22년 삼성 AI 전문가과정
6월 8일 수요일 5교시
장병탁

인공지능

11차시 : Probabilistic Reasoning

서울대학교 컴퓨터공학부
담당 교수: 장병탁

Seoul National University
Byoung-Tak Zhang

Lecture Overview

인공지능

11차시 : Probabilistic Reasoning

서울대학교 컴퓨터공학부
담당 교수: 장병탁

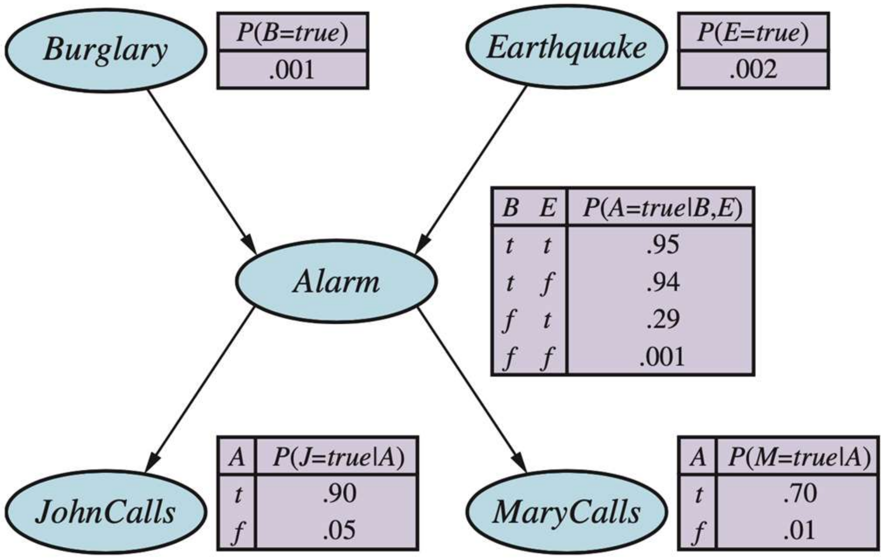
Seoul National University
Byoung-Tak Zhang

Introduction

- **Quantifying Uncertainty and Information Theory** (Previous lecture)
  - The right thing to do—the rational decision—depends on both the relative importance of various goals and the likelihood.
  - Other solutions for estimation of uncertainty: entropy, joint entropy, conditional entropy, mutual information, cross entropy, relative entropy, etc.
- **Probabilistic Representations of the World** (This lecture)
  - **How to represent** dependency relationships explicitly in **Bayesian networks.** Syntax and semantics
  - How to **capture uncertain knowledge** in a natural and efficient way
- **Probabilistic Reasoning** (This lecture)
  - How **probabilistic inference** can be done efficiently in many practical situations
  - A variety of **approximate inference** algorithms (vs. exact inference)

Bayesian Networks

- Bayesian network is directed acyclic graph (DAG) representing a full joint probability distribution of random variables.
  - Node: random variables (X_i)
  - Edges: X_i is a parent of X_j
  - CPT (conditional probability table)
- Representing the full joint distribution
  - $P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | Parents(X_i))$
  - $P(j, m, a, \neg b, \neg e)$
    $= P(j|a)P(m|a)P(a|\neg b \wedge \neg e)P(\neg b)P(\neg e)$
    $= 0.90 \times 0.70 \times 0.001 \times 0.999 \times 0.998$
    $= 0.000628$



<출처> Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4th Edition). Pearson

MCMC Sampling for Bayesian Networks

Metropolis-Hastings (MH) Sampling

- Most broadly applicable MCMC algorithm.
- Generate samples x according to target probabilities $\pi(x)$.

- MH has two stages as follows:

1. **Sample** a new state x' from a proposal distribution $q(x'|x)$, given the current state x.
2. Probabilistically **accept or reject** x' according to acceptance probability

$$a(x'|x) = \min\left(1, \frac{\pi(x')q(x|x')}{\pi(x)q(x'|x)}\right)$$

If the proposal is rejected, the state remains at x.

