Probabilistic Reasoning System

Informatics Engineering Study Program
School of Electrical Engineering and Informatics
Institute of Technology Bandung

Contents

Review

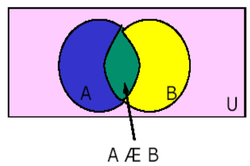
1. Deterministic
   - Problem Solving agent
   - Knowledge Based agent
2. Learning
3. Non Deterministic
   - Probabilistic & Bayes' Rule

Probability

- Logic represents uncertainty by disjunction
- But, cannot tell us how likely the different conditions are
- Probability theory provides a quantitative way of encoding likelihood

Axioms of Probability

- Universe of atomic events (like interpretations in logic).
- Events are sets of atomic events
- P: events → [0,1]
  - P(true) = 1 = P(U)
  - P(false) = 0 = P()
  - P(A v B) = P(A) +



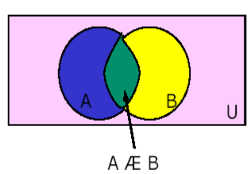
- Bayesian → Subjectivist
  - Probability is a model of your degree of belief

Examples of Human Probability Reasoning

Jane is from Berkeley. She was active in anti-war protests in the 60's. She lives in a commune.

- Which is more probable?

1. Jane is a bank teller
2. Jane is a feminist bank teller



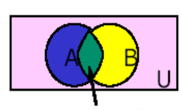
Random Variables

- Random variables
  - Function: discrete domain → [0, 1]
  - Sums to 1 over the domain
    - Raining is a propositional random variable
    - Raining(true) = 0.2
      - P(Raining = true) = 0.2
    - Raining(false) = 0.8
      - P(Raining = false) = 0.8
- Joint distribution
  - Probability assignment to all combinations of values of random variables

Joint Probability Distribution

	Toothache	¬Toothache
Cavity	0.04	0.06
¬Cavity	0.01	0.89

- The sum of the entries in this table has to be 1
- Given this table, one can answer all the probability questions about this domain
- P(cavity) = 0.1 [add elements of cavity row]
- P(toothache) = 0.05 [add elements of toothache column]
- P(A | B) = P(A ∧ B)/P(B) [prob of A when U is limited to B]
- P(cavity | toothache) = 0.04/0.05 = 0.8



A Æ B

Bayes' Rule

- Bayes' Rule
  - P(A | B) = P(B | A) P(A) / P(B)
  - P(disease | symptom)
    = P(symptom | disease) P(disease)/ P(symptom)
  - Imagine
    - disease = BSE
    - symptom = paralysis
    - P(disease | symptom) is different in England vs US
    - P(symptom | disease) should be the same
    - It is more useful to learn P(symptom | disease)
- Conditioning
  - $P(A) = P(A \mid B)\, P(B) + P(A \mid \neg B)\, P(\neg B)$
    $= P(A \wedge B) + P(A \wedge \neg B)$

Independence

- A and B are independent iff
  - $P(A \wedge B) = P(A) \cdot P(B)$
  - P(A | B) = P(A)
  - P(B | A) = P(B)
- Independence is essential for efficient probabilistic reasoning
- A and B are conditionally independent given C iff
  - P(A | B, C) = P(A | C)
  - P(B | A, C) = P(B | C)
  - $P(A \wedge B \mid C) = P(A \mid C) \cdot P(B \mid C)$

Example of Conditional Independence

- X is late (X)
- Traffic Jam (T)
- Y is late (Y)
- None of these propositions are independent of one other
- X and Y are conditionally independent given T

- Radio plays (R)
- Battery is dead (B)
- Starter turns over (S)
- None of these propositions are independent of one another
- R and S are conditionally independent given B

Bayesian Network

- To do probabilistic reasoning, you need to know the joint probability distribution

- But, in a domain with N binary propositional variables (2 possibilities value), one needs 2^N numbers to specify the joint probability distribution

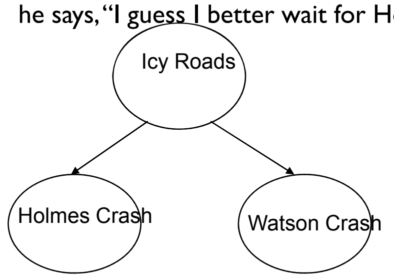
- We want to exploit independences in the domain

