CS 188: Artificial Intelligence
Spring 2010

Lecture 17: Bayes' Nets IV – Inference

3/16/2010

Pieter Abbeel – UC Berkeley

Many slides over this course adapted from Dan Klein, Stuart Russell, Andrew Moore

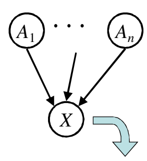
Announcements

- Assignments
  - W4 back today in lecture
  - Any assignments you have not picked up yet
    - In bin in 283 Soda [same room as for submission drop-off]
- Midterm
  - 3/18, 6-9pm, 0010 Evans --- no lecture on Thursday
  - We have posted practice midterms (and finals)
  - One note letter-size note sheet (two sides), non-programmable calculators [strongly encouraged to compose your own!]
  - Topics go through last Thursday
  - **Section this week: midterm review**

Bayes' Net Semantics

- Let's formalize the semantics of a Bayes' net
- A set of nodes, one per variable X
- A directed, acyclic graph
- A conditional distribution for each node
  - A collection of distributions over X, one for each combination of parents' values

    $P(X|a_1 \ldots a_n)$
  - CPT: conditional probability table
  - Description of a noisy "causal" process



$P(X|A_1 \ldots A_n)$

A Bayes net = Topology (graph) + Local Conditional Probabilities

Probabilities in BNs

- For all joint distributions, we have (chain rule):

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

- Bayes' nets implicitly encode joint distributions
  - As a product of local conditional distributions
  - To see what probability a BN gives to a full assignment, multiply all the relevant conditionals together:

$$P(x_1, x_2, \ldots x_n) = \prod_{i=1}^{n} P(x_i|\text{parents}(X_i))$$

- This lets us reconstruct any entry of the full joint
- Not every BN can represent every joint distribution
  - The topology enforces certain conditional independencies

Bayes Nets Representation Summary

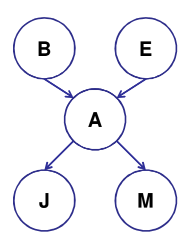
- Bayes nets compactly encode joint distributions
- Guaranteed independencies of distributions can be deduced from BN graph structure
- D-separation gives precise conditional independence guarantees from graph alone
- A Bayes' net's joint distribution may have further (conditional) independence that is not detectable until you inspect its specific distribution

Inference

- Inference: calculating some useful quantity from a joint probability distribution
- Examples:
  - Posterior probability:

    $P(Q|E_1 = e_1, \ldots E_k = e_k)$
  - Most likely explanation:

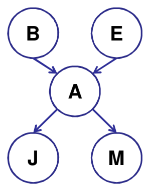
    $\text{argmax}_q \; P(Q = q|E_1 = e_1 \ldots)$



Inference by Enumeration

- Given unlimited time, inference in BNs is easy
- Recipe:
  - State the marginal probabilities you need
  - Figure out ALL the atomic probabilities you need
  - Calculate and combine them
- Example:

$$P(+b|+j,+m) = \frac{P(+b,+j,+m)}{P(+j,+m)}$$

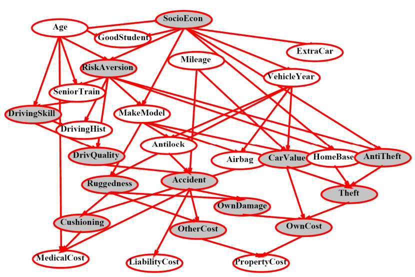


Example: Enumeration

- In this simple method, we only need the BN to synthesize the joint entries

$$P(+b,+j,+m) =$$
$$P(+b)P(+e)P(+a|+b,+e)P(+j|+a)P(+m|+a)+$$
$$P(+b)P(+e)P(-a|+b,+e)P(+j|-a)P(+m|-a)+$$
$$P(+b)P(-e)P(+a|+b,-e)P(+j|+a)P(+m|+a)+$$
$$P(+b)P(-e)P(-a|+b,-e)P(+j|-a)P(+m|-a)$$

Inference by Enumeration?



