# FIT 3080: Intelligent Systems

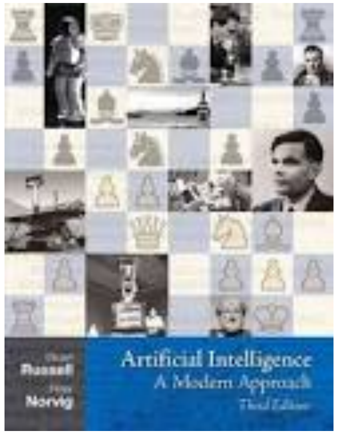
# Bayesian Networks: Independence

Gholamreza Haffari – Monash University

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

# Announcements

- Readings
  - Sections 14.3
  - Sections 14.4-5



# Outline

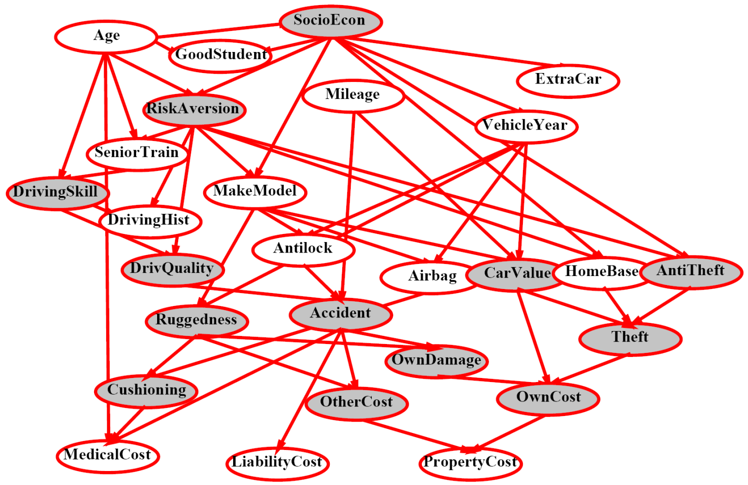
- Semantics of BNs
- Encoded (Conditional) Independencies in BNs
- Reasoning (aka Inference)

# Outline

- Semantics of BNs
- Encoded (Conditional) Independencies in BNs
- Reasoning (aka Inference)

# Bayes' Nets

- A Bayes' net is an efficient encoding of a probabilistic model of a domain



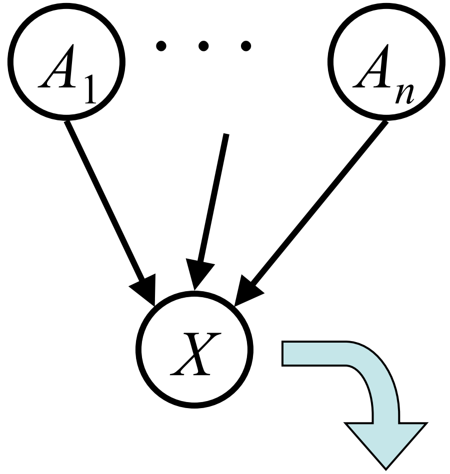

- Questions we can ask:
  - Inference: given a fixed BN, what is P(X | e)?
  - Representation: given a BN graph, what kinds of distributions can it encode?
  - Modeling: what BN is most appropriate for a given domain?

# 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

$$P(X|a_1 \dots a_n)$$

  - CPT: conditional probability table
  - Description of a noisy "causal" process

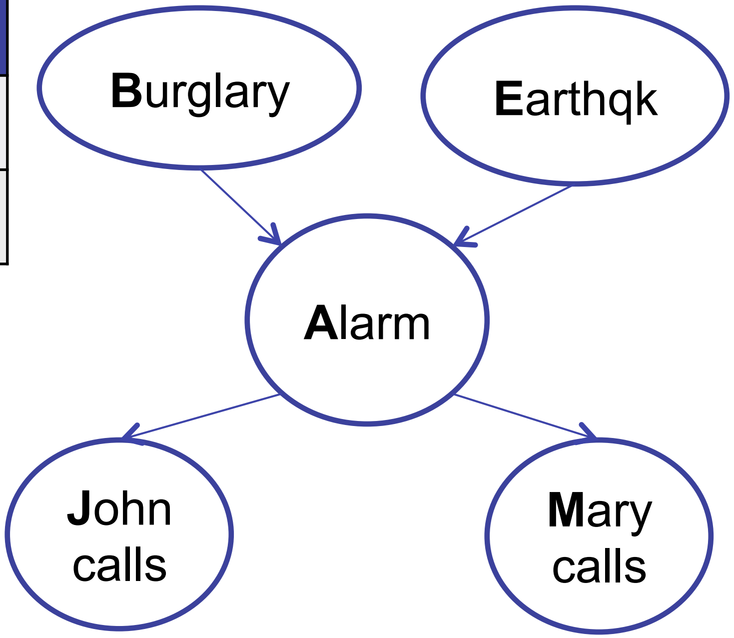



$$P(X|A_1 \dots A_n)$$

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

# Example: Alarm Network

| B | P(B) |
|---|---|
| +b | 0.001 |
| ¬b | 0.999 |




| E | P(E) |
|---|---|
| +e | 0.002 |
| ¬e | 0.998 |

| A | J | P(J\|A) |
|---|---|---|
| +a | +j | 0.9 |
| +a | ¬j | 0.1 |
| ¬a | +j | 0.05 |
| ¬a | ¬j | 0.95 |

