Reasoning under Uncertainty Part II

Artificial Intelligence, 2015
TIN172/DIT410

Prasanth Kolachina
based on slides by
Poole, Mackworth and slides from 2015

Chalmers University of Technology

April 29, 2016

Quick recap: Random Variables

- Upper case: X.
- Value is subject to chance.
  - Values: lower case.
  - Could represent the outcome of an experiment.
- A probability $\in [0, 1]$ is associated to each value that X can take.

Quick recap: Probability Distributions

- Describes the behaviour of a random variable.
- $P(X)$ is the probability measure of X.
- More than one variable:
  - Joint: $P(X, Y, Z)$
  - Marginal: $P(X) = \sum_Y P(X, Y)$
  - Conditional: $P(X|Y) = \frac{P(X,Y)}{P(Y)}$

Example: Probability Distributions

X - The outcome of a coin toss

X	$P(X)$
heads	0.5
tails	0.5

- This is called the **Binomial distribution**
- $P(X = heads)$ - the probability that coin comes up heads
- $P(X = tails) = 1 - P(X = heads)$

Chain Rule for Probabilities

$$P(X, Y, Z) = P(X|Y, Z)P(Y, Z)$$

Chain Rule for Probabilities

$$
\begin{aligned}
P(X,Y,Z) &= P(X|Y,Z)P(Y,Z) \\
&= P(X|Y,Z)P(Y|Z)P(Z)
\end{aligned}
$$

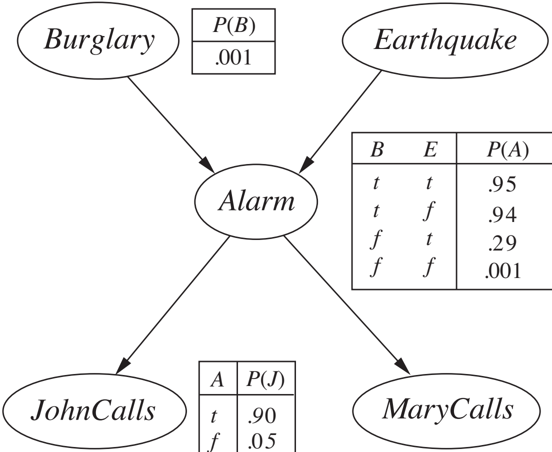
Conditional Independence

$$X \perp Y | Z \rightarrow P(X|Y, Z) = P(X|Z)$$

Probability Distributions ctd.

- Belief networks.
    - Nodes: random variables.
    - Arcs: causal dependence
    - Network encodes independence.
    - Flow of influence.

Example Belief Network



$P(B)$
.001

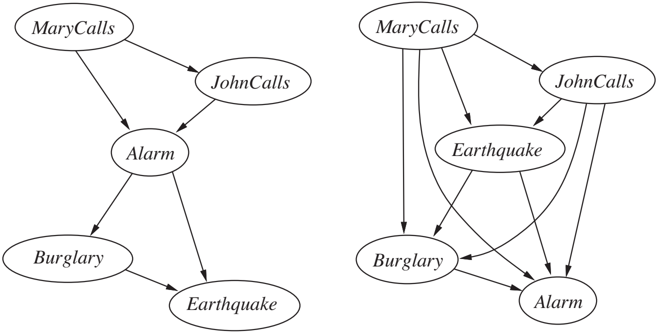
$P(E)$
.002

B	E	$P(A)$
t	t	.95
t	f	.94
f	t	.29
f	f	.001

A	$P(J)$
t	.90
f	.05

A	$P(M)$
t	.70
f	.01

Alternate Formulation



(a) (b)

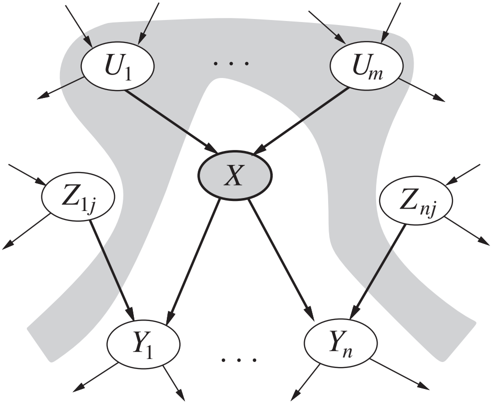
Chain Rule for Bayesian Networks

- Chain rule for Bayesian Networks:

$$P(X_1, X2, X3, ..., X_n) = \prod_i P(X_i | parents(X_i))$$

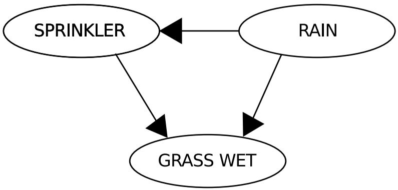
- P factorizes over the network.

