Probabilistic Graphical Models
Part I: Bayesian Belief Networks

Selim Aksoy

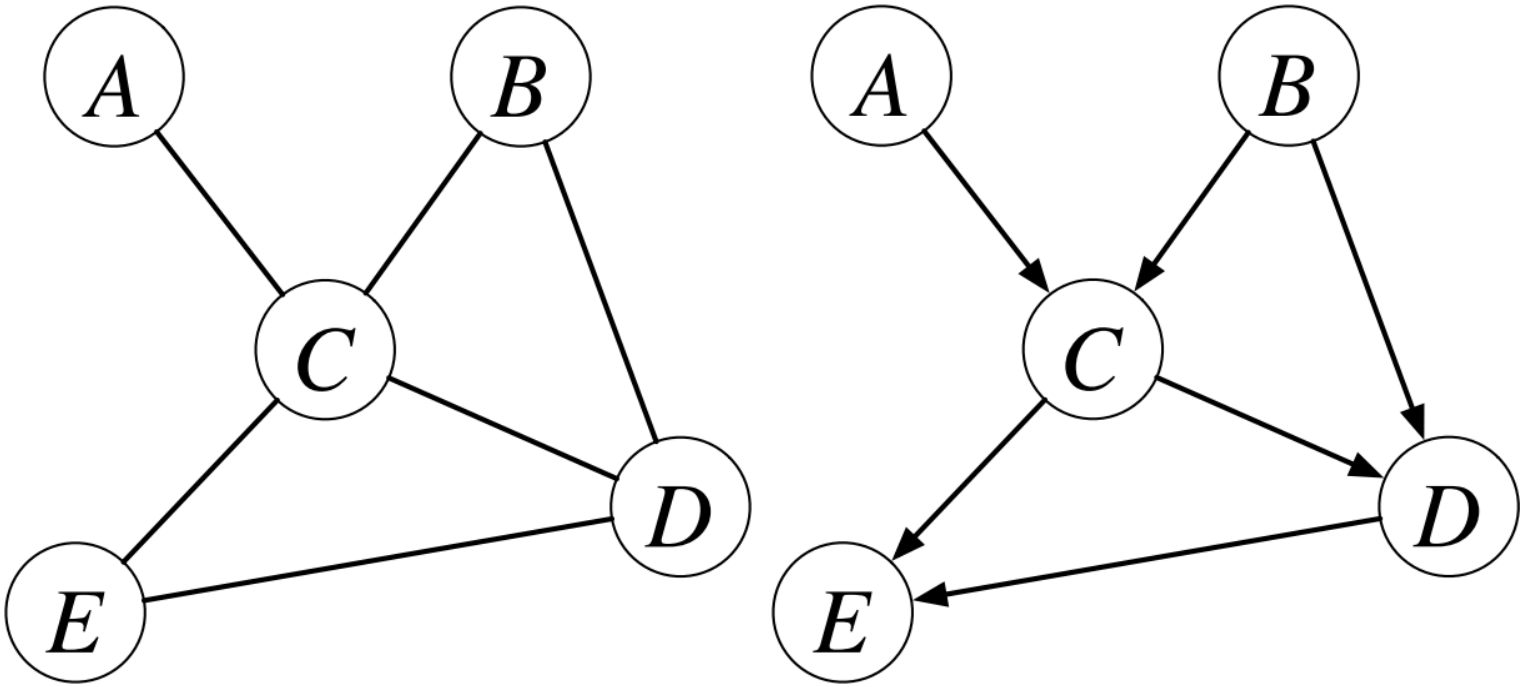
Department of Computer Engineering
Bilkent University
saksoy@cs.bilkent.edu.tr

CS 551, Spring 2012

Introduction

- Graphs are an intuitive way of representing and visualizing the relationships among many variables.
- Probabilistic graphical models provide a tool to deal with two problems: uncertainty and complexity.
- Hence, they provide a compact representation of joint probability distributions using a combination of graph theory and probability theory.
- The graph structure specifies statistical dependencies among the variables and the local probabilistic models specify how these variables are combined.

Introduction



(a) Undirected graph (b) Directed graph

Figure 1: Two main kinds of graphical models. Nodes correspond to random variables. Edges represent the statistical dependencies between the variables.