| A | M | P(M\|A) |
|---|---|---|
| +a | +m | 0.7 |
| +a | ¬m | 0.3 |
| ¬a | +m | 0.01 |
| ¬a | ¬m | 0.99 |

| B | E | A | P(A\|B,E) |
|---|---|---|---|
| +b | +e | +a | 0.95 |
| +b | +e | ¬a | 0.05 |
| +b | ¬e | +a | 0.94 |
| +b | ¬e | ¬a | 0.06 |
| ¬b | +e | +a | 0.29 |
| ¬b | +e | ¬a | 0.71 |
| ¬b | ¬e | +a | 0.001 |
| ¬b | ¬e | ¬a | 0.999 |

# Building the (Entire) Joint

- We can take a Bayes' net and build any entry from the full joint distribution it encodes

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

  - Typically, there's no reason to build ALL of it
  - We build what we need on the fly

- To emphasize: every BN over a domain implicitly defines a joint distribution over that domain, specified by local probabilities and graph structure

# Size of a Bayes' Net

- How big is a joint distribution over N Boolean variables?

  $2^N$

- How big is an N-node net if nodes have up to k parents?

  $O(N * 2^{k+1})$

- Both give you the power to calculate $P(X_1, X_2, \ldots X_n)$
- BNs: Huge space savings!
- Also easier to elicit local CPTs
- Also turns out to be faster to answer queries (coming)

# Bayes' Nets So Far

- We now know:
  - What is a Bayes' net?
  - What joint distribution does a Bayes' net encode?
- Now: properties of that joint distribution (independence)
  - Key idea: conditional independence
  - Last class: assembled BNs using an intuitive notion of conditional independence as causality
  - Today: formalize these ideas
  - Main goal: answer queries about conditional independence and influence
- Next: how to compute posteriors quickly (inference)

# Outline

- Semantics of BNs
- Encoded (Conditional) Independencies in BNs
- Reasoning (aka Inference)

# Conditional Independence

- Reminder: independence
  - X and Y are independent if

$$\forall x, y \;\; P(x,y) = P(x)P(y) \dashrightarrow \;\; X \perp\!\!\!\perp Y$$

  - X and Y are conditionally independent given Z

$$\forall x, y, z \;\; P(x,y|z) = P(x|z)P(y|z) \dashrightarrow X \perp\!\!\!\perp Y|Z$$

  - (Conditional) independence is a property of a distribution

# Example: Independence

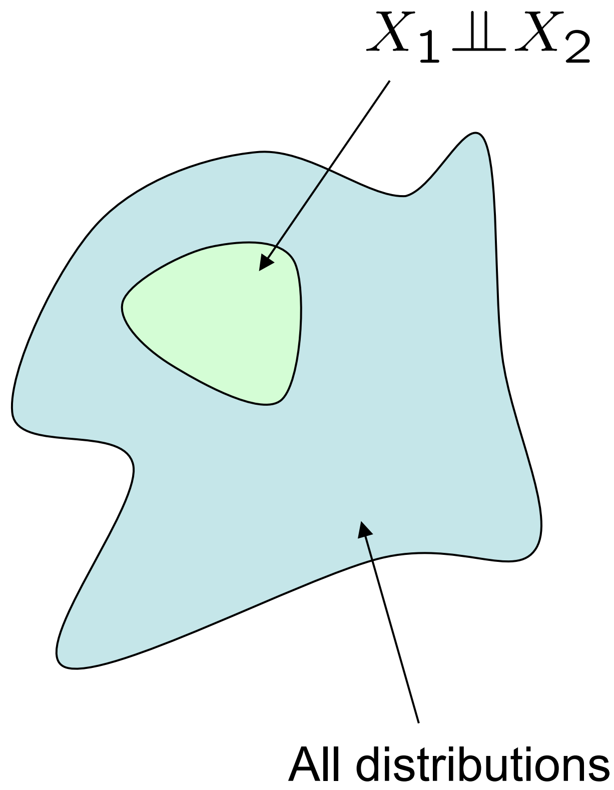
- For this graph, you can fiddle with θ (the CPTs) all you want, but you won't be able to represent any distribution in which the flips are dependent!