Conditional Independence



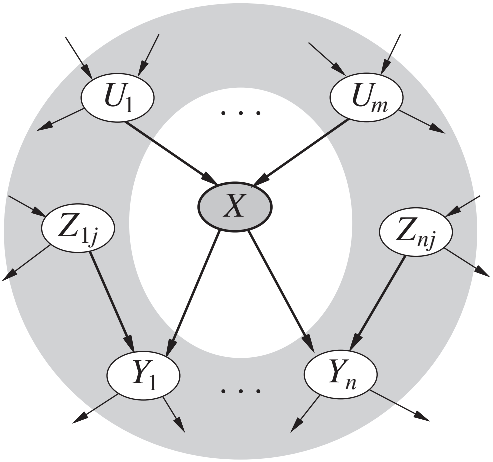
- Defined by the semantics of belief networks

Example Belief Network



- Using the chain rule: $P(Grass, Sprinkler, Rain) = P(Grass|Sprinkler, Rain)P(Sprinkler|Rain)P(Rain)$
- A factorization of P.

Markov Blanket



Querying Belief Networks

- Query variable(s): Q
- Observed evidence: $E_1 = e_1, E_2 = e_2, \ldots, E_n = e_n$
- How do you calculate $P(Q|e)$?

Inference in Belief Networks

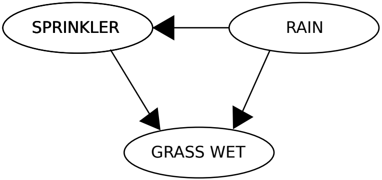
1. Set observed evidence: $E_1 = e_1, E_2 = e_2, \ldots, E_n = e_n$
2. Marginalize out non-query variables W .
3. $P(Q|e) \propto \sum_W P(Q, W, E = e)$
4. Renormalize.

Renormalization

- $\tilde{P}(X)$ - an unnormalized probability measure.
- $P(X) = \frac{\tilde{P}(X)}{\sum_X \tilde{P}(X)}$ - renormalized probability distribution over X.
- The denominator, $\sum_X \tilde{P}(X)$ is merely a constant.

Example Belief Network

RAIN	SPRINKLER T	SPRINKLER F
F	0.4	0.6
T	0.01	0.99



RAIN T	RAIN F
0.2	0.8

SPRINKLER	RAIN	GRASS WET T	GRASS WET F
F	F	0.0	1.0
F	T	0.8	0.2
T	F	0.9	0.1
T	T	0.99	0.01

Factors in general

Function: $f(X_1, \ldots, X_j)$.
Assignments:

- $f(X_1 = x_1, X_2, \ldots, X_j)$, is a factor on $X_2, \ldots, X_j$.
- $f(X_1 = x_1, X_2 = x_2, \ldots, X_j = x_j)$

Example factors

$r(X, Y, Z)$:

X	Y	Z	val
t	t	t	0.1
t	t	f	0.9
t	f	t	0.2
t	f	f	0.8
f	t	t	0.4
f	t	f	0.6
f	f	t	0.3
f	f	f	0.7

$r(X{=}t, Y, Z)$:

Y	Z	val
t	t	0.1
t	f	
f	t	
f	f	

Example factors

$r(X, Y, Z)$:

X	Y	Z	val
t	t	t	0.1
t	t	f	0.9
t	f	t	0.2
t	f	f	0.8
f	t	t	0.4
f	t	f	0.6
f	f	t	0.3
f	f	f	0.7

$r(X{=}t, Y, Z)$:

Y	Z	val
t	t	0.1
t	f	0.9
f	t	0.2
f	f	0.8

$r(X{=}t, Y, Z{=}f)$:

Example factors

$r(X,Y,Z)$:

X	Y	Z	val
t	t	t	0.1
t	t	f	0.9
t	f	t	0.2
t	f	f	0.8
f	t	t	0.4
f	t	f	0.6
f	f	t	0.3
f	f	f	0.7

$r(X=t,Y,Z)$:

Y	Z	val
t	t	0.1
t	f	
f	t	
f	f	

$r(X=t,Y,Z=f)$:

