

Reasoning Under Uncertainty: Variable Elimination

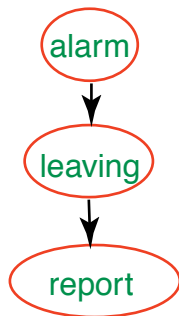
CPSC 322 Lecture 28

March 23, 2007
Textbook §9.4

Lecture Overview

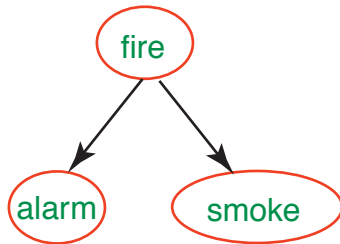
- 1 Recap
- 2 Factors
- 3 Variable Elimination

Chain



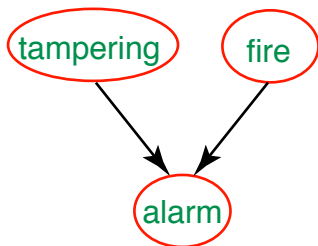
- *alarm* and *report* are independent: **false**.
- *alarm* and *report* are independent given *leaving*: **true**.
- Intuitively, the only way that the *alarm* affects *report* is by affecting *leaving*.

Common ancestors



- *alarm* and *smoke* are independent: **false**.
- *alarm* and *smoke* are independent given *fire*: **true**.
- Intuitively, *fire* can **explain** *alarm* and *smoke*; learning one can affect the other by changing your belief in *fire*.

Common descendants



- *tampering* and *fire* are independent: **true**.
- *tampering* and *fire* are independent given *alarm*: **false**.
- Intuitively, *tampering* can **explain away** *fire*

Belief Network Inference

- Our goal: compute probabilities of variables in a belief network
- Two cases:
 - ① the unconditional (prior) distribution over one or more variables
 - ② the posterior distribution over one or more variables, conditioned on one or more observed variables
- To address both cases, we only need a computational solution to case 1
- Our method: exploiting the structure of the network to efficiently eliminate (sum out) the non-observed, non-query variables one at a time.

Evidence

- If we want to compute the posterior probability of Z given evidence $Y_1 = v_1 \wedge \dots \wedge Y_j = v_j$:

$$\begin{aligned} P(Z|Y_1 = v_1, \dots, Y_j = v_j) &= \frac{P(Z, Y_1 = v_1, \dots, Y_j = v_j)}{P(Y_1 = v_1, \dots, Y_j = v_j)} \\ &= \frac{P(Z, Y_1 = v_1, \dots, Y_j = v_j)}{\sum_Z P(Z, Y_1 = v_1, \dots, Y_j = v_j)}. \end{aligned}$$

- So the computation reduces to the probability of $P(Z, Y_1 = v_1, \dots, Y_j = v_j)$.

Factors

- A **factor** is a representation of a function from a tuple of random variables into a number.
- We will write factor f on variables X_1, \dots, X_j as $f(X_1, \dots, X_j)$.
- A factor denotes a distribution over the given tuple of variables in some (unspecified) context
 - e.g., $P(X_1, X_2)$ is a factor $f(X_1, X_2)$
 - e.g., $P(X_1, X_2, X_3 = v_3)$ is a factor $f(X_1, X_2)$
 - e.g., $P(X_1, X_3 = v_3 | X_2)$ is a factor $f(X_1, X_2)$

Manipulating Factors

- We can make new factors out of an existing factor
- Our first operation: we can assign some or all of the variables of a factor.
 - $f(X_1 = v_1, X_2, \dots, X_j)$, where $v_1 \in \text{dom}(X_1)$, is a factor on X_2, \dots, X_j .
 - $f(X_1 = v_1, X_2 = v_2, \dots, X_j = v_j)$ is a number that is the value of f when each X_i has value v_i .
- The former is also written as
$$f(X_1, X_2, \dots, X_j)_{X_1 = v_1, \dots, X_j = v_j}$$

Lecture Overview

- 1 Recap
- 2 **Factors**
- 3 Variable Elimination

Summing out variables

Our second operation: we can **sum out** a variable, say X_1 with domain $\{v_1, \dots, v_k\}$, from factor $f(X_1, \dots, X_j)$, resulting in a factor on X_2, \dots, X_j defined by:

$$\begin{aligned} & \left(\sum_{X_1} f \right) (X_2, \dots, X_j) \\ &= f(X_1 = v_1, \dots, X_j) + \dots + f(X_1 = v_k, \dots, X_j) \end{aligned}$$

Summing out a variable example

f_3 :

A	B	C	val
t	t	t	0.03
t	t	f	0.07
t	f	t	0.54
t	f	f	0.36
f	t	t	0.06
f	t	f	0.14
f	f	t	0.48
f	f	f	0.32

