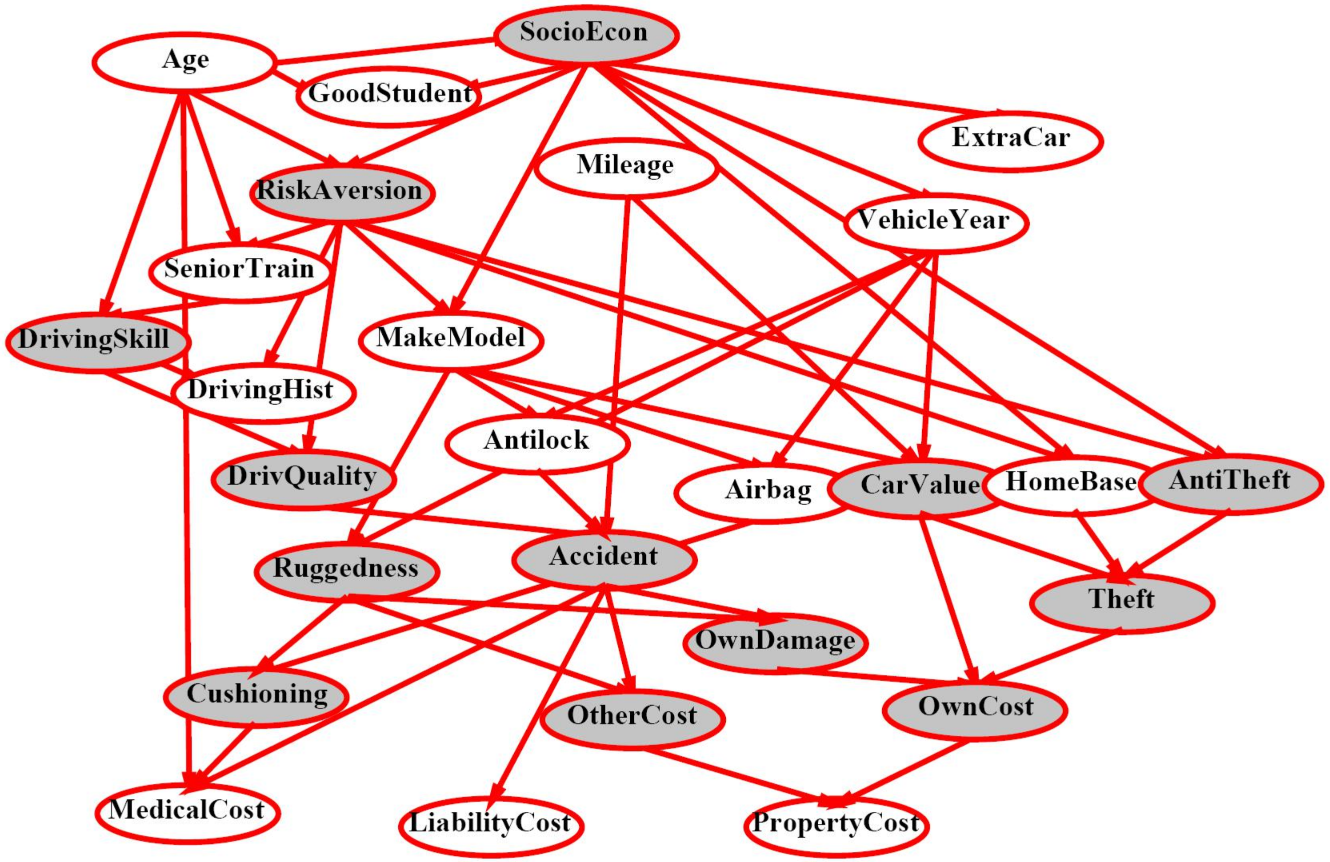
Artificial Intelligence

CS4365 --- Fall 2022

Bayes' Net: Variable Elimination

Instructor: Yunhui Guo

Inference by Enumeration



$$P(Antilock|observed\ variables) = ?$$

Variable Elimination

- Why is inference by enumeration so slow?
  - You join up the whole joint distribution before you sum out the hidden variables
- Idea: interleave joining and marginalizing!
  - Called "Variable Elimination"
  - Still NP-hard, but usually much faster than inference by enumeration

