Probabilistic Reasoning

(Mostly using Bayesian Networks)

Introduction: Why probabilistic reasoning?

- The world is not deterministic. (Usually because information is limited.)
- Ways of coping with uncertainty include default logic, fuzzy logic, and probability theory.
- Probability theory is well-studied, well-defined, well-supported.
- Also, systems based on it seem to work in practice.

Outline

- A running example
- Review of probability
- A naïve approach to probabilistic inference
- Bayesian networks!

An Example: The Burglar Alarm Domain

Imagine you live in a nice house in Los Angeles. While you're at work, your neighbor John calls to say your burglar alarm is ringing, but neighbor Mary doesn't call. Sometimes the alarm is set off by minor earthquakes. Is there a burglar?

Uncertainty clearly occurs here. Let us handle it using probability theory…

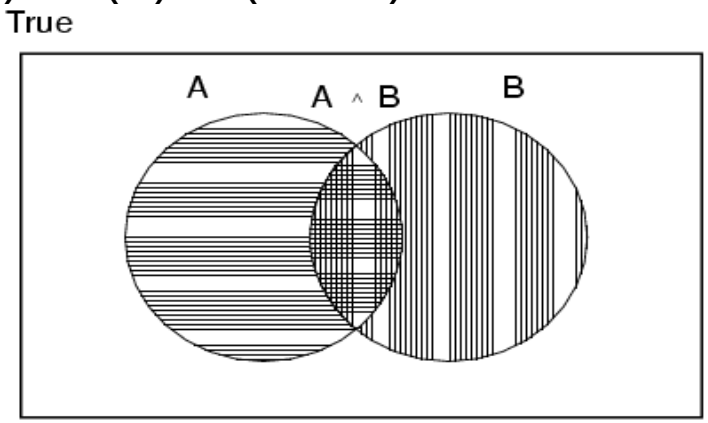
Review of Probability

- A ***random variable*** is just a variable. Can be boolean (e.g., *Earthquake* $\in$ [t, f]), range over a discrete set of values (e.g., *Earthquake* $\in$ [*none, moderate, severe*]), or range over integers or reals (e.g., over the Richter scale).
- A ***sample space*** is a set Ω of possible configurations of the world, where each configuration contains different values for the random variables. (E.g., *Earthquake=t, Burglar=f, Alarm=t.)*
- An ***event*** is any subset of Ω, and can be described using a ***proposition***. (E.g., *Earthquake=t* $\vee$ *Burglar=f.)*
- A ***probability model*** assigns a probability P(ω) to each $\omega \in \Omega$ (and so to each possible event.)

Axioms of probability

For all propositions A, B

- $0 \leq P(A) \leq 1$
- P(true) = 1 and P(false) = 0
- $P(A \cup B) = P(A) + P(B) - P(A \cup B)$



- They imply that $\Sigma\, P(\omega) = 1.0$.
- (If a distribution doesn't follow them, it is ill-defined.)

The Burglar Alarm domain

Here is a well-defined ***joint probability distribution*** for part of the example domain:

	Earthquake		¬Earthquake	
	Burglar	¬Burglar	Burglar	¬Burglar
Alarm	0.0000019	0.00057942	0.00093812	0.000997002
¬Alarm	0.0000001	0.00141858	0.00005988	0.996005000

It can be used to answer any question about these three variables.

“Inference by enumeration”

Inferring the probability of simple propositions, no evidence:

	Earthquake		¬Earthquake	
	Burglar	¬Burglar	Burglar	¬Burglar
Alarm	0.0000019	0.00057942	0.00093812	0.000997002
¬Alarm	0.0000001	0.00141858	0.00005988	0.996005000

For any proposition, sum up the entries where it is true.

P(*earthquake*) = 0.002

Prior/Posterior probabilities

- Probabilities relate propositions to agent's own state of knowledge, and change with new evidence:

  P(*alarm*) = 0.002516442

  P(*alarm*|*earthquake*) = 0.29

  So, these are not assertions about the world.

- The probability of a proposition before the arrival of any evidence is a ***prior*** or ***unconditional*** probability.

- The probability of a proposition given some evidence is a ***posterior*** or ***conditional*** probability.

  But what does ‘given some evidence’ mean, mathematically?