Approximate Inference for Bayesian Networks

Basic Idea

- Draw N samples from a sampling distribution S
- Compute an approximate posterior probability $\hat{P}$
- Show this converges to the true probability P

Methods

- Direct sampling
- Rejection sampling
- Importance sampling (likelihood weighting)
- Gibbs sampling
- Markov chain Monte Carlo (MCMC)
- Metropolis-Hastings algorithm

Outline (Lecture 11)

Stuart Russell & Peter Norvig (2021), Artificial Intelligence: A Modern Approach (4th Edition)

11.1 Representing Knowledge in an Uncertain Domain

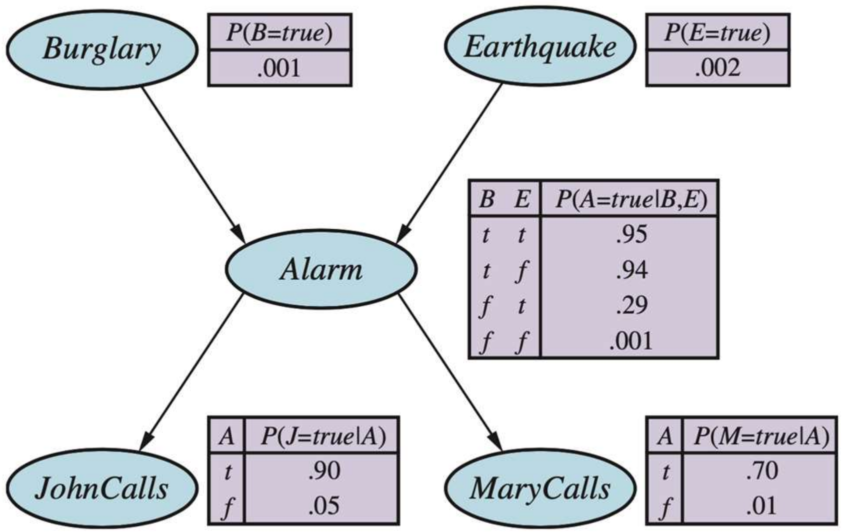
11.1 Representing Knowledge in an Uncertain Domain (1/2)

Bayesian Network

- Represent the dependencies among variables.
- Directed graph in which each node is annotated with quantitative probability information.
- Details are as follows:
  - Each node corresponds to a random variable, which may be discrete or continuous.
  - Directed links or arrows connect pairs of nodes.
  - Each node has associated probability information that quantifies the effect of the parents of the node using a finite number of parameters.

11.1 Representing Knowledge in an Uncertain Domain (2/2)

- Example of Bayesian Network, with both topology and the conditional probability tables (CPTs)
- Directed acyclic graph (DAG) representing a full joint probability distribution of random variables.
  - Node: random variables (X_i)
  - Edges: X_i is a parent of X_j
- Associated with each node is a CPT representing a conditional probability distribution that quantifies the effect of the parents on the node.



11.2 The Semantics of Bayesian Networks

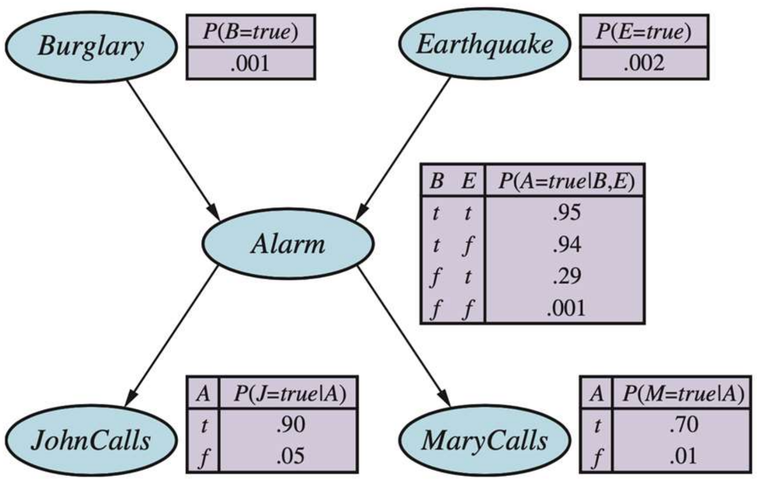
11.2 The Semantics of Bayesian Networks (1/5)

Semantics of Bayesian networks

- Numerical semantics: The network as a representation of the *joint* probability distribution.
- Topological semantics: The network as an encoding of a collection of *conditional independence* statements.