- Two components: structure and numerical parameters

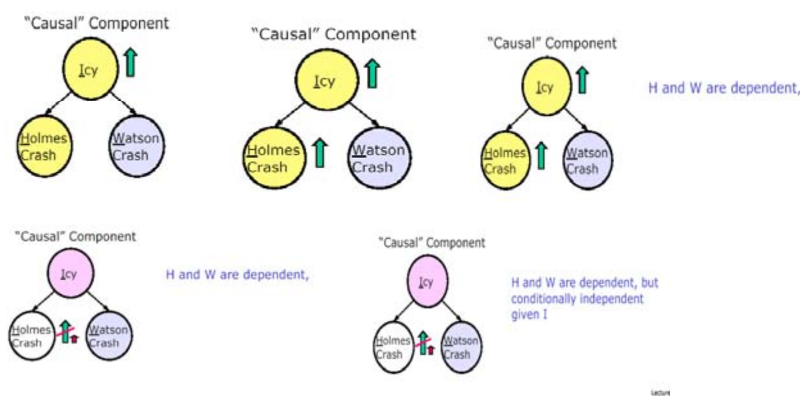
Example of Bayesian Network

- Icy Roads

Inspector Smith is waiting for Holmes and Watson, who are driving (separately) to meet him. It is winter. His secretary tells him that Watson has had an accident. He says, "It must be that the roads are icy. I bet that Holmes will have an accident too. I should go to lunch." But, his secretary says, "No, the roads are not icy, look at the window." So, he says, "I guess I better wait for Holmes."



Icy Roads (con't)

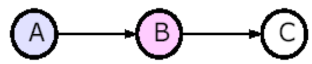


Connections

A = battery dead

B = car won't start

C = car won't move



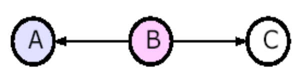
- Forward Serial Connection
  - Knowing about A will tell us something about C
  - But if we know B then knowing about A will not tell us anything about C
- Backward Serial Connection
  - Knowing about C will tell us something about A
  - But if we know B then knowing about C will not tell us anything about A

Connections (con't)

A = Watson Crash

B = Icy

C = Holmes Crash



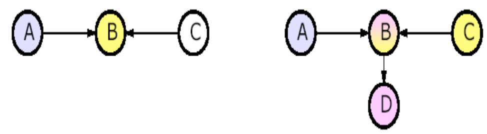
- Diverging Connection
  - Knowing about A will tell us something about C
  - Knowing about C will tell us something about A
  - But if we know B then knowing about A will not tell us anything new about C, and vice versa

Connections (con't)

A = Bacterial Infection

B = Sore Throat

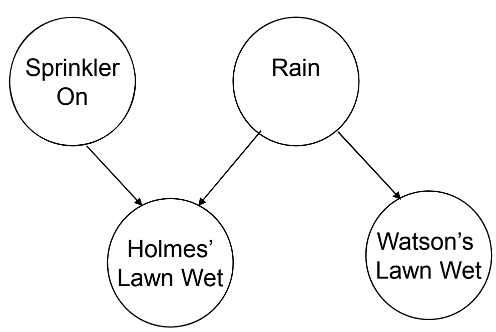
C = Viral Infection



- Converging Connection
  - Without knowing B finding A does not tell us something about C
  - If we see evidence for B, then A and C becomes dependent (potential for "explaining away"). If we find bacteria in patient with a sore throat, then viral infection is less likely.

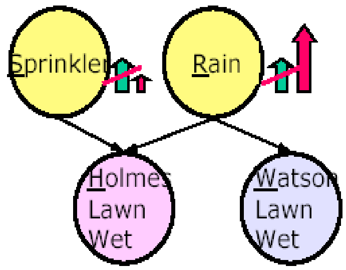
Connections (con't)

Holmes and Watson have moved to LA. He wakes up to find his lawn wet. He wonders if it has rained or if he left his sprinkler on. He looks at his neighbor Watson's lawn and he sees it is wet too. So, he concludes it must have rained.