$P(X_1)$

| | |
|---|---|
| h | 0.5 |
| t | 0.5 |

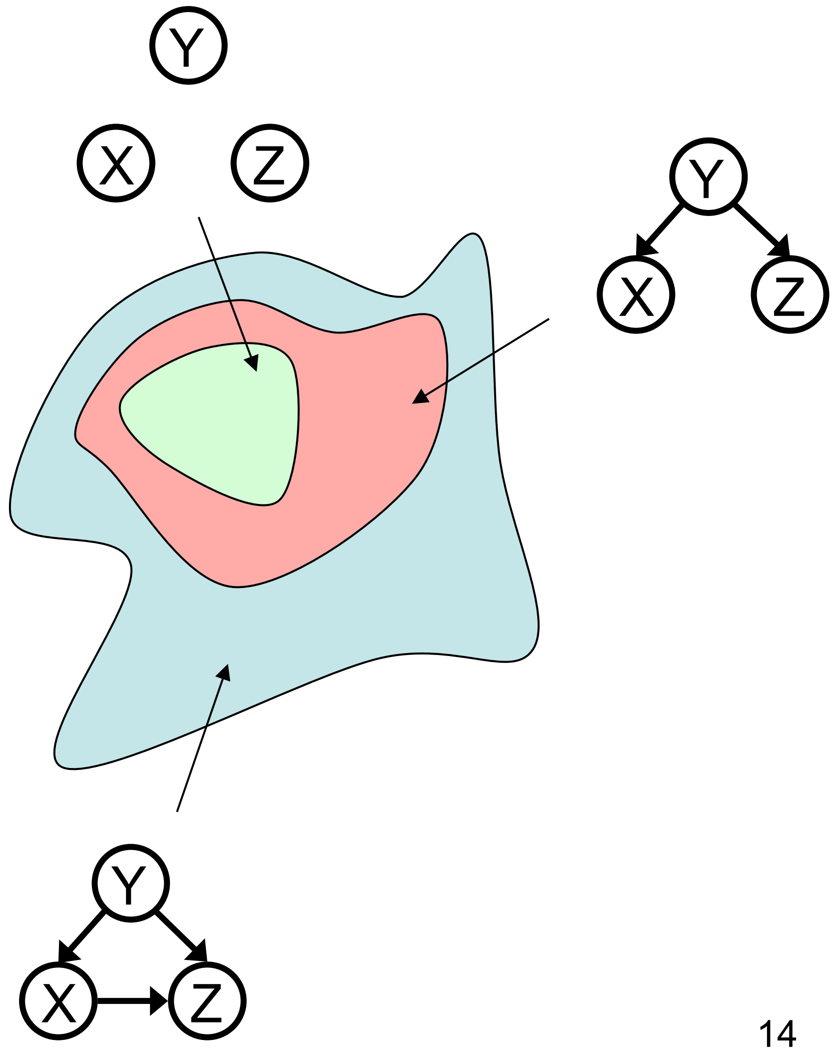
$P(X_2)$

| | |
|---|---|
| h | 0.5 |
| t | 0.5 |



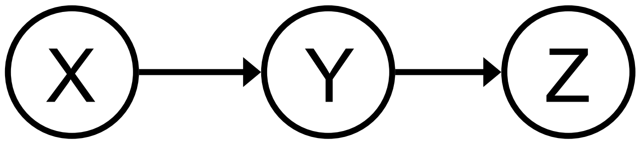
# Topology Limits Distributions

- Given some graph topology G, only certain joint distributions can be encoded
- The graph structure guarantees certain (conditional) independences
- (There might be more independence)
- Adding arcs increases the set of distributions, but has several costs
- Full conditioning can encode any distribution



# Independence in a BN

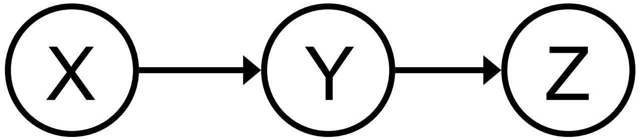
- Important question about a BN:
  - Are two nodes independent given certain evidence?
  - If yes, can prove using algebra (tedious in general)
  - If no, can prove with a counter example
  - Example:




  - Question: are X and Z necessarily independent?
    - Answer: no.  Example: low pressure causes rain, which causes traffic.
    - X can influence Z, Z can influence X (via Y)
    - Addendum: they *could* be independent: how?

# Causal Chains

- This configuration is a "causal chain"




X: Low pressure

Y: Rain

Z: Traffic

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

  - Is X independent of Z given Y?

