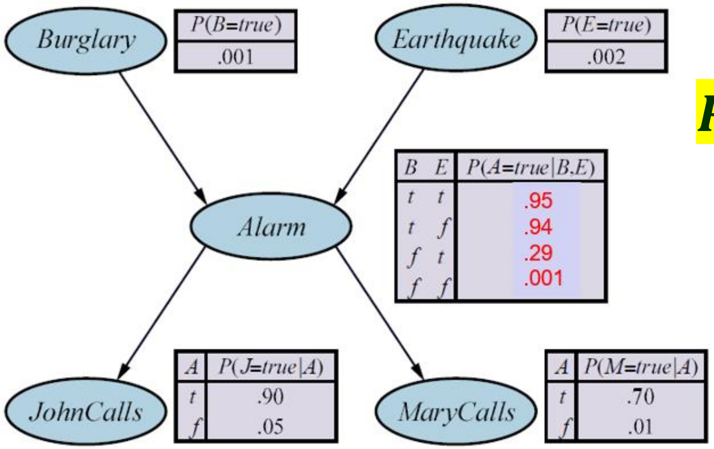
$$\mathbf{P}(X \mid \mathbf{e}) = \alpha \mathbf{P}(X, \mathbf{e}) = \alpha \sum_{\mathbf{y}} \mathbf{P}(X, \mathbf{e}, \mathbf{y})$$

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(X_i))$$

computable as products of conditional probabilities from the BN.

- Answer the query $\mathbf{P}(X \mid \mathbf{e})$ using a BN.

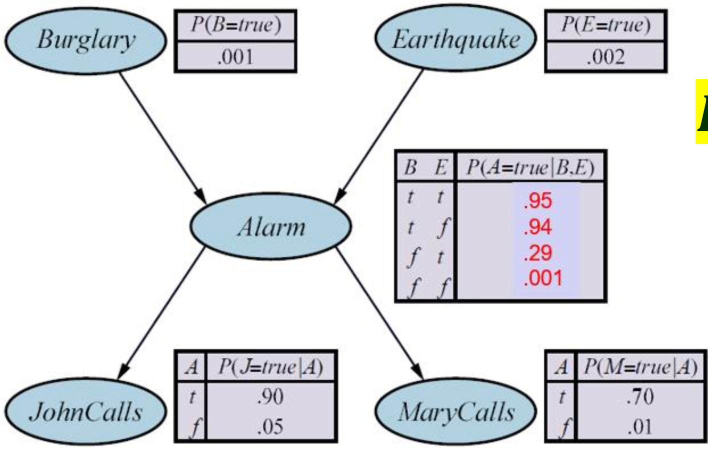
Burglary Example (revisited)



$\mathbf{P}(Burglary \mid JohnCalls = true, MaryCalls = true)?$

Hidden variables: $Y = \{Earthquake, Alarm\}$

Burglary Example (revisited)

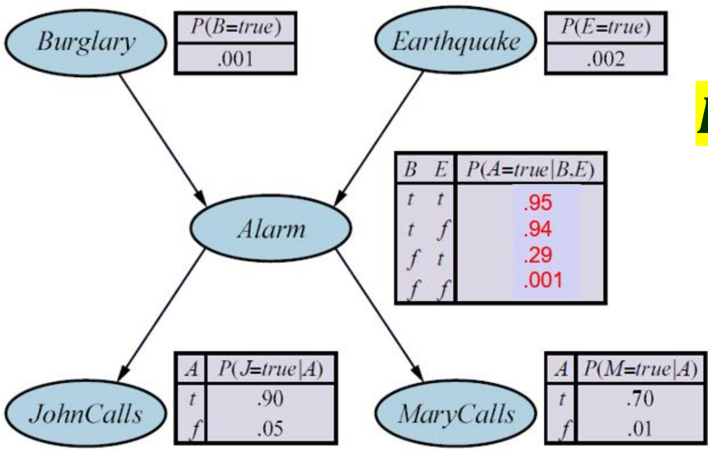


$\boldsymbol{P}(\mathit{Burglary} \mid \mathit{JohnCalls} = \mathit{true}, \mathit{MaryCalls} = \mathit{true})$?

Hidden variables: $Y = \{\mathit{Earthquake}, \mathit{Alarm}\}$

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{P}(B, j, m) = \alpha \sum_{E \in \{e, \neg e\}} \sum_{A \in \{a, \neg a\}} \boldsymbol{P}(B, j, m, E, A)$$

Burglary Example (revisited)



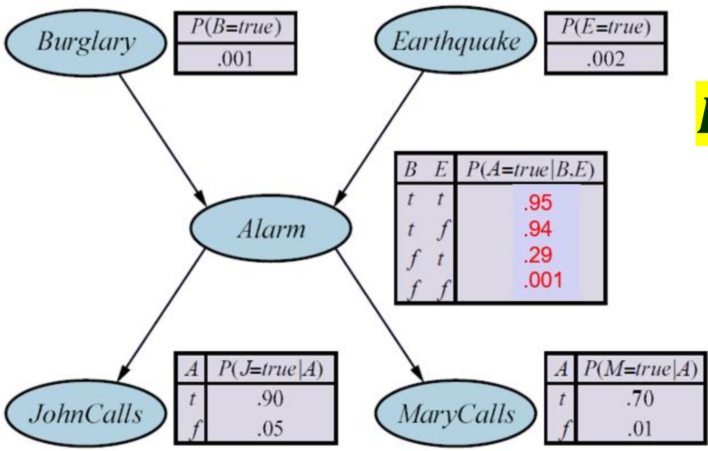
$\mathbf{P}(Burglary \mid JohnCalls = true, MaryCalls = true)$?

Hidden variables: $Y = \{Earthquake, Alarm\}$

$$\mathbf{P}(B \mid j, m) = \alpha \mathbf{P}(B, j, m) = \alpha \sum_{E \in \{e, \neg e\}} \sum_{A \in \{a, \neg a\}} \mathbf{P}(B, j, m, E, A)$$

// the textbook uses e, a instead of E, A in the summation.
// this is incorrect because e is a constant value standing
// for $Earthquake = true$. the random variable E can take
// on both e and $\neg e$ needed for the summation. you may
// also call this an "abuse" of the notation e and a.

Burglary Example (revisited)



$\boldsymbol{P}(Burglary \mid JohnCalls = true, MaryCalls = true)$?

Hidden variables: $Y = \{Earthquake, Alarm\}$

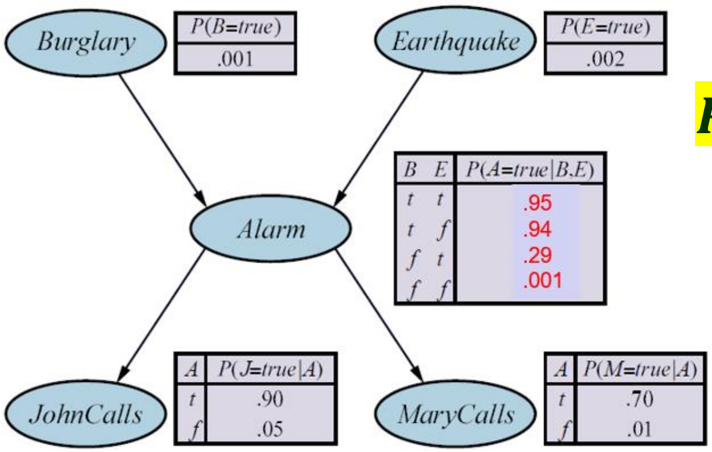
$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{P}(B, j, m) = \alpha \underbrace{\sum_{E \in \{e, \neg e\}} \sum_{A \in \{a, \neg a\}}} \boldsymbol{P}(B, j, m, E, A)$$

evaluated for b (*Burglary* = *true*) based on

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(X_i))$$

// the textbook uses e, a instead of E, A in the summation.
// this is incorrect because e is a constant value standing
// for *Earthquake* = *true*. the random variable E can take
// on both e and $\neg e$ needed for the summation. you may
// also call this an "abuse" of the notation e and a.

Burglary Example (revisited)



$\mathbf{P}(Burglary \mid JohnCalls = true, MaryCalls = true)$?

Hidden variables: $Y = \{Earthquake, Alarm\}$

$$\mathbf{P}(B \mid j, m) = \alpha \mathbf{P}(B, j, m) = \alpha \sum_{E \in \{e, \neg e\}} \sum_{A \in \{a, \neg a\}} \mathbf{P}(B, j, m, E, A)$$

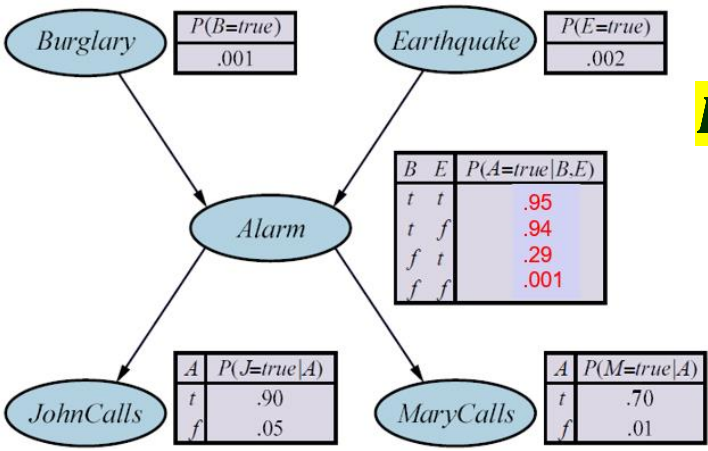
// the textbook uses e, a instead of E, A in the summation.
// this is incorrect because e is a constant value standing
// for *Earthquake* = *true*. the random variable E can take
// on both e and $\neg e$ needed for the summation. you may
// also call this an "abuse" of the notation e and a.

evaluated for b (*Burglary* = *true*) based on

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(X_i))$$

$$P(b \mid j, m) = \alpha \sum_E \sum_A P(b) P(E) P(A \mid b, E) P(j \mid A) P(m \mid A)$$

Burglary Example (revisited)



$\boldsymbol{P}(Burglary \mid JohnCalls = true, MaryCalls = true)$?

Hidden variables: $Y = \{Earthquake, Alarm\}$

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{P}(B, j, m) = \alpha \sum_{E \in \{e, \neg e\}} \sum_{A \in \{a, \neg a\}} \boldsymbol{P}(B, j, m, E, A)$$

// the textbook uses e, a instead of E, A in the summation.
// this is incorrect because e is a constant value standing
// for *Earthquake* = *true*. the random variable E can take
// on both e and $\neg e$ needed for the summation. you may
// also call this an “abuse” of the notation e and a.

evaluated for b (*Burglary* = *true*) based on

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(X_i))$$

$$P(b \mid j, m) = \alpha \sum_{E} \sum_{A} P(b) P(E) P(A \mid b, E) P(j \mid A) P(m \mid A)$$

In the general case with n variables, there are 2^n summands, each as a product requires $O(n)$ computation time.

Expression Tree

Take advantage of the nested structure to move summations inwards as far as possible.

$$P(b \mid j, m) = \alpha \sum_{E} \sum_{A} P(b)P(E)P(A \mid b, E)P(j \mid A)P(m \mid A)$$

Expression Tree

Take advantage of the nested structure to move summations inwards as far as possible.

$$P(b \mid j, m) = \alpha \sum_E \sum_A P(b) P(E) P(A \mid b, E) P(j \mid A) P(m \mid A)$$

Expression Tree

Take advantage of the nested structure to move summations inwards as far as possible.

