COMP3308/3608, Lecture 11

ARTIFICIAL INTELLIGENCE

Probabilistic Reasoning.

Bayesian Networks and Inference.

Reference: Russell and Norvig, ch.13 and ch.14

Witten, Frank, Hall and Pal, ch.9 pp.336-352

Outline

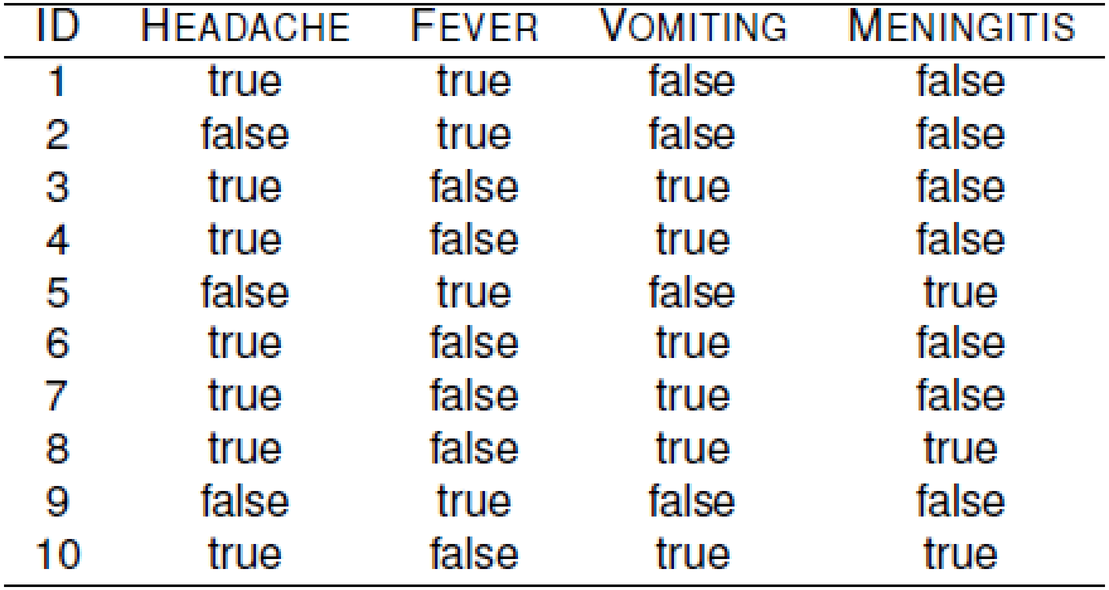
- **Probabilistic reasoning**
  - **Probability basics**
  - **Inference using full joint distribution**
  - **Conditional independence**
  - **Product rule, chain rule, theorem of total probability**
  - **Bayes Theorem and its use for inference**

- **Bayesian networks**
  - **Representation and assumptions**
  - **Exact inference by enumeration**
  - **Exact inference by variable elimination**
  - **Relation between Naïve Bayes and Bayesian networks**
  - **Learning CPT from data**

Probabilistic Reasoning

Probability Basics - Example

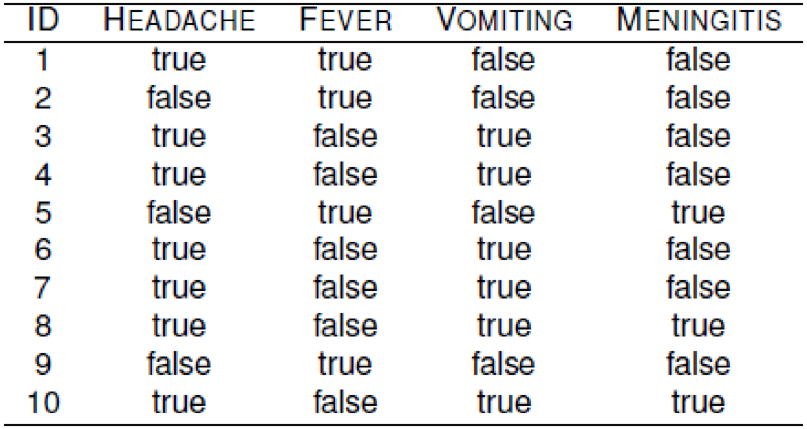
- Let’s start with an example:
  - 4 Boolean variables: Headache, Fever, Vomiting and Meningitis
  - Headaches, fever and vomiting are common symptoms of the disease meningitis



ID	HEADACHE	FEVER	VOMITING	MENINGITIS
1	true	true	false	false
2	false	true	false	false
3	true	false	true	false
4	true	false	true	false
5	false	true	false	true
6	true	false	true	false
7	true	false	true	false
8	true	false	true	true
9	false	true	false	false
10	true	false	true	true

Probability Basics (1)

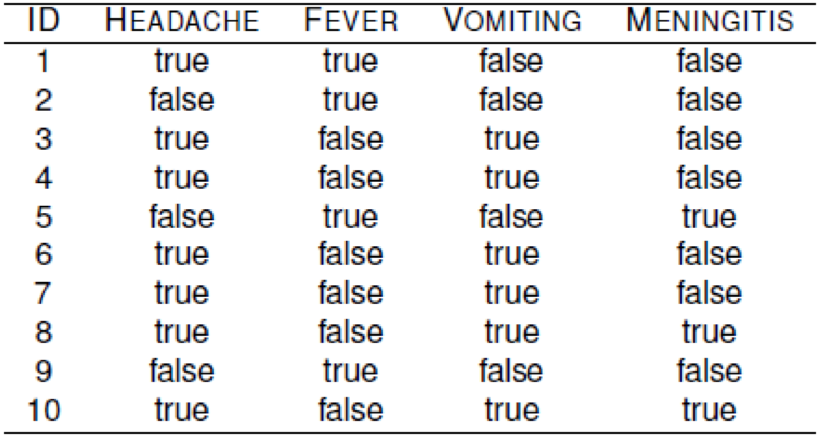
- We have 4 *random variables*: Headache, Fever, Vomiting and Meningitis
- *Domain of a random variable* – the values it can take
  - Our variables are binary, so their domain is {true, false}
  - Domain of a coin flip: {head, tail}
  - Domain of a die roll: {1, 2, 3, 4, 5, 6}
- *Event (proposition)* – an assignment of a set of variables with values, e.g. Headache=true ∧ Vomiting=false



ID	Headache	Fever	Vomiting	Meningitis
1	true	true	false	false
2	false	true	false	false
3	true	false	true	false
4	true	false	true	false
5	false	true	false	true
6	true	false	true	false
7	true	false	true	false
8	true	false	true	true
9	false	true	false	false
10	true	false	true	true

Probability Basics (2)

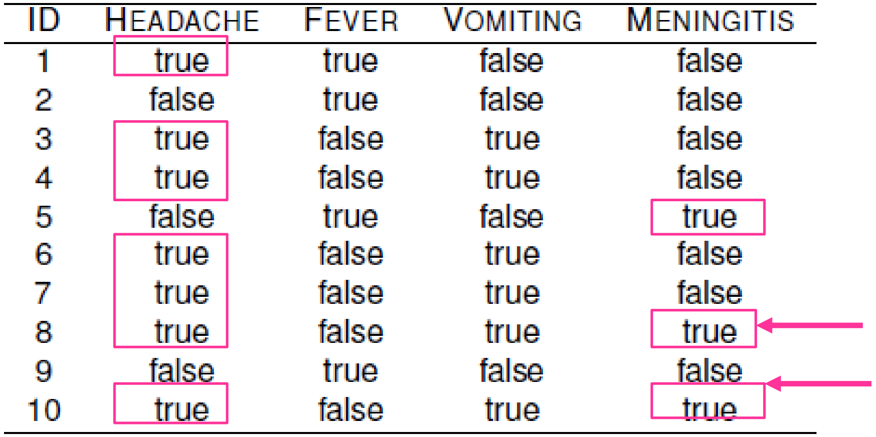
- ***Probability*** **P(A=a) is the fraction of times A takes the value a**
  - **P(Fever=true)=4/10, P(Headache=false)=3/10**
  - **P(head)=P(tail)=0.5 for a fair coin**



ID	HEADACHE	FEVER	VOMITING	MENINGITIS
1	true	true	false	false
2	false	true	false	false
3	true	false	true	false
4	true	false	true	false
5	false	true	false	true
6	true	false	true	false
7	true	false	true	false
8	true	false	true	true
9	false	true	false	false
10	true	false	true	true

Probability Basics (3)