Conditional Probability

- Definition of conditional probability:
  $P(a \mid b) = P(a \wedge b) / P(b)$ if $P(b) > 0$
- ***Product rule*** gives an alternative formulation:
  $P(a \wedge b) = P(a \mid b)\, P(b) = P(b \mid a)\, P(a)$
- A general version holds for whole distributions, e.g.,
  $\mathbf{P}(Earthquake, Alarm) = \mathbf{P}(Alarm \mid Earthquake)\, \mathbf{P}(Earthquake)$
  (View as a set of 4 $\times$ 2 equations, not matrix multiplication.)
- ***Chain rule*** is derived by successive application of product rule:

$$\begin{aligned}\mathbf{P}(X_1, \ldots, X_n) &= \mathbf{P}(X_1,\ldots,X_{n-1})\, \mathbf{P}(X_n \mid X_1,\ldots,X_{n-1}) \\ &= \mathbf{P}(X_1,\ldots,X_{n-2})\, \mathbf{P}(X_{n-1} \mid X_1,\ldots,X_{n-2})\, \mathbf{P}(X_n \mid X_1,\ldots,X_{n-1}) \\ &= \ldots \\ &= \prod_{i=1}^{n} \mathbf{P}(X_i \mid X_1, \ldots, X_{i-1})\end{aligned}$$

Inference by enumeration, cont.

For any proposition, sum the entries where it is true.

	Earthquake		¬Earthquake	
	Burglar	¬Burglar	Burglar	¬Burglar
Alarm	0.0000019	0.00057942	0.00093812	0.000997002
¬Alarm	0.0000001	0.00141858	0.00005988	0.996005000

P(*alarm*|*earthquake*) =

P(*alarm*,*earthquake*)/P(*earthquake*)

= 0.00058132/0.002 = 0.29

Inference, generalized

Typically, we are interested in the posterior joint distribution of the query variables **Y** given specific values e for the evidence variables **E**

Let the hidden variables be **H** = **X** - **Y** - **E**

Then the required summation of joint entries is done by summing out the hidden variables:

$\mathbf{P}(\mathbf{Y} \mid \mathbf{E} = \mathbf{e}) = \mathbf{P}(\mathbf{Y},\mathbf{E} = \mathbf{e}) / \mathbf{P}(\mathbf{E} = e) \propto \Sigma \mathbf{P}(\mathbf{Y},\mathbf{E}= \mathbf{e}, \mathbf{H} = \mathbf{h})$

(The terms in the summation are joint entries because **Y**, **E** and **H** together exhaust the set of random variables)

Problems

Obvious problems with doing inference by summing over the joint:

1. Space complexity to store the joint distribution is $O(d^n)$, where d is the largest arity.
2. Worst-case time complexity is $O(d^n)$.
3. How to find the numbers for $O(d^n)$ entries?
   (Learning is easier when parameters are few.)

We want to make use of regularities in the domain to express that large table more compactly…

Independence

A and *B* are ***independent*** iff

$\mathbf{P}(A|B) = \mathbf{P}(A)$ or $\mathbf{P}(B|A) = \mathbf{P}(B)$ or $\mathbf{P}(A, B) = \mathbf{P}(A)\,\mathbf{P}(B)$

E.g., $\mathbf{P}(\textit{Earthquake, Burglar}) = \mathbf{P}(\textit{Earthquake})\mathbf{P}(\textit{Burglar})$

So, instead of a table with three entries, we could have two tables with just one entry each.

Wouldn't it be nice if we could factor the whole joint like this?

But absolute independence is rare…

Independence, cont

Conditional independence occurs when two variables are independent given another variable.

E.g. If John and Mary do not communicate when deciding to call you about your alarm, then *MaryCalls* and J*ohnCalls* are conditionally independent given *Alarm*.

P(*MaryCalls* ,J*ohnCalls* | *Alarm)* =
P(*MaryCalls* | *Alarm)* **P**(J*ohnCalls* | *Alarm)*

Taking all the independencies into account, we can rewrite the full joint as a product of smaller terms:

P(*MaryCalls*, J*ohnCalls*, *Alarm, Earthquake, Burglar)* =

P(*MaryCalls* | *Alarm)* **P**(J*ohnCalls* | *Alarm)* **P**(*Alarm* |*Earthquake, Burglar)* **P**(*Earthquake*)**P**(*Burglar*)

The full joint would have needed 31 entries. How many does this factorization need?