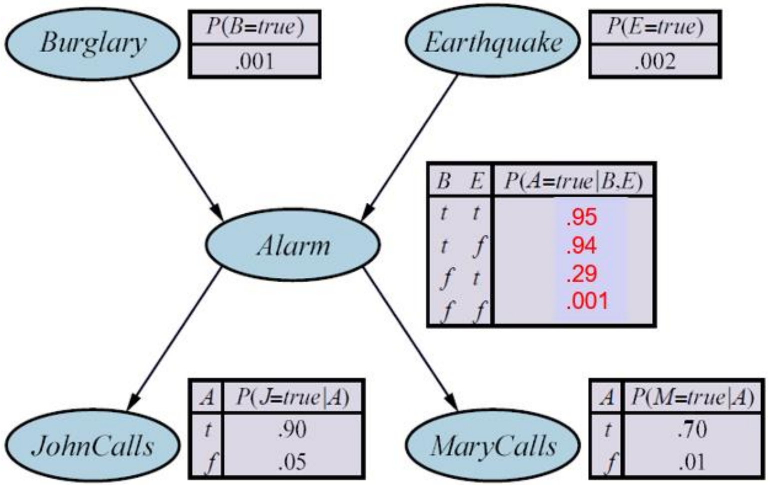
$$P(b \mid j, m) = \alpha \sum_E \sum_A P(b) P(E) P(A \mid b, E) P(j \mid A) P(m \mid A)$$

Expression Tree

$$P(b \mid j,m) = \alpha P(b) \sum_{E} P(E) \sum_{A} P(A \mid b,E) P(j \mid A) P(m \mid A)$$

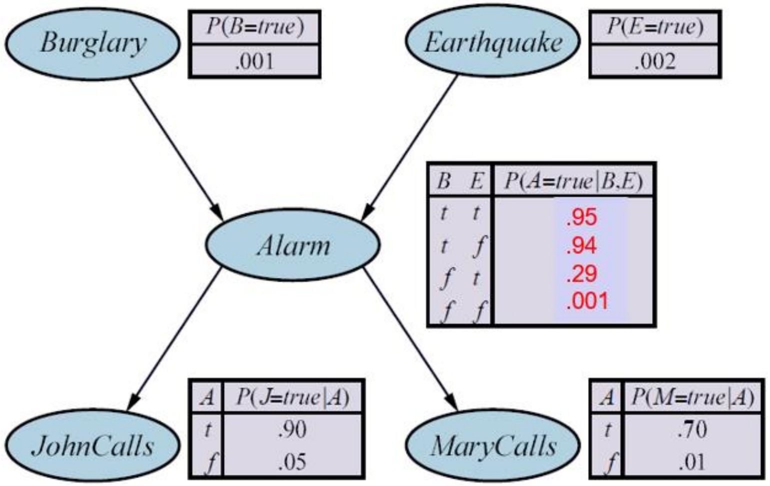
Expression Tree

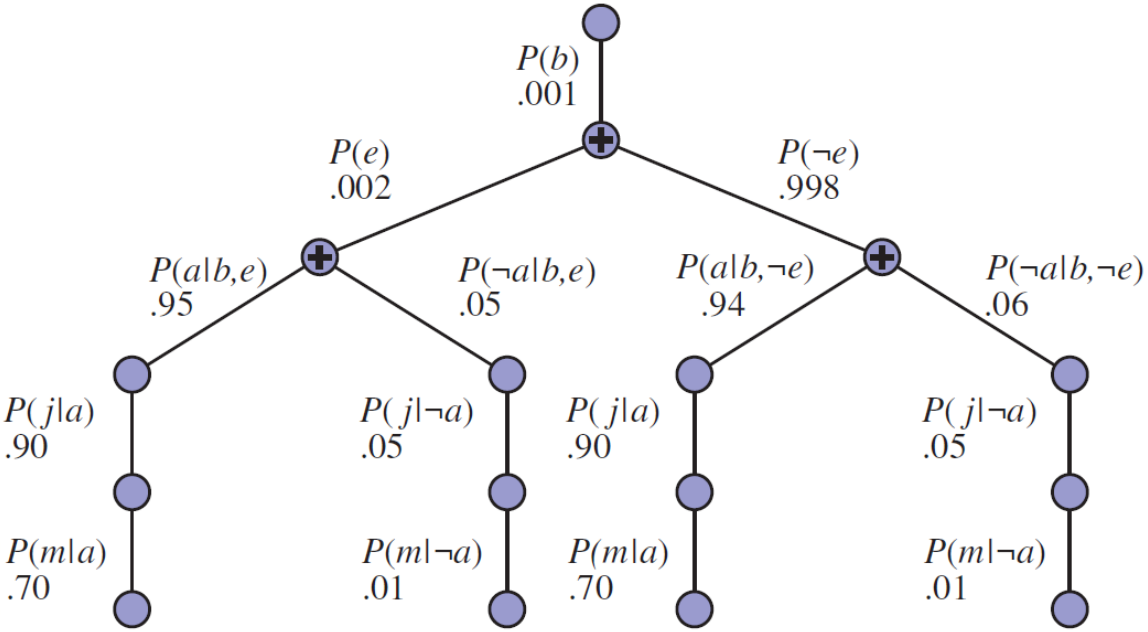
$$P(b \mid j, m) = \alpha P(b) \sum_{E} P(E) \sum_{A} P(A \mid b, E) P(j \mid A) P(m \mid A)$$



Expression Tree

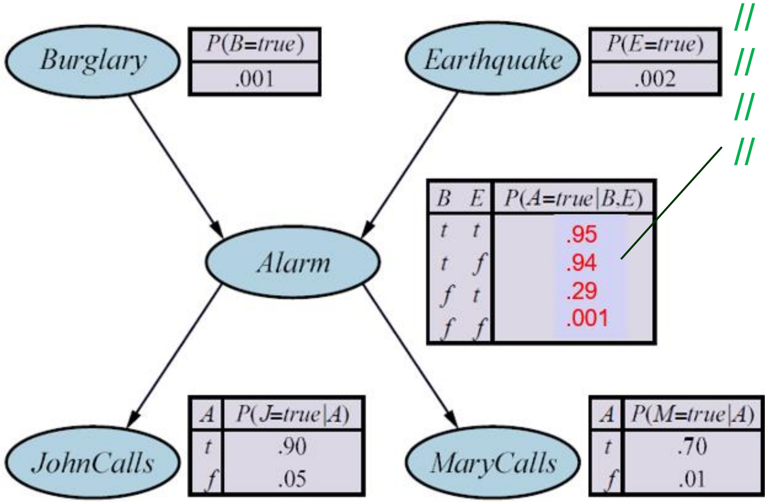
$$P(b \mid j,m) = \alpha P(b) \sum_E P(E) \sum_A P(A \mid b,E) P(j \mid A) P(m \mid A)$$



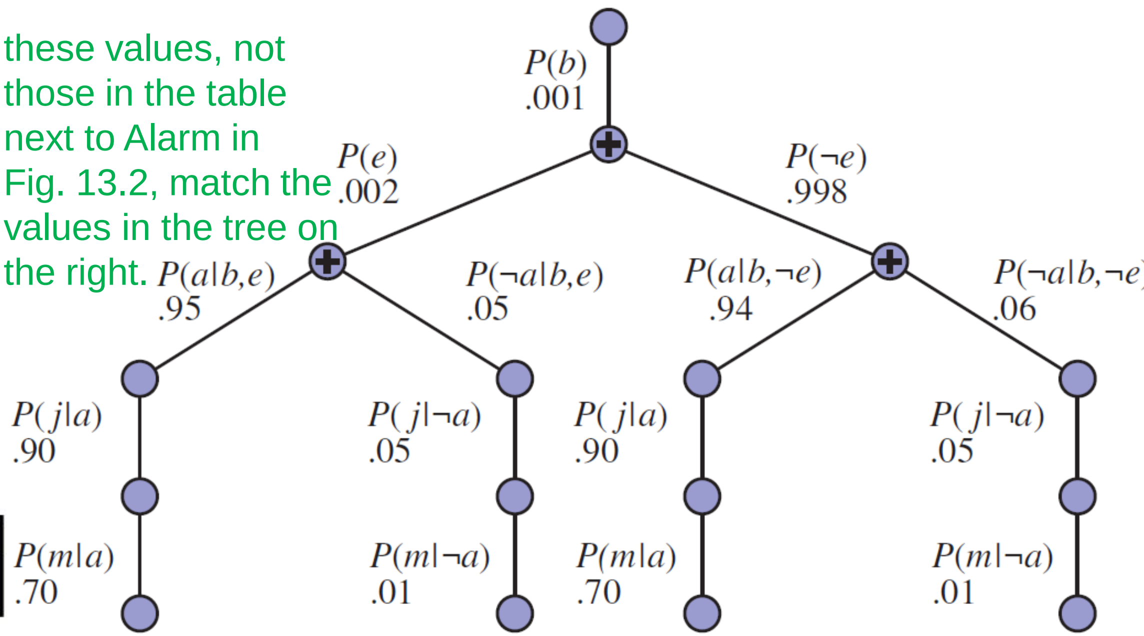


Expression Tree

$$P(b \mid j,m) = \alpha P(b) \sum_{E} P(E) \sum_{A} P(A \mid b,E) P(j \mid A) P(m \mid A)$$



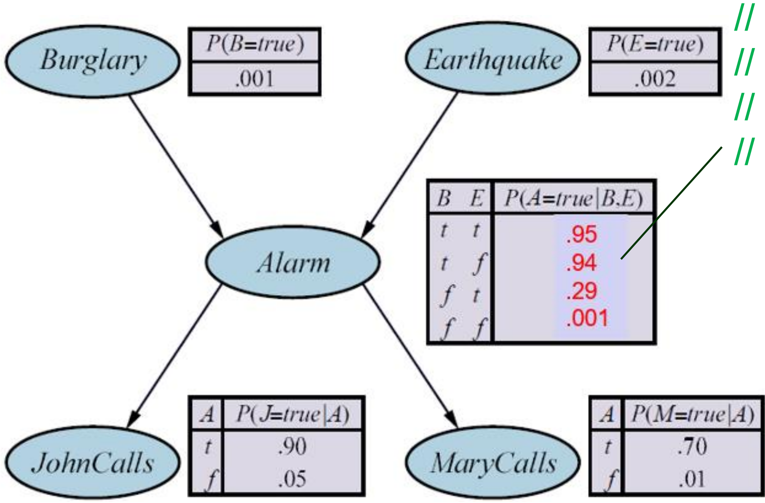
// these values, not
// those in the table
// next to Alarm in
// Fig. 13.2, match the
// values in the tree on
// the right.

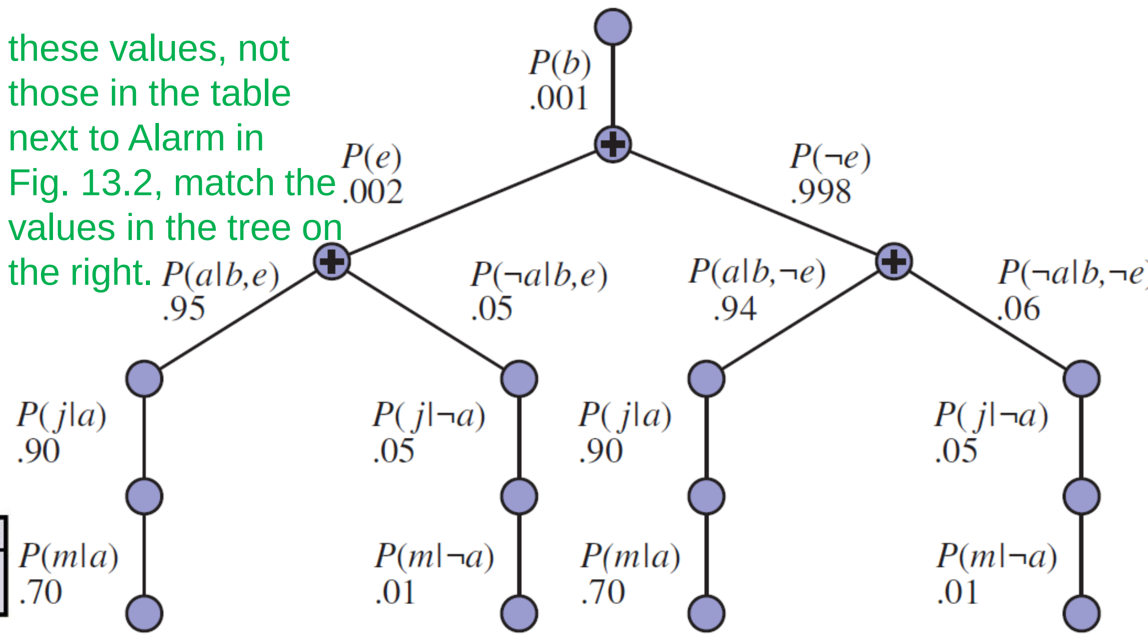


Expression Tree

$$P(b \mid j,m) = \alpha P(b) \sum_{E} P(E) \sum_{A} P(A \mid b,E) P(j \mid A) P(m \mid A)$$

// these values, not
// those in the table
// next to Alarm in
// Fig. 13.2, match the
// values in the tree on
// the right.