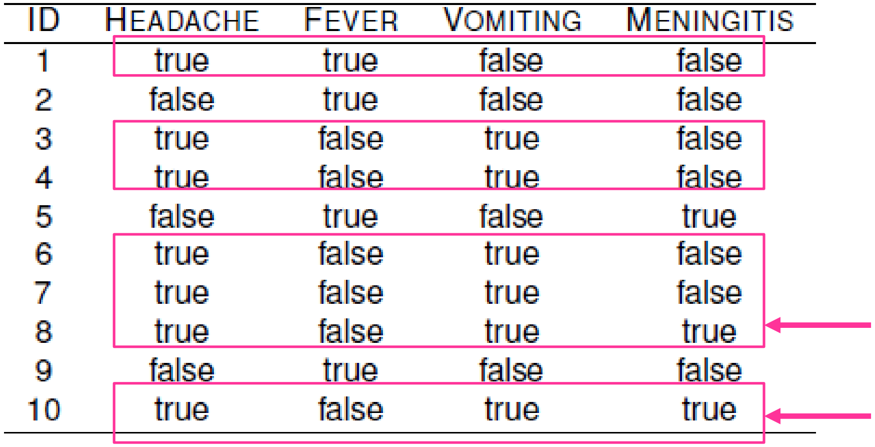
- ***Joint probability of 2 variables*** **P(A=a, B=b) is the probability of A=a and B=b to occur together, i.e. of both variables to take specific values**
  - **P(Meningitis=true, Headache=true)= 2/10**
- **Note that P(A=a, B=b) is a shorthand for P(A=a ∧ B=b)**
- **Joint probability can be similarly defined for more than 2 variables**



ID	HEADACHE	FEVER	VOMITING	MENINGITIS
1	true	true	false	false
2	false	true	false	false
3	true	false	true	false
4	true	false	true	false
5	false	true	false	true
6	true	false	true	false
7	true	false	true	false
8	true	false	true	true
9	false	true	false	false
10	true	false	true	true

Probability Basics (4)

- ***Conditional probability*** **P(A=a| B=b) is the probability of A=a given that we already know that B=b**
    - **P(Meningitis=true| Headache=true)= 2/7**



ID	HEADACHE	FEVER	VOMITING	MENINGITIS
1	true	true	false	false
2	false	true	false	false
3	true	false	true	false
4	true	false	true	false
5	false	true	false	true
6	true	false	true	false
7	true	false	true	false
8	true	false	true	true
9	false	true	false	false
10	true	false	true	true

- ***Probability distribution*** **of a variable A is a <u>data structure</u> that contains the probability for each possible value of A**
    - **P(Meningitis)=<0.3, 0.7>** ← the values should sum to 1

bold

1st value is for true, 2nd is for false

Probability Basics (5)

- ***Joint probability distribution*** **of 2 variables A and B is a data structure (a table) that shows the probability for all combinations of values of A and B**
  - **P(Vomiting, Meningitis):**

Vomiting	Meningitis	P(Vomiting, Meningitis)
T	T	1/10
T	F	5/10
F	T	1/10
F	F	3/10

Probability Basics (6)

- The *full joint probability distribution* (i.e. of all variables) for our example is:

H	F	V	M	P(H,F,V,M)
T	T	T	T	0/10
T	T	T	F	0/10
T	T	F	T	…
T	T	F	F	…
T	F	T	T	…
T	F	T	F	4/10
T	F	F	T	…
T	F	F	F	…

H	F	V	M	P(H,F,V,M)
F	T	T	T	…
F	T	T	F	…
F	T	F	T	…
F	T	F	F	…
F	F	T	T	…
F	F	T	F	…
F	F	F	T	…
F	F	F	F	…

- All numbers must sum to 1 as this is the *full* joint probability table
- Given N variables, each with k values => k^N entries in the table (this is a lot!)

Axioms and Theorems of Probability

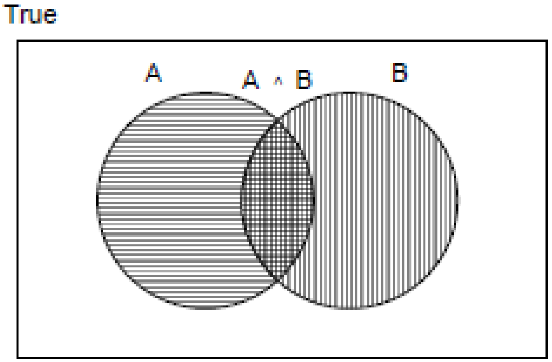
Axioms:

1) All probabilities are between 0 and 1: $P(A) \in [0,1]$

2) Valid propositions have a probability of 1 and invalid propositions have a probability of 0, i.e. P(true)=1 and P(false)=0

3) Probability of a disjunction:

$$P(A \vee B) = P(A) + P(B) - P(A \wedge B)$$



Some theorems:

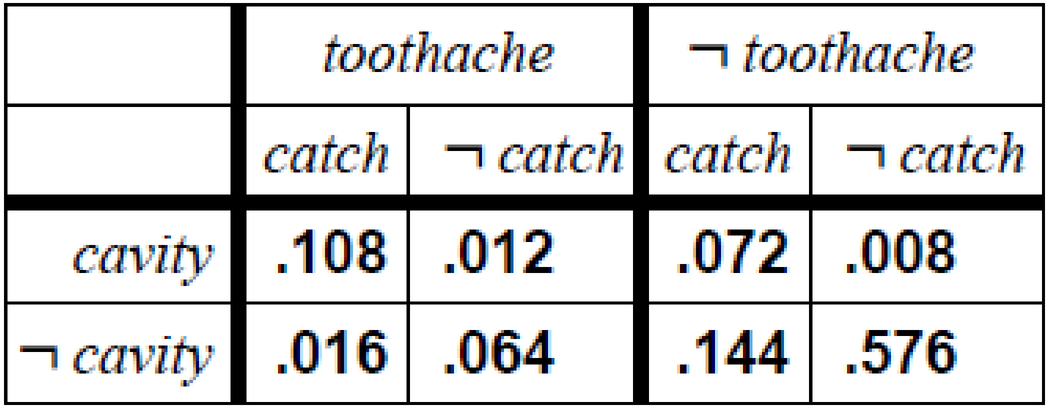
- For an event A: $P(\sim A)=1-P(A)$
- For a random variable X with k different values $x_1, \dots, x_k$:

  $P(X=x_1)+\dots+P(X=x_k)=1$
- => the probability distribution of a single variable must sum to 1

Inference Using the Full Joint Distribution

- **Given the full joint probability table, we can compute the probability of any event in the domain by summing over the cells (atomic events) where the event is true**
- **Example: 3 Boolean variables: Toothache, Cavity and Catch**
  - **(What is Catch? Catch=true – the dentist's steel probe catches in our tooth)**

- **P(toothache)=?**
- **This is a shorthand for P(Toothache=true)**



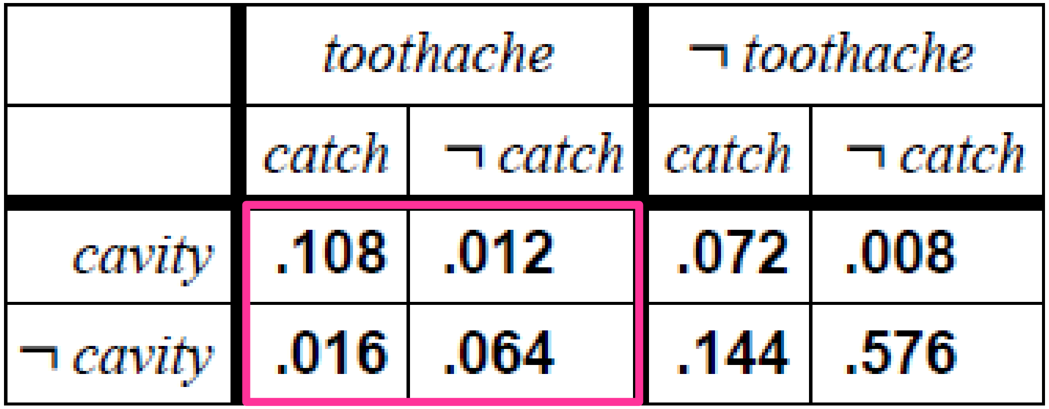
	toothache		*¬ toothache*	
	catch	*¬ catch*	*catch*	*¬ catch*
cavity	.108	.012	.072	.008
¬ cavity	.016	.064	.144	.576

http://cartoonsmix.com/cartoons/cartoon-tooth-pain.html

Inference Using the Full Joint Distribution

- Given the full joint probability table, we can compute the probability of any event in the domain by summing over the cells (atomic events) where the event is true
- Example: 3 Boolean variables: Toothache, Cavity and Catch
    - (What is Catch? Catch=true – the dentist's steel probe catches in our tooth)

- P(toothache) = 0.108+0.12+0.16+0.064 = 0.2
- This is a shorthand for P(Toothache=true)

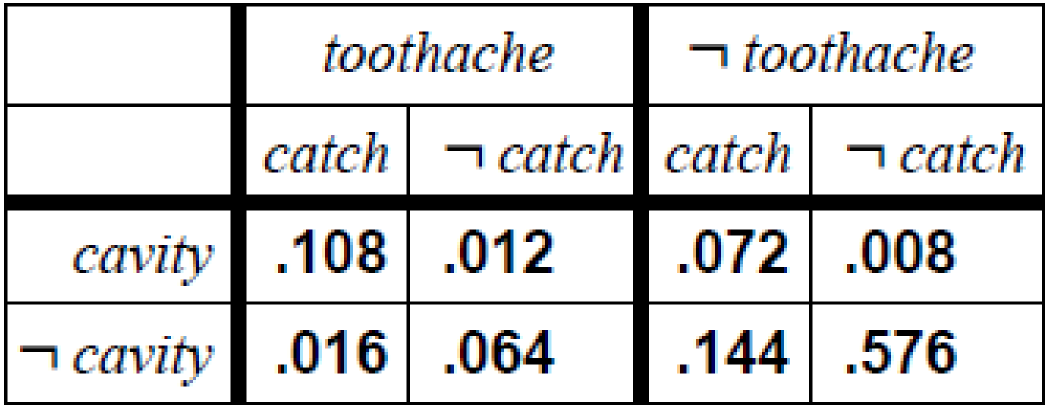


	toothache		*¬ toothache*	
	catch	*¬ catch*	*catch*	*¬ catch*
cavity	.108	.012	.072	.008
¬ cavity	.016	.064	.144	.576