Factor I

- Joint distribution: P(X,Y)
  - Entries P(x,y) for all x, y
  - Sums to 1
- Selected joint: P(x,Y)
  - A slice of the joint distribution
  - Entries P(x,y) for fixed x, all y
  - Sums to P(x)
- Number of capitals = dimensionality of the table

$P(T, W)$

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

$P(cold, W)$

T	W	P
cold	sun	0.2
cold	rain	0.3

Factor II

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

$P(W|cold)$

T	W	P
cold	sun	0.4
cold	rain	0.6

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

$P(W|T)$

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

$P(W|hot)$

$P(W|cold)$

Factor III

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

$P(rain|T)$

T	W	P
hot	rain	0.2
cold	rain	0.6

$P(rain|hot)$

$P(rain|cold)$

Summary

- 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 $P(y_1 \ldots y_N \mid x_1 \ldots x_M)$
  - Any assigned (=lower-case) X or Y is a dimension missing (selected) from the array

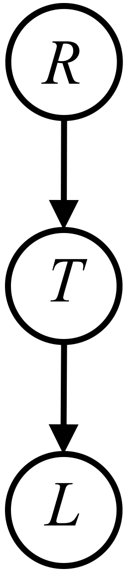
Example: Tracffic Domain

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

$$P(L) = ?$$

$$= \sum_{r,t} P(r, \mathbf{t}, L)$$

$$= \sum_{r,t} P(r)P(\mathbf{t}|r)P(L|\mathbf{t})$$



$P(R)$

+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

Inference by Enumeration: Procedural 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

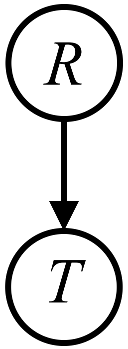
$P(+\ell|T)$

+t	+l	0.3
-t	+l	0.1

- Procedure: Join all factors, then eliminate all hidden variables

Operation 1: Join Factors

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



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

$P(R)$

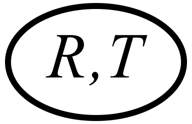
+r	0.1
-r	0.9

$P(T|R)$

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

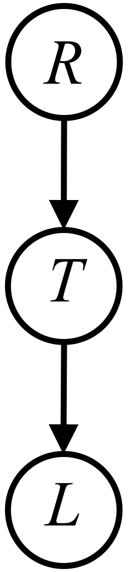
$P(R,T)$

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



- Computation for each entry: pointwise products $\forall r, t : \quad P(r,t) = P(r) \cdot P(t|r)$

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

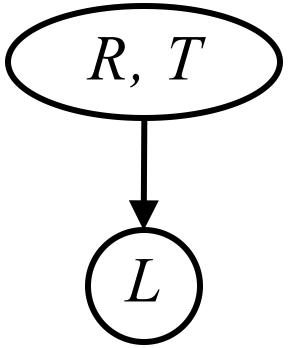
Join R

$P(R, T)$

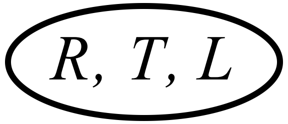
+r	+t	0.08
+r	-t	0.02
-r	+t	0.09
-r	-t	0.81

$P(L|T)$

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



Join T



$P(R, T, L)$

+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

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:

$P(R,T)$

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

$\xrightarrow{\text{sum } R}$

$P(T)$

+t	0.17
-t	0.83

Multiple Elimination

R, T, 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

Sum out R

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

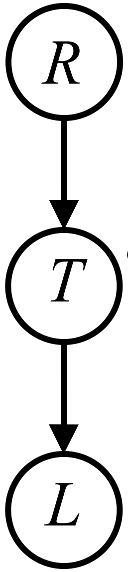
$P(L)$

+l	0.134
-l	0.886

Summary

- Multiple Join, Multiple Eliminate (= Inference by Enumeration)
- Marginalizing Early (= Variable Elimination)

Traffic Domain



$$P(L) = ?$$

- Inference by Enumeration

$$= \sum_t \sum_r \underbrace{P(L|t)\underbrace{P(r)P(t|r)}_{\text{Join on r}}}_{\text{Join on t}}$$