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

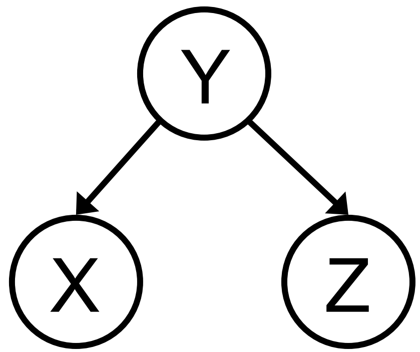
$$= P(z|y)$$

*Yes!*

  - Evidence along the chain "blocks" the influence

# Common Cause

- Another basic configuration: two effects of the same cause
  - Are X and Z independent?
  - Are X and Z independent given Y?




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

$$= P(z|y)$$

*Yes!*

Y: Project due

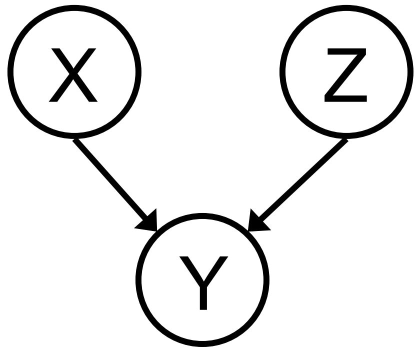
X: Newsgroup busy

Z: Lab full

  - Observing the cause blocks influence between effects.

# Common Effect

- Last configuration: two causes of one effect (v-structures)
  - Are X and Z independent?
    - Yes: the ballgame and the rain cause traffic, but they are not correlated
    - Still need to prove they must be (try it!)
  - Are X and Z independent given Y?
    - No: seeing traffic puts the rain and the ballgame in competition as explanation?
  - This is backwards from the other cases
    - Observing an effect activates influence between possible causes.




X: Raining

Z: Ballgame

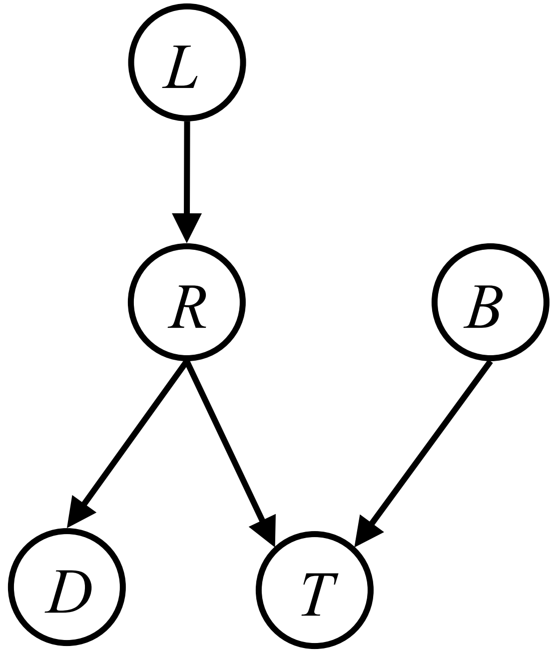
Y: Traffic

# The General Case

- Any complex example can be analyzed using these three canonical cases
- General question: in a given BN, are two variables independent (given evidence)?
- Solution: analyze the graph