Y	val
t	
f	

$r(X=t,Y=f,Z=f) =$

Example factors

$r(X, Y, Z)$:

X	Y	Z	val
t	t	t	0.1
t	t	f	0.9
t	f	t	0.2
t	f	f	0.8
f	t	t	0.4
f	t	f	0.6
f	f	t	0.3
f	f	f	0.7

$r(X{=}t, Y, Z)$:

Y	Z	val
t	t	0.1
t	f	
f	t	
f	f	

$r(X{=}t, Y, Z{=}f)$:

Y	val
t	0.9
f	0.8

$$r(X{=}t, Y{=}f, Z{=}f) = 0.8$$

Multiplying factors

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:

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

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	
t	f	t	
t	f	f	
f	t	t	
f	t	f	
f	f	t	
f	f	f	

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

Variable Elimination Algorithm

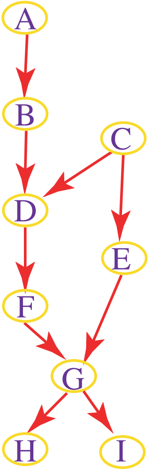
Compute the distribution of some query variable X_q

1. Use chain rule to get factorization.
2. Set observed variables.
3. Elimination: Marginalize all variables except X_q:
    - "Push in" the summations:

$$\sum_Y P(X)P(Y) = P(X)\sum_Y P(Y)$$

4. Multiply the remaining factors.
5. Renormalize.

Variable elimination example: $P(D|H)$



$$\left.\begin{array}{l} P(A) \\ P(B|A) \end{array}\right\} \xrightarrow{\text{elim } A} f_1(B)$$

$$\left.\begin{array}{l} P(C) \\ P(D|B,C) \\ P(E|C) \end{array}\right\} \xrightarrow{\text{elim } C} f_2(BDE)$$

$$P(F|D)$$

$$P(G|F,E)$$

$$P(H|G) \;\}\; \xrightarrow{\text{obs } H} f_3(G)$$

$$P(I|G) \;\}\; \xrightarrow{\text{elim } I} f_4(G)$$

Variable Elimination example: $P(D|H)$

$$P(D, H = h) = \frac{\sum_{A,B,C,E,F,G,I} P(A, B, C, D, E, F, G, H = h, I)}{Z}$$

$$= \frac{\sum_{A,B,C,E,F,G,I} P(I|G)P(H{=}h|G)P(G|F,E)P(F|D)P(E|C)P(D|B,C)P(C)P(B|A)P(A)}{Z}$$

$$= \frac{\sum_{I,G} P(I|G)P(H = h|G) \sum_{E,F} P(G|F, E)P(F|D) \sum_{C} P(E|C) \sum_{B} P(D|B, C)P(C) \sum_{A} P(B|A)P(A)}{Z}$$

Z is the (re)normalizing constant.

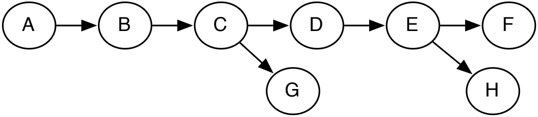
Variable Elimination example, ctd.

$$\underbrace{\sum_{G} f_4(G) f_3(G) \underbrace{\sum_{E,F} P(G|F,E) P(F|D) \underbrace{\sum_{B} f_2(B,D,E) f_1(B)}_{f_5(D,E)}}_{f_6(D,G)}}_{f_7(D)}$$

$$P(D, H = h) = \frac{f_7(D)}{Z}$$

Z is the (re)normalizing constant. f_1, f_2, f_3, see previous slide.

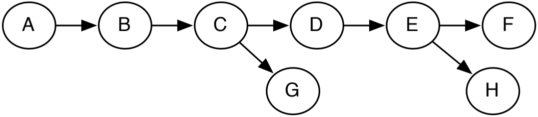
Variable Elimination example



Query: $P(G|f)$; elimination ordering: A, H, E, D, B, C

$$P(G|f) \quad \propto$$

Variable Elimination example



Query: $P(G|f)$; elimination ordering: A, H, E, D, B, C

$$P(G|f) \propto \sum_C \sum_B \sum_D \sum_E \sum_H \sum_A P(A)P(B|A)P(C|B) P(D|C)P(E|D)P(f|E)P(G|C)P(H|E)$$

$$= \sum_C \left(\sum_B \left(\sum_A P(A)P(B|A) \right) P(C|B) \right) P(G|C) \left(\sum_D P(D|C) \left(\sum_E P(E|D)P(f|E) \sum_H P(H|E) \right) \right)$$

Markov chain

- A Markov chain is a special case of belief network:

What probabilities need to be specified? What Independence assumptions are made?