Introduction

- Marginal independence:

$$X \perp Y \Leftrightarrow X \perp Y|\emptyset \Leftrightarrow P(X,Y) = P(X)P(Y)$$

- Conditional independence:

$$X \perp Y|V \Leftrightarrow P(X|Y,V) = P(X|V) \quad \text{when } P(Y,V) > 0$$

$$X \perp Y|V \Leftrightarrow P(X,Y|V) = P(X|V)P(Y|V)$$

$$\mathcal{X} \perp \mathcal{Y}|\mathcal{V} \Leftrightarrow \{X \perp Y|\mathcal{V},\ \forall X \in \mathcal{X} \text{ and } \forall Y \in \mathcal{Y}\}$$

Introduction

- Marginal and conditional independence examples:
  - Amount of speeding fine $\perp$ Type of car | Speed
  - Lung cancer $\perp$ Yellow teeth | Smoking
  - $(\text{Position}, \text{Velocity})_{t+1} \perp$
    $(\text{Position}, \text{Velocity})_{t-1} \mid (\text{Position}, \text{Velocity})_t, \text{Acceleration}_t$
  - Child's genes $\perp$ Grandparents' genes | Parents' genes
  - Ability of team A $\perp$ Ability of team B
  - not(Ability of team A $\perp$
    Ability of team B | Outcome of A vs B game)

Bayesian Networks

- *Bayesian networks (BN)* are probabilistic graphical models that are based on directed acyclic graphs.
- There are two components of a BN model: $\mathcal{M} = \{\mathcal{G}, \Theta\}$.
  - Each node in the graph $\mathcal{G}$ represents a random variable and edges represent conditional independence relationships.
  - The set Θ of parameters specifies the probability distributions associated with each variable.

Bayesian Networks

- Edges represent "causation" so no directed cycles are allowed.
- Markov property: Each node is conditionally independent of its ancestors given its parents.

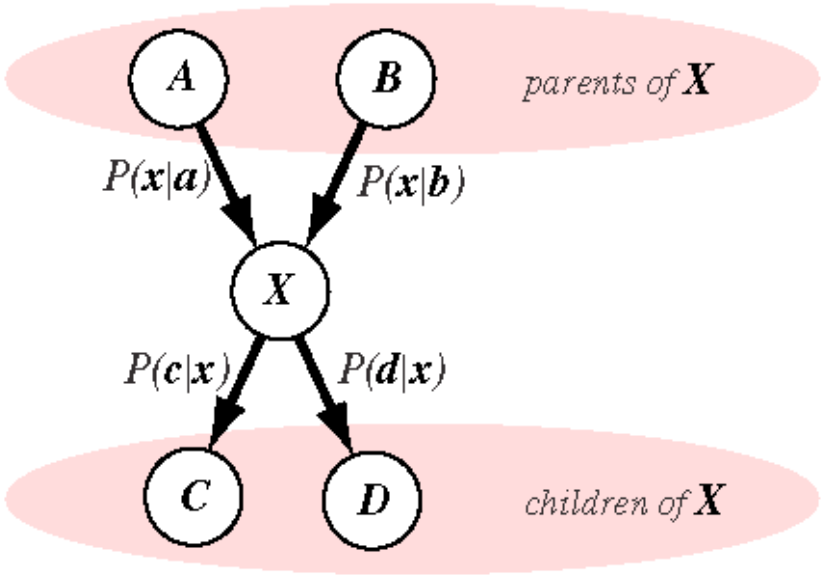


Figure 2: An example BN.

Bayesian Networks

- The joint probability of a set of variables $x_1, \ldots, x_n$ is given as

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

  using the chain rule.

- The conditional independence relationships encoded in the Bayesian network state that a node x_i is conditionally independent of its ancestors given its parents $\boldsymbol{\pi_i}$. Therefore,

$$P(x_1, \ldots, x_n) = \prod_{i=1}^{n} P(x_i | \boldsymbol{\pi_i}).$$

- Once we know the joint probability distribution encoded in the network, we can answer all possible inference questions about the variables using marginalization.

