Bayes
Quiz 2 Review

Russell and Norvig: Chapter 13
CSE 240: Winter 2023
Lecture 12

Announcements

- Quiz 2 opens on today after class
  - No time limit (should take ~30 minutes)
  - Deadline: 5pm on Friday.
  - Mixed multiple choice + open ended answers.
  - Open book open note.
- Assignment 3 is out.
- Working on regrades

Agenda and Topics

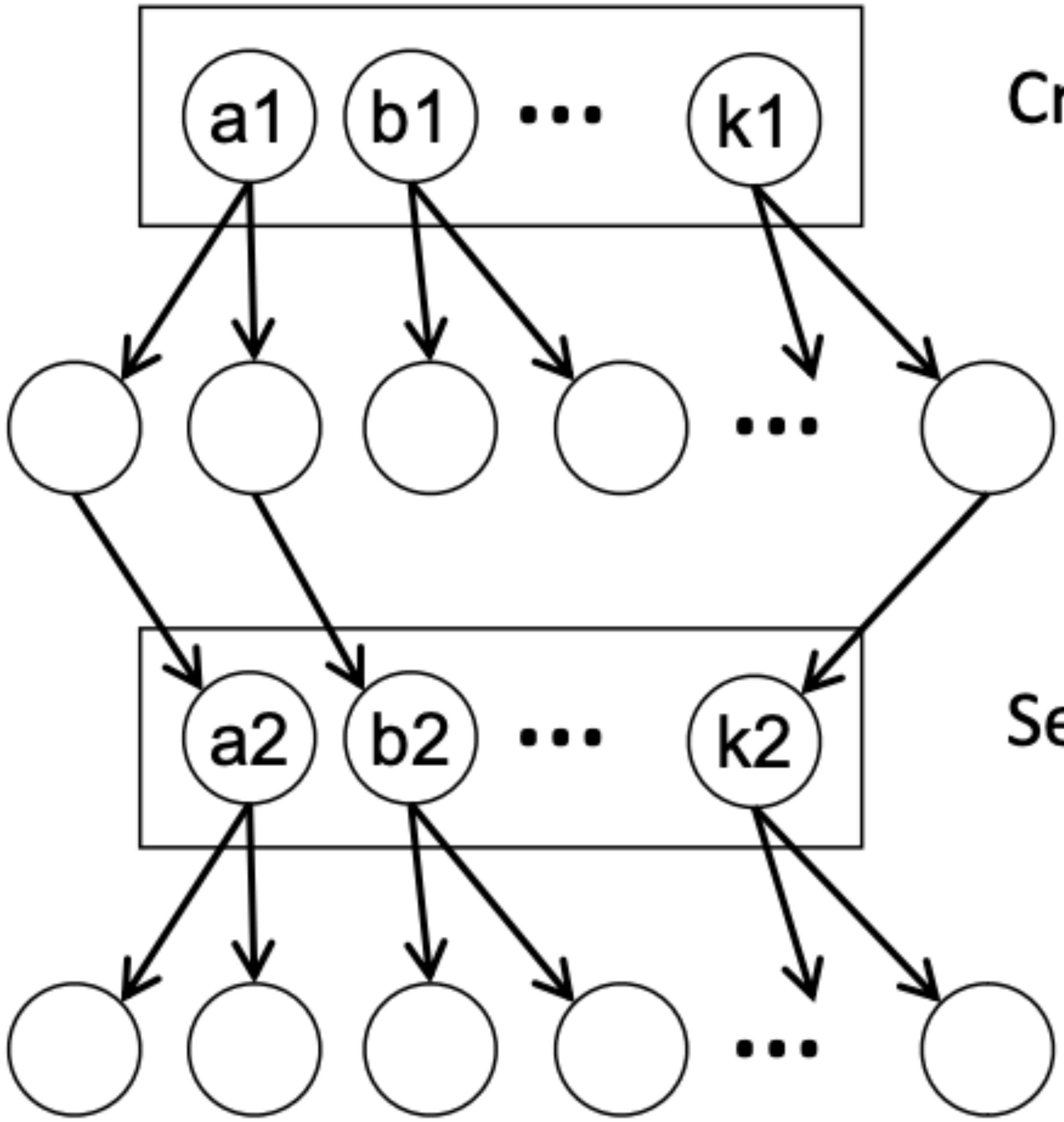
- Quiz/Assignment review
  - Local Beam search (pseudocode QA)
  - CSP Tree-structuring question
  - Topics
- Probability
  - Product rule, chain rule, Bayes rule
  - Inference
  - Independence and conditional independence

Quiz Review

Local Beam Search

- Keep track of *k* states rather than just one
- Start with *k* randomly generated states
- At each iteration, all the successors of all *k* states are generated
- If any one is a goal state, stop; else select the *k* best successors from the complete list and repeat.