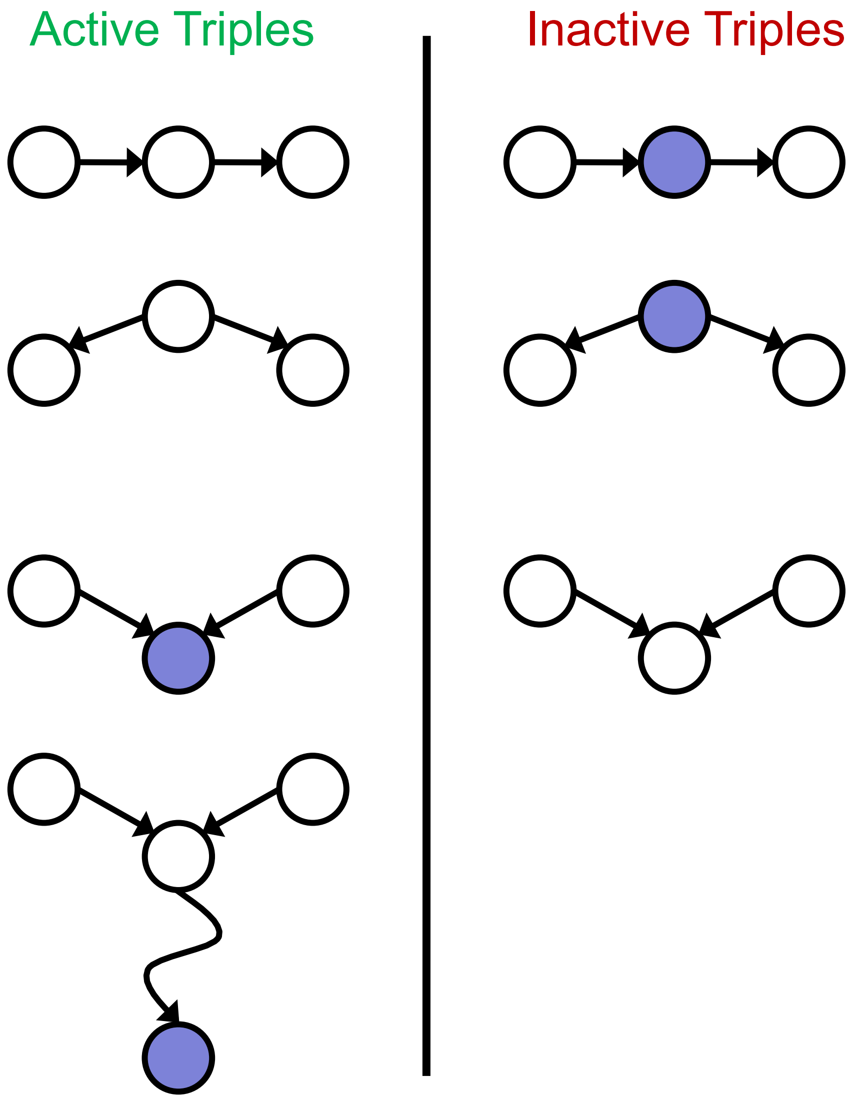
# Reachability

- Recipe: shade evidence nodes
- Attempt 1: if two nodes are connected by an undirected path not blocked by a shaded node, they are conditionally independent
- Almost works, but not quite
  - Where does it break?
  - Answer: the v-structure at T doesn't count as a link in a path unless "active"



# Reachability (D-Separation)

- Question: Are X and Y conditionally independent given evidence vars {Z}?
  - Yes, if X and Y “separated” by Z
  - Look for active paths from X to Y
  - No active paths = independence!
- A path is active if each triple is active:
  - Causal chain $A \rightarrow B \rightarrow C$ where B is unobserved (either direction)
  - Common cause $A \leftarrow B \rightarrow C$ where B is unobserved
  - Common effect (aka v-structure) $A \rightarrow B \leftarrow C$ where B *or one of its descendents* is observed
- All it takes to block a path is a single inactive segment