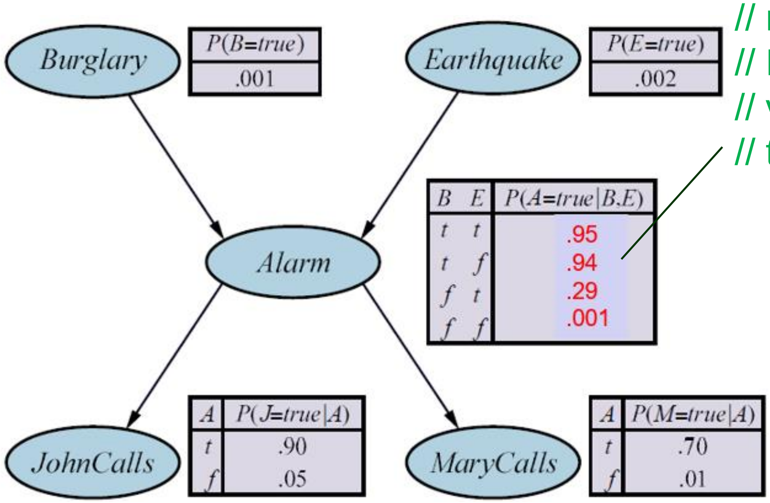




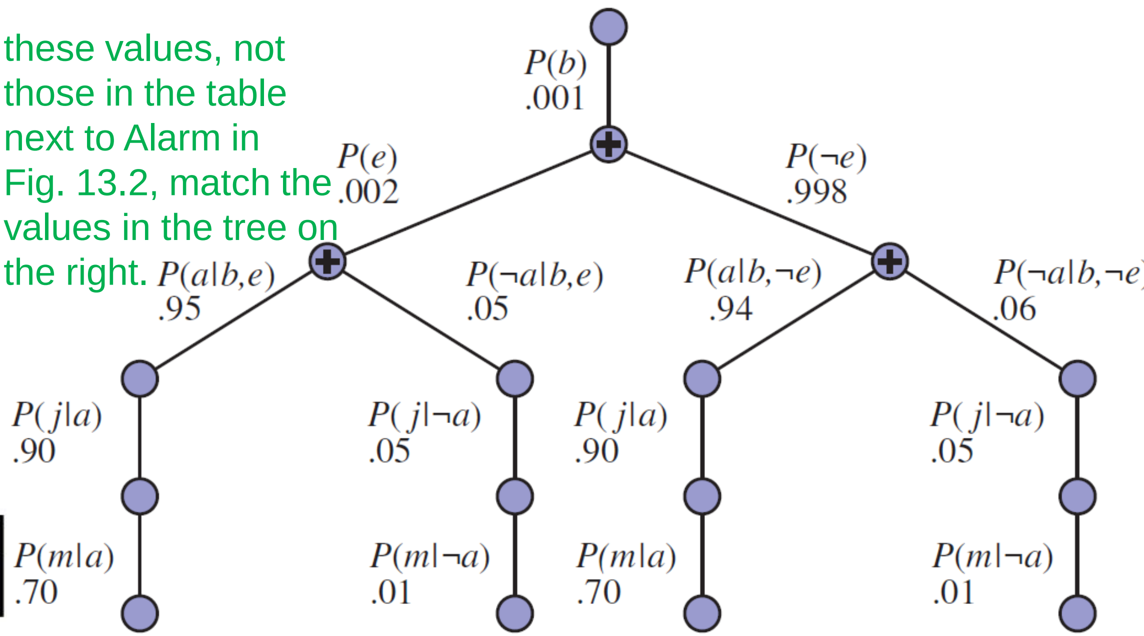
- Multiply values along each path.
- Sum at the “+” nodes.

Expression Tree

$$P(b \mid j,m) = \alpha P(b) \sum_{E} P(E) \sum_{A} P(A \mid b,E) P(j \mid A) P(m \mid A)$$



// these values, not
// those in the table
// next to Alarm in
// Fig. 13.2, match the
// values in the tree on
// the right.



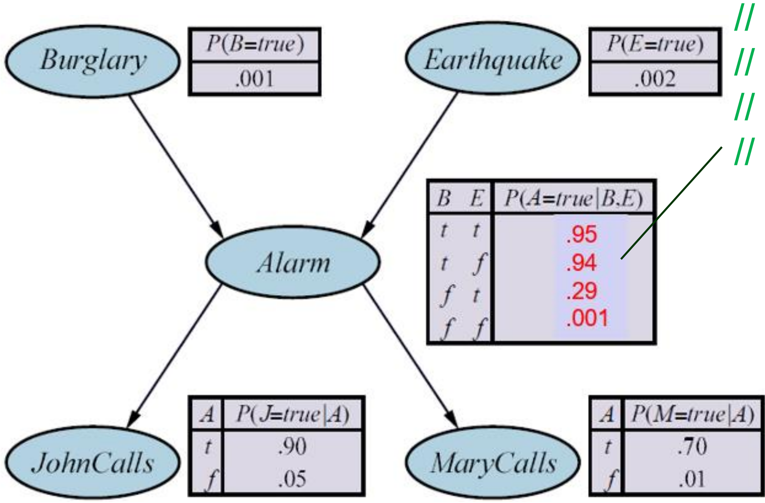
- Multiply values along each path.
- Sum at the “+” nodes.

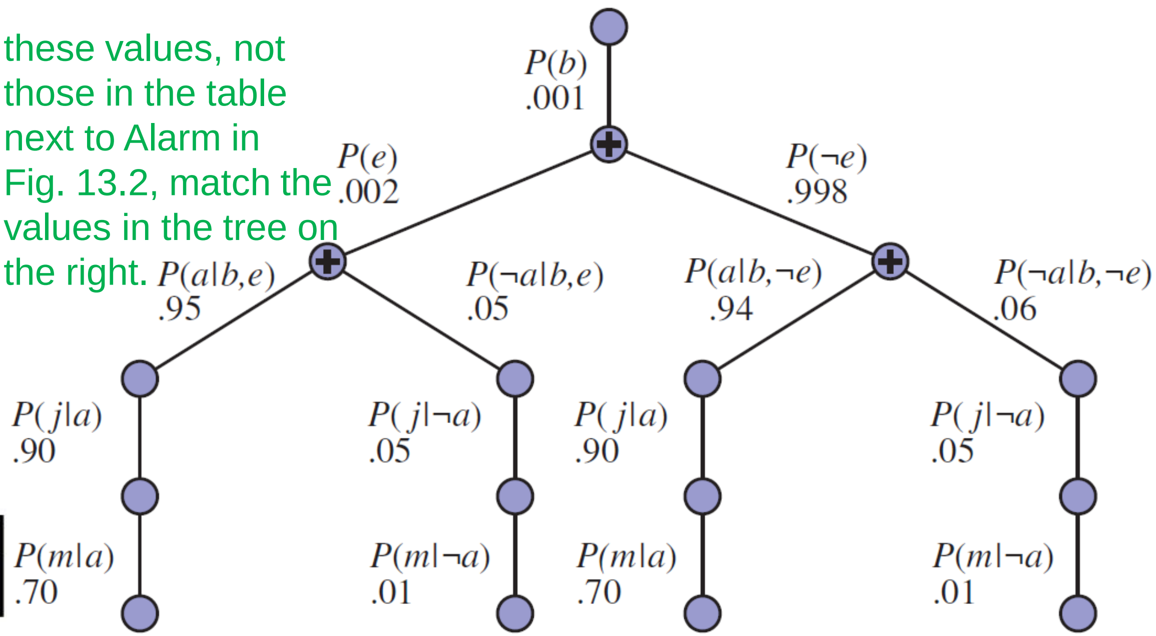
$$P(b \mid j,m) = \alpha \times 0.00059224$$

Expression Tree

$$P(b \mid j, m) = \alpha P(b) \sum_E P(E) \sum_A P(A \mid b, E) P(j \mid A) P(m \mid A)$$

// these values, not
// those in the table
// next to Alarm in
// Fig. 13.2, match the
// values in the tree on
// the right.





- Multiply values along each path.
- Sum at the “+” nodes.

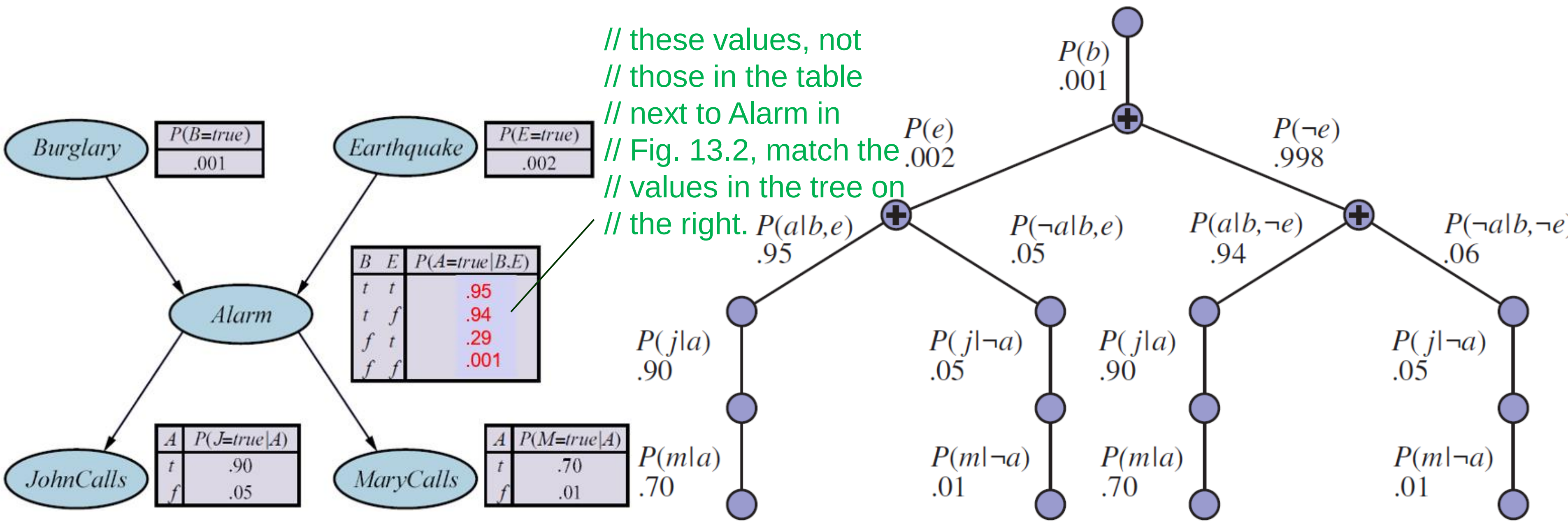
$$P(b \mid j, m) = \alpha \times 0.00059224$$

$$P(\neg b \mid j, m) = \alpha \times 0.0014919$$

(computed using a tree of the same structure but with b replaced by $\neg b$ and probabilities changed)

