Reasoning Under Uncertainty

CPS 270

Ron Parr

What We Know So Far

- Probabilities are a good way to represent our beliefs under conditions of uncertainty
- To do this correctly we need a full, joint probability distribution
- A full, joint probability distribution is unwieldy – exponential in no. of variables
- Use *independence* to make this tractable

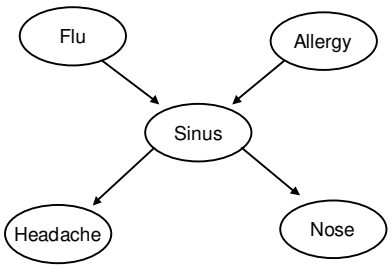
Where This Is Going

- Want: Some clever data structures and algorithms to circumvent the combinatorial explosion in the size of the joint
- Bayesian networks: our primary data structure
- Use familiar algorithms to manipulate this data structure

Conditional Independence

- Suppose we know the following:
  - The flu causes sinus inflammation
  - Allergies cause sinus inflammation
  - Sinus inflammation causes a runny nose
  - Sinus inflammation causes headaches
- How are these connected?

Causal Structure



Knowing sinus separates the variables from each other.

Conditional Independence

- We say that two variables, A and B, are conditionally independent given C if:
  - $P(A|BC) = P(A|C)$
- How does this help?
- We store only a conditional probability table (CPT) of each variable given its parents
- Naïve Bayes (spam assassin)

Notation Reminder

- P(A|B) is a conditional prob. distribution
  - It is a function!
  - P(A=true|B=true), P(A=true|B=false), P(A=false|B=True), P(A=false|B=true)
- P(A|b) is a probability distribution, function
- P(a|B) is a function, not a distribution
- P(a|b) is a number

Getting More Formal

- What is a Bayes net?
  - A directed acyclic graph (DAG)
  - Parents chosen such that

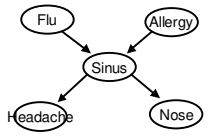
$$P(X_1 \ldots X_n) = \prod_i P(X_i \mid \text{parents}(X_i))$$

  - Given the parents, each variable is independent of non-descendents
  - For each node X_i, store $P(X_i|\text{parents}(X_i))$
  - Represent as table called a CPT

Real Applications of Bayes Nets

- Diagnosis of lymph node disease
- Used in Microsoft office and Windows
  - http://www.research.microsoft.com/research/dtg/
- Used by robots to identify meteorites to study
- Study the human genome:Alex Hartemink et al.
- Many other applications…

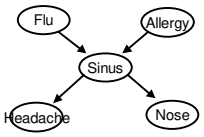
Space Efficiency



- Entire joint as 32 (31) entries
  - P(H|S),P(N|S) have 4 (2)
  - P(S|AF) has 8 (4)
  - P(A) has 2 (1)
  - Total is 20 (10)
- This can require exponentially less space
- Space problem is solved for "most" problems

Atomic Event Probabilities

$$P(X_1 \ldots X_n) = \prod_i P(X_i \mid \text{parents}(X_i))$$



Note that this is true if we construct the net so that for each new variable added, we connect all influencing variables as parents

Doing Things the Hard Way

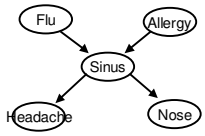
$$P(f \mid h) = \frac{P(fh)}{P(h)} = \frac{\sum_{SAN} P(fhSAN)}{\sum_{SANF} P(hSANF)}$$

defn. of conditional probability

marginalization

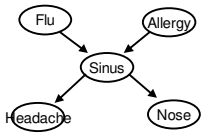
Doing this naïvely, we need to sum over all atomic events defined over these variables. There are exponentially many of these.