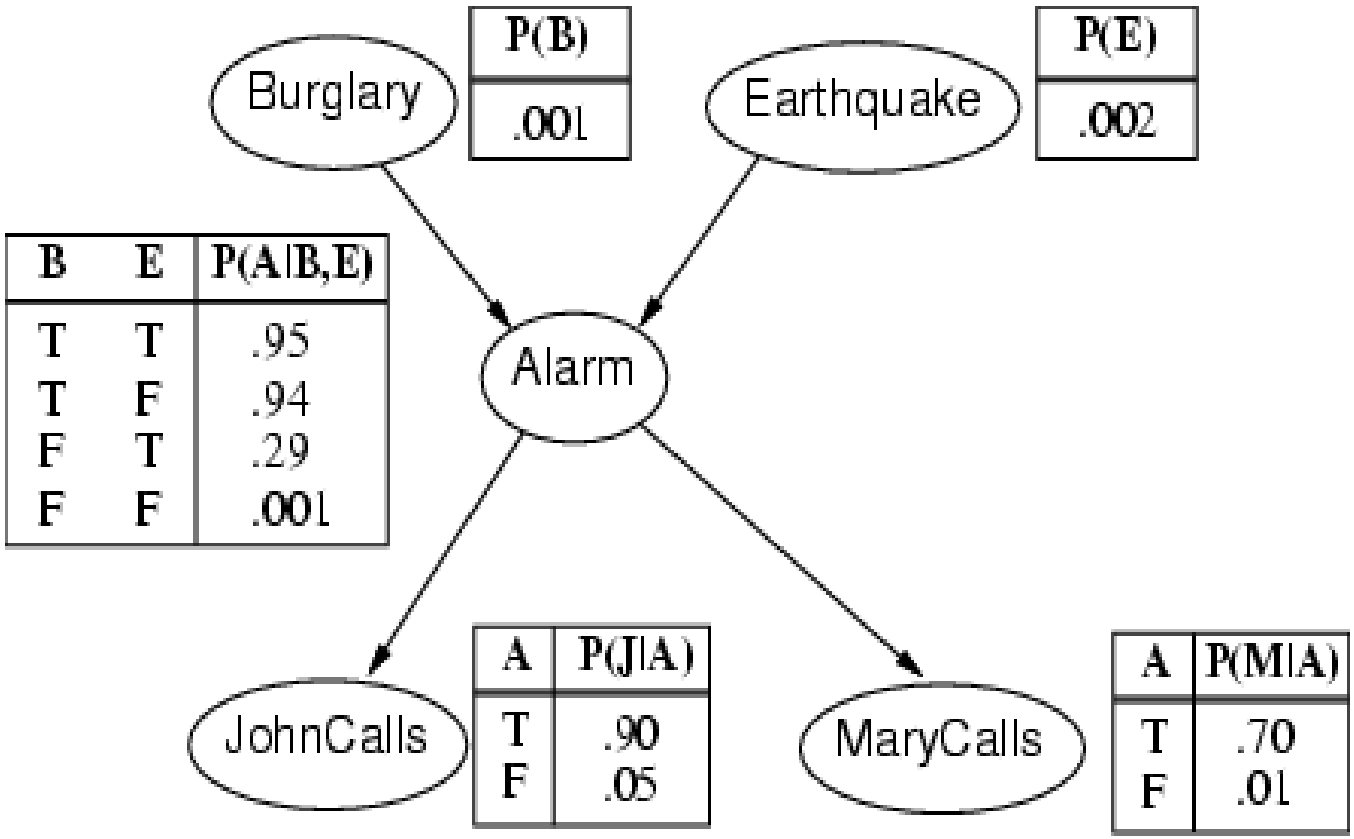
Bayesian networks

- Independence lets us express a joint compactly as a product of conditional distributions.
- This product can be represented using a directed, acyclic graph where:
  - Graph nodes represent probabilistic variables
  - Graph edges represent "direct influences"
  - Each node X_i has an associated conditional distribution:

    $$\mathbf{P}(X_i \mid \text{Parents}(X_i))$$

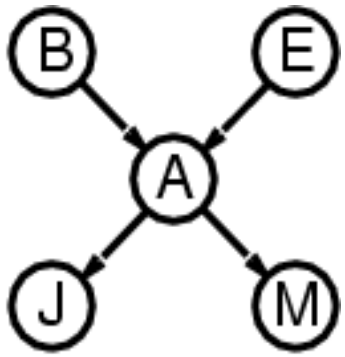
    (In the simplest case, these will be represented as ***conditional probability tables***, or ***CPT***s.)
- Such graphs are known as ***Bayesian networks*** or ***belief networks*** (***BN***s.) They are a type of ***graphical model***.