Expression Tree

$$P(b \mid j,m) = \alpha P(b) \sum_{E} P(E) \sum_{A} P(A \mid b,E) P(j \mid A) P(m \mid A)$$



- Multiply values along each path.
- Sum at the "+" nodes.

$$P(b \mid j,m) = \alpha \times 0.00059224$$
$$P(\neg b \mid j,m) = \alpha \times 0.0014919$$

$\Longrightarrow$ $\mathbf{P}(B \mid j,m) = \alpha\langle 0.00059224, 0.0014919\rangle \approx \langle 0.284, 0.716\rangle$

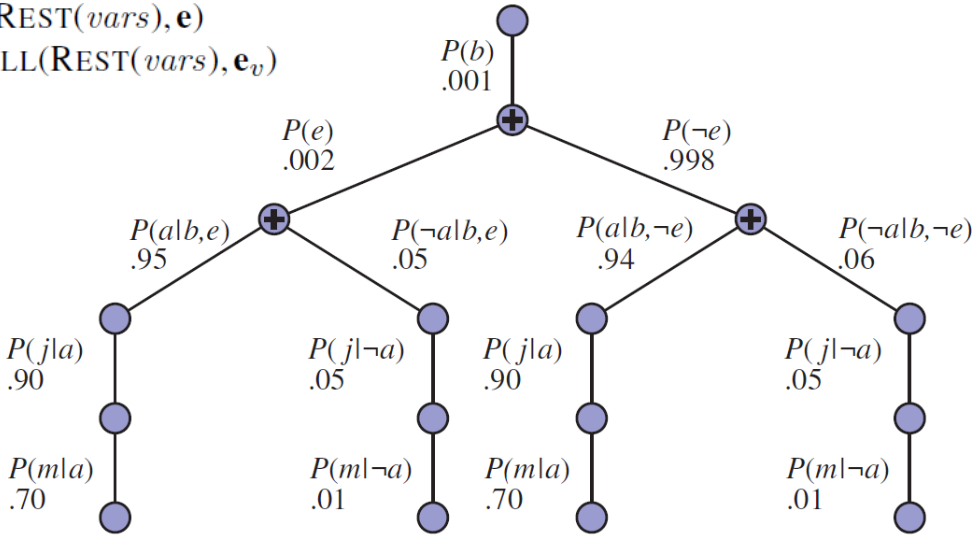
(computed using a tree of the same structure but with b replaced by $\neg b$ and probabilities changed)

Enumeration Algorithm

function ENUMERATION-ASK(X, $\mathbf{e}$, bn) **returns** a distribution over X
  inputs: X, the query variable
    $\mathbf{e}$, observed values for variables $\mathbf{E}$
    bn, a Bayes net with variables $vars$

  $\mathbf{Q}(X) \leftarrow$ a distribution over X, initially empty
  for each value x_i of X **do**
    $\mathbf{Q}(x_i) \leftarrow$ ENUMERATE-ALL($vars$, $\mathbf{e}_{x_i}$)
      where $\mathbf{e}_{x_i}$ is $\mathbf{e}$ extended with $X = x_i$
  return NORMALIZE($\mathbf{Q}(X)$)

function ENUMERATE-ALL($vars$, $\mathbf{e}$) **returns** a real number
  if EMPTY?($vars$) **then return** 1.0
  $V \leftarrow$ FIRST($vars$)
  if V is an evidence variable with value v in $\mathbf{e}$
    then return $P(v \mid parents(V)) \times$ ENUMERATE-ALL(REST($vars$), $\mathbf{e}$)
    else return $\sum_v P(v \mid parents(V)) \times$ ENUMERATE-ALL(REST($vars$), $\mathbf{e}_v$)
      where $\mathbf{e}_v$ is $\mathbf{e}$ extended with $V = v$



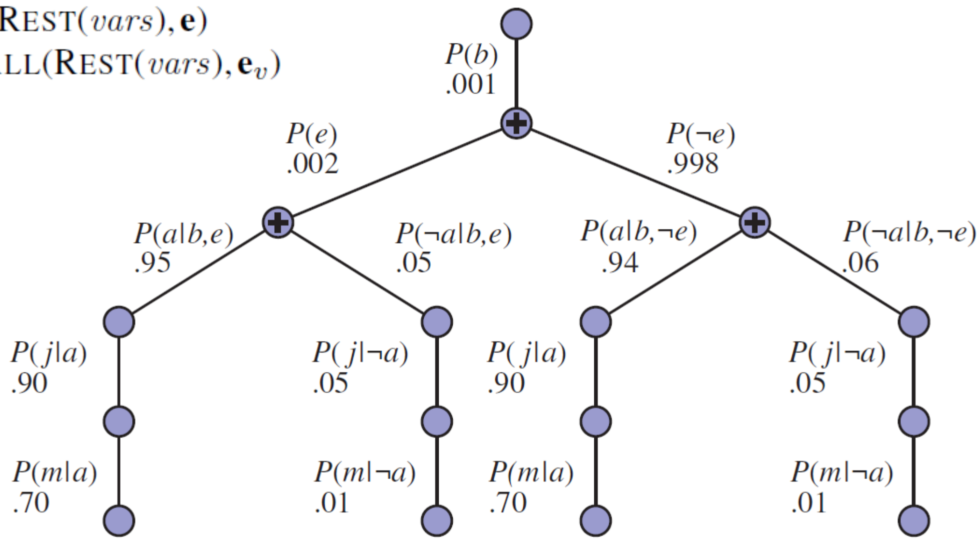
Enumeration Algorithm

function ENUMERATION-ASK(X, **e**, bn) **returns** a distribution over X
  inputs: X, the query variable
    e, observed values for variables **E**
    bn, a Bayes net with variables $vars$

  $\mathbf{Q}(X) \leftarrow$ a distribution over X, initially empty
  for each value x_i of X **do**
    $\mathbf{Q}(x_i) \leftarrow$ ENUMERATE-ALL($vars$, $\mathbf{e}_{x_i}$)
      where $\mathbf{e}_{x_i}$ is **e** extended with $X = x_i$
  return NORMALIZE($\mathbf{Q}(X)$)

function ENUMERATE-ALL($vars$, **e**) **returns** a real number
  if EMPTY?($vars$) **then return** 1.0
  $V \leftarrow$ FIRST($vars$)
  if V is an evidence variable with value v in **e**
    then return $P(v \mid parents(V)) \times$ ENUMERATE-ALL(REST($vars$), **e**)
    else return $\sum_v P(v \mid parents(V)) \times$ ENUMERATE-ALL(REST($vars$), $\mathbf{e}_v$)
      where $\mathbf{e}_v$ is **e** extended with $V = v$

- Depth-first recursion



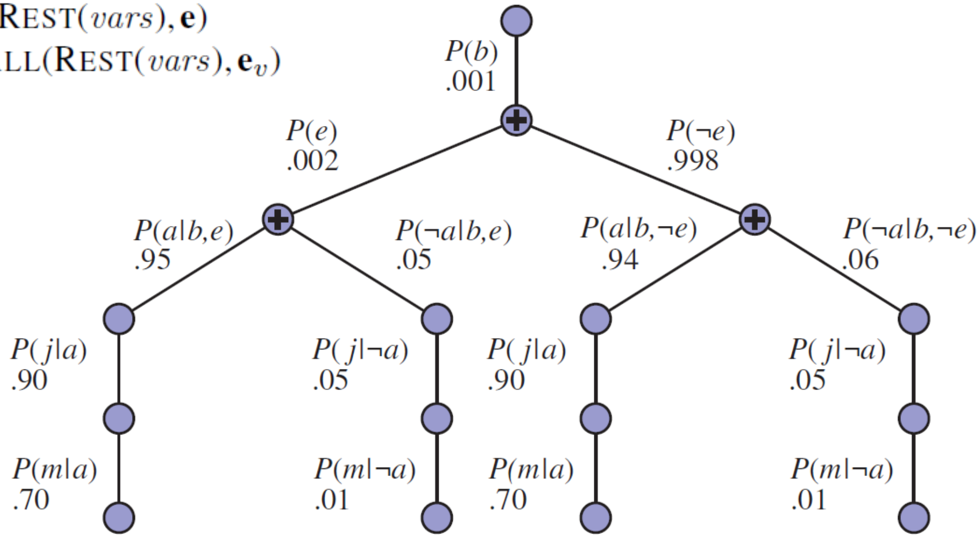
Enumeration Algorithm

function ENUMERATION-ASK(X, $\mathbf{e}$, bn) **returns** a distribution over X
  inputs: X, the query variable
    $\mathbf{e}$, observed values for variables $\mathbf{E}$
    bn, a Bayes net with variables $vars$

  $\mathbf{Q}(X) \leftarrow$ a distribution over X, initially empty
  for each value x_i of X **do**
    $\mathbf{Q}(x_i) \leftarrow$ ENUMERATE-ALL($vars$, $\mathbf{e}_{x_i}$)
      where $\mathbf{e}_{x_i}$ is $\mathbf{e}$ extended with $X = x_i$
  return NORMALIZE($\mathbf{Q}(X)$)

function ENUMERATE-ALL($vars$, $\mathbf{e}$) **returns** a real number
  if EMPTY?($vars$) **then return** 1.0
  $V \leftarrow$ FIRST($vars$)
  if V is an evidence variable with value v in $\mathbf{e}$
    then return $P(v \mid parents(V)) \times$ ENUMERATE-ALL(REST($vars$), $\mathbf{e}$)
    else return $\sum_v P(v \mid parents(V)) \times$ ENUMERATE-ALL(REST($vars$), $\mathbf{e}_v$)
      where $\mathbf{e}_v$ is $\mathbf{e}$ extended with $V = v$

- Depth-first recursion
- $O(2^n)$ time for n variables



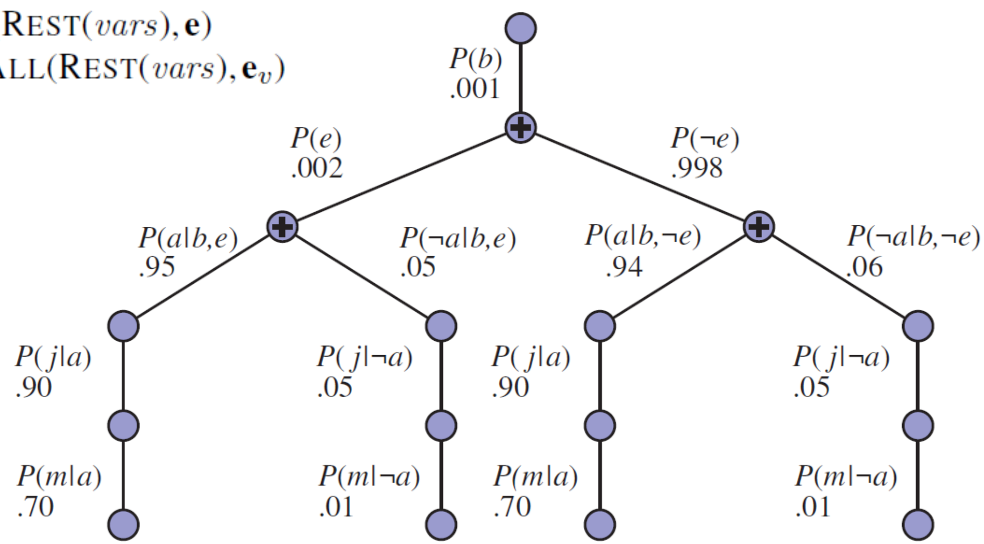
Enumeration Algorithm

function ENUMERATION-ASK(X, $\mathbf{e}$, bn) **returns** a distribution over X
  inputs: X, the query variable
    $\mathbf{e}$, observed values for variables $\mathbf{E}$
    bn, a Bayes net with variables $vars$