Eliminate r

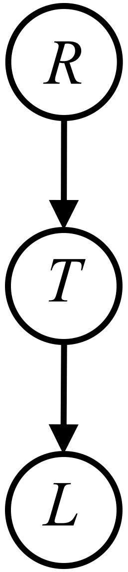
Eliminate t

Variable Elimination

$$= \sum_t \underbrace{P(L|t) \underbrace{\sum_r \underbrace{P(r)P(t|r)}_{\text{Join on r}}}_{\text{Eliminate r}}}_{\text{Join on t}}$$

Eliminate t

Marginalizing Early! (aka VE)



$P(R)$

+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

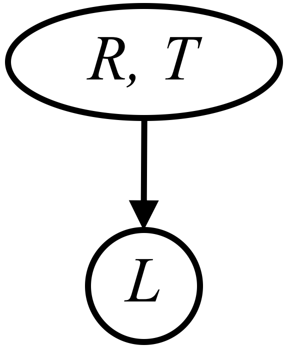
$P(L|T)$

+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



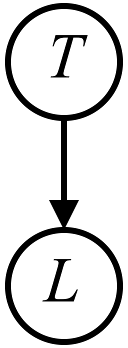
$P(L|T)$

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

Sum out R

$P(T)$

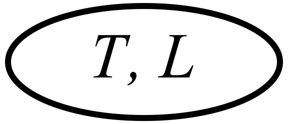
+t	0.17
-t	0.83



$P(L|T)$

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

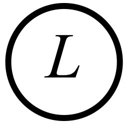
Join T



$P(T,L)$

+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	0.866

Evidence

- 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

$P(+r)$

+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

  - 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 would 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!

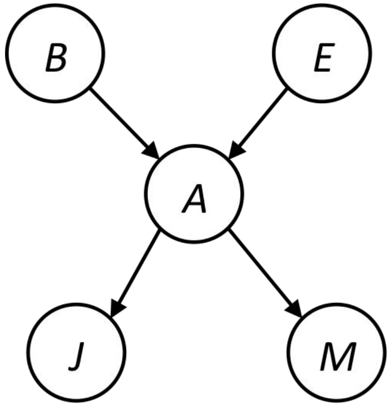
General Variable Elimination

- 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
- Join all remaining factors and normalize

Example

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

$$\boxed{P(B) \qquad P(E) \qquad P(A|B,E) \qquad P(j|A) \qquad P(m|A)}$$

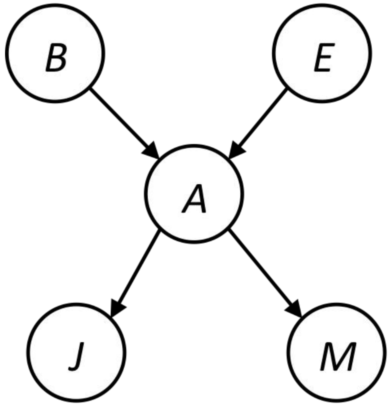


Choose A

$$\begin{matrix} P(A|B,E) \\ P(j|A) \\ P(m|A) \end{matrix} \xrightarrow{\times} P(j,m,A|B,E) \xrightarrow{\sum} P(j,m|B,E)$$

$$\boxed{P(B) \qquad P(E) \qquad P(j,m|B,E)}$$

Example



$$\boxed{P(B) \qquad P(E) \qquad P(j, m|B, E)}$$

Choose E

$$\begin{matrix} P(E) \\ P(j, m|B, E) \end{matrix} \xrightarrow{\times} P(j, m, E|B) \xrightarrow{\sum} P(j, m|B)$$

$$\boxed{P(B) \qquad P(j, m|B)}$$

Finish with B

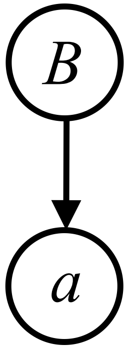
$$\begin{matrix} P(B) \\ P(j, m|B) \end{matrix} \xrightarrow{\times} P(j, m, B) \xrightarrow{\text{Normalize}} P(B|j, m)$$

Example 2: P(B|a)