An example BN



An Example BN, cont.

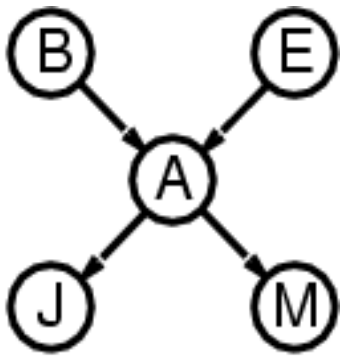
- Topology of network encodes conditional independence assertions:



- *JohnCalls* and *MaryCalls* are conditionally independent given *Alarm.*
- *Burglar* and *Earthquake* are independent a priori, but dependent given *Alarm.* (This is known as 'explaining away.')

Local semantics

General local semantics:



Each node is conditionally independent of its nondescendants given its parents.

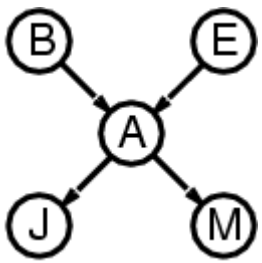
Each node is conditionally independent of the rest of the network given its Markov blanket: its parents, its children, and its children's parents.

(These concepts play a large role during inference.)

Global Semantics

The full joint distribution is defined as the product of the local conditional distributions:

$$\boldsymbol{P}(X_1, \ldots, X_n) = \prod_{i=1}^{n} \boldsymbol{P}(X_i \mid Parents(X_i))$$



As we said, $\boldsymbol{P}(j \wedge m \wedge a \wedge \neg b \wedge \neg e)$

$= \boldsymbol{P}(j \mid a)\, \boldsymbol{P}(m \mid a)\, \boldsymbol{P}(a \mid \neg b, \neg e)\, \boldsymbol{P}(\neg b)\, \boldsymbol{P}(\neg e)$

Interpreting Bayesian networks

- We constructed our burglar alarm network by thinking about the inter-variable dependencies.
- The links seem to reflect causal relationships.
- One of the advantages of Bayesian networks is that they can make sense in this way. This makes them easier for domain experts to construct, or interpret.

  (Not all types of graphical models share this appealing feature!)

- Nevertheless, here is an all-purpose algorithm for constructing a Bayesian network by reasoning about dependencies.

Constructing Bayesian networks

- 1. Choose an ordering of variables $X_1, \ldots, X_n$
- 2. For $i = 1$ to n
  - add X_i to the network
  - select parents from $X_1, \ldots, X_{i-1}$ such that

$$\boldsymbol{P}(X_i \mid Parents(X_i)) = \boldsymbol{P}(X_i \mid X_1, \ldots X_{i-1})$$

This choice of parents guarantees:

$$\boldsymbol{P}(X_1, \ldots, X_n) = \prod_{i=1}^{n} \boldsymbol{P}(X_i \mid X_1, \ldots, X_{i-1}) \quad \text{(chain rule)}$$

$$= \prod_{i=1}^{n} \boldsymbol{P}(X_i \mid Parents(X_i)) \quad \text{(by construction)}$$

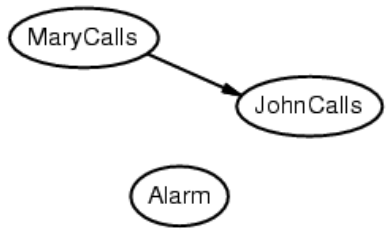
Example

- Suppose we choose the ordering *M, J, A, B, E*

***P**(J | M) = **P**(J)?*

Example

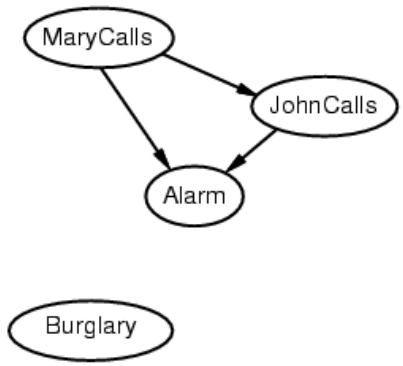
- Suppose we choose the ordering *M, J, A, B, E*