Representing the full joint distribution

- $P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | Parents(X_i))$
- $P(j, m, a, \neg b, \neg e)$
  $= P(j|a)P(m|a)P(a|\neg b \land \neg e)P(\neg b)P(\neg e)$
  $= 0.90 \times 0.70 \times 0.001 \times 0.999 \times 0.998$
  $= 0.000628$



P(B=true)
.001

P(E=true)
.002

B	E	P(A=true|B,E)
t	t	.95
t	f	.94
f	t	.29
f	f	.001

A	P(J=true|A)
t	.90
f	.05

A	P(M=true|A)
t	.70
f	.01

<출처> Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4th Edition). Pearson

11.2 The Semantics of Bayesian Networks (2/5)

Joint probability as a product of conditional probabilities

- **Chain rule**:

$$P(x_1, \dots, x_n) = P(x_n|x_{n-1}, .., x_1)P(x_{n-1}, \dots, x_1)$$

$$= P(x_n|x_{n-1}, .., x_1)P(x_{n-1}|x_{n-2}, .., x_1) \dots P(x_2|x_1)P(x_1) = \prod_{i=1}^{n} P(x_i|x_{i-1}, \dots, x_1)$$

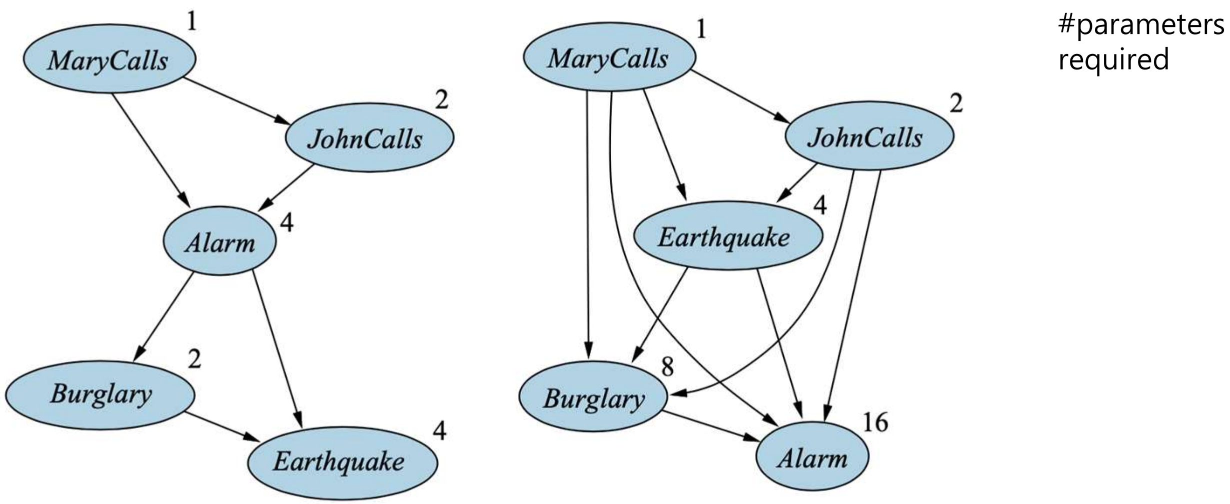
A Method for constructing Bayesian networks

- **Nodes**: First determine the set of variables that are required to model the domain. Now order them, $\{X_1, \dots, X_n\}$. Any order will work, but the resulting network will be more compact if the variables are ordered such that causes precede effects.
- **Links**: For $i = 1\ to\ n$ do:
  - Choose a minimal set of parents for $X_1, \dots, X_{i-1}$.
  - For each parent insert a link from the parent to X_i.
  - CPTs: Write down the conditional probability table, $P(X_i|Parents(X_i))$

11.2 The Semantics of Bayesian Networks (3/5)

Compactness and node ordering

- Ordering 1 (Figure a): <*MaryCalls, JohnCalls, Alarm, Burglary, Earthquake*>
- Ordering 2 (Figure b): <*Marycalls, JohnCalls, Earthquake, Burglary, Alarm*>