Variable Elimination

- Why is inference by enumeration so slow?
  - You join up the whole joint distribution before you sum out the hidden variables
  - You end up repeating a lot of work!
- Idea: interleave joining and marginalizing!
  - Called "Variable Elimination"
  - Still NP-hard, but usually much faster than inference by enumeration
- We'll need some new notation to define VE

Factor Zoo I

- Joint distribution: P(X,Y)
  - Entries P(x,y) for all x, y
  - Sums to 1

$P(T, W)$ f(W,T)

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

factor on T and W

- Selected joint: P(x,Y)
  - A slice of the joint distribution
  - Entries P(x,y) for fixed x, all y
  - Sums to P(x)

$P(cold, W)$

T	W	P
cold	sun	0.2
cold	rain	0.3

factor on cold and W

Factor Zoo II

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

- Family of conditionals: P(X |Y)
  - Multiple conditionals
  - Entries P(x | y) for all x, y
  - Sums to |Y|

$P(W|T)$

factor on W and T
f(W,T)

T	W	P
hot	sun	0.8
hot	rain	0.2
cold	sun	0.4
cold	rain	0.6

$P(W|hot)$ (hot rows), $P(W|cold)$ (cold rows)

- Single conditional: P(Y | x)
  - Entries P(y | x) for fixed x, all y
  - Sums to 1

$P(W|cold)$ f(W, cold)

T	W	P
cold	sun	0.4
cold	rain	0.6

Factor Zoo III

f(W|T)

- Specified family: P(y | X)
  - Entries P(y | x) for fixed y, but for all x
  - Sums to … who knows!

$P(rain|T)$ f(rain, T)

T	W	P
hot	rain	0.2
cold	rain	0.6

$P(rain|hot)$

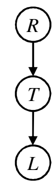
$P(rain|cold)$

- In general, when we write $P(Y_1 \ldots Y_N \mid X_1 \ldots X_M)$
  - It is a "factor," a multi-dimensional array
  - Its values are all $P(y_1 \ldots y_N \mid x_1 \ldots x_M)$
  - Any assigned X or Y is a dimension missing (selected) from the array

13

Example: Traffic Domain

- Random Variables
  - R: Raining
  - T: Traffic
  - L: Late for class!



Joins

Summations

$P(R)$ f(R)

+r	0.1
-r	0.9

$P(T|R)$ f(T|R)

+r	+t	0.8
+r	-t	0.2
-r	+t	0.1
-r	-t	0.9

$P(L|T)$ f(L|T)

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9

14

Variable Elimination Outline

- Track objects called factors
- Initial factors are local CPTs (one per node)

$P(R)$

+r	0.1
-r	0.9

$P(T|R)$

+r	+t	0.8
+r	-t	0.2
-r	+t	0.1
-r	-t	0.9

$P(L|T)$

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9

- Any known values are selected
  - E.g. if we know $L = +\ell$, the initial factors are

$P(R)$

+r	0.1
-r	0.9

$P(T|R)$

+r	+t	0.8
+r	-t	0.2
-r	+t	0.1
-r	-t	0.9

$P(+\ell|T)$

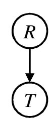
+t	+l	0.3
-t	+l	0.1

- VE: Alternately join factors and eliminate variables

15

Operation 1: Join Factors

- First basic operation: joining factors
- Combining factors:
  - Just like a database join
  - Get all factors over the joining variable
  - Build a new factor over the union of the variables involved
- Example: Join on R



$P(R) \times P(T|R) \Rightarrow P(R,T)$

+r	0.1
-r	0.9

1 entry per value for R

+r	+t	0.8
+r	-t	0.2
-r	+t	0.1
-r	-t	0.9

2 entries per value for R

+r	+t	0.08
+r	-t	0.02
-r	+t	0.09
-r	-t	0.81

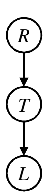
2 entries per value for R in resulting table