Bayesian Network Examples

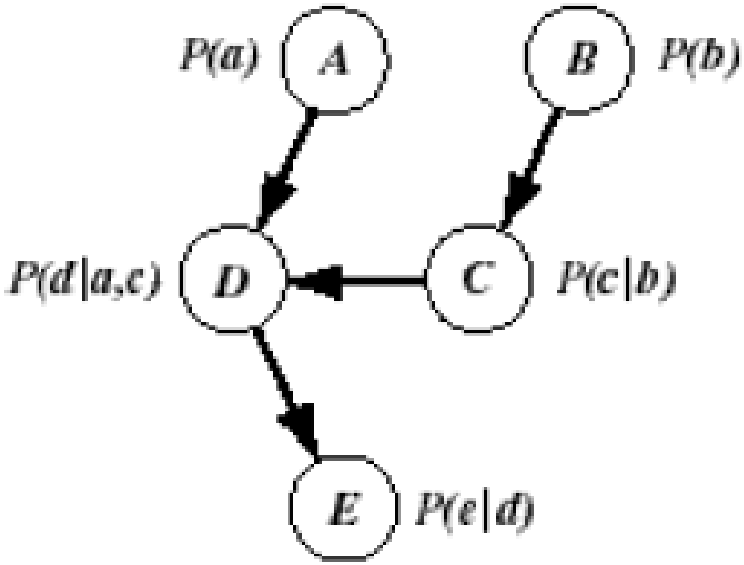


Figure 3: $P(a, b, c, d, e) = P(a)P(b)P(c|b)P(d|a, c)P(e|d)$

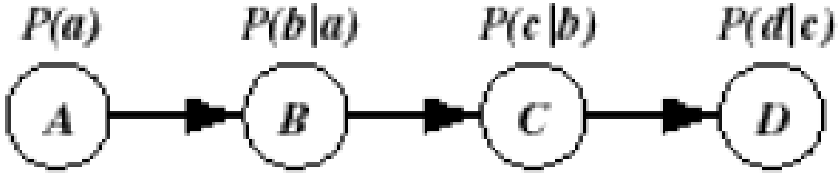


Figure 4: $P(a, b, c, d) = P(a)P(b|a)P(c|b)P(d|c)$

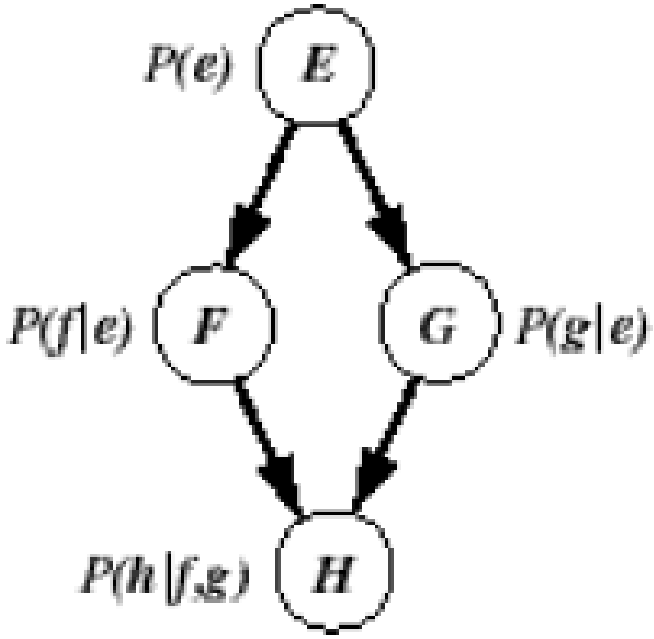


Figure 5: $P(e, f, g, h) = P(e)P(f|e)P(g|e)P(h|f, g)$

Bayesian Network Examples

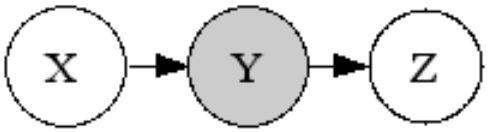


Figure 6: When y is given, x and z are conditionally independent. Think of x as the past, y as the present, and z as the future.

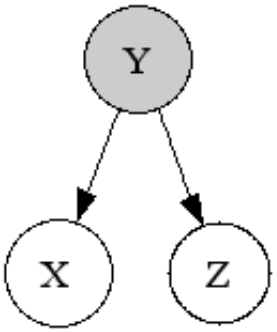


Figure 7: When y is given, x and z are conditionally independent. Think of y as the common cause of the two independent effects x and z.