$\sum_B f_3$:

A	C	val
t	t	0.57
t	f	0.43
f	t	0.54
f	f	0.46

Multiplying factors

- Our third operation: factors can be multiplied together.
- The **product** of factor $f_1(\overline{X}, \overline{Y})$ and $f_2(\overline{Y}, \overline{Z})$, where \overline{Y} are the variables in common, is the factor $(f_1 \times f_2)(\overline{X}, \overline{Y}, \overline{Z})$ defined by:

$$(f_1 \times f_2)(\overline{X}, \overline{Y}, \overline{Z}) = f_1(\overline{X}, \overline{Y})f_2(\overline{Y}, \overline{Z}).$$

Multiplying factors example

f_1 :

A	B	val
t	t	0.1
t	f	0.9
f	t	0.2
f	f	0.8

f_2 :

B	C	val
t	t	0.3
t	f	0.7
f	t	0.6
f	f	0.4

$f_1 \times f_2$:

A	B	C	val
t	t	t	0.03
t	t	f	0.07
t	f	t	0.54
t	f	f	0.36
f	t	t	0.06
f	t	f	0.14
f	f	t	0.48
f	f	f	0.32

Lecture Overview

- 1 Recap
- 2 Factors
- 3 Variable Elimination**

Probability of a conjunction

- Suppose the variables of the belief network are X_1, \dots, X_n .
- What we **want to compute**: the factor $P(Z, Y_1 = v_1, \dots, Y_j = v_j)$
- We can compute $P(Z, Y_1 = v_1, \dots, Y_j = v_j)$ by summing out the variables $Z_1, \dots, Z_k = \{X_1, \dots, X_n\} \setminus \{Z, Y_1, \dots, Y_j\}$.
- We sum out these variables one at a time
 - the order in which we do this is called our **elimination ordering**.

$$\begin{aligned} &P(Z, Y_1 = v_1, \dots, Y_j = v_j) \\ &= \sum_{Z_k} \cdots \sum_{Z_1} P(X_1, \dots, X_n)_{Y_1 = v_1, \dots, Y_j = v_j} \end{aligned}$$

Probability of a conjunction

- What we **know**: the factors $P(X_i|pX_i)$.
- Using the chain rule and the definition of a belief network, we can write $P(X_1, \dots, X_n)$ as $\prod_{i=1}^n P(X_i|pX_i)$. Thus:

$$\begin{aligned} &P(Z, Y_1 = v_1, \dots, Y_j = v_j) \\ &= \sum_{Z_k} \cdots \sum_{Z_1} P(X_1, \dots, X_n)_{Y_1 = v_1, \dots, Y_j = v_j} \cdot \\ &= \sum_{Z_k} \cdots \sum_{Z_1} \prod_{i=1}^n P(X_i|pX_i)_{Y_1 = v_1, \dots, Y_j = v_j} \cdot \end{aligned}$$

Computing sums of products

Computation in belief networks thus reduces to computing the sums of products.

- It takes 14 multiplications or additions to evaluate the expression $ab + ac + ad + aeh + afh + agh$. How can this expression be evaluated more efficiently?

Computing sums of products

Computation in belief networks thus reduces to computing the sums of products.

- It takes 14 multiplications or additions to evaluate the expression $ab + ac + ad + aeh + afh + agh$. How can this expression be evaluated more efficiently?
 - factor out the a and then the h giving $a(b + c + d + h(e + f + g))$
 - this takes only 7 multiplications or additions

Computing sums of products

Computation in belief networks thus reduces to computing the sums of products.

- It takes 14 multiplications or additions to evaluate the expression $ab + ac + ad + aeh + afh + agh$. How can this expression be evaluated more efficiently?
 - factor out the a and then the h giving $a(b + c + d + h(e + f + g))$
 - this takes only 7 multiplications or additions
- How can we compute $\sum_{Z_1} \prod_{i=1}^n P(X_i | pX_i)$ efficiently?

Computing sums of products

Computation in belief networks thus reduces to computing the sums of products.

- It takes 14 multiplications or additions to evaluate the expression $ab + ac + ad + aeh + afh + agh$. How can this expression be evaluated more efficiently?
 - factor out the a and then the h giving $a(b + c + d + h(e + f + g))$
 - this takes only 7 multiplications or additions
- How can we compute $\sum_{Z_1} \prod_{i=1}^n P(X_i | pX_i)$ efficiently?
- Factor out those terms that don't involve Z_1 :

$$\left(\prod_{i|Z_1 \notin \{X_i\} \cup pX_i} P(X_i | pX_i) \right) \left(\sum_{Z_1} \prod_{i|Z_1 \in \{X_i\} \cup pX_i} P(X_i | pX_i) \right)$$

(terms that do not involve Z_1)
(terms that involve Z_1)