Start / Select

$P(B)$

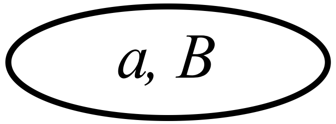
B	P
+b	0.1
¬b	0.9



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

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

Join on B



$P(a, B)$

A	B	P
+a	+b	0.08
+a	¬b	0.09

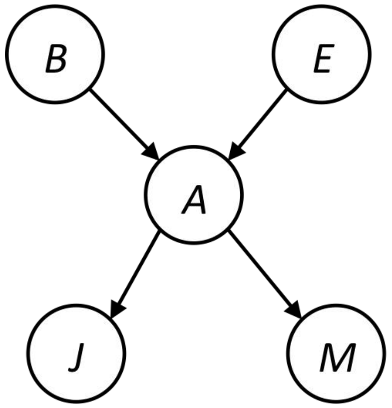
Normalize

$P(B|a)$

A	B	P
+a	+b	8/17
+a	¬b	9/17

Same Example in Equations

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



$$\boxed{P(B) \qquad P(E) \qquad P(A|B,E) \qquad P(j|A) \qquad P(m|A)}$$

$$\begin{aligned}
P(B|j,m) &\propto P(B,j,m) \\
&= \sum_{e,a} P(B,j,m,e,a) && \text{marginal can be obtained from joint by summing out} \\
&= \sum_{e,a} P(B)P(e)P(a|B,e)P(j|a)P(m|a) && \text{use Bayes' net joint distribution expression} \\
&= \sum_{e} P(B)P(e)\sum_{a} P(a|B,e)P(j|a)P(m|a) && \text{use } x*(y+z) = xy + xz \\
&= \sum_{e} P(B)P(e)f_1(B,e,j,m) && \text{joining on a, and then summing out gives } f_1 \\
&= P(B)\sum_{e} P(e)f_1(B,e,j,m) && \text{use } x*(y+z) = xy + xz \\
&= P(B)f_2(B,j,m) && \text{joining on e, and then summing out gives } f_2
\end{aligned}$$

All we are doing is exploiting uwy + uwz + uxy + uxz + vwy + vwz + vxy +vxz = (u+v)(w+x)(y+z) to improve computational efficiency!

Another Variable Elimination Example

Query: $P(X_3|Y_1 = y_1, Y_2 = y_2, Y_3 = y_3)$

Start by inserting evidence, which gives the following initial factors:

$$p(Z)p(X_1|Z)p(X_2|Z)p(X_3|Z)p(y_1|X_1)p(y_2|X_2)p(y_3|X_3)$$

Eliminate X_1, this introduces the factor $f_1(Z, y_1) = \sum_{x_1} p(x_1|Z)p(y_1|x_1)$, and we are left with:

$$p(Z)f_1(Z, y_1)p(X_2|Z)p(X_3|Z)p(y_2|X_2)p(y_3|X_3)$$

Eliminate X_2, this introduces the factor $f_2(Z, y_2) = \sum_{x_2} p(x_2|Z)p(y_2|x_2)$, and we are left with:

$$p(Z)f_1(Z, y_1)f_2(Z, y_2)p(X_3|Z)p(y_3|X_3)$$

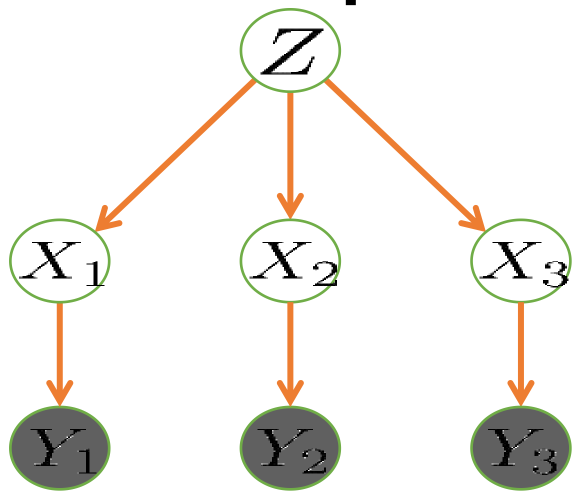
Eliminate Z, this introduces the factor $f_3(y_1, y_2, X_3) = \sum_z p(z)f_1(z, y_1)f_2(z, y_2)p(X_3|z)$, and we are left:

$$p(y_3|X_3), f_3(y_1, y_2, X_3)$$

No hidden variables left. Join the remaining factors to get:

$$f_4(y_1, y_2, y_3, X_3) = P(y_3|X_3)f_3(y_1, y_2, X_3).$$

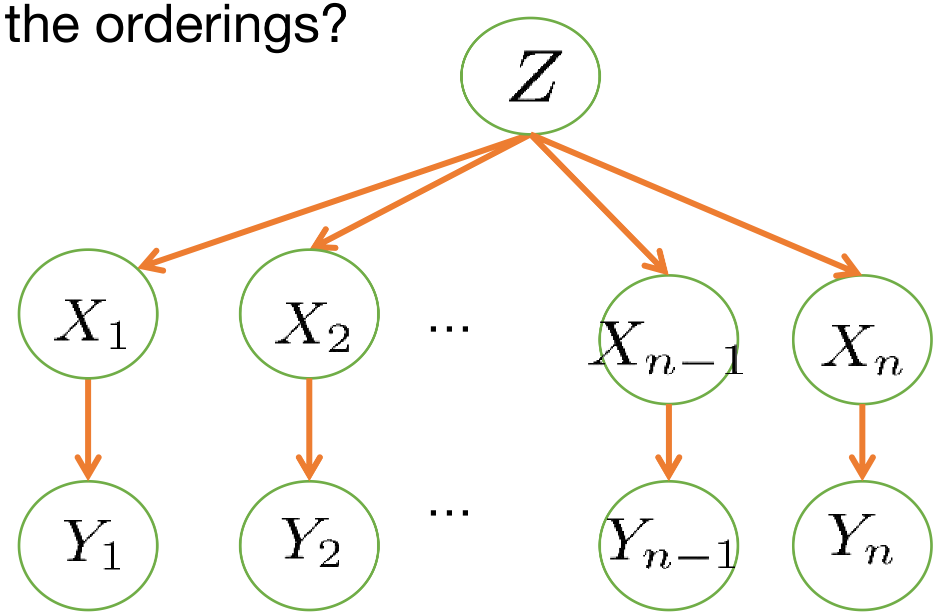
Normalizing over X_3 gives $P(X_3|y_1, y_2, y_3)$.



- Computational complexity critically depends on the largest factor being generated in this process.
- Size of factor = number of entries in table.
- In example above (assuming binary) all factors generated are of size 2 --- as they all only have one variable (Z, Z, and X_3 respectively).

Variable Elimination Ordering

- For the query $P(X_n|y_1,\ldots,y_n)$ work through the following two different orderings as done in previous slide: $Z, X_1, \ldots, X_{n-1}$ and $X_1, \ldots, X_{n-1}, Z$. What is the size of the maximum factor generated for each of the orderings?



- Answer: 2^{n+1} versus 2^2 (assuming binary)
- In general: the ordering can greatly affect efficiency.

VE: Computational and Space Complexity

- The computational and space complexity of **variable elimination** is determined by the largest factor
- The elimination ordering can greatly affect the size of the largest factor.
  - E.g., previous slide's example 2^{n+1} vs. 2^2
- Does there always exist an ordering that only results in small factors?
  - No!

Worst Case Complexity?

- 3-SAT:

$$(x_1 \vee x_2 \vee \neg x_3) \wedge (\neg x_1 \vee x_3 \vee \neg x_4) \wedge (x_2 \vee \neg x_2 \vee x_4) \wedge (\neg x_3 \vee \neg x_4 \vee \neg x_5) \wedge (x_2 \vee x_5 \vee x_7) \wedge (x_4 \vee x_5 \vee x_6) \wedge (\neg x_5 \vee x_6 \vee \neg x_7) \wedge (\neg x_5 \vee \neg x_6 \vee x_7)$$

$$P(X_i = 0) = P(X_i = 1) = 0.5$$

$$Y_1 = X_1 \vee X_2 \vee \neg X_3$$

...