This is called ***marginalization***, *or* ***summing out*** because the variables other than Toothache were summed out (their values were added)

Inference Using the Full Joint Distribution (2)

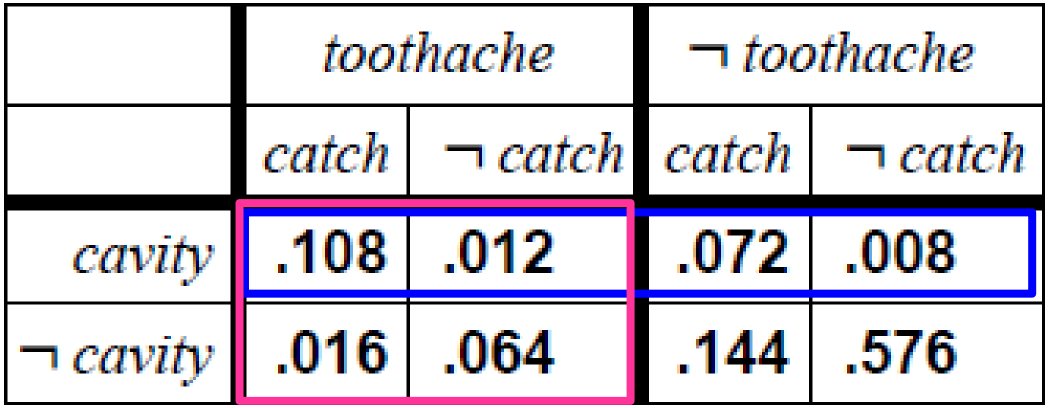
- **P(cavity ∨ toothache) = ?**
- **This is a shorthand for P(Cavity=true ∨ Toothache=true)**



	toothache		*¬ toothache*	
	catch	*¬ catch*	*catch*	*¬ catch*
cavity	.108	.012	.072	.008
¬ cavity	.016	.064	.144	.576

Inference Using the Full Joint Distribution (2)

- **P(cavity ∨ toothache) = 0.108+0.012+0.072+0.008+0.016+0.064 =0.28**
- **This is a shorthand for P(Cavity=true ∨ Toothache=true)**



	toothache		*¬ toothache*	
	catch	*¬ catch*	*catch*	*¬ catch*
cavity	.108	.012	.072	.008
¬ cavity	.016	.064	.144	.576

Conditional Probability, Product and Chain Rules

1) Definition of *conditional probability*:

$$P(a|b)=\frac{P(a,b)}{P(b)}, if\ P(b)\neq 0$$

NB: , means $\wedge$, so this is the same:

$$P(a|b)=\frac{P(a\wedge b)}{P(b)}, if\ P(b)\neq 0$$

2) The *product rule* (follows from 1):

$$P(a,b)=P(a|b)P(b)$$

$$P(a,b)=P(b,a)=P(b|a)P(a)$$

<- It works the other way too

commutative property

3) *Chain rule* – derived by successive application of the product rule:

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

The decomposition holds for any order of the variables

$$P(a,b)=P(a|b)P(b)$$

$$P(a,b,c)=P(a|b,c)P(b|c)P(c)$$

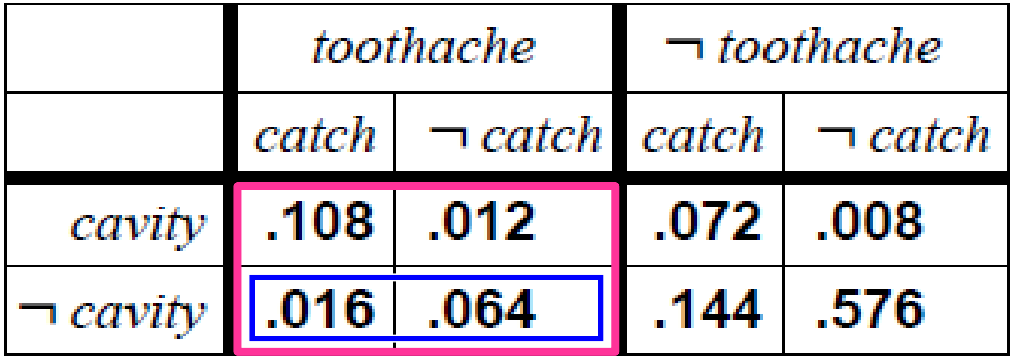
$$P(a,b,c,d)=P(a|b,c,d)P(b|c,d)P(c|d)P(d)$$

Let’s do Some Reasoning Using Conditional Probability!

- **P(~cavity| toothache) = ?**
- **This is a shorthand for P(Cavity=false| Toothache=true)**

$$P(\neg cavity \mid toothache) = \frac{P(\neg cavity, toothache)}{P(toothache)} =$$

$$= \frac{0.016 + 0.064}{0.108 + 0.012 + 0.016 + 0.064} = \frac{0.08}{0.2} = 0.4$$



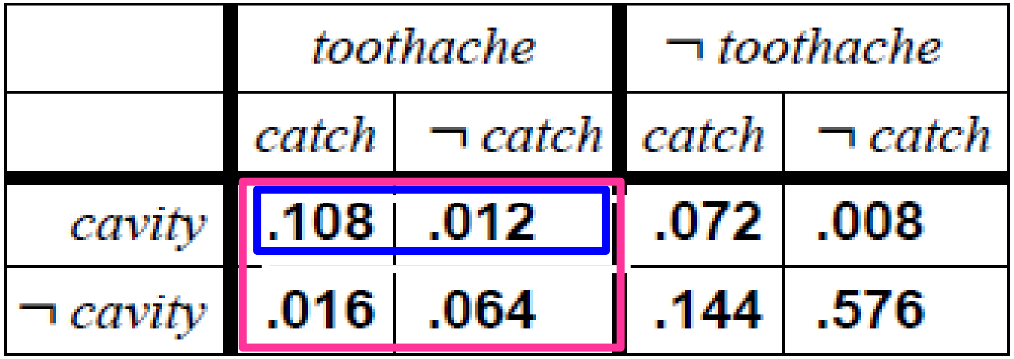
	toothache		*¬ toothache*	
	catch	*¬ catch*	*catch*	*¬ catch*
cavity	.108	.012	.072	.008
¬ cavity	.016	.064	.144	.576

Let’s do Some Reasoning Using Conditional Probability! (2)

- And let’s calculate the opposite: P(cavity| toothache) = ?
- This is a shorthand for P(Cavity=true| Toothache=true)

$$P(cavity \mid toothache) = \frac{P(cavity, toothache)}{P(toothache)} =$$

$$= \frac{0.108 + 0.012}{0.108 + 0.012 + 0.016 + 0.064} = \frac{0.12}{0.2} = 0.6$$



	toothache		¬ toothache	
	catch	¬ catch	catch	¬ catch
cavity	.108	.012	.072	.008
¬ cavity	.016	.064	.144	.576

- Good - 0.6 and 0.4 sum up to 1 as expected!

=> P(Cavity| toothache) = <0.6, 0.4>

Normalization

- **The denominator is the same and can be viewed as a normalization constant α:**

$$P(cavity \mid toothache) = \alpha P(cavity, toothache) =$$

$$= \alpha * (0.108 + 0.012) = \alpha * 0.12$$

$$P(\neg cavity \mid toothache) = \alpha P(\neg cavity, toothache) =$$

$$\alpha * (0.016 + 0.064) = \alpha * 0.08$$

- **The probability distribution with α :**

$$\mathbf{P}(Cavity \mid toothache) = \alpha < 0.12, 0.08 >$$

- **Computing the normalization constant:**

α =1/(0.12+0.08)= 1/0.2

- **Final probability distribution without α :**

$$\mathbf{P}(Cavity \mid toothache) =< 0.12/0.2,\ 0.08/0.2 >=< 0.6, 0.4 >$$

Normalization – More Generally:

- **Another way to write this using the P notation:**

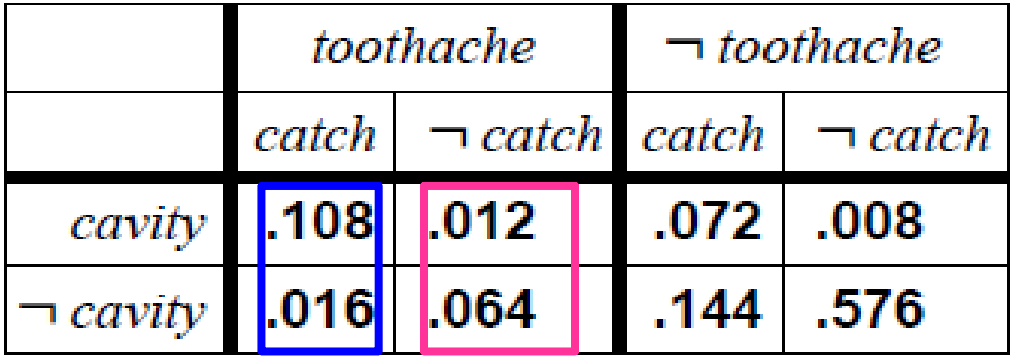
$$\mathbf{P}(Cavity \mid toothache) = \alpha \mathbf{P}(Cavity, toothache) =$$

$$= \alpha[\mathbf{P}(Cavity, toothache, catch) + \mathbf{P}(Cavity, toothache, \neg catch)] =$$

$$= \alpha[< 0.108, 0.016 > + < 0.012, 0.0064 >] =$$

$$= \alpha < 0.12, 0.08 >= \quad < 0.12/(0.12+0.08), 0.08/(0.12+006) >=$$

$$=< 0.6, 0.4 >$$



	toothache		*¬ toothache*	
	catch	*¬ catch*	*catch*	*¬ catch*
cavity	.108	.012	.072	.008
¬ cavity	.016	.064	.144	.576

Inference by Enumeration Using Full Joint Distribution

General rule:

- **X – query variable (e.g. Cavity)**
- **E – evidence (e.g. Toothache)**
- **e – the observed values for E (e.g. true)**
- **H – the remaining (*hidden*) variables: H = X - E**

$$\mathbf{P}(X|E=e) = \alpha\mathbf{P}(X,E=e) = \alpha\sum_{h}\mathbf{P}(X,E=e,H=h)$$

The summation is over the values of the hidden variables, i.e. we are *summing out* the hidden variables

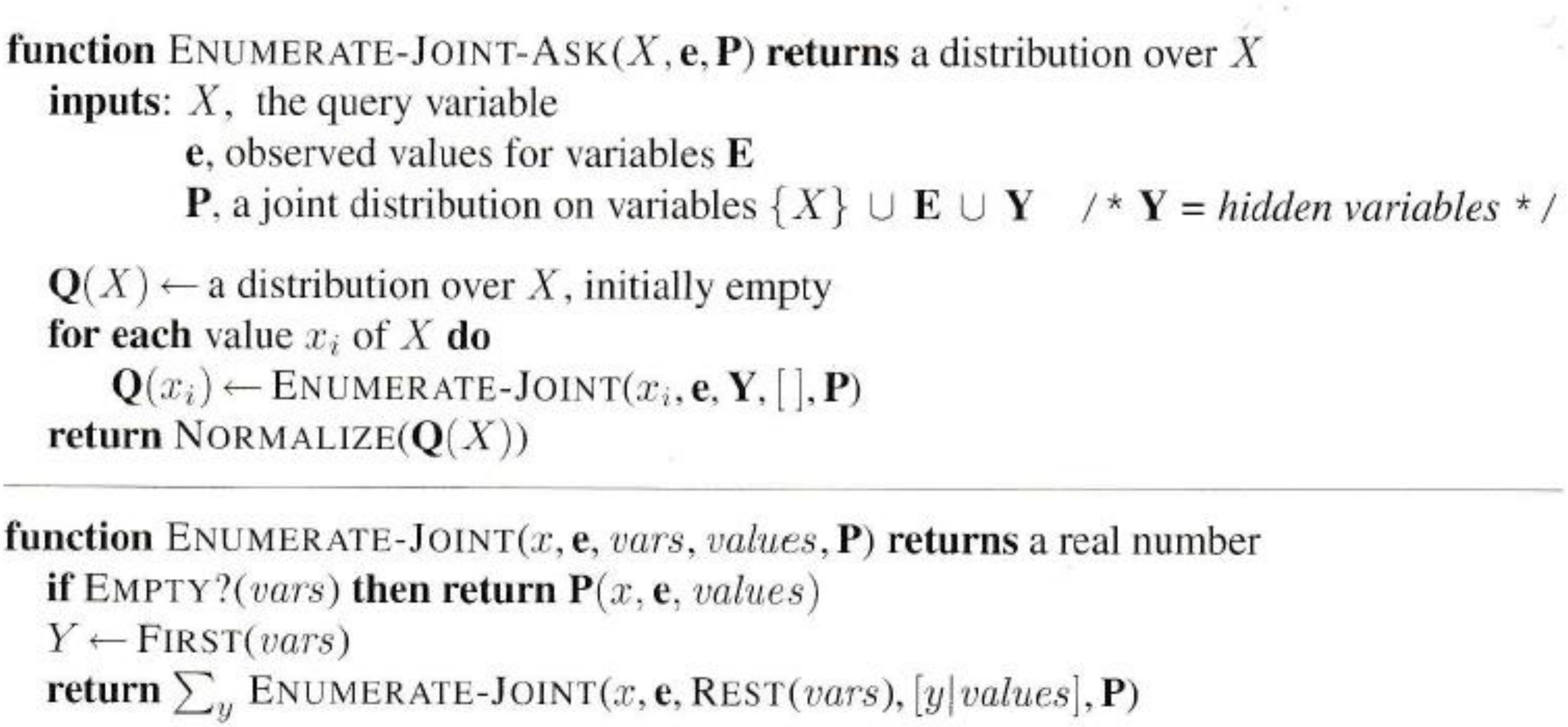
$$\mathbf{P}(Cavity \mid Toothache = true) = \alpha\mathbf{P}(Cavity \mid Toothache = true) =$$

$$= \alpha \sum_{h=true,\ false} \mathbf{P}(Cavity, Toothache = true, Catch = h)$$

$$= \alpha[\mathbf{P}(Cavity, Toothache = true, Catch = true) +$$

$$\mathbf{P}(Cavity, Toothache = true, Catch = false)]$$

Pseudocode



```
function ENUMERATE-JOINT-ASK($X$, e, P) returns a distribution over $X$
   inputs: $X$, the query variable
           e, observed values for variables E
           P, a joint distribution on variables $\{X\} \cup \mathbf{E} \cup \mathbf{Y}$   /* Y = hidden variables */

   Q($X$) ← a distribution over $X$, initially empty
   for each value $x_i$ of $X$ do
       Q($x_i$) ← ENUMERATE-JOINT($x_i$, e, Y, [ ], P)
   return NORMALIZE(Q($X$))
```

```
function ENUMERATE-JOINT($x$, e, $vars$, $values$, P) returns a real number
   if EMPTY?($vars$) then return P($x$, e, $values$)
   $Y$ ← FIRST($vars$)
   return $\sum_y$ ENUMERATE-JOINT($x$, e, REST($vars$), $[y|values]$, P)
```

- **It doesn’t scale well:**
  - **For n Boolean variables it requires a joint probability table of size O(2^n) and O(2^n) time to process it**
  - **Impractical for real problems: hundreds of random variables**

Independence

- 2 events A and B are independent, if:

  $P(A,B) = P(A)*P(B)$

- This condition is equivalent to:

  $P(A|B) = P(A)$ and

  $P(B|A) = P(B)$

  Can you show this?

- Independence is domain knowledge!
- It makes the inference easier

Independence - Example

product rule:

$P(A,B) = P(A|B)P(B)$

- **Dental example: Suppose that we have 1 more binary variable in the dental example: Weather with values *cloudy* and *sunny*. Task – compute:**

$$P(Weather = cloudy, toothache, catch, cavity) =$$

product rule

$$|= P(Weather = cloudy \mid toothache, catch, cavity) P(toothache, catch, cavity)$$

$$= P(Weather = cloudy)$$

- **However, a person's dental problems <u>do not influence</u> the weather**

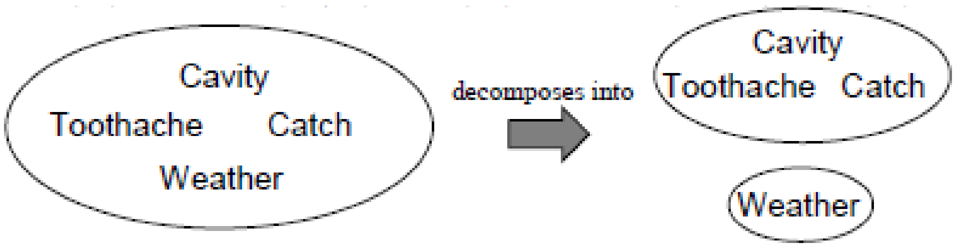
=> $P(Weather = cloudy)|toothache, catch, cavity) = P(Weather = cloudy)$

independence rule:

$P(A|B) = P(A)$

=> $$P(Weather = cloudy, toothache, catch, cavity) =$$
$$= P(Weather = cloudy)P(toothache, catch, cavity)$$

⇒ **Instead of storing a table with 2^4 entries, we can store $2^3 + 2^1$. This also reduces the complexity of the inference problem.**