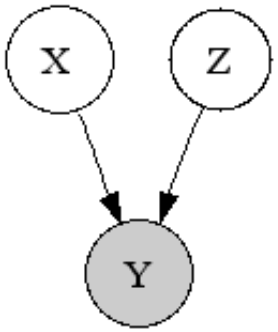


Figure 8: x and z are marginally independent, but when y is given, they are conditionally dependent. This is called explaining away.

Bayesian Network Examples

- You have a new burglar alarm installed at home.
- It is fairly reliable at detecting burglary, but also sometimes responds to minor earthquakes.
- You have two neighbors, Ali and Veli, who promised to call you at work when they hear the alarm.
- Ali always calls when he hears the alarm, but sometimes confuses telephone ringing with the alarm and calls too.
- Veli likes loud music and sometimes misses the alarm.
- Given the evidence of who has or has not called, we would like to estimate the probability of a burglary.

Bayesian Network Examples

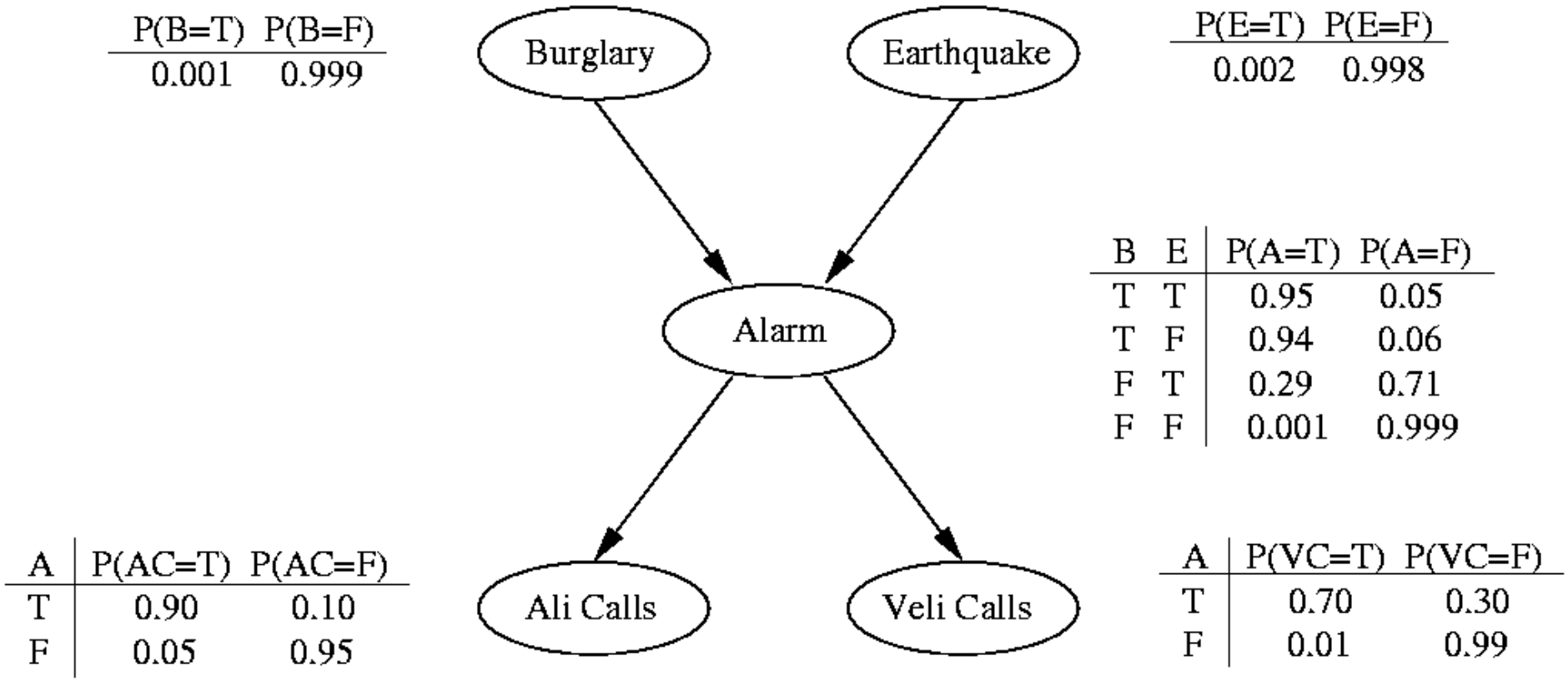


Figure 9: The Bayesian network for the burglar alarm example. Burglary (B) and earthquake (E) directly affect the probability of the alarm (A) going off, but whether or not Ali calls (AC) or Veli calls (VC) depends only on the alarm. (Russell and Norvig, Artificial Intelligence: A Modern Approach, 1995)