$$Y_8 = \neg X_5 \vee X_6 \vee X_7$$

$$Y_{1,2} = Y_1 \wedge Y_2$$

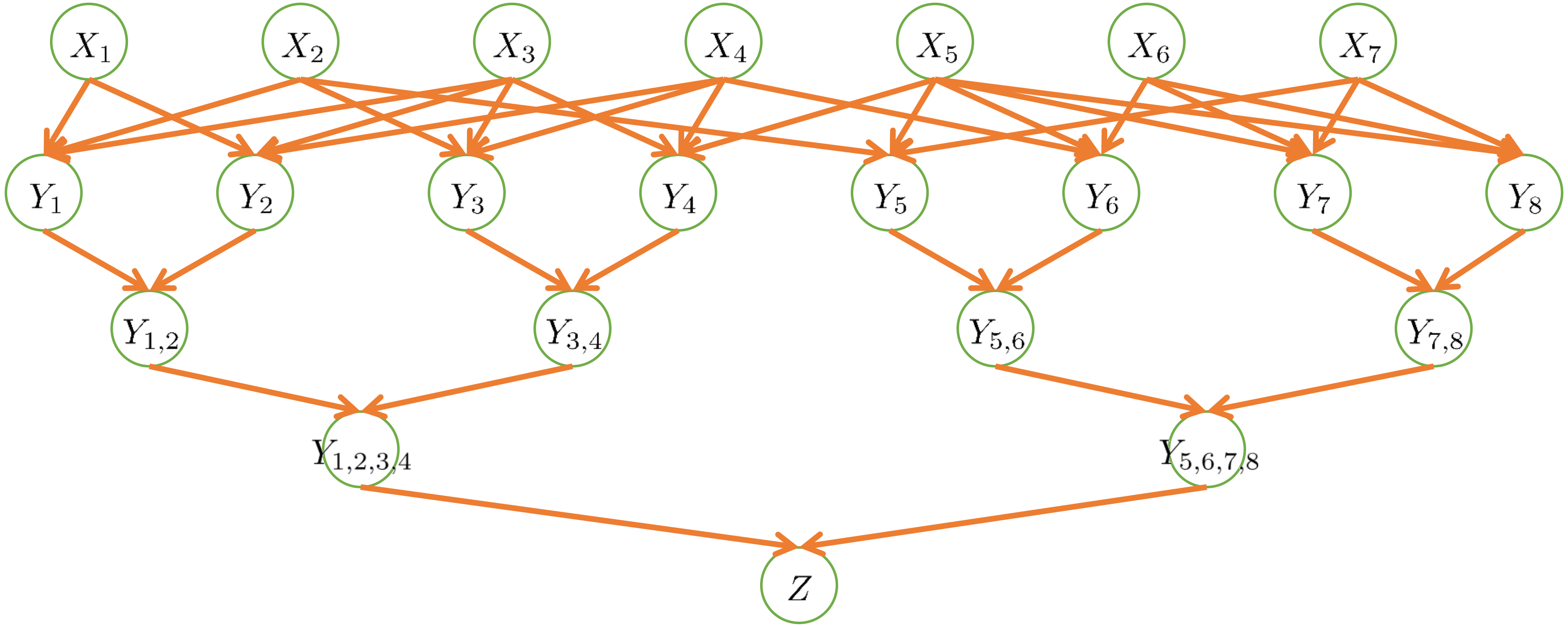
...

$$Y_{7,8} = Y_7 \wedge Y_8$$

$$Y_{1,2,3,4} = Y_{1,2} \wedge Y_{3,4}$$

$$Y_{5,6,7,8} = Y_{5,6} \wedge Y_{7,8}$$

$$Z = Y_{1,2,3,4} \wedge Y_{5,6,7,8}$$

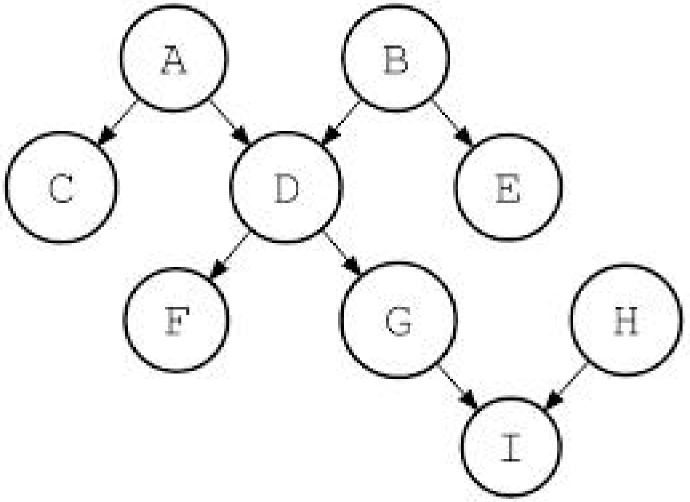


If we can answer $P(X_1, X_2, ..., X_n|z=1)$, we answered whether the 3-SAT problem has a solution.