Conditional Independence

- **Absolute independence is very useful property but it is rare in practice**
- **Although random variables are rarely absolutely independent, they are often *conditionally independent***
- **Example:**
  1) **If I have a cavity, the probability that the probe catches in my tooth doesn't depend on whether I have a toothache (it depends on the skill of the dentist):**

     $$P(catch \mid toothache, cavity) = P(catch \mid cavity)$$

  2) **The same independence holds if I do not have a cavity:**

     $$P(catch \mid toothache, \neg cavity) = P(catch \mid \neg cavity)$$

  - **In other words, Catch is conditionally independent of Toothache given Cavity:**

    $$\mathbf{P}(Catch \mid Toothache, Cavity) = \mathbf{P}(Catch \mid Cavity)$$

Conditional Independence (2)

- **Similarly, based on domain knowledge we can assert that Toothache is conditionally independent of Catch given Cavity:**

$$\mathbf{P}(Toothache \mid Catch, Cavity) = \mathbf{P}(Toothache \mid Cavity) \qquad (1)$$

- **Let's see how conditional independence simplifies inference:**

$$\begin{aligned}
&\mathbf{P}(Toothache, Catch, Cavity) = \\
&= \mathbf{P}(Toothache \mid Catch, Cavity)\mathbf{P}(Catch \mid Cavity)\mathbf{P}(Cavity) = && \text{chain rule} \\
&= \mathbf{P}(Toothache \mid Cavity)\mathbf{P}(Catch \mid Cavity)\mathbf{P}(Cavity) && \text{from (1)}
\end{aligned}$$

- **So we need $2^2+2^2+2^1=10$ probability numbers or 5 independent numbers (as counterparts sum to 1) which is less than without conditional independence**
- **In most cases, conditional independence reduces the size of the representation of the joint distribution from O(2^n) to O(n)**
  - **n is the num. of variables conditionally independent, given another variable**
- **=> Conditional independence assertions: 1) allow probabilistic systems to scale up and 2) are more available than absolute independence assertions**

Bayes Theorem

- **We know it already!**
- **But this time we will use it for reasoning not classification**
- **We can see where it comes from – from the product rule:**

$$P(a,b) = P(a \mid b)P(b)$$

$$P(a,b) = P(b,a) = P(b \mid a)P(a)$$ **commutativity**

- **Equating the 2 right-hand sides:**

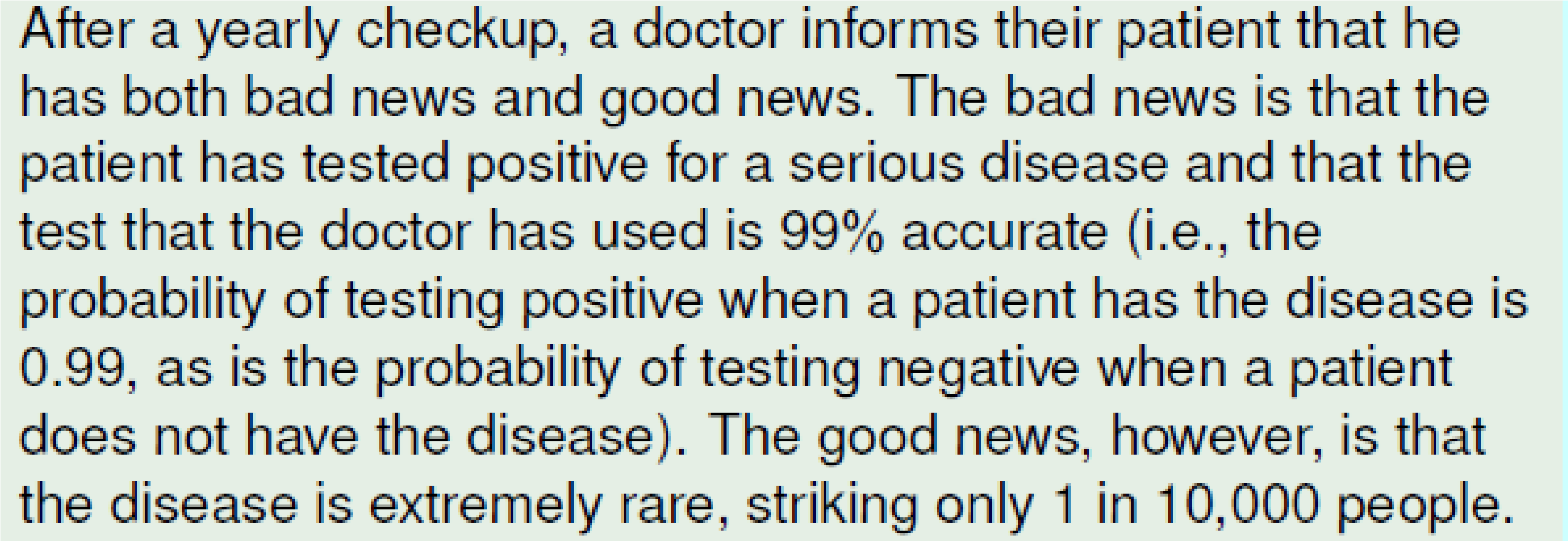
$$P(a \mid b)P(b) = P(b \mid a)P(a)$$

=> $$P(a \mid b) = \frac{P(b \mid a)P(a)}{P(b)}$$ **Bayes Theorem**

$$\mathbf{P}(A \mid B) = \frac{\mathbf{P}(B \mid A)\mathbf{P}(A)}{\mathbf{P}(B)}$$ **<- More general form for multivalued variables using the P notation**

$$\mathbf{P}(A \mid B) = \alpha \mathbf{P}(B \mid A)\mathbf{P}(A)$$ **<- General form with α**

Using the Bayes Theorem for Reasoning - Example



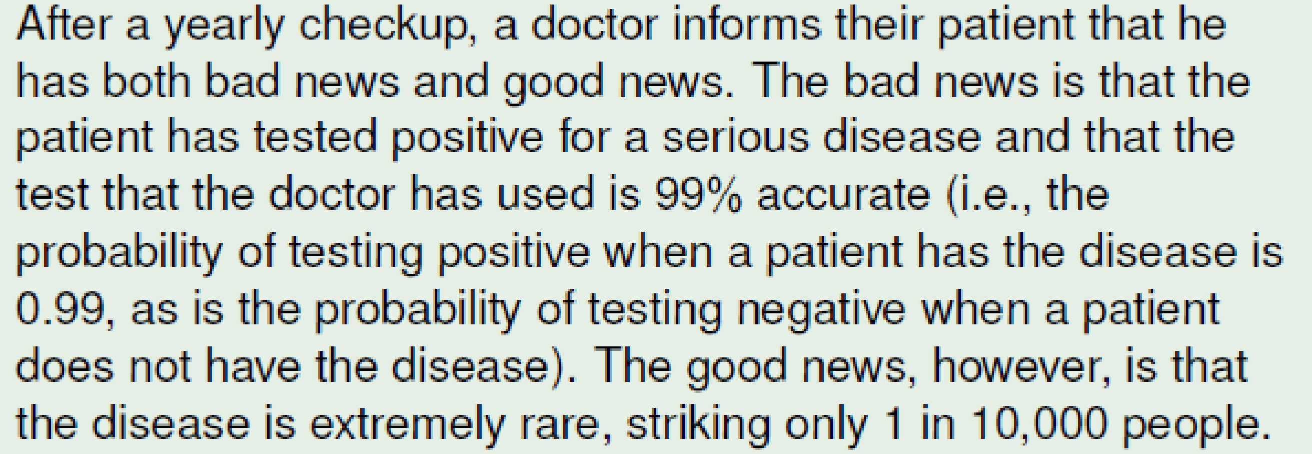
After a yearly checkup, a doctor informs their patient that he has both bad news and good news. The bad news is that the patient has tested positive for a serious disease and that the test that the doctor has used is 99% accurate (i.e., the probability of testing positive when a patient has the disease is 0.99, as is the probability of testing negative when a patient does not have the disease). The good news, however, is that the disease is extremely rare, striking only 1 in 10,000 people.

Q1. What is the probability that the patient has the disease?

Q2. Why is the rarity of the disease good news given that the patient has tested positive for it?

Example from Kelleher et al. 2015, MIT Press

Answering Q1



After a yearly checkup, a doctor informs their patient that he has both bad news and good news. The bad news is that the patient has tested positive for a serious disease and that the test that the doctor has used is 99% accurate (i.e., the probability of testing positive when a patient has the disease is 0.99, as is the probability of testing negative when a patient does not have the disease). The good news, however, is that the disease is extremely rare, striking only 1 in 10,000 people.

Q1. What is the probability that the patient has the disease?

Variables: D – hasDisease {true, false} , T – positiveTest {true, false}

$P(d \mid t) = ?$ **This is a shorthand for:** $P(D = true \mid T = true) = ?$

$$P(d \mid t) = \frac{P(t \mid d)P(d)}{P(t)} = \frac{0.99 * 0.0001}{P(t)}$$

To calculate P(t) we will use the *Theorem of Total Probability*

Theorem (Rule) of Total Probability

- **The unconditional probability for any event X is:**

$$P(X) = \sum_i P(X \mid Y_i) P(Y_i)$$

where Y is another event and Y_i are all possible outcomes for Y

- **This is actually the summing out rule that we used before!**

$$P(t) = P(t \mid d)P(d) + P(t \mid \neg d)P(\neg d) =$$

$$= 0.99 * 0.0001 + 0.01 * 0.9999 = 0.0101$$

$$P(d \mid t) = \frac{P(t \mid d)P(d)}{P(t)} = \frac{0.99 * 0.0001}{0.0101} = 0.0098$$

=> The probability to have the disease, given that the test is positive is 0.98%. This is very low: <1%.