Bayesian Network Examples

- What is the probability that the alarm has sounded but neither a burglary nor an earthquake has occurred, and both Ali and Veli call?

$$
\begin{aligned}
&P(AC, VC, A, \neg B, \neg E) \\
&\quad = P(AC|A)P(VC|A)P(A|\neg B, \neg E)P(\neg B)P(\neg E) \\
&\quad = 0.90 \times 0.70 \times 0.001 \times 0.999 \times 0.998 \\
&\quad = 0.00062
\end{aligned}
$$

(capital letters represent variables having the value true, and $\neg$ represents negation)

Bayesian Network Examples

- What is the probability that there is a burglary given that Ali calls?

$$
\begin{aligned}
P(B|AC) &= \frac{P(B, AC)}{P(AC)} \\
&= \frac{\sum_{vc}\sum_{a}\sum_{e} P(AC|a)P(vc|a)P(a|B,e)P(B)P(e)}{P(B, AC) + P(\neg B, AC)} \\
&= \frac{0.00084632}{0.00084632 + 0.0513} \\
&= 0.0162
\end{aligned}
$$

- What about if Veli also calls right after Ali hangs up?

$$P(B|AC, VC) = \frac{P(B, AC, VC)}{P(AC, VC)} = 0.29$$

Bayesian Network Examples

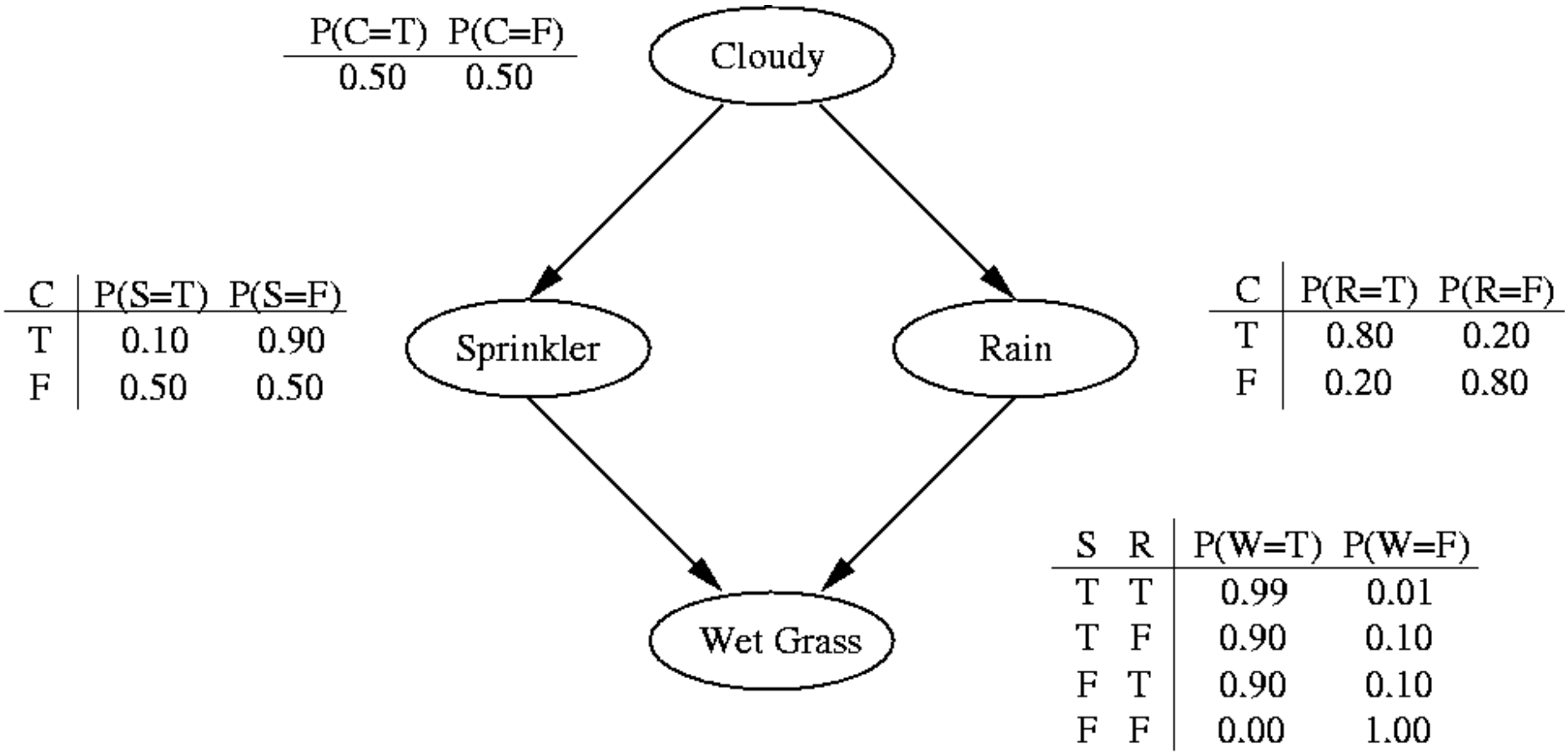


Figure 10: Another Bayesian network example. The event that the grass being wet (W = true) has two possible causes: either the water sprinkler was on (S = true) or it rained (R = true). (Russell and Norvig, Artificial Intelligence: A Modern Approach, 1995)

Bayesian Network Examples

- Suppose we observe the fact that the grass is wet. There are two possible causes for this: either it rained, or the sprinkler was on. Which one is more likely?

$$P(S|W) = \frac{P(S,W)}{P(W)} = \frac{0.2781}{0.6471} = 0.430$$

$$P(R|W) = \frac{P(R,W)}{P(W)} = \frac{0.4581}{0.6471} = 0.708$$

- We see that it is more likely that the grass is wet because it rained.

Applications of Bayesian Networks

- Example applications include:
  - Machine learning
  - Statistics
  - Computer vision
  - Natural language processing
  - Speech recognition
  - Error-control codes
  - Bioinformatics
  - Medical diagnosis
  - Weather forecasting
- Example systems include:
  - PATHFINDER medical diagnosis system at Stanford
  - Microsoft Office assistant and troubleshooters
  - Space shuttle monitoring system at NASA Mission Control Center in Houston

Two Fundamental Problems for BNs

- *Evaluation (inference) problem:* Given the model and the values of the observed variables, estimate the values of the hidden nodes.