Working Smarter



$$
\begin{aligned}
P(hSANF) &= P(hN \mid SAF)P(SAF) \\
&= P(hN \mid S)P(SAF) \\
&= P(h \mid S)P(N \mid S)P(SAF) \\
&= P(h \mid S)P(N \mid S)P(S \mid AF)P(AF) \\
&= P(h \mid S)P(N \mid S)P(S \mid AF)P(A)P(F)
\end{aligned}
$$

Working Smarter



$$
\begin{aligned}
P(h) &= \sum_{SANF} P(hSANF) \\
&= \sum_{SANF} P(hN \mid SAF)P(SAF) \\
&= \sum_{NS} P(hN \mid S)\sum_{AF} P(SAF) \\
&= \sum_{S} P(h \mid S)\sum_{N} P(N \mid S)\sum_{AF} P(SAF) \\
&= \sum_{S} P(h \mid S)\sum_{N} P(N \mid S)\sum_{AF} P(S \mid AF)P(A)P(F)
\end{aligned}
$$

Potential for exponential reduction in computation.

Where We Are

- BNs can give us an exponential reduction in the space required to represent a joint distribution.
- Storage is exponential in largest parent set.
- Claim: Parent sets are often reasonable.
- Claim: Inference cost is often reasonable.
- Question: Can we quantify relationship between structure and inference cost?

Computational Efficiency

$$
\begin{aligned}
&\sum_{SANF} P(hSANF) = \sum_{SANF} P(h \mid S)P(N \mid S)P(S \mid AF)P(A)P(F) \\
&= \sum_{S} P(h \mid S)\sum_{N} P(N \mid S)\sum_{AF} P(S \mid AF)P(A)P(F)
\end{aligned}
$$

The distributive law allows us to decompose the sum.

Potential for an exponential reduction in computation costs.

What Is a Bayes Net, Really?

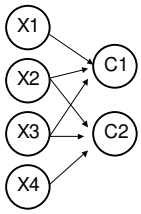
- A Bayes net is a data structure
- A Bayes net is not a method
- Probabilistic reasoning is a method
- Bayes nets don't solve real problems any more than b-trees solve real problems
- Q: How often can a bnet solve a computational efficiency problem?

Bayes Net Inference Is NP-hard

- Decision problem: Is $P(X_i)>0.5$?
- We reduce from 3SAT
- 3SAT variables map to BN variables
- Clauses become variables with the corresponding SAT variables as parents

Reduction

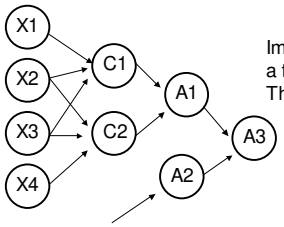
$(\overline{X}_1 \vee X_2 \vee X_3) \wedge (\overline{X}_2 \vee X_3 \vee X_4) \wedge ...$



Problem: What if we have a large number of clauses? How does this fit into our decision problem framework?

And Trees

We could make a single variable which is the AND of all of our clauses, but this would have CPT that is exponential in the number of clauses.



Implement as a tree of ANDs. This is polynomial.

Is BN Inference NP Complete?

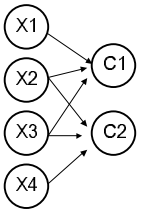
- Can show that BN inference is #P hard
- #P is counting the number of satisfying assignments
- Idea: Assign variables uniform probability
- Probability of conjunction of clauses tells us how many assignments are satisfying

Checkpoint

- BNs can be very compact
- Worst case: Inference is intractable
- We need to show that this worst case can be avoided in many circumstances if we have any hope of salvaging probabilistic reasoning as a tractable method

(Undirected) Trees

- Are the structures from our reduction trees?

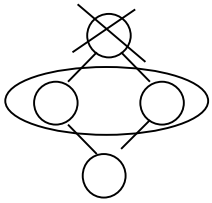


No!

Dealing With General DAGs

- Notice similarities between BNs and CSPs
- They are NP hard (reduction from 3SAT)
- We can solve trees in poly time
- Can we carry over our algorithms from CSPs?