  - Computation for each entry: pointwise products

$$\forall r, t: \quad P(r,t) = P(r) \cdot P(t|r)$$

16

Example: Multiple Joins



$P(R)$

+r	0.1
-r	0.9

$P(T|R)$

+r	+t	0.8
+r	-t	0.2
-r	+t	0.1
-r	-t	0.9

$P(L|T)$

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9

Join R

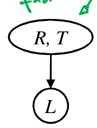
$P(R,T)$

+r	+t	0.08
+r	-t	0.02
-r	+t	0.09
-r	-t	0.81

$P(L|T)$

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9

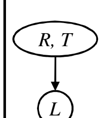
factor(R,T)



L|T

18

Example: Multiple Joins



$P(R,T)$ f(R,T)

+r	+t	0.08
+r	-t	0.02
-r	+t	0.09
-r	-t	0.81

2

$P(L|T)$ f(L,T)

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9

2

Join T

join on T

T, R, T, L

+t +r +l 0.024

$P(R,T,L)$ 0.056

+r	+t	+l	0.024
+r	+t	-l	0.056
+r	-t	+l	0.002
+r	-t	-l	0.018
-r	+t	+l	0.027
-r	+t	-l	0.063
-r	-t	+l	0.081
-r	-t	-l	0.729

4 entries / value of t

19

Operation 2: Eliminate

- Second basic operation: marginalization
- Take a factor and sum out a variable
  - Shrinks a factor to a smaller one
  - A projection operation
- Example: f(l,T)   f(T)

$P(R,T)$

+r	+t	0.08
+r	-t	0.02
-r	+t	0.09
-r	-t	0.81

sum R

$P(T)$

+t	0.17
-t	0.83

Multiple Elimination

join R, join T

R, T, L → T, L → L

$P(R, T, L)$

+r	+t	+l	0.024
+r	+t	-l	0.056
+r	-t	+l	0.002
+r	-t	-l	0.018
-r	+t	+l	0.027
-r	+t	-l	0.063
-r	-t	+l	0.081
-r	-t	-l	0.729

2?

Sum out R

f(T,L)

$P(T, L)$

+t	+l	0.051
+t	-l	0.119
-t	+l	0.083
-t	-l	0.747

Sum out T

$P(L)$

+l	0.134
-l	0.886

P(L) : Marginalizing Early!

$P(R)$

+r	0.1
-r	0.9

$P(T|R)$

+r	+t	0.8
+r	-t	0.2
-r	+t	0.1
-r	-t	0.9

$P(L|T)$

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9

Join R

$P(R, T)$

+r	+t	0.08
+r	-t	0.02
-r	+t	0.09
-r	-t	0.81

$P(L|T)$

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9

Sum out R

$P(T)$

+t	0.17
-t	0.83

$P(L|T)$

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9

Marginalizing Early (aka VE*)

$P(T)$

+t	0.17
-t	0.83

$P(L|T)$

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9

Join T

T, L

$P(T, L)$

+t	+l	0.051
+t	-l	0.119
-t	+l	0.083
-t	-l	0.747

Sum out T

L

$P(L)$

+l	0.134
-l	0.886

f(L)

* VE is variable elimination

Evidence

R → T → L

- If evidence, start with factors that select that evidence
  - No evidence uses these initial factors:

$P(R)$

+r	0.1
-r	0.9

$P(T|R)$

+r	+t	0.8
+r	-t	0.2
-r	+t	0.1
-r	-t	0.9

$P(L|T)$

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9

  - Computing $P(L|+r)$, the initial factors become:

join and sum over T

$P(+r)$

+r	0.1

$P(T|+r)$

+r	+t	0.8
+r	-t	0.2

$P(L|T)$

+t	+l	0.3
+t	-l	0.7
-t	+l	0.1
-t	-l	0.9

join over T
sum over T
join w/ P(+r)

- We eliminate all vars other than query + evidence

Evidence II

- Result will be a selected joint of query and evidence
  - E.g. for P(L | +r), we'd end up with:

$P(+r, L)$

+r	+l	0.026
+r	-l	0.074

Normalize

$P(L|+r)$

+l	0.26
-l	0.74

- To get our answer, just normalize this!
- That's it!