- *Learning problem:* Given training data and prior information (e.g., expert knowledge, causal relationships), estimate the network structure, or the parameters of the probability distributions, or both.

Bayesian Network Evaluation Problem

- If we observe the “leaves” and try to infer the values of the hidden causes, this is called diagnosis, or bottom-up reasoning.
- If we observe the “roots” and try to predict the effects, this is called prediction, or top-down reasoning.
- *Exact inference* is an NP-hard problem because the number of terms in the summations (integrals) for discrete (continuous) variables grows exponentially with increasing number of variables.

Bayesian Network Evaluation Problem

- Some restricted classes of networks, namely the singly connected networks where there is no more than one path between any two nodes, can be efficiently solved in time linear in the number of nodes.
- There are also clustering algorithms that convert multiply connected networks to single connected ones.
- However, *approximate inference* methods such as
  - sampling (Monte Carlo) methods
  - variational methods
  - loopy belief propagation

  have to be used for most of the cases.

Bayesian Network Learning Problem

- The simplest situation is the one where the network structure is completely known (either specified by an expert or designed using causal relationships between the variables).
- Other situations with increasing complexity are: known structure but unobserved variables, unknown structure with observed variables, and unknown structure with unobserved variables.

Table 1: Four cases in Bayesian network learning.

	Observability	
Structure	Full	Partial
Known	Maximum Likelihood Estimation	EM (or gradient ascent)
Unknown	Search through model space	EM + search through model space

Known Structure, Full Observability

- The joint pdf of the variables with parameter set Θ is

$$p(x_1, \ldots, x_n|\boldsymbol{\Theta}) = \prod_{i=1}^{n} p(x_i|\boldsymbol{\pi_i}, \boldsymbol{\theta_i})$$

  where $\boldsymbol{\theta_i}$ is the vector of parameters for the conditional distribution of x_i and $\boldsymbol{\Theta} = (\boldsymbol{\theta_1}, \ldots, \boldsymbol{\theta_n})$.