  $\mathbf{Q}(X) \leftarrow$ a distribution over X, initially empty
  for each value x_i of X **do**
    $\mathbf{Q}(x_i) \leftarrow$ ENUMERATE-ALL($vars$, $\mathbf{e}_{x_i}$)
      where $\mathbf{e}_{x_i}$ is $\mathbf{e}$ extended with $X = x_i$
  return NORMALIZE($\mathbf{Q}(X)$)

function ENUMERATE-ALL($vars$, $\mathbf{e}$) **returns** a real number
  if EMPTY?($vars$) **then return** 1.0
  $V \leftarrow$ FIRST($vars$)
  if V is an evidence variable with value v in $\mathbf{e}$
    then return $P(v \mid parents(V)) \times$ ENUMERATE-ALL(REST($vars$), $\mathbf{e}$)
    else return $\sum_v P(v \mid parents(V)) \times$ ENUMERATE-ALL(REST($vars$), $\mathbf{e}_v$)
      where $\mathbf{e}_v$ is $\mathbf{e}$ extended with $V = v$

- Depth-first recursion
- $O(2^n)$ time for n variables
- Repeated evaluations of the same subexpressions



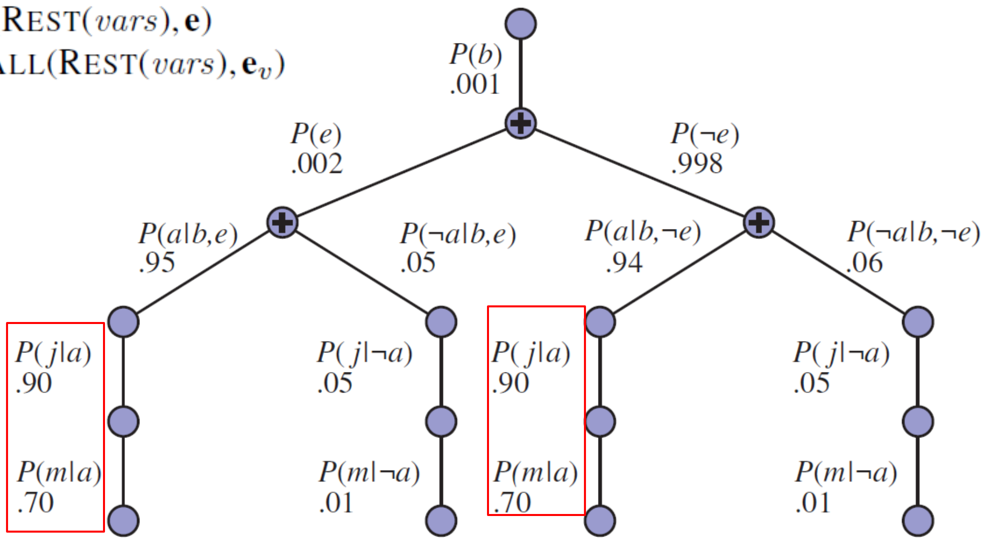
Enumeration Algorithm

function ENUMERATION-ASK(X, **e**, bn) **returns** a distribution over X
  inputs: X, the query variable
      e, observed values for variables **E**
      bn, a Bayes net with variables $vars$

  $\mathbf{Q}(X) \leftarrow$ a distribution over X, initially empty
  for each value x_i of X **do**
    $\mathbf{Q}(x_i) \leftarrow$ ENUMERATE-ALL($vars$, $\mathbf{e}_{x_i}$)
      where $\mathbf{e}_{x_i}$ is **e** extended with $X = x_i$
  return NORMALIZE($\mathbf{Q}(X)$)

function ENUMERATE-ALL($vars$, **e**) **returns** a real number
  if EMPTY?($vars$) **then return** 1.0
  $V \leftarrow$ FIRST($vars$)
  if V is an evidence variable with value v in **e**
    then return $P(v \mid parents(V)) \times$ ENUMERATE-ALL(REST($vars$), **e**)
    else return $\sum_v P(v \mid parents(V)) \times$ ENUMERATE-ALL(REST($vars$), $\mathbf{e}_v$)
      where $\mathbf{e}_v$ is **e** extended with $V = v$

- Depth-first recursion
- $O(2^n)$ time for n variables
- Repeated evaluations of the same subexpressions



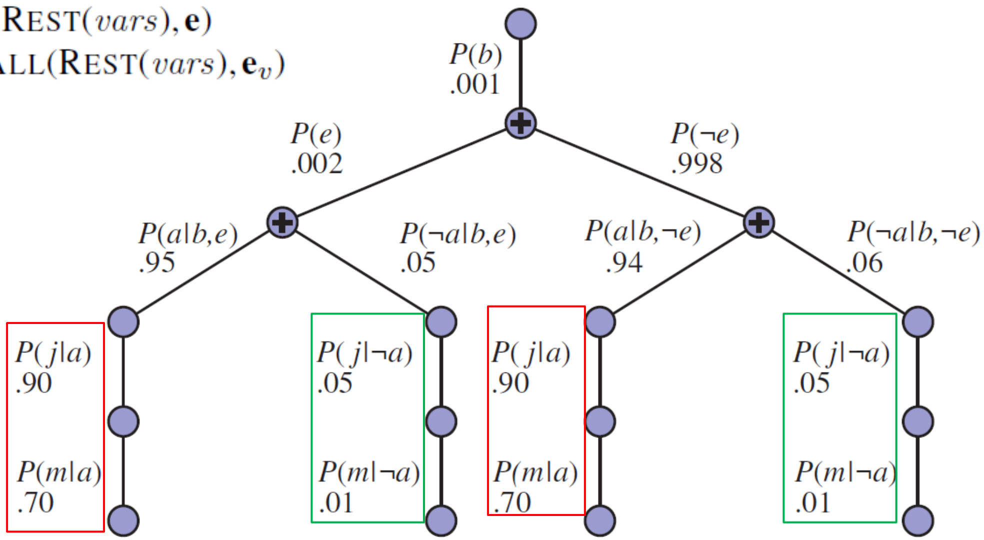
Enumeration Algorithm

function ENUMERATION-ASK(X, $\mathbf{e}$, bn) **returns** a distribution over X
  inputs: X, the query variable
    $\mathbf{e}$, observed values for variables $\mathbf{E}$
    bn, a Bayes net with variables $vars$

  $\mathbf{Q}(X) \leftarrow$ a distribution over X, initially empty
  for each value x_i of X **do**
    $\mathbf{Q}(x_i) \leftarrow$ ENUMERATE-ALL($vars$, $\mathbf{e}_{x_i}$)
      where $\mathbf{e}_{x_i}$ is $\mathbf{e}$ extended with $X = x_i$
  return NORMALIZE($\mathbf{Q}(X)$)

function ENUMERATE-ALL($vars$, $\mathbf{e}$) **returns** a real number
  if EMPTY?($vars$) **then return** 1.0
  $V \leftarrow$ FIRST($vars$)
  if V is an evidence variable with value v in $\mathbf{e}$
    then return $P(v \mid parents(V)) \times$ ENUMERATE-ALL(REST($vars$), $\mathbf{e}$)
    else return $\sum_v P(v \mid parents(V)) \times$ ENUMERATE-ALL(REST($vars$), $\mathbf{e}_v$)
      where $\mathbf{e}_v$ is $\mathbf{e}$ extended with $V = v$

- Depth-first recursion
- $O(2^n)$ time for n variables
- Repeated evaluations of the same subexpressions



III. Variable Elimination

Idea Do the calculation once and save the results for later use (like dynamic programming).

III. Variable Elimination

Idea Do the calculation once and save the results for later use (like dynamic programming).

- Evaluate an expression from right to left (i.e., bottom up in the expression tree).

$$P(b \mid j, m) = \alpha P(b) \sum_{E} P(E) \sum_{A} P(A \mid b, E) P(j \mid A) P(m \mid A)$$

III. Variable Elimination

Idea Do the calculation once and save the results for later use (like dynamic programming).

- Evaluate an expression from right to left (i.e., bottom up in the expression tree).

$$P(b \mid j, m) = \alpha P(b) \sum_E P(E) \sum_A P(A \mid b, E) P(j \mid A) P(m \mid A)$$

III. Variable Elimination

Idea Do the calculation once and save the results for later use (like dynamic programming).

- Evaluate an expression from right to left (i.e., bottom up in the expression tree).

$$P(b \mid j, m) = \alpha P(b) \sum_E P(E) \sum_A P(A \mid b, E) P(j \mid A) P(m \mid A)$$

$$\mathbf{P}(B \mid j, m) = \alpha \mathbf{P}(B) \sum_E P(E) \sum_A P(A \mid B, E) P(j \mid A) P(m \mid A)$$

III. Variable Elimination

Idea Do the calculation once and save the results for later use (like dynamic programming).

- Evaluate an expression from right to left (i.e., bottom up in the expression tree).

$$P(b \mid j, m) = \alpha P(b) \sum_E P(E) \sum_A P(A \mid b, E) P(j \mid A) P(m \mid A)$$

$$\mathbf{P}(B \mid j, m) = \alpha \underbrace{\mathbf{P}(B)}_{} \sum_E \underbrace{P(E)}_{} \sum_A \underbrace{P(A \mid B, E)}_{} \underbrace{P(j \mid A)}_{} \underbrace{P(m \mid A)}_{}$$

factors: $\boldsymbol{f}_1(B)$ $\boldsymbol{f}_2(E)$ $\boldsymbol{f}_3(A, B, E)$ $\boldsymbol{f}_4(A)$ $\boldsymbol{f}_5(A)$

III. Variable Elimination

Idea Do the calculation once and save the results for later use (like dynamic programming).

- Evaluate an expression from right to left (i.e., bottom up in the expression tree).

$$P(b \mid j, m) = \alpha P(b) \sum_E P(E) \sum_A P(A \mid b, E) P(j \mid A) P(m \mid A)$$

$$\mathbf{P}(B \mid j, m) = \alpha \underbrace{\mathbf{P}(B)}_{} \sum_E \underbrace{P(E)}_{} \sum_A \underbrace{P(A \mid B, E)}_{} \underbrace{P(j \mid A)}_{} \underbrace{P(m \mid A)}_{}$$

factors:	$\boldsymbol{f}_1(B)$	$\boldsymbol{f}_2(E)$	$\boldsymbol{f}_3(A, B, E)$	$\boldsymbol{f}_4(A)$	$\boldsymbol{f}_5(A)$
dimensions:	2×1	2×1	$2 \times 2 \times 2$	2×1	2×1

Evaluation Example

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{f}_{\mathbf{1}}(B) \times \sum_{E} \boldsymbol{f}_{\mathbf{2}}(E) \times \sum_{A} \boldsymbol{f}_{\mathbf{3}}(A, B, E) \times \boldsymbol{f}_{\mathbf{4}}(A) \times \boldsymbol{f}_{\mathbf{5}}(A)$$

Evaluation Example

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{f_1}(B) \times \sum_E \boldsymbol{f_2}(E) \times \sum_A \boldsymbol{f_3}(A, B, E) \times \boldsymbol{f_4}(A) \times \boldsymbol{f_5}(A)$$

- Sum out A from the product of $\boldsymbol{f}_3, \boldsymbol{f}_4, \boldsymbol{f}_5$.

$$\boldsymbol{f}_6(B, E) = \sum_{A \in \{a, \neg a\}} \boldsymbol{f_3}(A, B, E) \times \boldsymbol{f_4}(A) \times \boldsymbol{f_5}(A)$$

2×2

Evaluation Example

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{f_1}(B) \times \sum_E \boldsymbol{f_2}(E) \times \sum_A \boldsymbol{f_3}(A, B, E) \times \boldsymbol{f_4}(A) \times \boldsymbol{f_5}(A)$$

- Sum out A from the product of $\boldsymbol{f}_3, \boldsymbol{f}_4, \boldsymbol{f}_5$.