***P**(J | M) = **P**(J)?* **No**
***P**(A | J, M) = **P**(A | J)? **P**(A | J, M) = **P**(A)?*

Example

- Suppose we choose the ordering *M, J, A, B, E*



P(*J* | *M*) = ***P***(*J*)? **No**

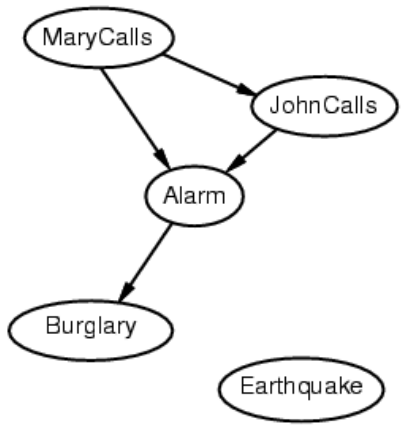
P(*A* | *J*, *M*) = ***P***(*A* | *J*)? ***P***(*A* | *J*, *M*) = ***P***(*A*)? **No**

P(*B* | *A*, *J*, *M*) = ***P***(*B* | *A*)?

P(*B* | *A*, *J*, *M*) = ***P***(*B*)?

Example

- Suppose we choose the ordering M, J, A, B, E



P(*J* | *M*) = ***P***(*J*)? **No**

P(*A* | *J*, *M*) = ***P***(*A* | *J*)? ***P***(*A* | *J*, *M*) = ***P***(*A*)? **No**

P(*B* | *A*, *J*, *M*) = ***P***(*B* | *A*)? **Yes**

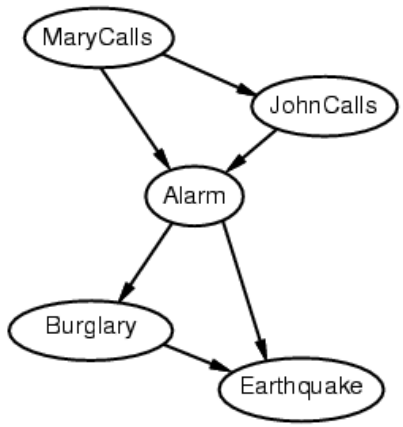
P(*B* | *A*, *J*, *M*) = ***P***(*B*)? **No**

P(*E* | *B*, *A* ,*J*, *M*) = ***P***(*E* | *A*)?

P(*E* | *B*, *A*, *J*, *M*) = ***P***(*E* | *A*, *B*)?

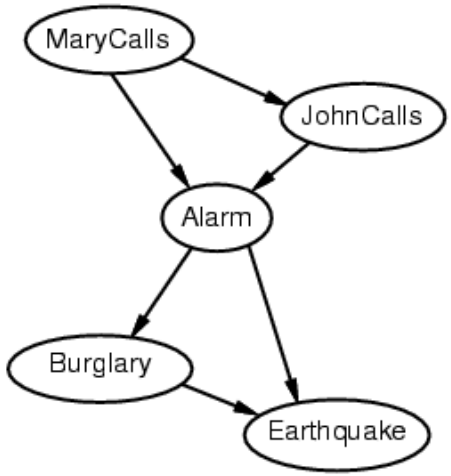
Example

- Suppose we choose the ordering M, J, A, B, E



P(*J* | *M*) = ***P***(*J*)? **No**
P(*A* | *J*, *M*) = ***P***(*A* | *J*)? ***P***(*A* | *J*, *M*) = ***P***(*A*)? **No**
P(*B* | *A*, *J*, *M*) = ***P***(*B* | *A*)? **Yes**
P(*B* | *A*, *J*, *M*) = ***P***(*B*)? **No**
P(*E* | *B*, *A* ,*J*, *M*) = ***P***(*E* | *A*)? **No**
P(*E* | *B*, *A*, *J*, *M*) = ***P***(*E* | *A*, *B*)? **Yes**

Example contd.