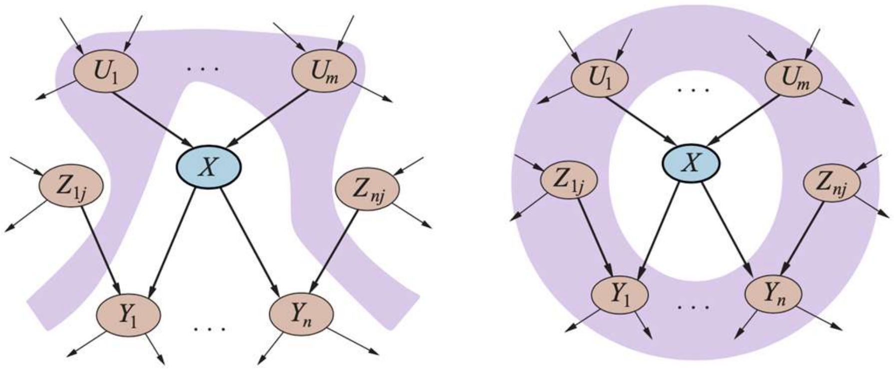
(a) Total: 13 parameters (b) Total: 31 parameters

<출처> Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4th Edition). Pearson

11.2 The Semantics of Bayesian Networks (4/5)

Conditional independence relations in Bayesian networks

- Numerical semantics: representation of the full joint distribution *(previous)*
- Topological semantics: conditional independence relationships
  - Each variable is conditionally independent of its non-descendants, given its parents.
  - A node is conditionally independent of all other nodes in the network, given its parents, children, and children's parents, that is, given its Markov blanket.



(a) (b)

<출처> Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4th Edition). Pearson

11.2 The Semantics of Bayesian Networks (5/5)

Efficient representation of conditional distributions

- Even if the max number of parents k is small, filling in the CPT for a node requires up to $O(2^k)$.
- Uncertain relationships can ofen be characterized by so-called noisy logical relationships.
  - Noisy-OR
  - *Fever* is true if and only if *Cold, Flu,* or *Malaria* are true.

Supposed the individual inhibition probabilities are:

$$q_{cold} = P(\neg fever|cold, \neg flu, \neg malaria) = 0.6$$
$$q_{flu} = P(\neg fever|\neg cold, flu, \neg malaria) = 0.2$$
$$q_{malaria} = P(\neg fever|\neg cold, \neg flu, malaria) = 0.1$$
$$P(x_i|parents(X_i)) = 1 - \prod_{\{j:X_j=true\}} q_j$$

Cold	*Flu*	*Malaria*	$P(fever\|\cdot)$	$P(\neg fever\|\cdot)$
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	$0.012 = 0.6 \times 0.2 \times 0.1$

<출처> Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4th Edition). Pearson

11.3 Exact Inference in Bayesian Networks

11.3 Exact Inference in Bayesian Networks (1/5)

Basic concept

- The basic task for any probabilistic inference system is to compute the posterior probability distribution for a set of query variables, given some observed event, i.e. some assignment of values to a set of evidence variables.
  - $\{X\} \cup E \cup Y$

    X: query variables

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

    $Y = \{Y_1, \ldots, Y_l\}$ non-evidence, non-query variables (hidden variables)
- This section will cover methods for appropriate inference.

11.3 Exact Inference in Bayesian Networks (2/5)

Inference by enumeration

- $\mathbf{P}(X|e) = \alpha\mathbf{P}(X,e) = \alpha \sum_y \mathbf{P}(X,e,y)$

  $\mathbf{P}(B|j,m) = \alpha\mathbf{P}(B,j,m) = \alpha \sum_e \sum_a \mathbf{P}(B,j,m,e,a)$