$$\underset{2 \times 2}{\boldsymbol{f}_6(B, E)} = \sum_{A \in \{a, \neg a\}} \boldsymbol{f_3}(A, B, E) \times \boldsymbol{f_4}(A) \times \boldsymbol{f_5}(A)$$

$$= \left(\underset{2 \times 2}{\boldsymbol{f}_3(a, B, E)} \times P(a) \times P(a)\right) + \left(\underset{2 \times 2}{\boldsymbol{f}_3(\neg a, B, E)} \times P(\neg a) \times P(\neg a)\right)$$

Evaluation Example

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{f_1}(B) \times \sum_E \boldsymbol{f_2}(E) \times \sum_A \boldsymbol{f_3}(A, B, E) \times \boldsymbol{f_4}(A) \times \boldsymbol{f_5}(A)$$

- Sum out A from the product of $\boldsymbol{f}_3, \boldsymbol{f}_4, \boldsymbol{f}_5$.

$$\underset{2 \times 2}{\boldsymbol{f}_6(B, E)} = \sum_{A \in \{a, \neg a\}} \boldsymbol{f_3}(A, B, E) \times \boldsymbol{f_4}(A) \times \boldsymbol{f_5}(A)$$

$$= \left(\underset{2 \times 2}{\boldsymbol{f}_3(a, B, E)} \times P(a) \times P(a)\right) + \left(\underset{2 \times 2}{\boldsymbol{f}_3(\neg a, B, E)} \times P(\neg a) \times P(\neg a)\right)$$

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{f_1}(B) \times \sum_E \boldsymbol{f}_2(E) \times \boldsymbol{f}_6(B, E)$$

Evaluation Example

$$\mathbf{P}(B \mid j, m) = \alpha \boldsymbol{f}_{\mathbf{1}}(B) \times \sum_{E} \boldsymbol{f}_{\mathbf{2}}(E) \times \sum_{A} \boldsymbol{f}_{\mathbf{3}}(A, B, E) \times \boldsymbol{f}_{\mathbf{4}}(A) \times \boldsymbol{f}_{\mathbf{5}}(A)$$

- Sum out A from the product of $\boldsymbol{f}_3, \boldsymbol{f}_4, \boldsymbol{f}_5$.

$$\underset{2\times 2}{\boldsymbol{f}_6(B,E)} = \sum_{A \in \{a, \neg a\}} \boldsymbol{f}_{\mathbf{3}}(A, B, E) \times \boldsymbol{f}_{\mathbf{4}}(A) \times \boldsymbol{f}_{\mathbf{5}}(A)$$

$$= \Big(\underset{2\times 2}{\boldsymbol{f}_3(a, B, E)} \times P(a) \times P(a)\Big) + \Big(\underset{2\times 2}{\boldsymbol{f}_3(\neg a, B, E)} \times P(\neg a) \times P(\neg a)\Big)$$

$$\mathbf{P}(B \mid j, m) = \alpha \boldsymbol{f}_{\mathbf{1}}(B) \times \sum_{E} \boldsymbol{f}_2(E) \times \boldsymbol{f}_6(B, E)$$

- Sum out E from the product of $\boldsymbol{f}_2$ and $\boldsymbol{f}_6$.

$$\underset{1\times 2}{\boldsymbol{f}_7(B)} = \sum_{E \in \{e, \neg e\}} \boldsymbol{f}_{\mathbf{2}}(E) \times \boldsymbol{f}_{\mathbf{6}}(B, E) = P(e) \times \underset{1\times 2}{\boldsymbol{f}_6(B, e)} + P(\neg e) \times \underset{1\times 2}{\boldsymbol{f}_6(B, \neg e)}$$

Evaluation Example

$$\mathbf{P}(B \mid j, m) = \alpha \boldsymbol{f}_{\mathbf{1}}(B) \times \sum_{E} \boldsymbol{f}_{\mathbf{2}}(E) \times \sum_{A} \boldsymbol{f}_{\mathbf{3}}(A, B, E) \times \boldsymbol{f}_{\mathbf{4}}(A) \times \boldsymbol{f}_{\mathbf{5}}(A)$$

- Sum out A from the product of $\boldsymbol{f}_3, \boldsymbol{f}_4, \boldsymbol{f}_5$.

$$\underset{2 \times 2}{\boldsymbol{f}_6(B, E)} = \sum_{A \in \{a, \neg a\}} \boldsymbol{f}_{\mathbf{3}}(A, B, E) \times \boldsymbol{f}_{\mathbf{4}}(A) \times \boldsymbol{f}_{\mathbf{5}}(A)$$

$$= \Big(\underset{2 \times 2}{\boldsymbol{f}_3(a, B, E)} \times P(a) \times P(a)\Big) + \Big(\underset{2 \times 2}{\boldsymbol{f}_3(\neg a, B, E)} \times P(\neg a) \times P(\neg a)\Big)$$

$$\mathbf{P}(B \mid j, m) = \alpha \boldsymbol{f}_{\mathbf{1}}(B) \times \sum_{E} \boldsymbol{f}_2(E) \times \boldsymbol{f}_6(B, E)$$

- Sum out E from the product of $\boldsymbol{f}_2$ and $\boldsymbol{f}_6$.

$$\underset{1 \times 2}{\boldsymbol{f}_7(B)} = \sum_{E \in \{e, \neg e\}} \boldsymbol{f}_{\mathbf{2}}(E) \times \boldsymbol{f}_{\mathbf{6}}(B, E) = P(e) \times \underset{1 \times 2}{\boldsymbol{f}_6(B, e)} + P(\neg e) \times \underset{1 \times 2}{\boldsymbol{f}_6(B, \neg e)}$$

- Finally, carry out the following pointwise product:

$$\mathbf{P}(B \mid j, m) = \alpha \boldsymbol{f}_{\mathbf{1}}(B) \times \boldsymbol{f}_{\mathbf{7}}(B)$$

Pointwise Product of Two Factors

$$\boldsymbol{f}(X_1,\ldots,X_j,Y_1,\ldots,Y_k)\times \boldsymbol{g}(Y_1,\ldots,Y_k,Z_1,\ldots,Z_l)=\boldsymbol{h}(X_1,\ldots,X_j,Y_1,\ldots,Y_k,Z_1,\ldots,Z_l)$$

X	Y	$\mathbf{f}(X,Y)$	Y	Z	$\mathbf{g}(Y,Z)$	X	Y	Z	$\mathbf{h}(X,Y,Z)$
t	t	.3	t	t	.2	t	t	t	$.3 \times .2 = .06$
t	f	.7	t	f	.8	t	t	f	$.3 \times .8 = .24$
f	t	.9	f	t	.6	t	f	t	$.7 \times .6 = .42$
f	f	.1	f	f	.4	t	f	f	$.7 \times .4 = .28$
						f	t	t	$.9 \times .2 = .18$
						f	t	f	$.9 \times .8 = .72$
						f	f	t	$.1 \times .6 = .06$
						f	f	f	$.1 \times .4 = .04$

Pointwise Product of Two Factors

$$\boldsymbol{f}(X_1, \ldots, X_j, Y_1, \ldots, Y_k) \times \boldsymbol{g}(Y_1, \ldots, Y_k, Z_1, \ldots, Z_l) = \boldsymbol{h}(X_1, \ldots, X_j, Y_1, \ldots, Y_k, Z_1, \ldots, Z_l)$$

common variables

X	Y	$\mathbf{f}(X,Y)$	Y	Z	$\mathbf{g}(Y,Z)$	X	Y	Z	$\mathbf{h}(X,Y,Z)$
t	t	.3	t	t	.2	t	t	t	$.3 \times .2 = .06$
t	f	.7	t	f	.8	t	t	f	$.3 \times .8 = .24$
f	t	.9	f	t	.6	t	f	t	$.7 \times .6 = .42$
f	f	.1	f	f	.4	t	f	f	$.7 \times .4 = .28$
						f	t	t	$.9 \times .2 = .18$
						f	t	f	$.9 \times .8 = .72$
						f	f	t	$.1 \times .6 = .06$
						f	f	f	$.1 \times .4 = .04$

Pointwise Product of Two Factors

$$\boldsymbol{f}(X_1, \ldots, X_j, Y_1, \ldots, Y_k) \times \boldsymbol{g}(Y_1, \ldots, Y_k, Z_1, \ldots, Z_l) = \boldsymbol{h}(X_1, \ldots, X_j, Y_1, \ldots, Y_k, Z_1, \ldots, Z_l)$$

common variables

X	Y	$\mathbf{f}(X,Y)$	Y	Z	$\mathbf{g}(Y,Z)$	X	Y	Z	$\mathbf{h}(X,Y,Z)$
t	t	.3	t	t	.2	t	t	t	$.3 \times .2 = .06$
t	f	.7	t	f	.8	t	t	f	$.3 \times .8 = .24$
f	t	.9	f	t	.6	t	f	t	$.7 \times .6 = .42$
f	f	.1	f	f	.4	t	f	f	$.7 \times .4 = .28$
						f	t	t	$.9 \times .2 = .18$
						f	t	f	$.9 \times .8 = .72$
						f	f	t	$.1 \times .6 = .06$
						f	f	f	$.1 \times .4 = .04$

Pointwise Product of Two Factors

$$\boldsymbol{f}(X_1,\dots,X_j,Y_1,\dots,Y_k) \times \boldsymbol{g}(Y_1,\dots,Y_k,Z_1,\dots,Z_l) = \boldsymbol{h}(X_1,\dots,X_j,Y_1,\dots,Y_k,Z_1,\dots,Z_l)$$

common variables

X	Y	$\mathbf{f}(X,Y)$	Y	Z	$\mathbf{g}(Y,Z)$	X	Y	Z	$\mathbf{h}(X,Y,Z)$
t	t	.3	t	t	.2	t	t	t	$.3 \times .2 = .06$
t	f	.7	t	f	.8	t	t	f	$.3 \times .8 = .24$
f	t	.9	f	t	.6	t	f	t	$.7 \times .6 = .42$
f	f	.1	f	f	.4	t	f	f	$.7 \times .4 = .28$
						f	t	t	$.9 \times .2 = .18$
						f	t	f	$.9 \times .8 = .72$
						f	f	t	$.1 \times .6 = .06$
						f	f	f	$.1 \times .4 = .04$

Pointwise Product of Two Factors

$$\boldsymbol{f}(X_1, \ldots, X_j, Y_1, \ldots, Y_k) \times \boldsymbol{g}(Y_1, \ldots, Y_k, Z_1, \ldots, Z_l) = \boldsymbol{h}(X_1, \ldots, X_j, Y_1, \ldots, Y_k, Z_1, \ldots, Z_l)$$

common variables

X	Y	$\mathbf{f}(X,Y)$	Y	Z	$\mathbf{g}(Y,Z)$	X	Y	Z	$\mathbf{h}(X,Y,Z)$
t	t	.3	t	t	.2	t	t	t	$.3 \times .2 = .06$
t	f	.7	t	f	.8	t	t	f	$.3 \times .8 = .24$
f	t	.9	f	t	.6	t	f	t	$.7 \times .6 = .42$
f	f	.1	f	f	.4	t	f	f	$.7 \times .4 = .28$
						f	t	t	$.9 \times .2 = .18$
						f	t	f	$.9 \times .8 = .72$
						f	f	t	$.1 \times .6 = .06$
						f	f	f	$.1 \times .4 = .04$

Summing Out a Variable

This operation over a product of factors is carried out by adding up the submatrices, each for one value of the same variable.

$$\boldsymbol{l}(Y,Z) = \sum_X \boldsymbol{h}(X,Y,Z) = \boldsymbol{h}(x,Y,Z) + \boldsymbol{h}(\neg x,Y,Z)$$