Variable Elimination

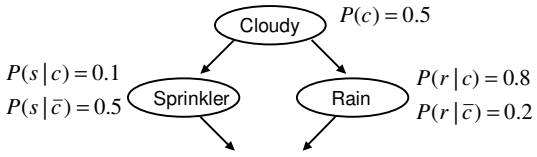


Recall that in variable elimination for CSPs, we eliminated variables and created new supervariables

BN Variable Elimination

- The same trick applies to Bayes nets
- Observation:
  - Every variable elimination ordering corresponds to a rearrangement of the summation in the marginalization computation

Another Example Network



$P(w \mid s\bar{r}) = 0.9$

$P(w \mid \bar{s}r) = 0.9$

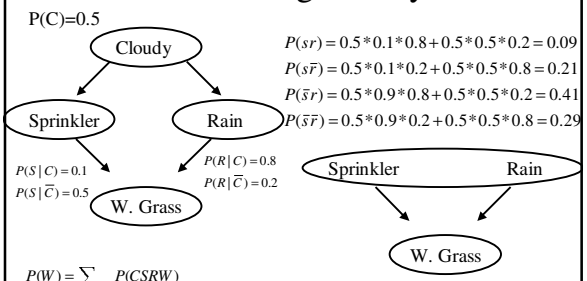
$P(w \mid \bar{s}\bar{r}) = 0.0$

Marginal Probabilities

Suppose we want P(W):

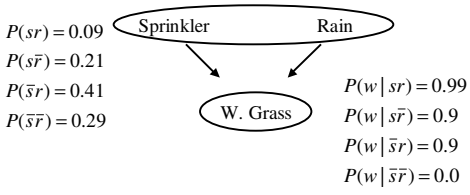
$$P(W) = \sum_{CSR} P(CSRW)$$
$$= \sum_{CSR} P(C)P(S \mid C)P(R \mid C)P(W \mid RS)$$
$$= \sum_{SR} P(W \mid RS) \sum_{C} P(S \mid C)P(C)P(R \mid C)$$

Eliminating Cloudy



$$P(W) = \sum_{CSR} P(CSRW)$$
$$= \sum_{CSR} P(C)P(S \mid C)P(R \mid C)P(W \mid RS)$$
$$= \sum_{SR} P(W \mid RS) \sum_{C} P(S \mid C)P(C)P(R \mid C)$$

Eliminating Sprinkler/Rain



$$P(w) = \sum_{SR} P(w \mid RS)P(RS)$$
$$= 0.09*0.99 + 0.21*0.9 + 0.41*0.9 + 0.29*0$$
$$= 0.6471$$

Dealing With Evidence

Suppose we have observed that the grass is wet?
What is the probability that it has rained?

$$P(R \mid W) = \alpha P(RW)$$
$$= \alpha \sum_{CS} P(CSRW)$$
$$= \alpha \sum_{CS} P(C)P(S \mid C)P(R \mid C)P(W \mid RS)$$
$$= \alpha \sum_{C} P(R \mid C)P(C) \sum_{S} P(S \mid C)P(W \mid RS)$$

Is there a more clever way to deal with w?

The Variable Elimination Algorithm

```
Elim(bn, query)
If bn.vars = query
  return bn
Else
  x = pick_variable(bn)
  newbn.vars = bn.vars - x
  newbn.vars = newbn.vars - neighbors(x)
  newbn.vars = newbn.vars + newvar
  newbn.vars(newvar).function =
```

$$\sum_{X} \prod_{Y \in X \cup neighbors(X)} \text{bn.vars(Y) .function}$$

```
  return(elim(newbn, query))
```

Facts About Variable Elimination

- Picking variables in optimal order is NP hard
- For some networks, there will be no elimination ordering that results in a poly time solution (Must be the case unless P=NP)
- Polynomial for trees
- Need to get a little fancier if there are a large number of query variables or evidence variables

Bayes Net Summary

- Bayes are a data structure for representing a joint probability distribution
- Can give exponential reduction in storage
- Variable elimination is a simple, elegant method that is efficient for many networks
- For some networks, must use approximation methods, e.g., sampling
- Hot topic: Approximation methods for Bayes nets

Bayes Nets as a model for "Neat" AI

- Assuming that we can get the correct probabilities:
  - Bayes nets do exactly the right thing
  - Bayes nets turn an ill-formed "reasoning nder uncertainty" problem into a well-formed computer science problem
  - Even our approximation methods have well-understood limiting behaviors
  - "It's better to approximate the right thing…