Answering Q2

- **Q2. Why is the rarity of the disease good news given that the patient has tested positive for it?**
- **Because P(d|t) is proportional to P(d), so a lower P(d) means a lower P(d|t)**

$$P(d \mid t) = \frac{P(t \mid d)P(d)}{P(t)} = \frac{0.99 * \boxed{0.0001}}{0.0101} = 0.0098$$

- **In other words, the Bayes theorem explicitly includes the prior probability of an event when calculating the likelihood of that event based on evidence**
- **Hence, for our example, when the disease is much rarer than the accuracy of the test, a positive test result does <u>not</u> mean the disease is likely**

Summary

- **Probability is a rigorous formalism for representing uncertain knowledge**
- **Given the full joint probability distribution, we can compute the probability of any event in the domain by summing over the cells (atomic events) where the event is true**
- **However, the full joint distribution is usually too large to store and use for inference**
- **Independence and conditional independence allow probabilistic systems to scale up**
- **We also saw how to use several rules and theorems for probabilistic reasoning, e.g. the definition of conditional probability, the product rule, the chain rule, the Bayes theorem and the theorem of total probability**

Bayesian Networks

Bayesian Networks - Motivation

- **To do probabilistic reasoning, we need to know the joint probability distribution of the variables in the domain**
- **But if we have N binary variables, we need 2^N numbers to specify the joint probability distribution**
- **As we saw, usually there are some variables that are independent (fully independent or conditionally independent), so we don't need all these 2^N probability values**
- **We would like to explicitly represent and exploit this independence**
- **Bayesian networks allow us to do this**

Bayesian Networks

- **Bayesian networks are graph-based models that encode structural relationships between variables such as direct influence and conditional independence**
- **Hence, they provide a more compact representation than a full joint probability distribution**

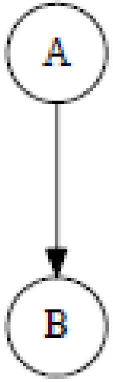
- **Bayesian networks are directed graphs without cycles (DAG – Directed Acyclic Graphs) composed of 3 elements:**
  - **Nodes – there is 1 node for each variable**
  - **Links – each link denotes dependence between 2 nodes: "directly influences"**
  - **Conditional Probability Tables (CPT) for each node, given its parents**

Bayesian Network – Example 1

- 2 binary variables: A and B
- Meaning of the link: A directly influences B (= the values of A directly influence the values of B)
- Terminology: A is a parent node of B, B is a child node of A
- A has no parents, so CPT only shows the unconditional probability distribution
- Each row in CPT sums to 1 => for a binary variable X we can simply state the probability for true P(X=T), as P(X=F)=1-P(X=F)
  - => we don’t need the second CPT column = simplified form
  - The simplified form is typically used

Example from Kelleher et al. 2015, MIT Press

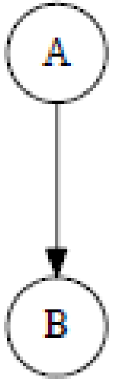
P(A=T)	P(A=F)
0.4	0.6



A	P(B=T|A)	P(B=F|A)
T	0.3	0.7
F	0.4	0.6

simplified CPT

P(A=T)	~~P(A=F)~~
0.4	~~0.6~~

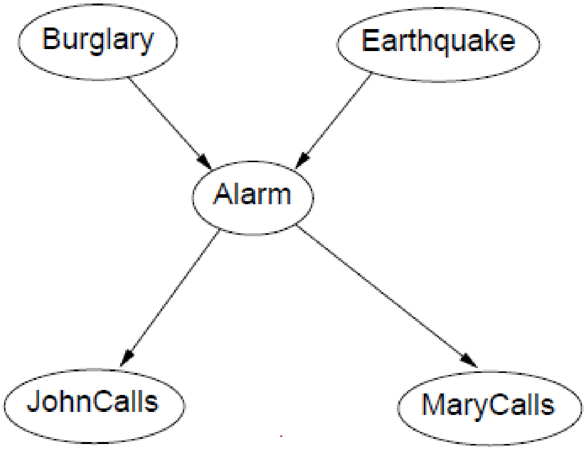


A	P(B=T|A)	~~P(B=F|A)~~
T	0.3	~~0.7~~
F	0.4	~~0.6~~

Bayesian Network – Example 2

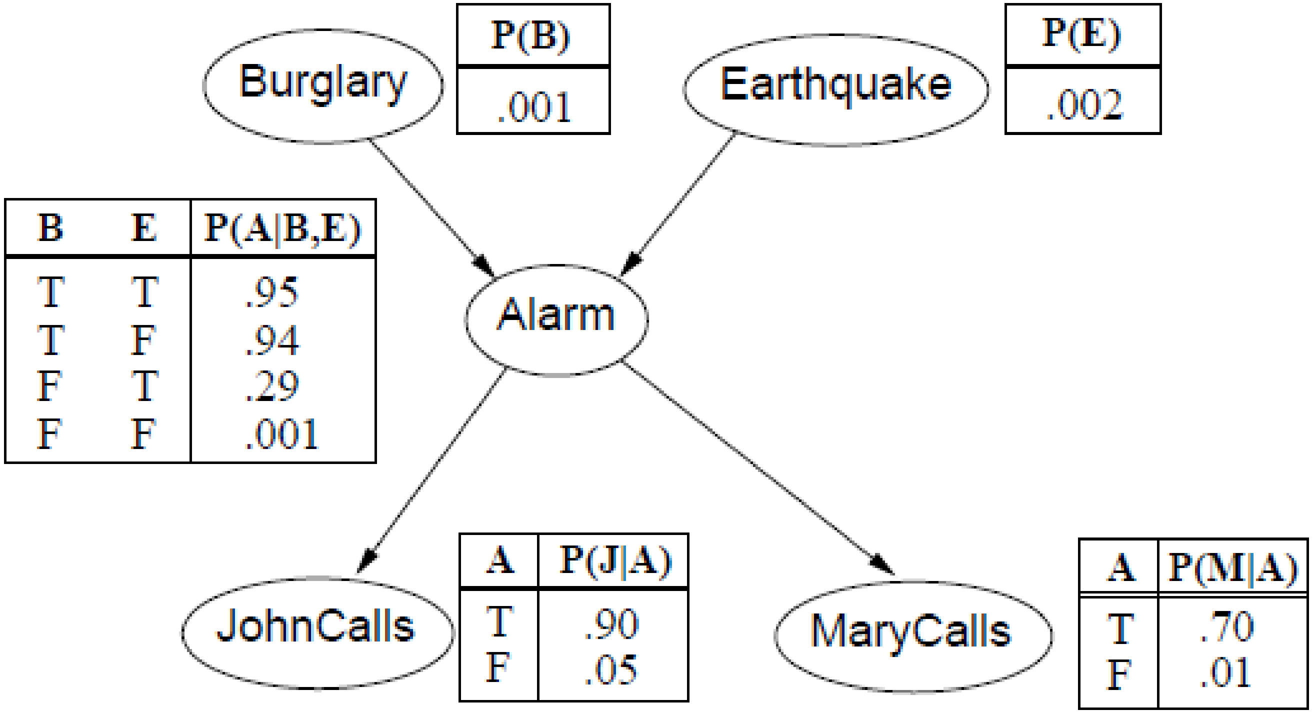
- I am at work, neighbour John calls to say my alarm is ringing, but neighbour Mary doesn't call. Sometimes the alarm is set off by minor earthquakes. Is there a burglar?

- Variables: Burglar, Earthquake, Alarm, JohnCalls, MaryCalls
- The network topology reflects *causal* knowledge:
  - A burglar can set the alarm off
  - An earthquake can set the alarm off
  - The alarm can cause Mary to call
  - The alarm can cause John to call



Example 2 with CPTs

- **For the burglary example we are also given these CPTs:**



P(B)
.001

P(E)
.002

B	E	P(A|B,E)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J|A)
T	.90
F	.05

A	P(M|A)
T	.70
F	.01

Compactness

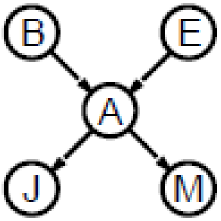
- **Consider a domain with *n* Boolean variables**
- **We have created a Bayesian network**
- **The CPT for a variable X with *k* parents will have 2^k rows**
  - **1 row for each combination of parent values, e.g. Alarm has 2 parents and will have 4 rows**
  - **Each row has 1 number *p* for X=*true*, the number for X=*false* is 1-*p***
- **If each variable has at most *k* parents, the complete Bayesian network requires $O(n^* 2^k)$ numbers, i.e. it grows linearly with *n***
- **For comparison, the full joint distribution table requires $O(2^n)$ numbers which is much bigger – grows exponentially with *n***

- **For our burglary example: 1+1+4+2+2=10 numbers in the CPT of the Bayesian net vs 2^5=32 in the full joint distribution table**