Connections (con't)

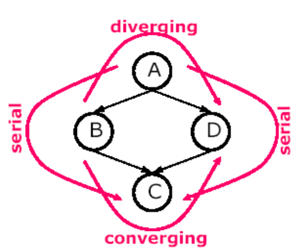
Holmes and Watson have moved to LA. He wakes up to find his lawn wet. He wonders if it has rained or if he left his sprinkler on. He looks at his neighbor Watson's lawn and he sees it is wet too. So, he concludes it must have rained.



Given W, P(R) goes up and P(S) goes down – "explaining away"

D Separation

- Two variables A and B are d-separated iff for every path between them, there is an intermediate variable V such that either
  - The connection is serial or diverging and V is known
  - The connection is converging and neither V nor any descendant is instantiated
  - Two variables are d-connected iff they are not d-separated



- A-B-C: serial, blocked when B is known, connected otherwise
- A-D-C: serial, blocked when D is known, connected otherwise
- B-A-D: diverging, blocked when A is known, connected otherwise
- B-C-D: converging, blocked when C has no evidence, connected otherwise

Bayesian (Belief) Network

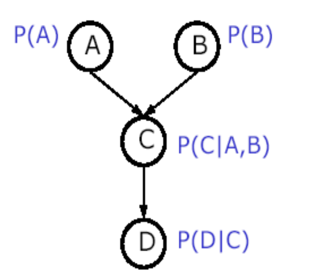
- Set of variables, each has a finite set of values
- Set of directed arcs between them forming acyclic graph
- Every node A, with parents B1, …, Bn, has P(A |B1,…,Bn) specified

Theorem: If A and B are d-separated given evidence e, then P(A | e) = P(A | B, e)

Chain Rule

- Variables: $V_1, \ldots, V_n$
- Values: $v_1, \ldots, v_n$
- $P(V_1=v_1, V_2=v_2, \ldots, V_n=v_n) = \prod_i P(V_i=v_i \mid \text{parents}(V_i))$

P(ABCD) = P(A=true, B=true, C=true, D=true)



P(ABCD) =

P(D|ABC)P(ABC) =

P(D|C) P(ABC) =

P(D|C) P(C|AB) P(AB) =

P(D|C) P(C|AB) P(A)P(B)

A d-separated from D given C

B d-separated from D given C

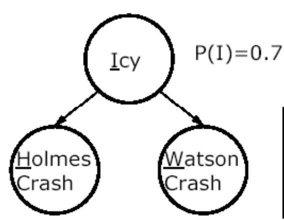
A d-separated from B

Inference in Bayesian Network

- Exact inference
- Approximate inference
- Given a Bayesian Network, what questions might we want to ask?
  - Conditional probability query: P(x | e)
  - Maximum a posteriori probability:
    What value of x maximizes P(x|e) ?
- General question: What's the whole probability distribution over variable X given evidence e, P(X | e)?

Icy Roads with Numbers

	P(H| I)
I	0.8
¬I	0.1

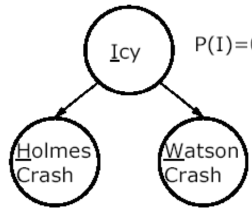


	P(W| I)
I	0.8
¬I	0.1

Probability that Watson Crashes:

$$P(W) = P(W| I)\, P(I) + P(W| \neg I)\, P(\neg I)$$
$$= 0.8 \cdot 0.7 + 0.1 \cdot 0.3$$
$$= 0.56 + 0.03$$
$$= 0.59$$

Icy Roads with Numbers (con’t)

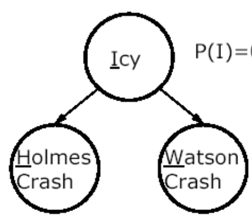


	P(H| I)
I	0.8
¬I	0.1

	P(W| I)
I	0.8
¬I	0.1

Probability of Icy given Watson (Bayes’ Rule):
P(I | W) = P(W | I) P(I) / P(W)
= 0.8 · 0.7 / 0.59
= 0.95
We started with P(I) = 0.7; knowing that Watson crashed raised the probability to 0.95