- Deciding conditional independence in noncausal directions is hard
- (Causal models and conditional independence seem hardwired for humans!)
- Network is less compact: 1 + 2 + 4 + 2 + 4 = 13 numbers needed

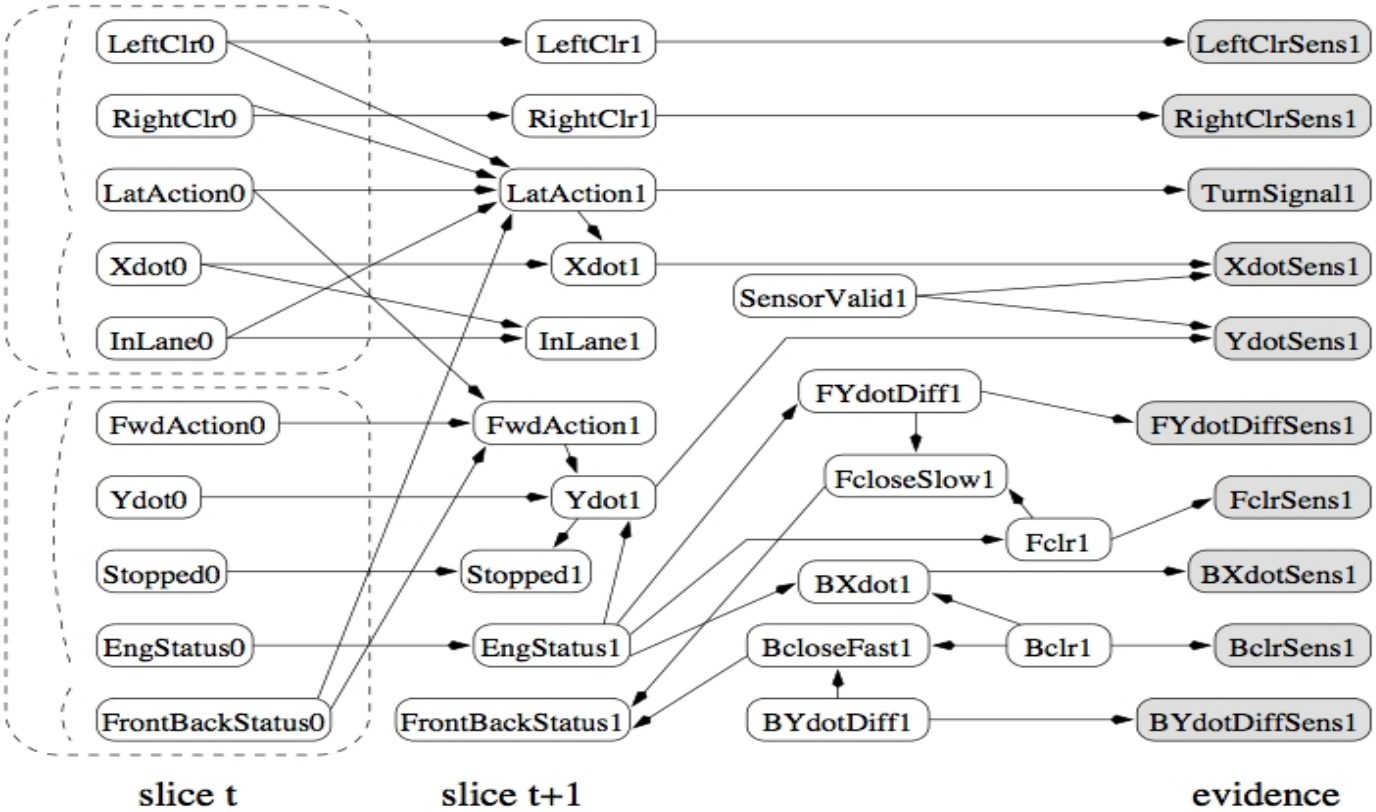
Compact conditional distributions

- CPTs grow exponentially with the number of parents
- CPTs become infinite with continuous-valued parents or children.
- Solution: use not tables, but canonical distributions, such as:
  - Deterministic functions
  - Probabilistic density functions, like Gaussians
  - Functions like Noisy-OR. (For discrete parents $Y_1 \ldots Y_n$ with independent failure probabilities q_i,

    $P(X|Y_1 \ldots Y_j, Y_{j+1} \ldots Y_n) = 1 - \Pi_{i=1} q_i$.)

  Could also encode context-specific independence using rules or trees.