  $P(b|j,m) = \alpha \sum_e \sum_a P(b)P(e)P(a|b,e)P(j|a)P(m|a)$

  $P(b|j,m) = \alpha P(b) \sum_e P(e) \sum_a P(a|b,e)P(j|a)P(m|a)$

  $\mathbf{P}(B|j,m) = \alpha < 0.00059224, 0.0014919 >$

  $\approx < 0.284, 0.716 >$

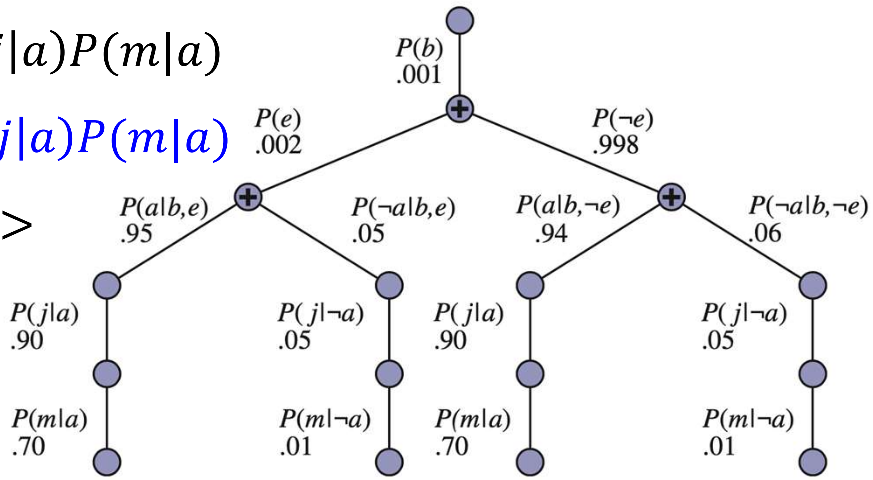


Figure 13.10 The structure of the expression shown in Equation (13.5). The evaluation proceeds top down, multiplying values along each path and summing at the "+" nodes. Notice the repetition of the paths for j and m.

<출처> Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4th Edition). Pearson

11.3 Exact Inference in Bayesian Networks (3/5)

The variable elimination algorithm

- $\mathbf{P}(B|j,m) = \alpha \underbrace{\mathbf{P}(B)}_{\mathbf{f}_1(B)} \sum_e \underbrace{P(e)}_{\mathbf{f}_2(E)} \sum_a \underbrace{\mathbf{P}(a|B,e)}_{\mathbf{f}_3(A|B,E)} \underbrace{P(j|a)}_{\mathbf{f}_4(A)} \underbrace{P(m|a)}_{\mathbf{f}_5(A)}$

$$\mathbf{f}_4(A) = \begin{pmatrix} p(j|a) \\ p(j|\neg a) \end{pmatrix} = \begin{pmatrix} 0.90 \\ 0.05 \end{pmatrix} \qquad \mathbf{f}_5(A) = \begin{pmatrix} p(m|a) \\ p(m|\neg a) \end{pmatrix} = \begin{pmatrix} 0.70 \\ 0.01 \end{pmatrix}$$

$$\mathbf{P}(B|j,m) = \alpha \mathbf{f}_1(B) \times \sum_e \mathbf{f}_2(E) \times \sum_a \mathbf{f}_3(A,B,E) \times \mathbf{f}_4(A) \times \mathbf{f}_5(A)$$

<출처> Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4th Edition). Pearson

11.3 Exact Inference in Bayesian Networks (4/5)

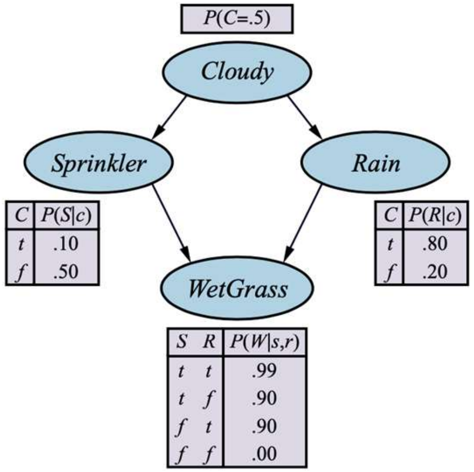
Variable ordering and variable relevance

- Every choice of ordering yields a valid algorithm.
- Different orderings cause different intermediate factors to be generated during the calculation.
- In general, the time and space requirements of variable elimination are dominated by the size of the largest factor constructed during the operation of the algorithm.
  - Determined by the order of elimination of variables and by the structure of the network.

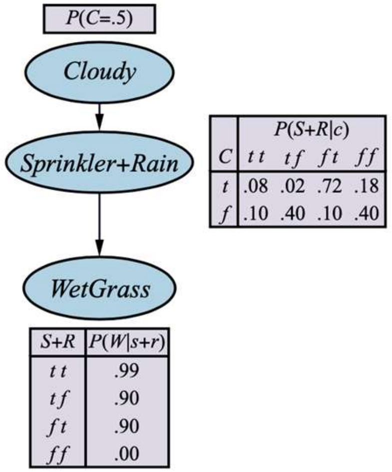
11.3 Exact Inference in Bayesian Networks (5/5)