Joint Probability Rule

- **Given a Bayesian network with *n* nodes, the probability of an event x_1, x_2, …, x_n can be computed as:**

$$P(x_1,\ldots,x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(x_i))$$



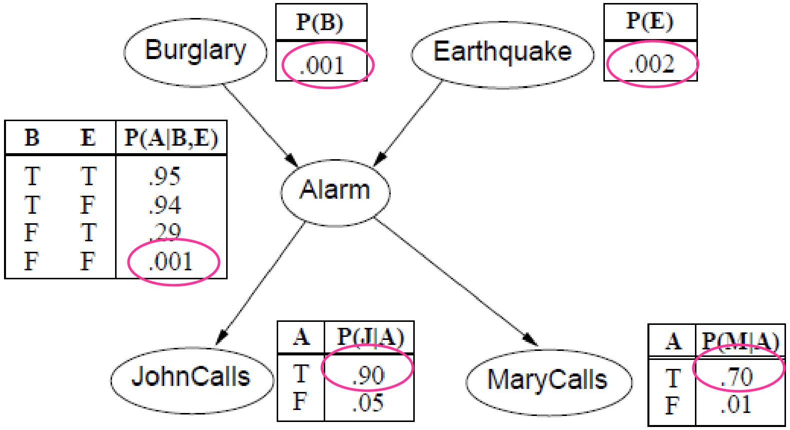
- **The full joint distribution is the product of the local conditional distributions**
- **Example for the burglar net:**

P(j, m, a, ~b, ~e) =

= P(j|a) P(m|a) P(a|~b,~e) P(~b)P(~e) =

= 0.9*0.7*0.001*(1-0.001)*(1-0.002) =

= 0.006281



P(B)
.001

P(E)
.002

B	E	P(A|B,E)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

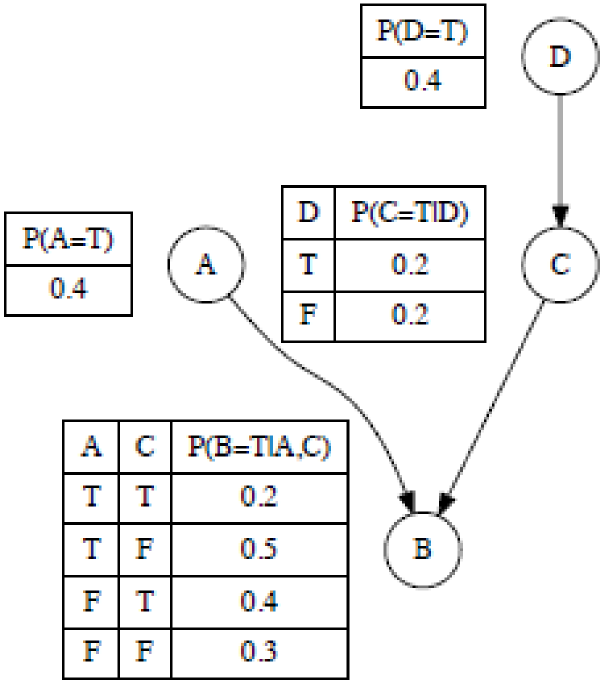
A	P(J|A)
T	.90
F	.05

A	P(M|A)
T	.70
F	.01

- **Notation: As before, j is a shorthand for JohnCalls=T, and ~j for JohnCalls=F**

Another Example

- **Given is the following Bayesian network where all nodes are binary**
- **Compute: P(a, ~b, ~c, d)**



P(D=T)
0.4

D	P(C=T|D)
T	0.2
F	0.2

P(A=T)
0.4

A	C	P(B=T|A,C)
T	T	0.2
T	F	0.5
F	T	0.4
F	F	0.3

Example from Kelleher et al. 2015, MIT Press

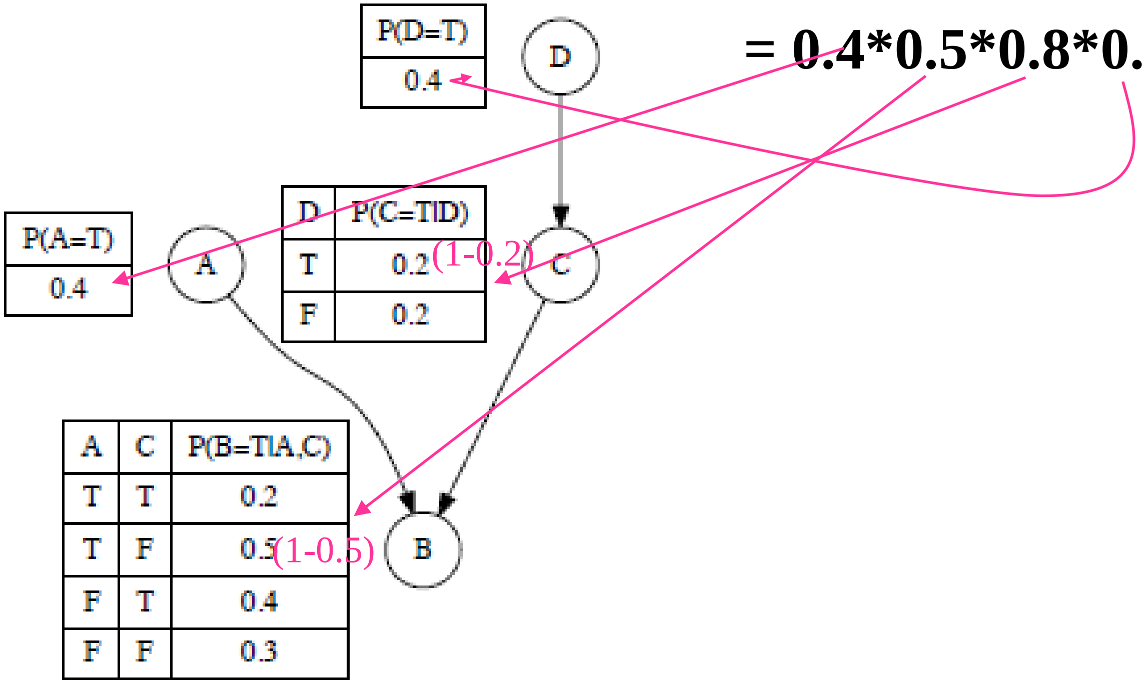
Solution

- **Given is the following Bayesian network where all nodes are binary**
- **Compute: P(a, ~b, ~c, d)**

P(a, ~b, ~c, d) =

= P(a) P(~b|a,~c) P(~c|d) P(d) =

= 0.4*0.5*0.8*0.4 = 0.064



P(D=T)
0.4

P(A=T)
0.4

D	P(C=T|D)
T	0.2
F	0.2

A	C	P(B=T|A,C)
T	T	0.2
T	F	0.5
F	T	0.4
F	F	0.3

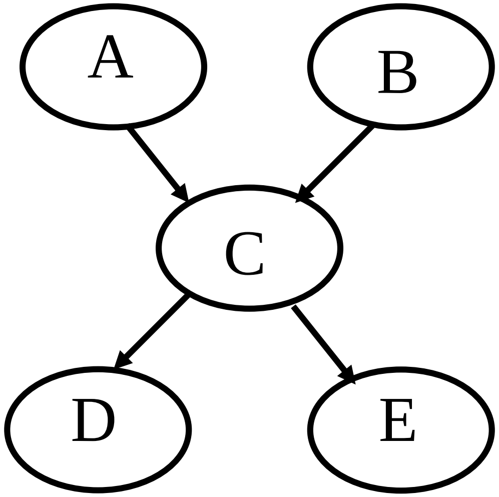
Assumption of Bayesian Networks

$$P(x_1,\ldots,x_n)=\prod_{i=1}^{n}P(x_i\mid Parents(x_i))$$

- **Why do we multiply these probabilities together?**
- **What is the assumption that we make in Bayesian networks, so that we can multiply these probabilities together?**

- **Bayesian network assumption: A node is conditionally independent of its *non-descendants*, given its parents (i.e. if the values of its parents are known)**
  - **Descendants = children, children of children, etc.**
  - **Non-descendants = everything else but we need to exclude the parents =>**
    - **all ancestors except parents, i.e. all grandparents, great-grandparents, etc.**
    - **other children of the parent, i.e. siblings**
-

Illustration



- D is conditionally independent of A, given C
- D is conditionally independent of B, given C

- E is conditionally independent of A, given C
- E is conditionally independent of B, given C

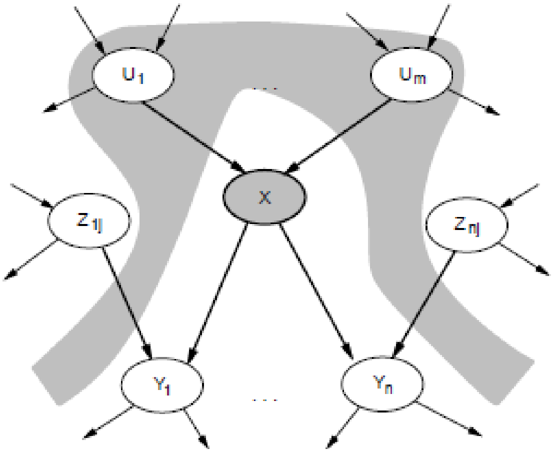
- D is conditionally independent of E, given C
- E is conditionally independent of D, given C