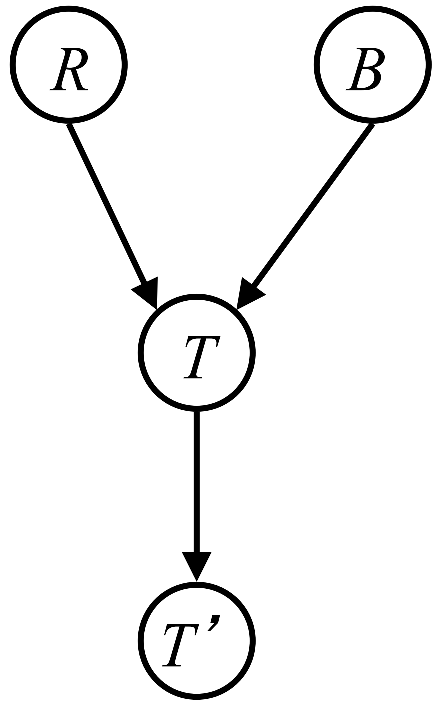


# Example

$R \perp\!\!\!\perp B$ *Yes*

$R \perp\!\!\!\perp B | T$

$R \perp\!\!\!\perp B | T'$



# Example

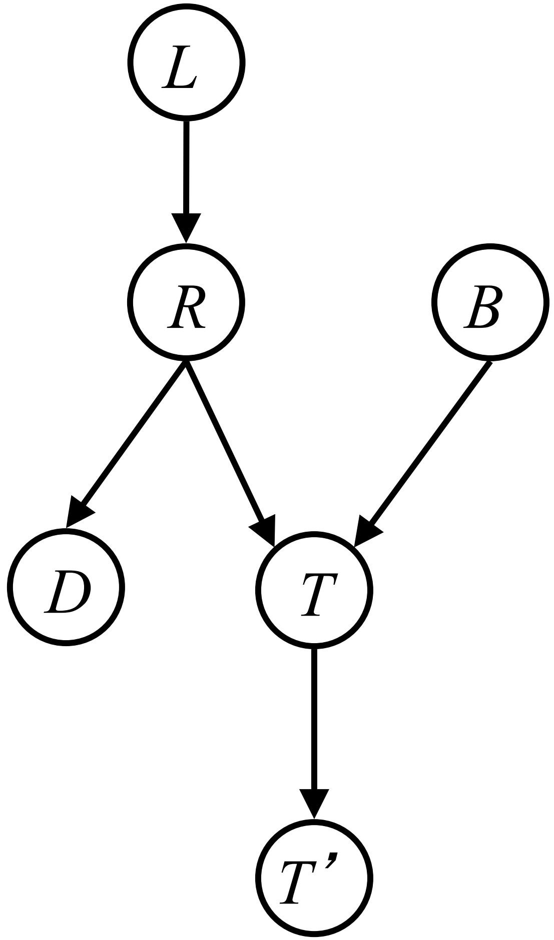
$L \perp\!\!\!\perp T' | T$ Yes

$L \perp\!\!\!\perp B$ Yes

$L \perp\!\!\!\perp B | T$

$L \perp\!\!\!\perp B | T'$

$L \perp\!\!\!\perp B | T, R$ Yes



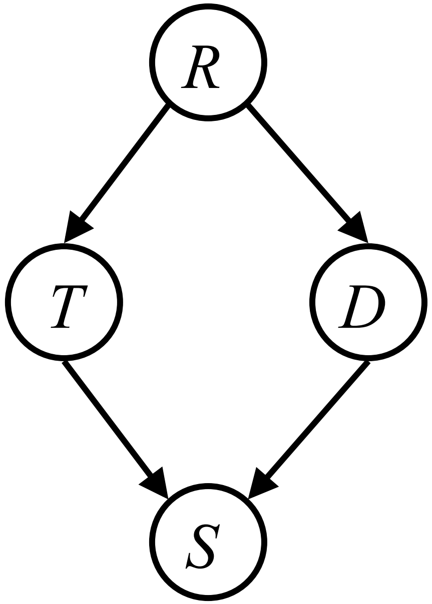
# Example

- Variables:
  - R: Raining
  - T: Traffic
  - D: Roof drips
  - S: I' m sad
- Questions:

$$T \perp\!\!\!\perp D$$

$$T \perp\!\!\!\perp D | R$$ Yes

$$T \perp\!\!\!\perp D | R, S$$



# Outline

- Semantics of BNs
- Encoded (Conditional) Independencies in BNs
- Reasoning (aka Inference)

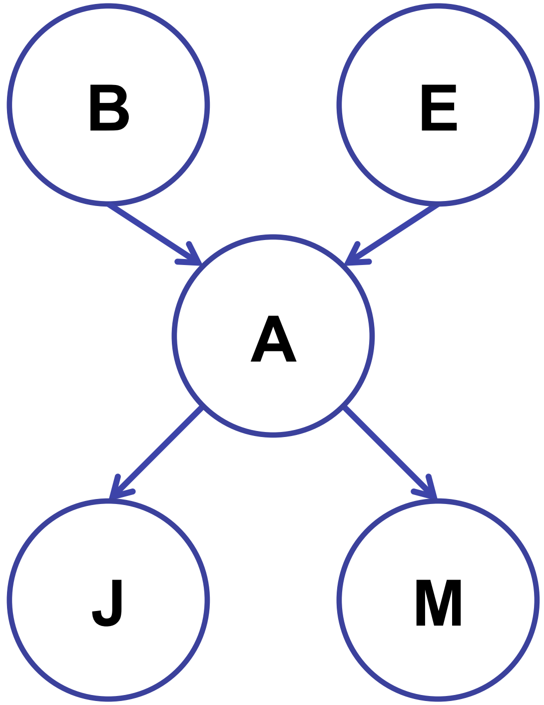
# Inference

- 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:

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

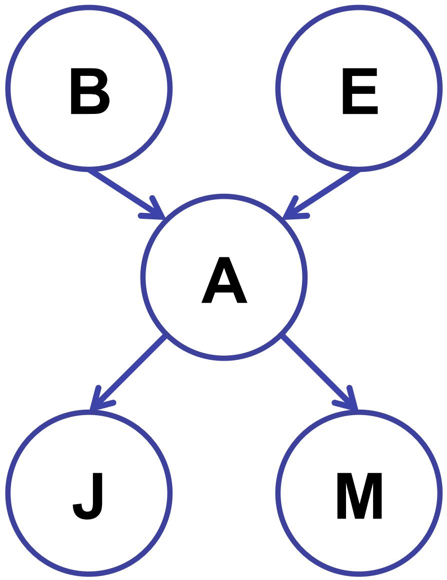


# Inference by Enumeration

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

$$P(+b|+j,+m) =$$

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



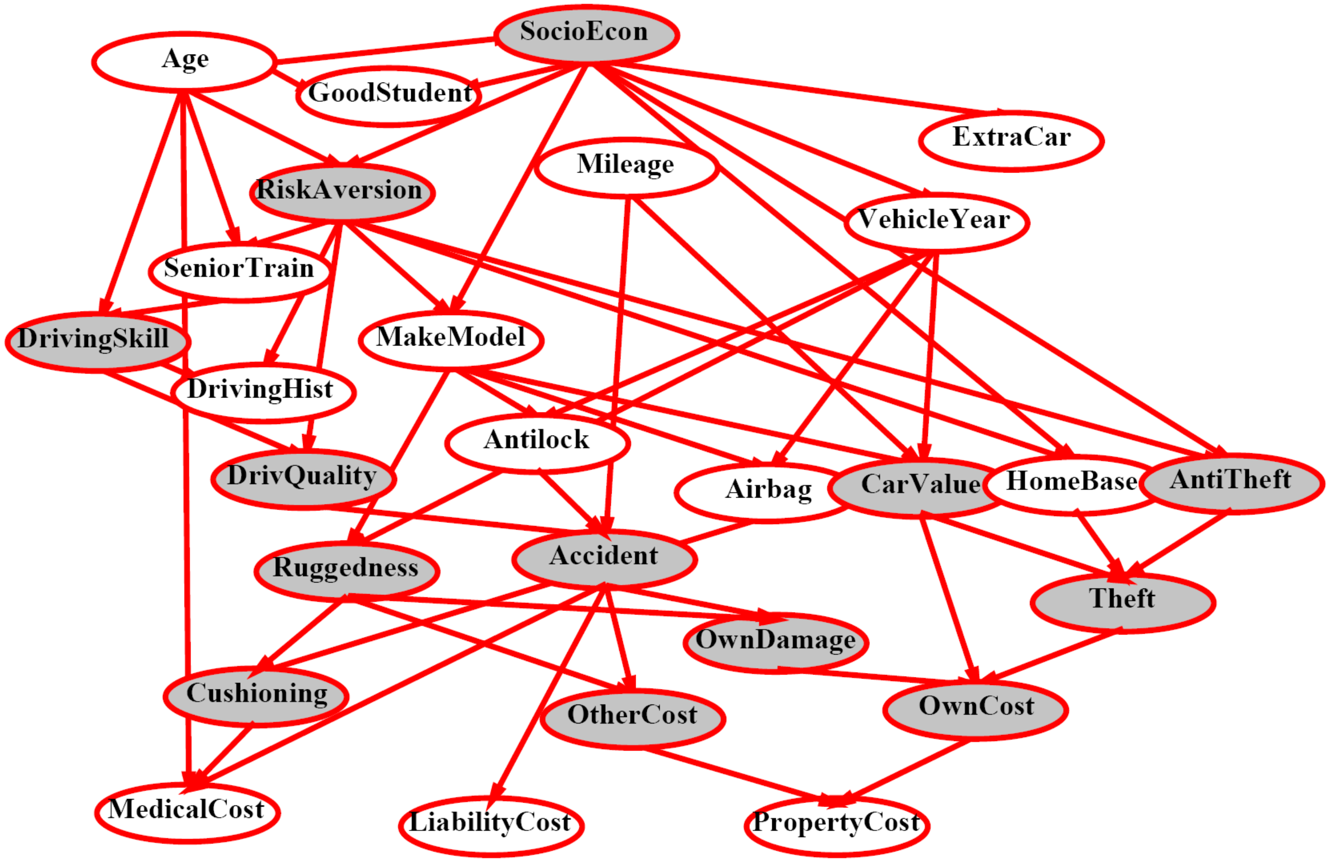
# Example: Enumeration

- 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

$$P(T, W)$$

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

- 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 Zoo II

- 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 |

$P(W|hot)$

$P(W|cold)$

- 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 |

# Factor Zoo 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 |

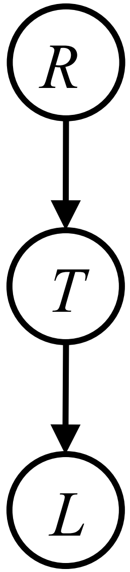
$P(rain|hot)$

$P(rain|cold)$

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

# Example: Traffic Domain

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




- First query: P(L)

$P(R)$

| | |
|---|---|
| +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 |

$P(L|R)$

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

# 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 |

$P(+\ell|T)$

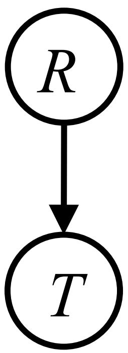
| | | |
|---|---|---|
| +t | +l | 0.3 |
| -t | +l | 0.1 |