Markov chain

- A Markov chain is a special case of belief network:

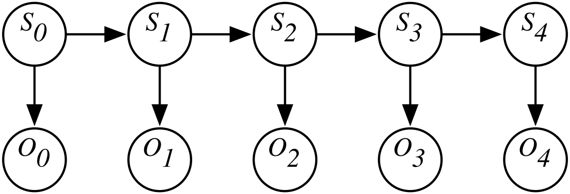
- $P(S_0)$ specifies initial conditions
- $P(S_{t+1}|S_t)$ specifies the dynamics
- $P(S_{t+1}|S_0, \ldots, S_t) = P(S_{t+1}|S_t)$.
- Often S_t represents the **state** at time t. Intuitively S_t conveys all of the information about the history that can affect the future states.
- "The future is independent of the past given the present."

Stationary Markov chain

- A **stationary Markov chain** is when for all $t > 0$, $t' > 0$, $P(S_{t+1}|S_t) = P(S_{t'+1}|S_{t'})$.
- We specify $P(S_0)$ and $P(S_{t+1}|S_t)$.
  - Simple model, easy to specify
  - Often the natural model
  - The network can extend indefinitely

Hidden Markov Model

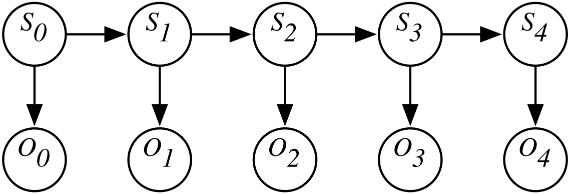
- A **Hidden Markov Model (HMM)** is a belief network:



The probabilities that need to be specified:

Hidden Markov Model

- A **Hidden Markov Model (HMM)** is a belief network:



The probabilities that need to be specified:

- $P(S_0)$ specifies initial conditions
- $P(S_{t+1}|S_t)$ specifies the dynamics
- $P(O_t|S_t)$ specifies the sensor model

Approximate Inference

- Complexity of Belief networks are connected to CSP
- In polytrees, linear to the size of the network
- In multiply connected, exponential in the worst case!
- Approximate inference through sampling
    - Rejection Sampling
    - Gibbs Sampling

Rejection sampling

$\hat{\mathbf{P}}(X|\mathbf{e})$ estimated from samples agreeing with $\mathbf{e}$

function REJECTION-SAMPLING($X, \mathbf{e}, bn, N$) **returns** an estimate of $P(X|\mathbf{e})$
  local variables: $\mathbf{N}$, a vector of counts over X, initially zero

  for $j = 1$ to N **do**
    $\mathbf{x} \leftarrow$ PRIOR-SAMPLE(bn)
    if $\mathbf{x}$ is consistent with $\mathbf{e}$ **then**
      $\mathbf{N}[x] \leftarrow \mathbf{N}[x]+1$ where x is the value of X in $\mathbf{x}$
  return NORMALIZE($\mathbf{N}[X]$)

E.g., estimate $\mathbf{P}(Rain|Sprinkler = true)$ using 100 samples
  27 samples have $Sprinkler = true$
    Of these, 8 have $Rain = true$ and 19 have $Rain = false$.

$\hat{\mathbf{P}}(Rain|Sprinkler = true) = \text{NORMALIZE}(\langle 8, 19 \rangle) = \langle 0.296, 0.704 \rangle$

Similar to a basic real-world empirical estimation procedure

Analysis of rejection sampling

$\hat{\mathbf{P}}(X|\mathbf{e}) = \alpha\mathbf{N}_{PS}(X,\mathbf{e})$ (algorithm defn.)
$= \mathbf{N}_{PS}(X,\mathbf{e})/N_{PS}(\mathbf{e})$ (normalized by $N_{PS}(\mathbf{e})$)
$\approx \mathbf{P}(X,\mathbf{e})/P(\mathbf{e})$ (property of PriorSample)
$= \mathbf{P}(X|\mathbf{e})$ (defn. of conditional probability)

Hence rejection sampling returns consistent posterior estimates

Problem: hopelessly expensive if $P(\mathbf{e})$ is small

$P(\mathbf{e})$ drops off exponentially with number of evidence variables!