BNs in Action: Driving a Car



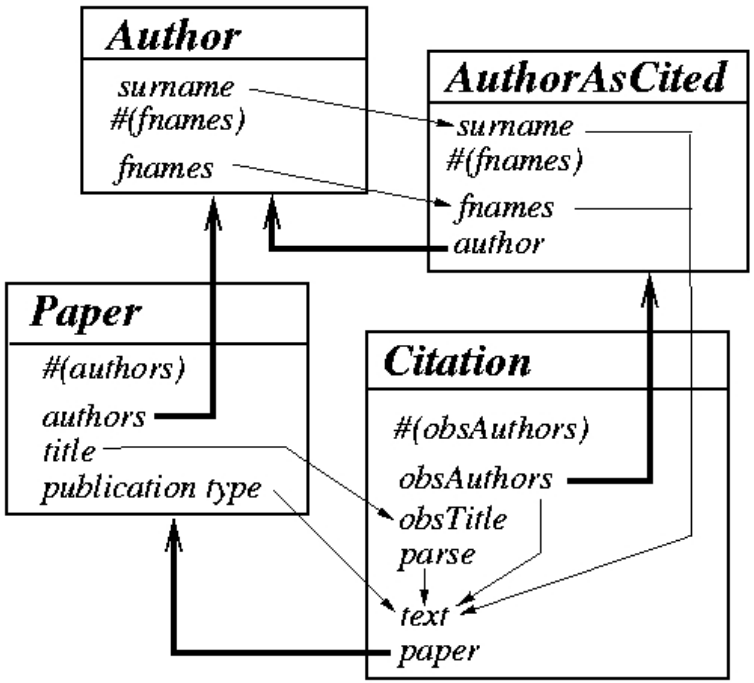
An example of a ***Dynamic Bayesian Network*** (DBN.)

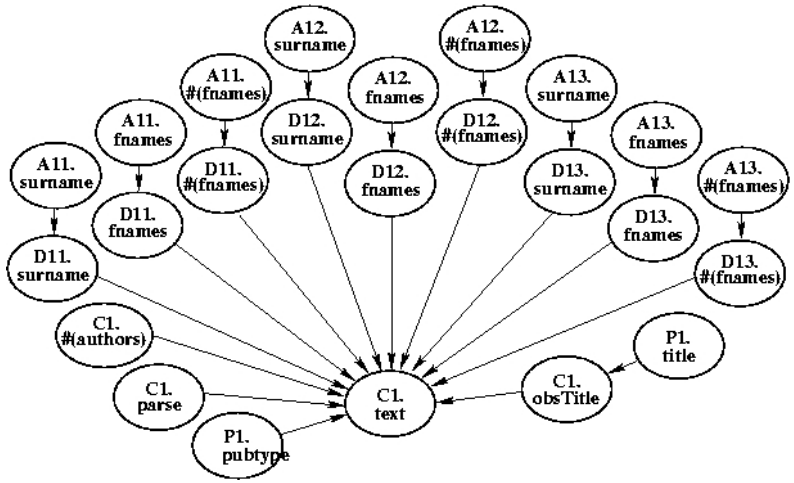
Relational BNs

- In logic, first-order representations:
  - Can encode things more compactly than propositional ones. (E.g., rules of chess.)
  - Are applicable in new situations, where the set of objects is different.
- We want to do the same thing with probabilistic representations
- A general approach: defining network fragments that quantify over objects and putting them together once the set of objects is known.

  In the burglar alarm example, we might quantify over people, their alarms, and their neighbours.
- One specific approach: ***Relational Probabilistic Models*** (RPMs.)

BNs in action: Relational BNs





A relational system for reasoning about papers and citations (as done by Citeseer.) The underlying set of papers is uncertain!

Summary

- Probability is a rigorous formalism for uncertain knowledge
- Joint probability distributions allow us to answer all queries by summing over possible worlds, but are impractical in nontrivial domains.
- Bayesian networks provide a compact representation of joint distributions
- …and a natural representation for (causally induced) conditional independence

Generative Models

- The models we have looked at today are all examples of ***generative models***.

  They define a full distribution over everything in the domain, so possible worlds can be generated from them.

  Generative models can answer any query.
- An alternative is ***discriminative models***.

  There, the query and evidence are decided upon in advance and only the conditional distribution P(Q|E) is learnt.
- Which type is “better”? Current opinion is divided.

A Leading Question

- Bayesian networks help us represent things compactly… but how can that translate into better inference algorithms?
- How would you do inference using a Bayes net?