X	Y	$\mathbf{f}(X,Y)$	Y	Z	$\mathbf{g}(Y,Z)$	X	Y	Z	$\mathbf{h}(X,Y,Z)$
t	t	.3	t	t	.2	t	t	t	$.3 \times .2 = .06$
t	f	.7	t	f	.8	t	t	f	$.3 \times .8 = .24$
f	t	.9	f	t	.6	t	f	t	$.7 \times .6 = .42$
f	f	.1	f	f	.4	t	f	f	$.7 \times .4 = .28$
						f	t	t	$.9 \times .2 = .18$
						f	t	f	$.9 \times .8 = .72$
						f	f	t	$.1 \times .6 = .06$
						f	f	f	$.1 \times .4 = .04$

Summing Out a Variable

This operation over a product of factors is carried out by adding up the submatrices, each for one value of the same variable.

$$\boldsymbol{l}(Y,Z) = \sum_X \boldsymbol{h}(X,Y,Z) = \boldsymbol{h}(x,Y,Z) + \boldsymbol{h}(\neg x,Y,Z)$$

X	Y	$\mathbf{f}(X,Y)$	Y	Z	$\mathbf{g}(Y,Z)$	X	Y	Z	$\mathbf{h}(X,Y,Z)$
t	t	.3	t	t	.2	t	t	t	$.3 \times .2 = .06$
t	f	.7	t	f	.8	t	t	f	$.3 \times .8 = .24$
f	t	.9	f	t	.6	t	f	t	$.7 \times .6 = .42$
f	f	.1	f	f	.4	t	f	f	$.7 \times .4 = .28$
						f	t	t	$.9 \times .2 = .18$
						f	t	f	$.9 \times .8 = .72$
						f	f	t	$.1 \times .6 = .06$
						f	f	f	$.1 \times .4 = .04$

Summing Out a Variable

This operation over a product of factors is carried out by adding up the submatrices, each for one value of the same variable.

$$\boldsymbol{l}(Y,Z) = \sum_X \boldsymbol{h}(X,Y,Z) = \boldsymbol{h}(x,Y,Z) + \boldsymbol{h}(\neg x,Y,Z)$$

X	Y	$\mathbf{f}(X,Y)$	Y	Z	$\mathbf{g}(Y,Z)$	X	Y	Z	$\mathbf{h}(X,Y,Z)$
t	t	.3	t	t	.2	t	t	t	$.3 \times .2 = .06$
t	f	.7	t	f	.8	t	t	f	$.3 \times .8 = .24$
f	t	.9	f	t	.6	t	f	t	$.7 \times .6 = .42$
f	f	.1	f	f	.4	t	f	f	$.7 \times .4 = .28$
						f	t	t	$.9 \times .2 = .18$
						f	t	f	$.9 \times .8 = .72$
						f	f	t	$.1 \times .6 = .06$
						f	f	f	$.1 \times .4 = .04$

Summing Out a Variable

This operation over a product of factors is carried out by adding up the submatrices, each for one value of the same variable.

$$\boldsymbol{l}(Y,Z) = \sum_X \boldsymbol{h}(X,Y,Z) = \boldsymbol{h}(x,Y,Z) + \boldsymbol{h}(\neg x,Y,Z)$$

$$= \begin{pmatrix} .06 & .24 \\ .42 & .28 \end{pmatrix}$$

X	Y	$\mathbf{f}(X,Y)$	Y	Z	$\mathbf{g}(Y,Z)$	X	Y	Z	$\mathbf{h}(X,Y,Z)$
t	t	.3	t	t	.2	t	t	t	$.3 \times .2 = .06$
t	f	.7	t	f	.8	t	t	f	$.3 \times .8 = .24$
f	t	.9	f	t	.6	t	f	t	$.7 \times .6 = .42$
f	f	.1	f	f	.4	t	f	f	$.7 \times .4 = .28$
						f	t	t	$.9 \times .2 = .18$
						f	t	f	$.9 \times .8 = .72$
						f	f	t	$.1 \times .6 = .06$
						f	f	f	$.1 \times .4 = .04$

Summing Out a Variable

This operation over a product of factors is carried out by adding up the submatrices, each for one value of the same variable.

$$\boldsymbol{l}(Y,Z) = \sum_{X} \boldsymbol{h}(X,Y,Z) = \boldsymbol{h}(x,Y,Z) + \boldsymbol{h}(\neg x,Y,Z)$$

$$= \begin{pmatrix} .06 & .24 \\ .42 & .28 \end{pmatrix}$$

X	Y	$\mathbf{f}(X,Y)$	Y	Z	$\mathbf{g}(Y,Z)$	X	Y	Z	$\mathbf{h}(X,Y,Z)$
t	t	.3	t	t	.2	t	t	t	$.3 \times .2 = .06$
t	f	.7	t	f	.8	t	t	f	$.3 \times .8 = .24$
f	t	.9	f	t	.6	t	f	t	$.7 \times .6 = .42$
f	f	.1	f	f	.4	t	f	f	$.7 \times .4 = .28$
						f	t	t	$.9 \times .2 = .18$
						f	t	f	$.9 \times .8 = .72$
						f	f	t	$.1 \times .6 = .06$
						f	f	f	$.1 \times .4 = .04$

Summing Out a Variable

This operation over a product of factors is carried out by adding up the submatrices, each for one value of the same variable.

$$\boldsymbol{l}(Y,Z) = \sum_X \boldsymbol{h}(X,Y,Z) = \boldsymbol{h}(x,Y,Z) + \boldsymbol{h}(\neg x,Y,Z)$$

$$= \begin{pmatrix} .06 & .24 \\ .42 & .28 \end{pmatrix}$$

X	Y	$\mathbf{f}(X,Y)$	Y	Z	$\mathbf{g}(Y,Z)$	X	Y	Z	$\mathbf{h}(X,Y,Z)$
t	t	.3	t	t	.2	t	t	t	$.3 \times .2 = .06$
t	f	.7	t	f	.8	t	t	f	$.3 \times .8 = .24$
f	t	.9	f	t	.6	t	f	t	$.7 \times .6 = .42$
f	f	.1	f	f	.4	t	f	f	$.7 \times .4 = .28$
						f	t	t	$.9 \times .2 = .18$
						f	t	f	$.9 \times .8 = .72$
						f	f	t	$.1 \times .6 = .06$
						f	f	f	$.1 \times .4 = .04$

Summing Out a Variable

This operation over a product of factors is carried out by adding up the submatrices, each for one value of the same variable.

$$\boldsymbol{l}(Y,Z) = \sum_X \boldsymbol{h}(X,Y,Z) = \boldsymbol{h}(x,Y,Z) + \boldsymbol{h}(\neg x,Y,Z)$$

$$= \begin{pmatrix} .06 & .24 \\ .42 & .28 \end{pmatrix} + \begin{pmatrix} .18 & .72 \\ .06 & .04 \end{pmatrix}$$

X	Y	$\mathbf{f}(X,Y)$	Y	Z	$\mathbf{g}(Y,Z)$	X	Y	Z	$\mathbf{h}(X,Y,Z)$
t	t	$.3$	t	t	$.2$	t	t	t	$.3 \times .2 = .06$
t	f	$.7$	t	f	$.8$	t	t	f	$.3 \times .8 = .24$
f	t	$.9$	f	t	$.6$	t	f	t	$.7 \times .6 = .42$
f	f	$.1$	f	f	$.4$	t	f	f	$.7 \times .4 = .28$
						f	t	t	$.9 \times .2 = .18$
						f	t	f	$.9 \times .8 = .72$
						f	f	t	$.1 \times .6 = .06$
						f	f	f	$.1 \times .4 = .04$

Summing Out a Variable

This operation over a product of factors is carried out by adding up the submatrices, each for one value of the same variable.

$$\boldsymbol{l}(Y,Z) = \sum_X \boldsymbol{h}(X,Y,Z) = \boldsymbol{h}(x,Y,Z) + \boldsymbol{h}(\neg x,Y,Z)$$

$$= \begin{pmatrix} .06 & .24 \\ .42 & .28 \end{pmatrix} + \begin{pmatrix} .18 & .72 \\ .06 & .04 \end{pmatrix} = \begin{pmatrix} .24 & .96 \\ .48 & .32 \end{pmatrix}$$

X	Y	$\mathbf{f}(X,Y)$	Y	Z	$\mathbf{g}(Y,Z)$	X	Y	Z	$\mathbf{h}(X,Y,Z)$
t	t	.3	t	t	.2	t	t	t	$.3 \times .2 = .06$
t	f	.7	t	f	.8	t	t	f	$.3 \times .8 = .24$
f	t	.9	f	t	.6	t	f	t	$.7 \times .6 = .42$
f	f	.1	f	f	.4	t	f	f	$.7 \times .4 = .28$
						f	t	t	$.9 \times .2 = .18$
						f	t	f	$.9 \times .8 = .72$
						f	f	t	$.1 \times .6 = .06$
						f	f	f	$.1 \times .4 = .04$

Variable Elimination Algorithm

Move outside the summation any factor independent of the variable to be summed out.

$$\sum_X \boldsymbol{f}(X,Y) \times \boldsymbol{g}(Y,Z) = \boldsymbol{g}(Y,Z) \times \sum_X \boldsymbol{f}(X,Y)$$

IV. Ordering Variables to Eliminate

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(X_i))$$

IV. Ordering Variables to Eliminate

- Every choice of ordering yields a valid algorithm. ⇐ $P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(X_i))$

IV. Ordering Variables to Eliminate

- Every choice of ordering yields a valid algorithm. ⇦ $P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(X_i))$

$$\mathbf{P}(B \mid j, m) = \alpha \boldsymbol{f}_1(B) \times \sum_E \boldsymbol{f}_2(E) \times \sum_A \boldsymbol{f}_3(A, B, E) \times \boldsymbol{f}_4(A) \times \boldsymbol{f}_5(A)$$

(Order: A, E)

IV. Ordering Variables to Eliminate

- Every choice of ordering yields a valid algorithm. ⇐ $P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(X_i))$

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{f_1}(B) \times \sum_E \boldsymbol{f_2}(E) \times \sum_A \boldsymbol{f_3}(A, B, E) \times \boldsymbol{f_4}(A) \times \boldsymbol{f_5}(A) \quad \text{(Order: } A, E\text{)}$$

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{f_1}(B) \times \sum_A \boldsymbol{f_4}(A) \times \boldsymbol{f_5}(A) \times \sum_E \boldsymbol{f_2}(E) \times \boldsymbol{f_3}(A, B, E) \quad \text{(Order: } E, A\text{)}$$

IV. Ordering Variables to Eliminate

- Every choice of ordering yields a valid algorithm. ⇐ $P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(X_i))$

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{f_1}(B) \times \sum_E \boldsymbol{f_2}(E) \times \sum_A \boldsymbol{f_3}(A, B, E) \times \boldsymbol{f_4}(A) \times \boldsymbol{f_5}(A) \quad \text{(Order: } A, E\text{)}$$

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{f_1}(B) \times \sum_A \boldsymbol{f_4}(A) \times \boldsymbol{f_5}(A) \times \sum_E \boldsymbol{f_2}(E) \times \boldsymbol{f_3}(A, B, E) \quad \text{(Order: } E, A\text{)}$$

- Different orderings generates different intermediate factors.

IV. Ordering Variables to Eliminate

- Every choice of ordering yields a valid algorithm. ⇐ $P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(X_i))$

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{f_1}(B) \times \sum_E \boldsymbol{f_2}(E) \times \sum_A \boldsymbol{f_3}(A, B, E) \times \boldsymbol{f_4}(A) \times \boldsymbol{f_5}(A) \qquad \text{(Order: } A, E\text{)}$$

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{f_1}(B) \times \sum_A \boldsymbol{f_4}(A) \times \boldsymbol{f_5}(A) \times \sum_E \boldsymbol{f_2}(E) \times \boldsymbol{f_3}(A, B, E) \qquad \text{(Order: } E, A\text{)}$$

- Different orderings generates different intermediate factors.