Local Beam Search



Create *k* random initial states

Generate their children

Select the *k* best children

Repeat indefinitely…

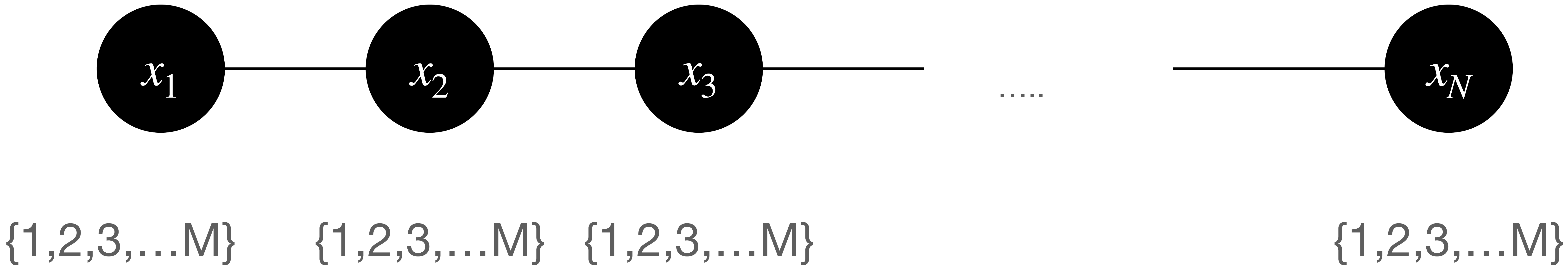
Pseudocode

```
local_beam_search(objective, bounds, step_size, k, n_iterations):
  beam = [ ]
  while len(beam) < k:
    point <- random
    add point to beam
  for iter in n_iterations:
     pool <- generate_neighbors for each beam
     check that each pool is in bounds (if not, drop it)
     sort the pool by the best objective
     set the beam to the k best members of the pool
  return the best member of the beam
```

Quiz 2 Overview

- Format
    - 8 questions
        - 3 True/False
        - 2 multiple choice
        - 3 *short* answer
- Questions
    - Minimax/Expectimax utility function
    - Evaluation function versus depth
    - Search problem types
    - Hill climbing pros/cons
    - Simulated Annealing properties
    - Solving a CSP
    - Tree-structured CSP (CE 11)
    - Tree-structured CSP algorithm and runtime