Hence inference in Bayes' nets is NP-hard. No known efficient probabilistic inference in general

Polytrees

- A polytree is a directed graph with no undirected cycles



- For poly-trees you can always find an ordering that is efficient
  - Try it!!

- Cut-set conditioning for Bayes' net inference

  - Choose set of variables such that if removed only a polytree remains

  - Exercise: Think about how the specifics would work out!

Summary: VE

- Time is exponential in size of largest factor
- Bad elimination order can generate huge factors
- NP Hard to find the best elimination order
- There are reasonable heuristics for picking an elimination order (such as choosing the variable that results in the smallest next factor)
- Inference in polytrees (nets with no cycles) is linear in size of the network (the largest CPT)

Probabilistic Reasoning

- Let p be a formula. We try to prove it from available information (evidence) that is known to be true.

- Let $q_1, \ldots, q_n$ be this evidence. Theorem proving can be used to determine if:

$$(q_1 \wedge q_2 \wedge \ldots \wedge q_n) \rightarrow p$$

When the evidence does not imply that p is true it may imply that p is false:

$$(q_1 \wedge q_2 \wedge \ldots \wedge q_n) \rightarrow \neg p$$

Probabilistic Reasoning

- But in many practical situations the evidence cannot be used to prove that p is either **true** or **false**. Still, it can always be used for probabilistic reasoning.
- The optimal Bayes decision rule is:

  Take p as true if:

$$P(p \mid q_1 \wedge q_2 \wedge \ldots \wedge q_n) \geq P(\neg p \mid q_1 \wedge q_2 \wedge \ldots \wedge q_n)$$

This is the same condition as:

Take p as true if:

$$P(p \mid q_1 \wedge q_2 \wedge \ldots \wedge q_n) \geq 0.5$$

Probabilistic Reasoning: Generalization of Logical Reasoning

- Observe that if $(q_1 \wedge q_2 \wedge \ldots \wedge q_n) \rightarrow p$, then

$$P(p \mid q_1 \wedge q_2 \wedge \ldots \wedge q_n) = 1$$

So, probabilistic reasoning generalizes logic reasoning.

Probabilistic Reasoning

- The optimal Bayes decision rule is:

  Take p as true if:

$$P(p \mid q_1 \wedge q_2 \wedge \ldots \wedge q_n) \geq P(\neg p \mid q_1 \wedge q_2 \wedge \ldots \wedge q_n)$$

- **Simplification using Bayes Rule**
  - Take p as true if:

$$P(q_1 \wedge q_2 \wedge \ldots \wedge q_n \mid p)\, P(p) \geq P(q_1 \wedge q_2 \wedge \ldots \wedge q_n \mid \neg p) P(\neg p)$$

Naïve Bayes assumption

- **Simplification using Naïve Bayes assumption**
  - The presence of a particular evidence is unrelated to the presence of any other evidences.

  - Take p as true if:

$$P(q_1|p)\,P(q_2|p)\,...\,P(q_n|p)P(p) \geq P(q_1|\neg p)\,P(q_2|\neg p)\,...\,P(q_3|\neg p)\,P(\neg p)$$

Naïve Bayes assumption

- Classifying documents by their content, for example into spam and non-spam e-mails
  - The document D is drawn from a number of classes (topics)
  - Each class C is modeled as sets of words, the probability that the i-th word occurs is $P(w_i \mid C)$
  - The probability of the document D given the classes C is

$$P(D|C) = \prod_i p(w_i|C)$$

Question: what is the probability that a given document D belongs to a given class C ? $P(C|D)$

Naïve Bayes assumption

- $P(D|C) = P(D \wedge C) / P(C)$ conditional distribution
- $P(C|D) = P(D \wedge C) / P(D)$ conditional distribution

  $= P(C)P(D|C) / P(D)$ product rule

Assume that there two classes: Spam (S) and not Spam (¬S)

$P(D|S) = \prod_i p(w_i|S)$ Assumption

$P(D|\neg S) = \prod_i p(w_i|\neg S)$ Assumption

$P(S|D) = P(S) \prod_i p(w_i|S) / P(D)$ Bayes rule

$P(\neg S|D) = P(\neg S) \prod_i p(w_i|\neg S) / P(D)$ Bayes rule

Assume $P(S) = P(\neg S) = 0.5$

$$P(S|D) / P(\neg S|D) = \prod_i p(w_i|S) / p(w_i|\neg S)$$