- Given training data $\mathcal{X} = \{\mathbf{x_1}, \ldots, \mathbf{x_m}\}$ where $\mathbf{x_l} = (x_{l1}, \ldots, x_{ln})^T$, the log-likelihood of Θ with respect to $\mathcal{X}$ can be computed as

$$\log L(\boldsymbol{\Theta}|\mathcal{X}) = \sum_{l=1}^{m} \sum_{i=1}^{n} \log p(x_{li}|\boldsymbol{\pi_i}, \boldsymbol{\theta_i}).$$

Known Structure, Full Observability

- The likelihood decomposes according to the structure of the network so we can compute the MLEs for each node independently.
- An alternative is to assign a prior probability density function $p(\boldsymbol{\theta_i})$ to each $\boldsymbol{\theta_i}$ and use the training data $\mathcal{X}$ to compute the posterior distribution $p(\boldsymbol{\theta_i}|\mathcal{X})$ and the Bayes estimate $E_{p(\boldsymbol{\theta_i}|\mathcal{X})}[\boldsymbol{\theta_i}]$.
- We will study the special case of discrete variables with discrete parents.

Known Structure, Full Observability

- Let each discrete variable x_i have r_i possible values (states) with probabilities

$$p(x_i = k | \boldsymbol{\pi_i} = j, \boldsymbol{\theta_i}) = \boldsymbol{\theta}_{ijk} > 0$$

where $k \in \{1, \ldots, r_i\}$, j is the state of x_i's parents and $\boldsymbol{\theta_i} = \{\boldsymbol{\theta}_{ijk}\}$ specifies the parameters of the multinomial distribution for every combination of $\boldsymbol{\pi_i}$.

- Given $\mathcal{X}$, the MLE of $\boldsymbol{\theta}_{ijk}$ can be computed as

$$\hat{\boldsymbol{\theta}}_{ijk} = \frac{N_{ijk}}{N_{ij}}$$

where N_{ijk} is the number of cases in $\mathcal{X}$ in which $x_i = k$ and $\boldsymbol{\pi_i} = j$, and $N_{ij} = \sum_{k=1}^{r_i} N_{ijk}$.

Known Structure, Full Observability

- Thus, learning just amounts to counting (in the case of multinomial distributions).
- For example, to compute the estimate for the W node in the water sprinkler example, we need to count

$$\#(W = T, S = T, R = T),$$
$$\#(W = T, S = T, R = F),$$
$$\#(W = T, S = F, R = T),$$
$$\vdots$$
$$\#(W = F, S = F, R = F).$$

Known Structure, Full Observability

- Note that, if a particular event is not seen, it will be assigned a probability of 0.
- We can avoid this using the Bayes estimate with a Dirichlet$(\alpha_{ij1}, \ldots, \alpha_{ijr_i})$ prior (the conjugate prior for the multinomial) that gives

$$\hat{\boldsymbol{\theta}}_{\boldsymbol{i}jk} = \frac{\alpha_{ijk} + N_{ijk}}{\alpha_{ij} + N_{ij}}$$

  where $\alpha_{ij} = \sum_{k=1}^{r_i} \alpha_{ijk}$ and $N_{ij} = \sum_{k=1}^{r_i} N_{ijk}$ as before.
- α_{ij} is sometimes called the equivalent sample size for the Dirichlet distribution.

Naive Bayesian Network

- When the dependencies among the features are unknown, we generally proceed with the simplest assumption that the features are conditionally independent given the class.
- This corresponds to the *naive Bayesian network* that gives the class-conditional probabilities

$$p(x_1, \ldots, x_n|w) = \prod_{i=1}^{n} p(x_i|w).$$

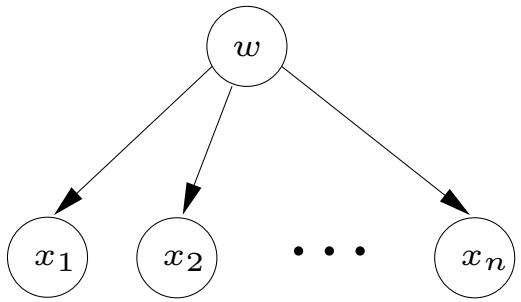


Figure 11: Naive Bayesian network structure. It looks like a very simple model but it often works quite well in practice.