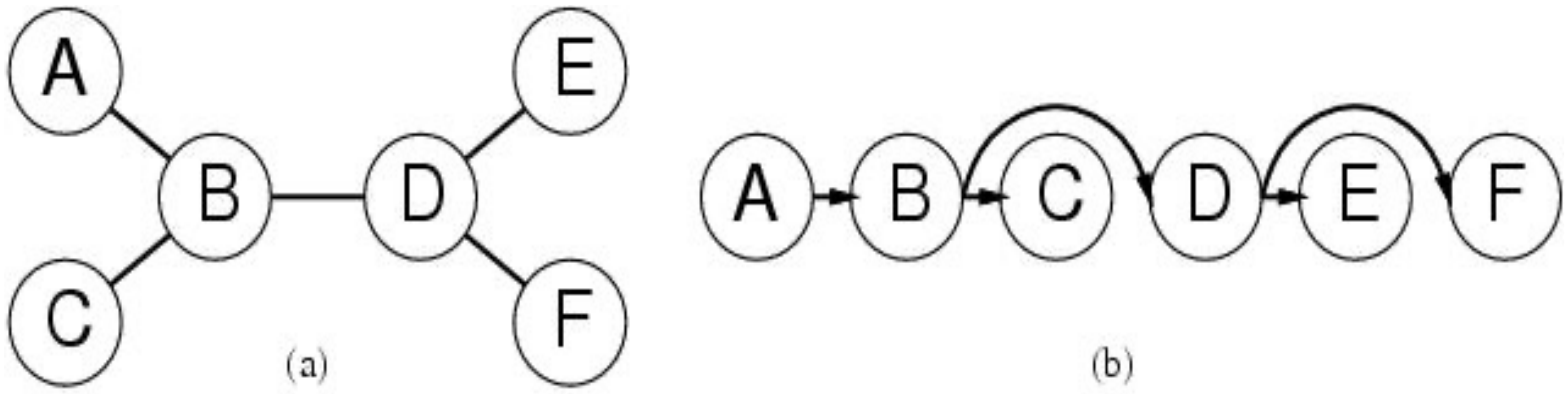
CSP Example



$X_i < X_{i+1}$

N=M

Tree-structured CSPs



- Any tree-structured CSP can be solved in time linear in the number of variables.
  - Choose a variable as root, order variables from root to leaves such that every node's parent precedes it in the ordering. (label var from X_1 to X_n)
  - For j from n down to 2, apply REMOVE-INCONSISTENT-VALUES(Parent(X_j),X_j)
  - For j from 1 to n assign X_j consistently with Parent(X_j)

Runtime: $O(nd^2)$ (why?)

Enforcing Arc Consistency in a CSP

```
function AC-3( csp) returns the CSP, possibly with reduced domains
   inputs: csp, a binary CSP with variables {X_1, X_2, ..., X_n}
   local variables: queue, a queue of arcs, initially all the arcs in csp

   while queue is not empty do
      (X_i, X_j) <- REMOVE-FIRST(queue)
      if REMOVE-INCONSISTENT-VALUES(X_i, X_j) then
         for each X_k in NEIGHBORS[X_i] do
            add (X_k, X_i) to queue
```

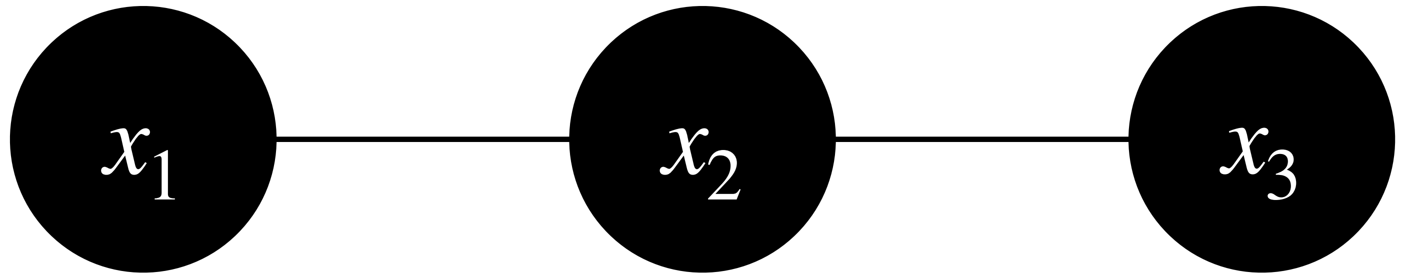
```
function REMOVE-INCONSISTENT-VALUES( X_i, X_j) returns true iff succeeds
   removed <- false
   for each x in DOMAIN[X_i] do
      if no value y in DOMAIN[X_j] allows (x,y) to satisfy the constraint X_i <-> X_j
         then delete x from DOMAIN[X_i]; removed <- true
   return removed
```

- Runtime: $O(n^2d^3)$

CSP Example

$X_i < X_{i+1}$

N=M



{1,2,3}	{1,2,3}	{1,2,3}
	{1,2}	{1,2,3}
{1}	{1,2}	{1,2,3}

Probability

Conditional Distributions

- Conditional distributions are probability distributions over some variables given fixed values of others

Conditional Distributions

$$P(W|T)$$

$$P(W|T = hot)$$

W	P
sun	0.8
rain	0.2

$$P(W|T = cold)$$

W	P
sun	0.4
rain	0.6

Joint Distribution

$$P(T, W)$$

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

Question: Conditional Probabilities

- $P(\text{ hot} \mid \text{sun})$?
- $P(\text{ cold} \mid \text{sun})$?
- $P(\text{rain} \mid \text{hot})$?