General Variable Elimination

$P(Q)?\ \sum_{a,b} P(A)P(B|A)P(C|B) = \sum_b P(C=c|B=b) \sum_a P(A=a)P(B=b|A=a)$ $f(C)$

- Query: $P(Q|E_1=e_1,\dots,E_k=e_k)$
- Start with initial factors:
  - Local CPTs (but instantiated by evidence)
- While there are still hidden variables (not Q or evidence):
  - Pick a hidden variable H
  - Join all factors mentioning H
  - Eliminate (sum out) H

$f(X_1,H)\ f(X_2,H)\ f(X_3,H)$ → $f(H,X_1,X_2,X_3)$ → $f(X_1,X_2,X_3)$

$f_1(Q,e_i,\dots)\ \ f_2(Q,e_k,e_i,\dots)$

- Join all remaining factors and normalize

$f(Q,e_1,\dots,e_k) = P(Q,e_1,\dots,e_k)$

Variable Elimination Bayes Rule

Start / Select | Join on B | Normalize

$P(B)$ B → a

B	P
+b	0.1
¬b	0.9

$P(A|B) \to P(a|B)$

B	A	P
+b	+a	0.8
~~b~~	~~¬a~~	~~0.2~~
¬b	+a	0.1
~~¬b~~	~~¬a~~	~~0.9~~

$P(B|a)$

a, B

$P(a,B)$

A	B	P
+a	+b	0.08
+a	¬b	0.09

$P(B|a)$

A	B	P
+a	+b	8/17
+a	¬b	9/17

Example

$P(B|j,m) \propto P(B,j,m)$

$P(B)$ $P(E)$ $P(A|B,E)$ $P(j|A)$ $P(m|A)$

Hidden variables: A, E, query variable: B, evidence vars: j,m

Choose A

$P(A|B,E)$, $P(j|A)$, $P(m|A)$ → × → $P(j,m,A|B,E)$ → Σ → $P(j,m|B,E)$

$f(j,m,A,B,E)$ $f(j,m,B,E)$

$P(B)$ $P(E)$ $P(j,m|B,E)$

Example

$P(B)\sum_E P(E)\sum_A P(A|B,E)P(j|A)P(m|A) = P(B|j,m)$

$f(B,j,m)$ $f(B,E,j,m)$

$P(B)$ $P(E)$ $P(j,m|B,E)$

Choose E

$P(E)$, $P(j,m|B,E)$ → × → $P(j,m,E|B)$ → $\sum_E$ → $P(j,m|B)$

$f(j,m,E,B)$ $f(j,m,B)$

$P(B)$ $P(j,m|B)$

Finish with B

$P(B)$, $P(j,m|B)$ → × → $P(j,m,B)$ → Normalize → $P(B|j,m)$

Variable Elimination

- What you need to know:
  - Should be able to run it on small examples, understand the factor creation / reduction flow
  - Better than enumeration: saves time by marginalizing variables as soon as possible rather than at the end

? what factors get generated depending on elimination ordering

- We will see special cases of VE later
  - On tree-structured graphs, variable elimination runs in polynomial time, like tree-structured CSPs
  - You'll have to implement a tree-structured special case to track invisible ghosts (Project 4)

2^n

$P(X_n|X_1=x_1)$ $X_2,X_3,X_4,\dots,X_{n-1}$ $X_{n-1},X_{n-2},\dots,X_2$ $X_{n/2},X_{n/4}$