Icy Roads with Numbers (con’t)

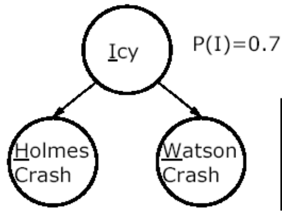


	P(H| I)
I	0.8
¬I	0.1

	P(W| I)
I	0.8
¬I	0.1

Probability of Holmes given Watson :
P(H|W) = P(H|W,I)P(I|W) + P(H|W,¬I) P(¬I| W)
= P(H|I)P(I|W) + P(H|¬I) P(¬I| W)
= 0.8 · 0.95 + 0.1 · 0.05
= 0.765
We started with P(H) = 0.59; knowing that Watson crashed raised the probability to 0.765

Icy Roads with Numbers (con't)



	P(H| I)
I	0.8
¬I	0.1

	P(W| I)
I	0.8
¬I	0.1

Probability of Holmes given Icy and Watson :
P(H|W, ¬I) = P(H|¬I) = 0.1

H and W are d-separated given I, so H and W are conditionally independent given I

Where do Bayesian Networks Come From?

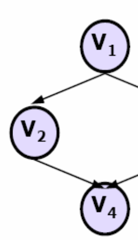
- Human Expert
  - Encoding rules obtained from expert
  - Very difficult in getting reliable probability estimates
- Learning From Data
  - Try to estimate the joint probability distribution
  - Looking for models that encode conditional independencies in data
  - Four cases →
    - Structure known or unknown
    - All variables are observable or some observable
- Combination of Both

Case 1: Structure is given

- Given nodes and arcs of a Bayesian network with m nodes
- Given a data set D = {<$v_1^1,\ldots,v_m^1$>,…, <$v_1^k,\ldots,v_m^k$>}
- Elements of D are assumed to be independent given M
- Find the model M that maximizes Pr(D|M)
- Known as the maximum likelihood model
- Humans are good at providing structure, data is good at providing numbers

Case 1: Estimates the Conditional Probability

- Use counts and definition of conditional probability
- Initializing all counters to 1 avoids 0 probabilities and converges on the maximum likelihood estimate



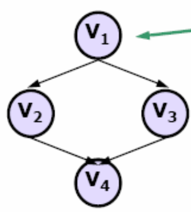
$$P(V_1) \approx \frac{\#(V_1{=}true)}{k}$$

$$P(V_3|V_1) \approx \frac{\#(V_3{=}true \wedge V_1{=}true)}{\#(V_1{=}true)}$$

$$P(V_3|\neg V_1) \approx \frac{\#(V_3{=}true \wedge V_1{=}false)}{\#(V_1{=}false)}$$

Case 1: Estimates the Conditional Probability

- Use counts and definition of conditional probability
- Initializing all counters to 1 avoids 0 probabilities and converges on the maximum likelihood estimate



$$P(V_1) \approx \frac{\#(V_1{=}true)+1}{k+2}$$

$$P(V_3|V_1) \approx \frac{\#(V_3{=}true \wedge V_1{=}true)+1}{\#(V_1{=}true)+2}$$

$$P(V_3|\neg V_1) \approx \frac{\#(V_3{=}true \wedge V_1{=}false)+1}{\#(V_1{=}false)+2}$$

generally, the number of possible values of the variable on the left of the bar

Case 2: Learning the Structure

- No direct way to find the best structure
- Too many to enumerate them all
- Start with some initial structure
- Do local search in structure space
  - neighborhood: add, delete, or reverse an arc
  - maintain no directed cycles
  - once you pick a structure, compute maximum-likelihood parameters, and then calculate the score of the model

Case 2: Learning the Structure

- no arcs
- choose random ordering V1 … Vn
  - variable Vi has all parents V1 … Vn-1
  - variable Vi has parents randomly chosen from V1 … Vn-1
- best tree network (can be computed in polynomial time)
  - compute pairwise mutual information between every pair of variables
  - find maximum-weight spanning tree

THANK YOU