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

Normalization Trick

- A trick to get a whole conditional distribution at once:
  - Select the joint probabilities matching the evidence
  - Normalize the selection (make it sum to one)

$P(T, W)$

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

Select

$P(T, r)$

T	W	P
hot	rain	0.1
cold	rain	0.3

Normalize

$P(T|r)$

T	P
hot	0.25
cold	0.75

- Why does this work? Sum of selection is P(evidence)! (P(r), here)

$$P(x_1|x_2) = \frac{P(x_1, x_2)}{P(x_2)} = \frac{P(x_1, x_2)}{\sum_{x_1} P(x_1, x_2)}$$

Probabilistic Inference

- Probabilistic inference: compute a desired probability from other known probabilities (e.g. conditional from joint)

- We generally compute conditional probabilities
  - P(on time | no reported accidents) = 0.90
  - These represent the agent's beliefs given the evidence

- Probabilities change with new evidence:
  - P(on time | no accidents, 5 a.m.) = 0.95
  - P(on time | no accidents, 5 a.m., raining) = 0.80
  - Observing new evidence causes beliefs to be updated

Inference by Enumeration

- P(sun)?
- 0.3+0.1+0.1+0.15= 0.65
- P(sun | winter)?

P(w) = 0.5
P(s,w) = 0.25
P(s|w) = 0.5

- P(sun | winter, hot)?

0.1/0.15 = 0.667

S	T	W	P
summer	hot	sun	0.30
summer	hot	rain	0.05
summer	cold	sun	0.10
summer	cold	rain	0.05
winter	hot	sun	0.10
winter	hot	rain	0.05
winter	cold	sun	0.15
winter	cold	rain	0.20

Inference by Enumeration

- General case:
  - Evidence variables: $E_1 \dots E_k = e_1 \dots e_k$
  - Query* variable: Q
  - Hidden variables: $H_1 \dots H_r$

$\left.\begin{matrix} E_1 \dots E_k = e_1 \dots e_k \\ Q \\ H_1 \dots H_r \end{matrix}\right\} X_1, X_2, \dots X_n$

All variables

- We want: $P(Q|e_1 \dots e_k)$

- First, select the entries consistent with the evidence
- Second, sum out H to get joint of Query and evidence:

$$P(Q, e_1 \dots e_k) = \sum_{h_1 \dots h_r} \underbrace{P(Q, h_1 \dots h_r, e_1 \dots e_k)}_{X_1, X_2, \dots X_n}$$

- Finally, normalize the remaining entries to conditionalize

- Obvious problems:
  - Worst-case time complexity $O(d^n)$
  - Space complexity $O(d^n)$ to store the joint distribution

* Works fine with multiple query variables, too

The Product Rule

- Sometimes have conditional distributions but want the joint

$$P(x|y) = \frac{P(x,y)}{P(y)} \quad \Leftrightarrow \quad P(x,y) = P(x|y)P(y)$$

- Example:

$P(W)$

W	P
sun	0.8
rain	0.2

$P(D|W)$

D	W	P
wet	sun	0.1
dry	sun	0.9
wet	rain	0.7
dry	rain	0.3

$\Leftrightarrow$

$P(D,W)$

D	W	P
wet	sun	0.08
dry	sun	0.72
wet	rain	0.14
dry	rain	0.06

The Chain Rule

- More generally, can always write any joint distribution as an incremental product of conditional distributions

$$P(x_1, x_2, x_3) = P(x_1)P(x_2|x_1)P(x_3|x_1, x_2)$$

$$P(x_1, x_2, \ldots x_n) = \prod_i P(x_i|x_1 \ldots x_{i-1})$$

Bayes Rule

Bayes' Rule

- Two ways to factor a joint distribution over two variables:

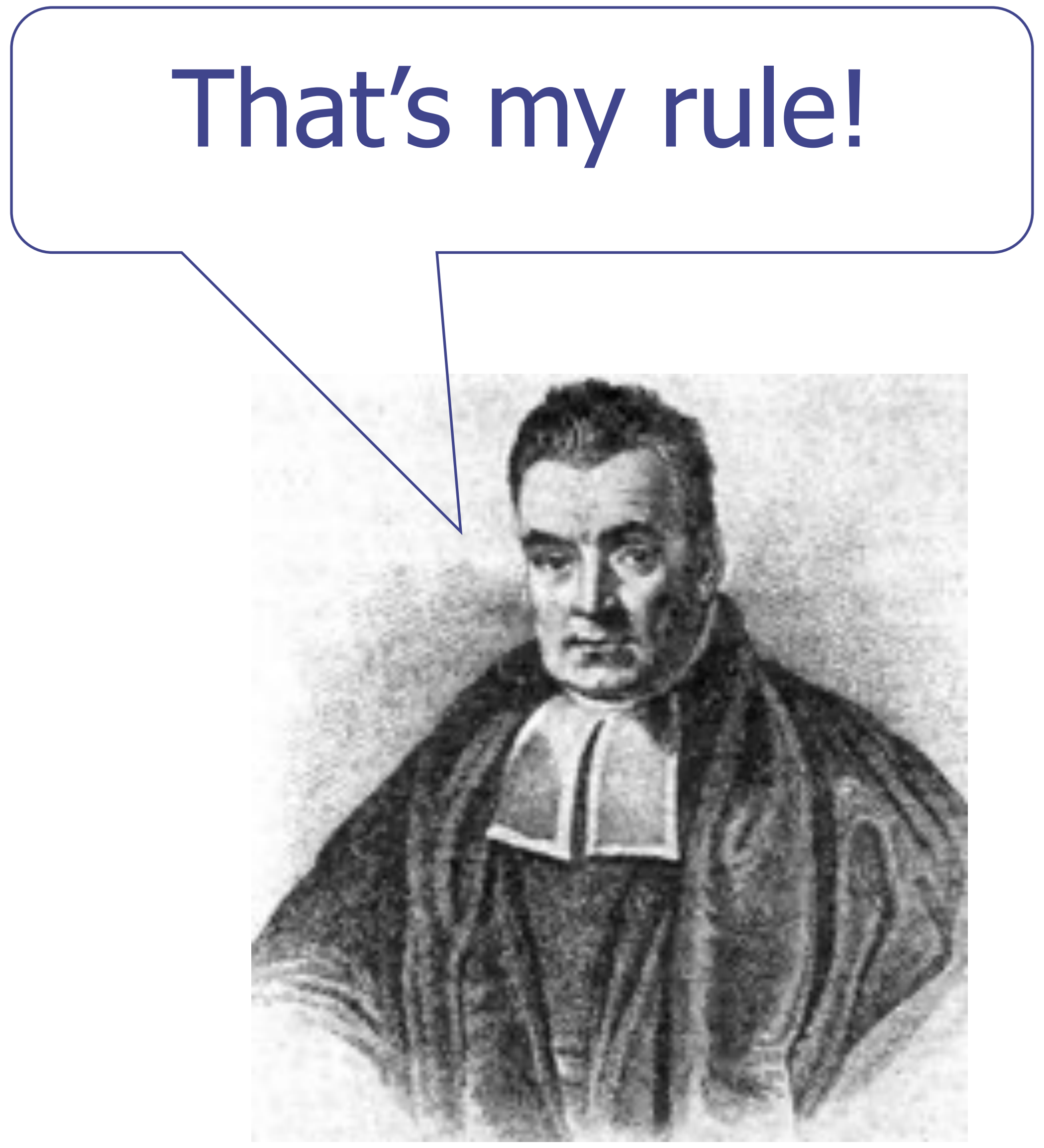
$$P(x, y) = P(x|y)P(y) = P(y|x)P(x)$$

- Dividing, we get:

$$P(x|y) = \frac{P(y|x)}{P(y)}P(x)$$

- Why is this at all helpful?
  - Lets us build one conditional from its reverse
  - Often one conditional is tricky but the other one is simple
  - Foundation of many systems

- In the running for most important AI equation!