- Time and space are dominated by the size of the largest factor constructed, which is subject to two factors:

IV. Ordering Variables to Eliminate

- Every choice of ordering yields a valid algorithm. ⇐ $P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(X_i))$

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{f_1}(B) \times \sum_E \boldsymbol{f_2}(E) \times \sum_A \boldsymbol{f_3}(A, B, E) \times \boldsymbol{f_4}(A) \times \boldsymbol{f_5}(A) \qquad \text{(Order: } A, E\text{)}$$

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{f_1}(B) \times \sum_A \boldsymbol{f_4}(A) \times \boldsymbol{f_5}(A) \times \sum_E \boldsymbol{f_2}(E) \times \boldsymbol{f_3}(A, B, E) \qquad \text{(Order: } E, A\text{)}$$

- Different orderings generates different intermediate factors.

- Time and space are dominated by the size of the largest factor constructed, which is subject to two factors:
    - ordering of variables
    - structure of the network

IV. Ordering Variables to Eliminate

- Every choice of ordering yields a valid algorithm. ⇐ $P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(X_i))$

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{f_1}(B) \times \sum_E \boldsymbol{f_2}(E) \times \sum_A \boldsymbol{f_3}(A, B, E) \times \boldsymbol{f_4}(A) \times \boldsymbol{f_5}(A) \qquad \text{(Order: } A, E\text{)}$$

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{f_1}(B) \times \sum_A \boldsymbol{f_4}(A) \times \boldsymbol{f_5}(A) \times \sum_E \boldsymbol{f_2}(E) \times \boldsymbol{f_3}(A, B, E) \qquad \text{(Order: } E, A\text{)}$$

- Different orderings generates different intermediate factors.

- Time and space are dominated by the size of the largest factor constructed, which is subject to two factors:
  - ordering of variables
  - structure of the network

- It is intractable to determine the optimal order.

IV. Ordering Variables to Eliminate

- Every choice of ordering yields a valid algorithm. ⇦ $P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(X_i))$

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{f_1}(B) \times \sum_E \boldsymbol{f_2}(E) \times \sum_A \boldsymbol{f_3}(A, B, E) \times \boldsymbol{f_4}(A) \times \boldsymbol{f_5}(A) \quad \text{(Order: } A, E\text{)}$$

$$\boldsymbol{P}(B \mid j, m) = \alpha \boldsymbol{f_1}(B) \times \sum_A \boldsymbol{f_4}(A) \times \boldsymbol{f_5}(A) \times \sum_E \boldsymbol{f_2}(E) \times \boldsymbol{f_3}(A, B, E) \quad \text{(Order: } E, A\text{)}$$

- Different orderings generates different intermediate factors.

- Time and space are dominated by the size of the largest factor constructed, which is subject to two factors:

  - ordering of variables
  - structure of the network

- It is intractable to determine the optimal order.

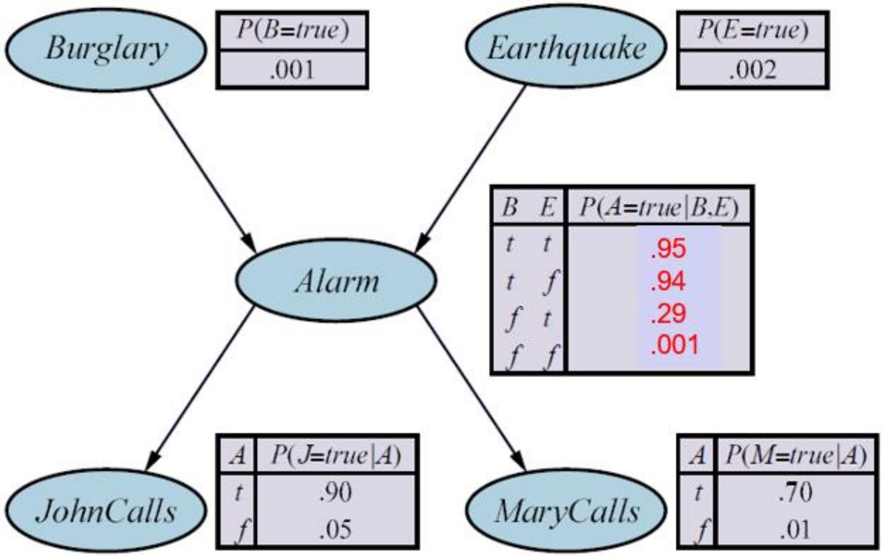
- Use a greedy heuristic: eliminate whichever variable minimizes the size of the next factor to be constructed.

Irrelevant Variable

$\mathbf{P}(JohnCalls \mid Burglary = true)$?

Irrelevant Variable

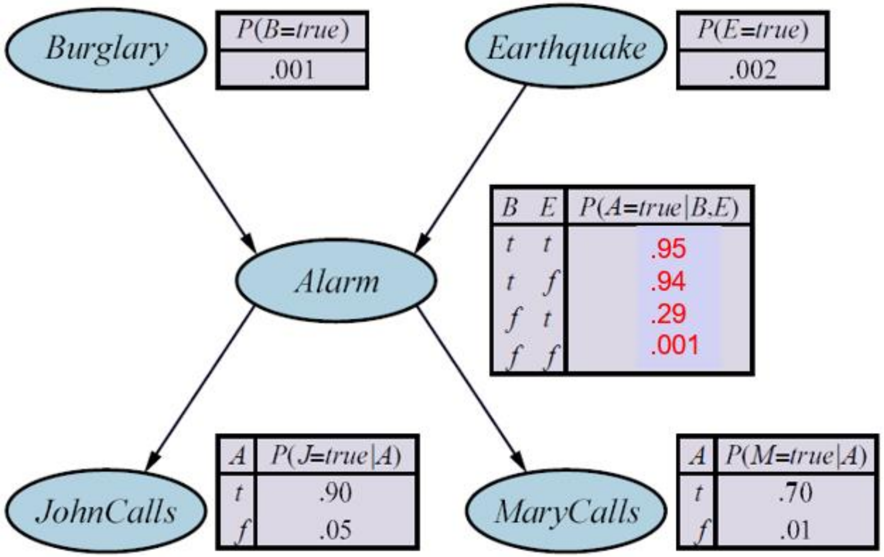
$\boldsymbol{P}(JohnCalls \mid Burglary = true)$?



Irrelevant Variable

P(*JohnCalls* | *Burglary* = *true*)?

$$\mathbf{P}(J \mid b) = \alpha P(b) \sum_{E} P(E) \sum_{A} P(A \mid b, E)\, \mathbf{P}(J \mid A) \sum_{M} P(M \mid A)$$

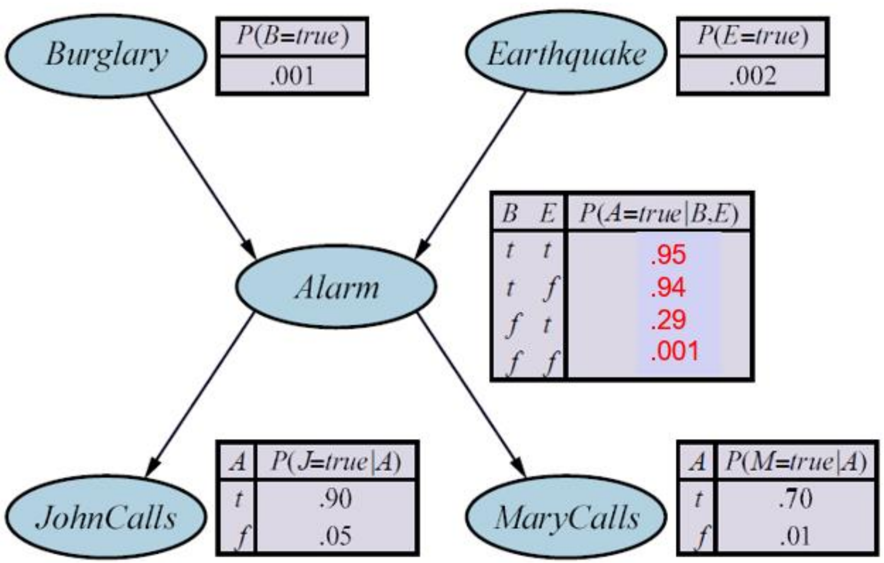


Irrelevant Variable

$\mathbf{P}(JohnCalls \mid Burglary = true)?$

$$\mathbf{P}(J \mid b) = \alpha P(b) \sum_E P(E) \sum_A P(A \mid b, E)\, \mathbf{P}(J \mid A) \sum_M P(M \mid A)$$

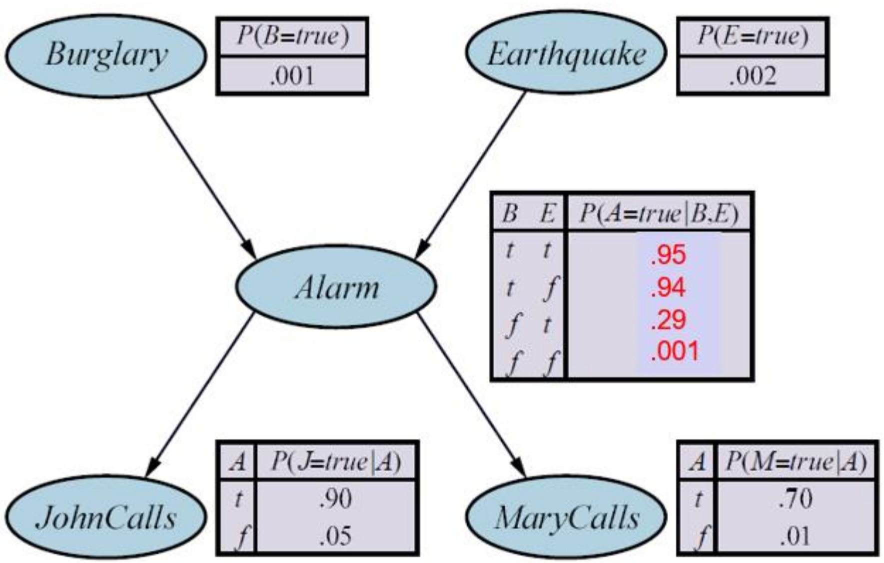
$$\sum_M P(M \mid A) = 1$$



Irrelevant Variable

$\mathbf{P}(JohnCalls \mid Burglary = true)$?

$$\mathbf{P}(J \mid b) = \alpha P(b) \sum_{E} P(E) \sum_{A} P(A \mid b, E)\, \mathbf{P}(J \mid A) \sum_{M} P(M \mid A)$$



$$\sum_{M} P(M \mid A) = 1$$

$\Downarrow$

The variable M is irrelevant to the query.

Inference with Elimination

- We can remove a leaf node (e.g., M) that is neither a query variable nor an evidence variable.

- After the removal, there may be more leaf nodes that are irrelevant. Remove them as well, and so on.

Inference with Elimination

- We can remove a leaf node (e.g., M) that is neither a query variable nor an evidence variable.

- After the removal, there may be more leaf nodes that are irrelevant. Remove them as well, and so on.

Every variable that is not an ancestor of a query variable or evidence variable is irrelevant to the query.

Inference with Elimination

- We can remove a leaf node (e.g., M) that is neither a query variable nor an evidence variable.

- After the removal, there may be more leaf nodes that are irrelevant. Remove them as well, and so on.

Every variable that is not an ancestor of a query variable or evidence variable is irrelevant to the query.

- ♦ Using *reverse topological order* for variables, exact inference with elimination can be 1,000 times faster than the enumeration algorithm.

Inference with Elimination

- We can remove a leaf node (e.g., M) that is neither a query variable nor an evidence variable.

- After the removal, there may be more leaf nodes that are irrelevant. Remove them as well, and so on.

Every variable that is not an ancestor of a query variable or evidence variable is irrelevant to the query.

- ♦ Using *reverse topological order* for variables, exact inference with elimination can be 1,000 times faster than the enumeration algorithm.

- ♦ If we want to compute posterior probabilities for all the variables rather than answer individual queries, we can use clustering algorithms (i.e., join tree algorithms).