Complexity of exact inference

- Single connected networks (polytrees):
  Burglary network $O(d^k n)$
- Multiply connected networks:
  - Can reduce 3SAT to exact inference
    → NP-hard
  - Equivalent to counting 3SAT models
    → #P-complete
- Polytree with meganodes
  (using clustering or joint tree algorithm)



Multiply connected network



Clustered equivalent (with meganodes)

<출처> Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4th Edition). Pearson

11.4 Approximate Inference for Bayesian Networks

11.4 Approximate Inference for Bayesian Networks (1/10)

Basic Idea

- Draw N samples from a sampling distribution S
- Compute an approximate posterior probability $\hat{P}$
- Show this converges to the true probability P

Methods

- Direct sampling
- Rejection sampling
- Likelihood weighting
- Markov chain Monte Carlo (MCMC)

11.4 Approximate Inference for Bayesian Networks (2/10)

Monte Carlo algorithms are randomized sampling algorithms that provide approximate answers whose accuracy depends on the number of samples generated.

Two families of algorithms: **direct sampling** and **Markov chain sampling**.

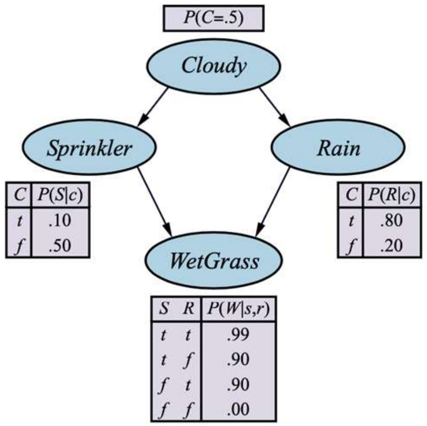
```
function PRIOR-SAMPLE(bn) returns an event sampled from the prior specified by bn
  inputs: bn, a Bayesian network specifying joint distribution P(X1,...,Xn)

  x ← an event with n elements
  for each variable Xi in X1,...,Xn do
     x[i] ← a random sample from P(Xi | parents(Xi))
  return x
```

Direct sampling methods

Ordering: <*Cloudy, Sprinkler, Rain, WetGrass*>

1. Sample from $\mathbf{P}(Cloudy) =< 0.5, 0.5 >$, **value is *true*.**
2. Sample from $\mathbf{P}(Sprinkler|Cloudy = True) =< 0.1, 0.9 >$, **value is *false*.**
3. Sample from $\mathbf{P}(Rain|Cloudy = True) =< 0.8, 0.2 >$, **value is *true*.**
4. Sample from $\mathbf{P}(WetGrass|Sprinkler = false, Rain = true) =$ $< 0.9, 0.1 >$, **value is *true*.**



<출처> Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4th Edition). Pearson

11.4 Approximate Inference for Bayesian Networks (3/10)

Direct sampling methods = PRIOR-SAMPLE algorithm (PS)

- $S_{PS}(x_1, \dots, x_n) = \prod_{i=1}^{n} P(x_i | parents(X_i))$

  $S_{PS}(x_1, \dots, x_n) = P(x_1, \dots, x_n)$

  $\lim_{N \to \infty} \frac{N_{PS}(x_1, \dots, x_n)}{N} = S_{PS}(x_1, \dots, x_n) = P(x_1, \dots, x_n)$

  $S_{PS}(true, false, true, true) = 0.5 \times 0.9 \times 0.8 \times 0.9 = 0.324$

  $P(x_1, \dots, x_m) \approx \frac{N_{PS}(x_1, \dots, x_m)}{N}$, where $m \leq b$

- Such an estimate is called consistent.

11.4 Approximate Inference for Bayesian Networks (4/10)

Rejection sampling in Bayesian networks

- Produce samples from a hard-to-sample distribution given an easy-to-sample distribution.
- $P(X|e)$ estimated from samples agreeing with $\boldsymbol{e}$.
- i.e. estimate $P(Rain|Sprinkler = true)$ using 100 samples
  - 27 samples have $Sprinkler = true$
  - Of these, 8 have $Rain = true$,
    19 have $Rain = false$
- $P(Rain|Sprinkler = true)$
  $= Normalize(< 8{,}19 >)$
  $=< 0.296{,}0.704 >$

function REJECTION-SAMPLING(X, **e**, bn, N) **returns** an estimate of $\mathbf{P}(X\,|\,\mathbf{e})$
  inputs: X, the query variable
    e, observed values for variables **E**
    bn, a Bayesian network
    N, the total number of samples to be generated
  local variables: **C**, a vector of counts for each value of X, initially zero