Inference with Bayes' Rule

- Example: Diagnostic probability from causal probability:

$$P(\text{Cause}|\text{Effect}) = \frac{P(\text{Effect}|\text{Cause})P(\text{Cause})}{P(\text{Effect})}$$

- Example:
  - m is meningitis, s is stiff neck

$$\left.\begin{aligned} P(s|m) &= 0.8 \\ P(m) &= 0.0001 \\ P(s) &= 0.1 \end{aligned}\right\} \text{Example givens}$$

$$P(m|s) = \frac{P(s|m)P(m)}{P(s)} = \frac{0.8 \times 0.0001}{0.1} = 0.0008$$

  - Note: posterior probability of meningitis still very small

CE 12: Bayes' Rule

Given

$P(W)$

R	P
sun	0.8
rain	0.2

$P(D|W)$

D	W	P
wet	sun	0.1
dry	sun	0.9
wet	rain	0.7
dry	rain	0.3

What is P(W | dry)?

Independence

Independent Random Variables

- Two variables X and Y are **independent** if
  - $P(X = x|Y = y) = P(X = x)$ for all values x,y
  - That is, learning the values of Y does not change prediction of X

- If X and Y are independent then
  - $P(X,Y) = P(X|Y)P(Y) = P(X)P(Y)$
  - notation: I(X,Y) $X \perp\!\!\!\perp Y$

- In general, if $X_1,\dots,X_n$ are independent, then
  - $P(X_1,\dots,X_n)= P(X_1)\dots P(X_n)$
  - Requires O(n) parameters vs ?

Conditional Independence

Conditional Independence

- Unfortunately, random variables of interest are rarely independent of each other
- A more suitable notion is that of **conditional independence**
- Two variables X and Y are **conditionally independent** given Z if
  - P(x| y,z) = P(x|z) for all values x,y,z
  - That is, learning the values of Y does not change prediction of X once we know the value of Z
  - Equivalently, P(x, y|z) = P(x|z)P(y|z) for all values x,y,z
- notation: I(X ; Y | Z), $X \perp\!\!\!\perp Y | Z$

Examples

Examples where:
I(A; B) is not true,
But I(A; B | C) is true

- What about:
  - Cavity, Toothache, Catch
  - I(Toothache; Catch)?
    - i.e., P(Toothache,Catch) = P(Toothache)P(Catch)?
  - I(Toothache; Catch | Cavity)?
    - i.e., P(Toothache | Cavity, Catch) = P(Toothache | Cavity)?
- What about:
  - Rain, Traffic, Umbrella
  - I(Umbrella; Traffic)?
    - i.e., P(Umbrella,Traffic) = P(Umbrella)P(Traffic)?
  - I(Umbrella;Traffic | Rain)?
    - i.e., P(Umbrella | Rain, Traffic) = P(Umbrella | Rain)?
- What about:
  - Fire, Smoke, Alarm
  - I(Smoke; Alarm)?
    - i.e., P(Smoke,Alarm) = P(Smoke)P(Alarm)?
  - I(Smoke; Alarm | Fire)?
    - P(Smoke | Fire, Alarm) = P(Smoke | Fire)?

More Examples

Example where:
I(A; B) is true,
But I(A; B | C) is not

- What about:
  - Gas, Battery, Starts
  - I(Battery, Gas)
    - P(Battery|Gas) = P(Battery)
      Gas and Battery are independent
  - I(Battery, Gas | Starts)?
    - P(Battery|Gas,Starts) ≠ P(Battery|Starts)
      Gas and Battery are not independent given Starts
- What about GetGoodGrade, AttendClass, DoHomework?

The Chain Rule

$$P(X_1, X_2, \ldots X_n) = P(X_1)P(X_2|X_1)P(X_3|X_1, X_2)\ldots$$

- Trivial decomposition:

$$P(\text{Traffic}, \text{Rain}, \text{Umbrella}) = \\ P(\text{Rain})P(\text{Traffic}|\text{Rain})P(\text{Umbrella}|\text{Rain}, \text{Traffic})$$

- With assumption of conditional independence:

$$P(\text{Traffic}, \text{Rain}, \text{Umbrella}) = \\ P(\text{Rain})P(\text{Traffic}|\text{Rain})P(\text{Umbrella}|\text{Rain})$$

- Bayes' nets / graphical models help us express conditional independence assumptions

Summary and Next Time

- This week:
  - Finish CSPs
  - Probability and Bayes
- This week
  - Naives Bayes