- VE: Alternately join factors and eliminate variables

# 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

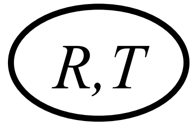



$$P(R) \times P(T|R) \Rightarrow P(R,T)$$

| | |
|---|---|
| +r | 0.1 |
| -r | 0.9 |

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

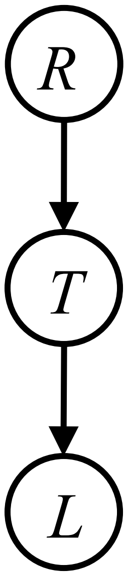
| | | |
|---|---|---|
| +r | +t | 0.08 |
| +r | -t | 0.02 |
| -r | +t | 0.09 |
| -r | -t | 0.81 |




  - 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 |

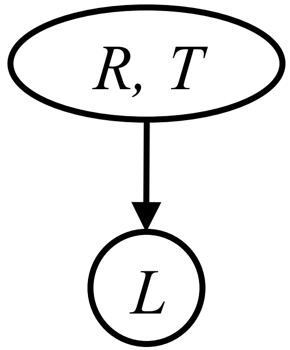
Join R

$P(R,T)$

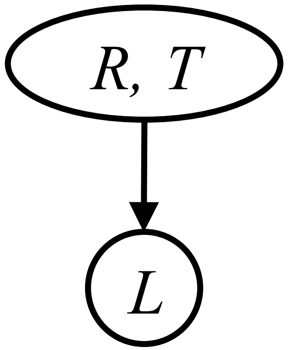
| | | |
|---|---|---|
| +r | +t | 0.08 |
| +r | -t | 0.02 |
| -r | +t | 0.09 |
| -r | -t | 0.81 |

$P(L|T)$

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



# Example: Multiple Joins



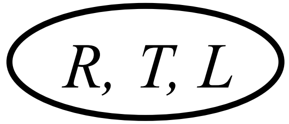

$P(R,T)$

| | | |
|---|---|---|
| +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 |

sum $R$ $\Rightarrow$

$P(T)$

| | |
|---|---|
| +t | 0.17 |
| -t | 0.83 |

# Multiple Elimination

$R, T, L$ → $T, L$ → $L$

$P(R,T,L)$

| | | | |
|---|---|---|---|
| +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 |

Sum out R

$P(T,L)$

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

Sum out T

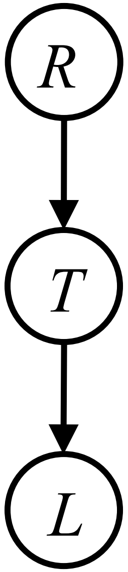
$P(L)$

| | |
|---|---|
| +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 |

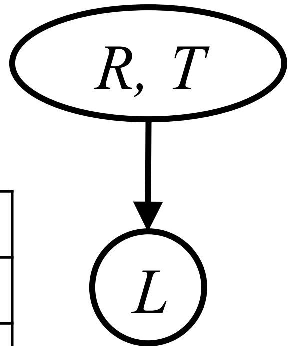
Join R

$P(R,T)$

| | | |
|---|---|---|
| +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 |



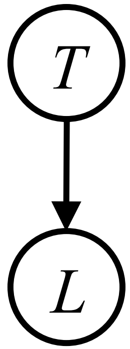

Sum out R

$P(T)$

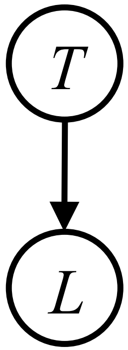
| | |
|---|---|
| +t | 0.17 |
| -t | 0.83 |

$P(L|T)$

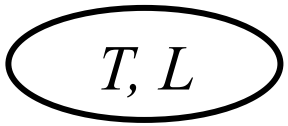
| | | |
|---|---|---|
| +t | +l | 0.3 |
| +t | -l | 0.7 |
| -t | +l | 0.1 |
| -t | -l | 0.9 |



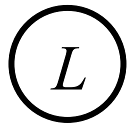
# Marginalizing Early (aka VE*)




Join T




Sum out T




$P(T)$

| | |
|---|---|
| +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 |

$P(T, L)$

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

$P(L)$

| | |
|---|---|
| +l | 0.134 |
| -l | 0.886 |

* VE is variable elimination

# 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'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

- 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

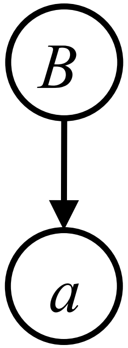
- Join all remaining factors and normalize

# Variable Elimination Bayes Rule

## Start / Select

$P(B)$

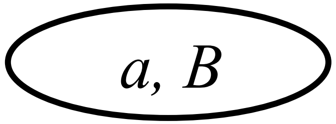
| B | P |
|---|---|
| +b | 0.1 |
| ¬b | 0.9 |




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

| 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 |

# 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

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

Choose E

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

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

Finish with B

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

# 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

# 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

- Variable elimination algorithm can be used for reasoning in BNs
  - Efficient computation of $P(Q|E_1 = e_1, \ldots E_k = e_k)$