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

<출처> Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4th Edition). Pearson

11.4 Approximate Inference for Bayesian Networks (5/10)

Importance sampling (likelihood weighting)

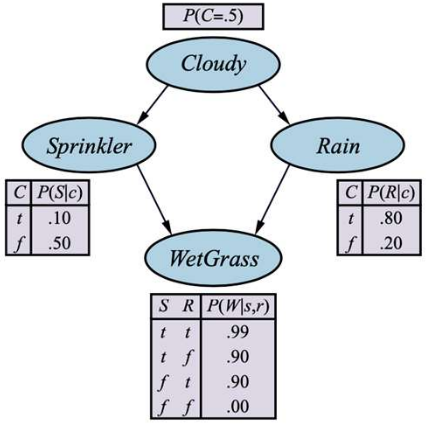
- Fix evidence variables **E**, sample only nonevidence variables, in topological order, each conditioned on its parents.
- Each event generated is consistent with the likelihood it accords the evidence.

- **Query**: $P(Rain|Cloudy = true, WetGrass = true)$
- **Ordering**: $< Cloudy, Sprinkler, Rain, WetGrass >$
- **Procedure**: Set weight $w \leftarrow 1.0$. Generate an event by:

(next slide)

11.4 Approximate Inference for Bayesian Networks (6/10)

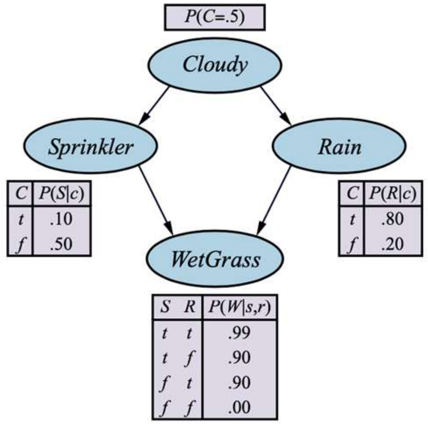
Likelihood weighting (cont'd)



1. $Cloudy$ is an evidence variable with value *true*. Therefore, we set
   $w \leftarrow w \times P(Cloudy = true) = 0.5$
2. $Sprinkler$ is not an evidence variable, so sample from
   $P(Sprinkler|Cloudy = true) =< 0.1, 0.9 >$; suppose return *false*.
3. Similarly, sample from $P(Rain|Cloudy = true) =< 0.8, 0.2 >$;
   suppose return *true*.
4. $WetGrass$ is an evidence variable with value true. Therefore, we set,
   $w \leftarrow w \times P(WetGrass = true|Sprinkler = false, Rain = true) = 0.45$

Weighted-sample returns the event [*true,false,true,true*] with weight *0.45*, and this is tallied under *Rain=true*.

11.4 Approximate Inference for Bayesian Networks (7/10)



Gibbs sampling in Bayesian Networks

Query: $P(Rain|Sprinkler = true, WetGrass = true)$

Initial state: [*true*, **true**, *false*, **true**]

Nonevidence variables Z_i are sampled repeatedly in an arbitrary order.

Example:

1. $Cloudy$ is chosen and then sampled, given the current values of its Markov blanket: in this case, we sample from $P(Cloudy|Sprinkler = true, Rain = false)$. Suppose the result is *Cloudy = false*.

2. $Rain$ is chose and then sampled, given the current values of its Markov blanket: in this case, we sampled from $P(Rain\ |Cloudy = false, Sprinkler = true, WetGrass = true)$. Suppose this yields *Rain = true*. The new current state is [*false*, **true**, *true*, **true**].

11.4 Approximate Inference for Bayesian Networks (8/10)

Approximate inference using MCMC

- State of network = current assignment to all variables.
- Generate next state by sampling one variable given Markov blanket.
- Sample each variable in turn, keeping evidence fixed.
- Can choose a variable to sample at random each time.

function GIBBS-ASK(X, $\mathbf{e}$, bn, N) **returns** an estimate of $\mathbf{P}(X \mid \mathbf{e})$
  local variables: $\mathbf{C}$, a vector of counts for each value of X, initially zero
    $\mathbf{Z}$, the nonevidence variables in bn
    $\mathbf{x}$, the current state of the network, initialized from $\mathbf{e}$

  initialize $\mathbf{x}$ with random values for the variables in $\mathbf{Z}$
  for $k = 1$ **to** N **do**
    choose any variable Z_i from $\mathbf{Z}$ according to any distribution $\rho(i)$
    set the value of Z_i in $\mathbf{x}$ by sampling from $\mathbf{P}(Z_i \mid mb(Z_i))$
    $\mathbf{C}[j] \leftarrow \mathbf{C}[j] + 1$ where x_j is the value of X in $\mathbf{x}$
  return NORMALIZE($\mathbf{C}$)

<출처> Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4th Edition). Pearson

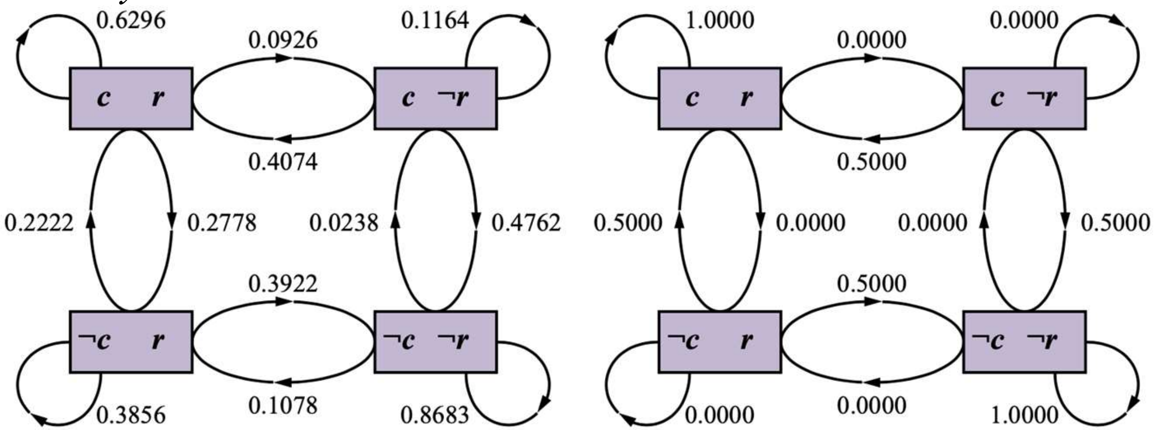
11.4 Approximate Inference for Bayesian Networks (9/10)

Analysis of Markov chains

- The state and transition probabilities of the Markov chain for the query

$$\mathbf{P}(Rain|Sprinkler = true, WetGrass = true)$$

- *(Left)* Self-loops: the state stays the same when either variable is chosen and then resamples the same value it already has.
- *(Right)* The transition probabilities when the CPT for *Rain* constrains it to have the same value as *Cloudy*.



<출처> Stuart J. Russell and Peter Norvig (2021). Artificial Intelligence: A Modern Approach (4th Edition). Pearson

11.4 Approximate Inference for Bayesian Networks (10/10)

Metropolis-Hastings (MH) sampling

- Most broadly applicable MCMC algorithm.
- Generate samples x according to target probabilities $\pi(x)$.

- MH has two stages as follows:

  1. Sample a new state x' from a proposal distribution $q(x'|x)$, given the current state $\boldsymbol{x}$.
  2. Probabilistically accept or reject x' according to acceptance probability

$$a(x'|x) = \min\left(1, \frac{\pi(x')q(x|x')}{\pi(x)q(x'|x)}\right)$$

  If the proposal is rejected, the state remains at x.

Summary

1. A Bayesian network is a directed acyclic graph whose nodes correspond to random variables; each node has a conditional distribution for the node, given its parents
2. Bayesian network provide a way to represent conditional independence relationship and specifies a joint probability distribution over its variable.
3. Inference in Bayesian networks means computing the probability distribution of a set of query variables, given a set of evidence variables. Exact inference algorithms, such as variable elimination, evaluate sum of products of conditional probabilities as efficiently as possible.
4. In polytrees, exact inference takes time linear in the size of the network.
5. Random sampling techniques such as likelihood weighting and Markov chain Monte Carlo can give reasonable estimates of the true posterior probabilities in a network.
6. Whereas Bayes nets capture probabilistic influences, causal networks capture causal relationships and allow prediction of the effects of interventions as well as observations.