- A is independent of B

Assumption of Bayesian Networks (2)

- **Bayesian Network (BN) assumption: A node is conditionally independent of its *non-descendants*, given its parents**
- **In other words, if we know the values of the parents, than knowing the values of any other non-descendant does not change the probability values of the nodes**
- **This can be written as:**

P(node | parents plus any other non-descendants)=P(node | parents)



X is conditionally independent of its non-descendants (nodes Z_{ij}) given its parents (nodes U_i – the gray area)

Assumption of Bayesian Networks (3)

- **BN assumption:**

  P(node | parents plus any other non-descendants)=P(node | parents)

- **Chain rule:**

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

- **Both together:**

$$P(x_1,\ldots,x_n) = \prod_{i=1}^{n} P(x_i \mid x_i -1,\ldots,x_1) = \prod_{i=1}^{n} P(x_i \mid Parents(x_i))$$

Chain rule

BN's assumption

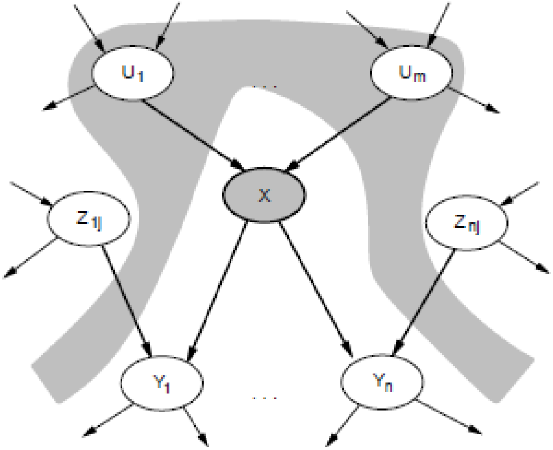
- **Result:** $P(x_1,\ldots,x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(x_i))$

The calculation of joint probabilities reduces to a multiplication of conditional probabilities from the BN

Markov Blanket

- **We saw this independence property:**

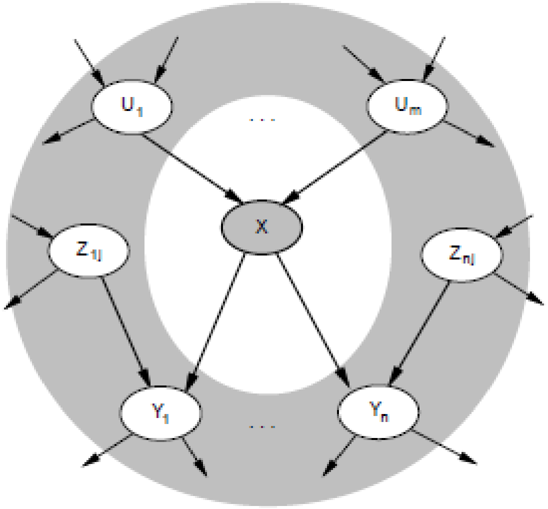
A node is conditionally independent of its *non-descendants*, given its parents



X is conditionally independent of its non-descendants (nodes Z_{ij}) given its parents (nodes U_i – the gray area)

- **Another important independence property:**

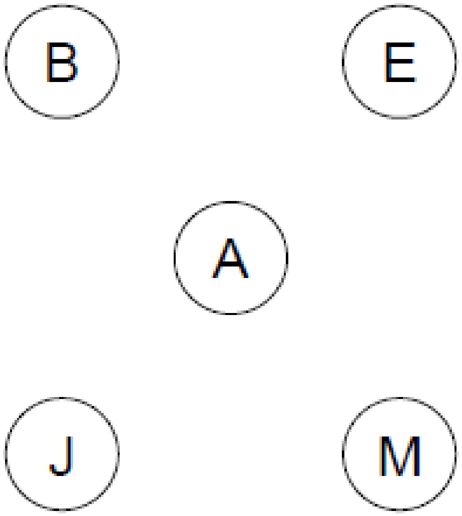
A node is conditionally independent of all other nodes in the network, given its parents, children and children's parents, i.e. given its *Markov blanket*



X is conditionally independent of all other nodes in the network, given its Markov blanket (the gray area)

Constructing Bayesian Networks

- **Bayesian networks are constructed by domain experts and this is relatively easy**
- **Step 1: Add the variables you want to include**



B: there's burglary in your house

E: there's an earthquake

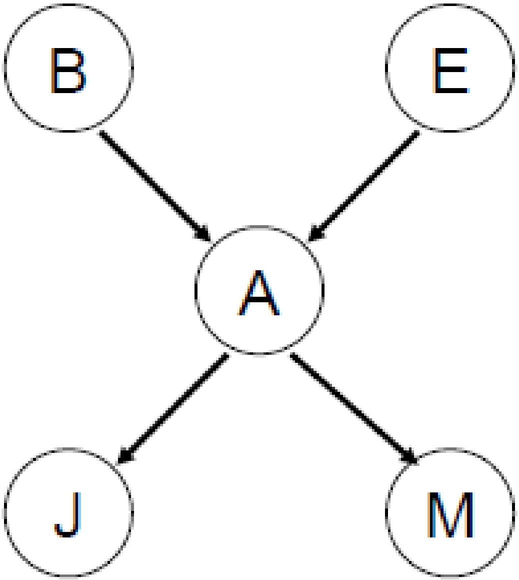
A: your alarm sounds

J: your neighbor John calls you

M: your other neighbor Mary calls you

Constructing Bayesian Networks (2)

- **Step 2: Add directed links**
  - **The graph must be acyclic**
  - **If node X is given parents Q1,…,Qm, this means that any variable that is not a descendant of X is conditionally independent of X given Q1,…,Qm**



B: there's burglary in your house

E: there's an earthquake

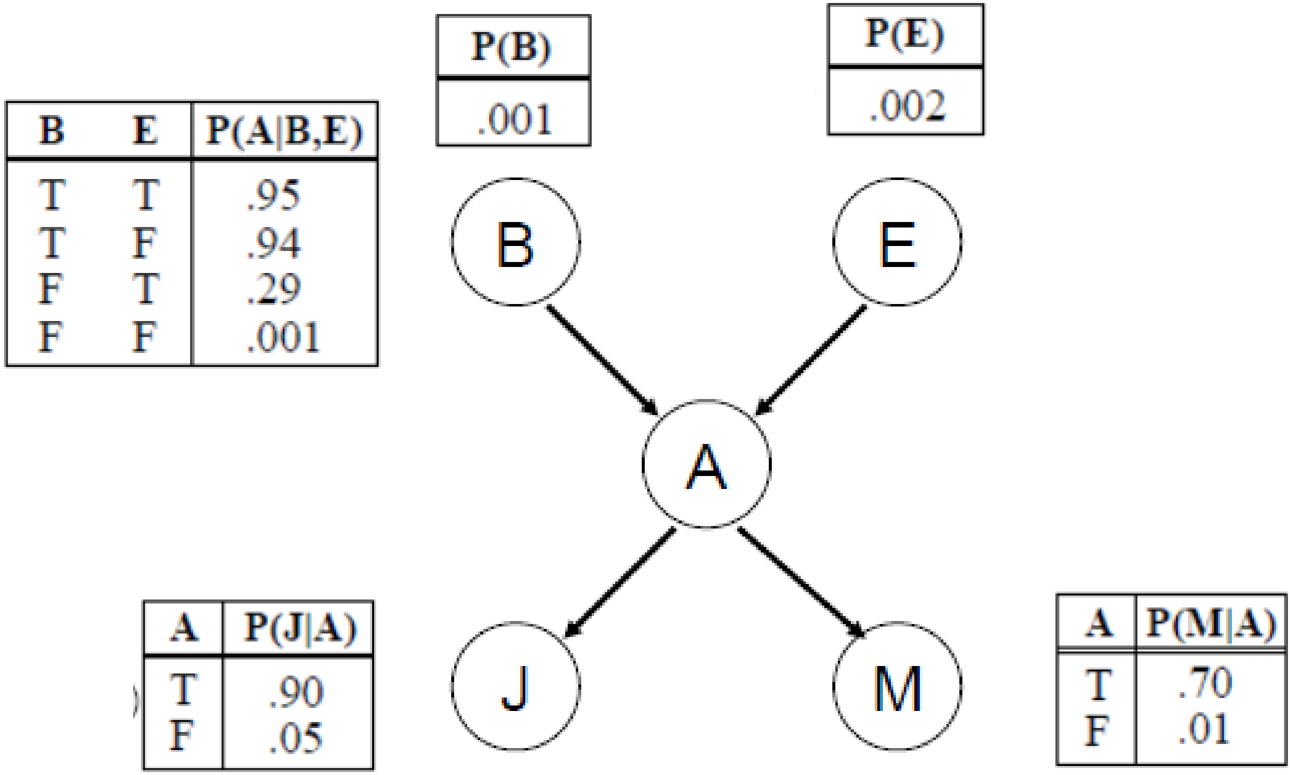
A: your alarm sounds

J: your neighbor John calls you

M: your other neighbor Mary calls you

Constructing Bayesian Networks (3)

- **Step 2: Add the conditional probability tables (CPTs)**
- **Each CPT must show P(X|Parent values) for all combinations of parent values**



P(B)
.001

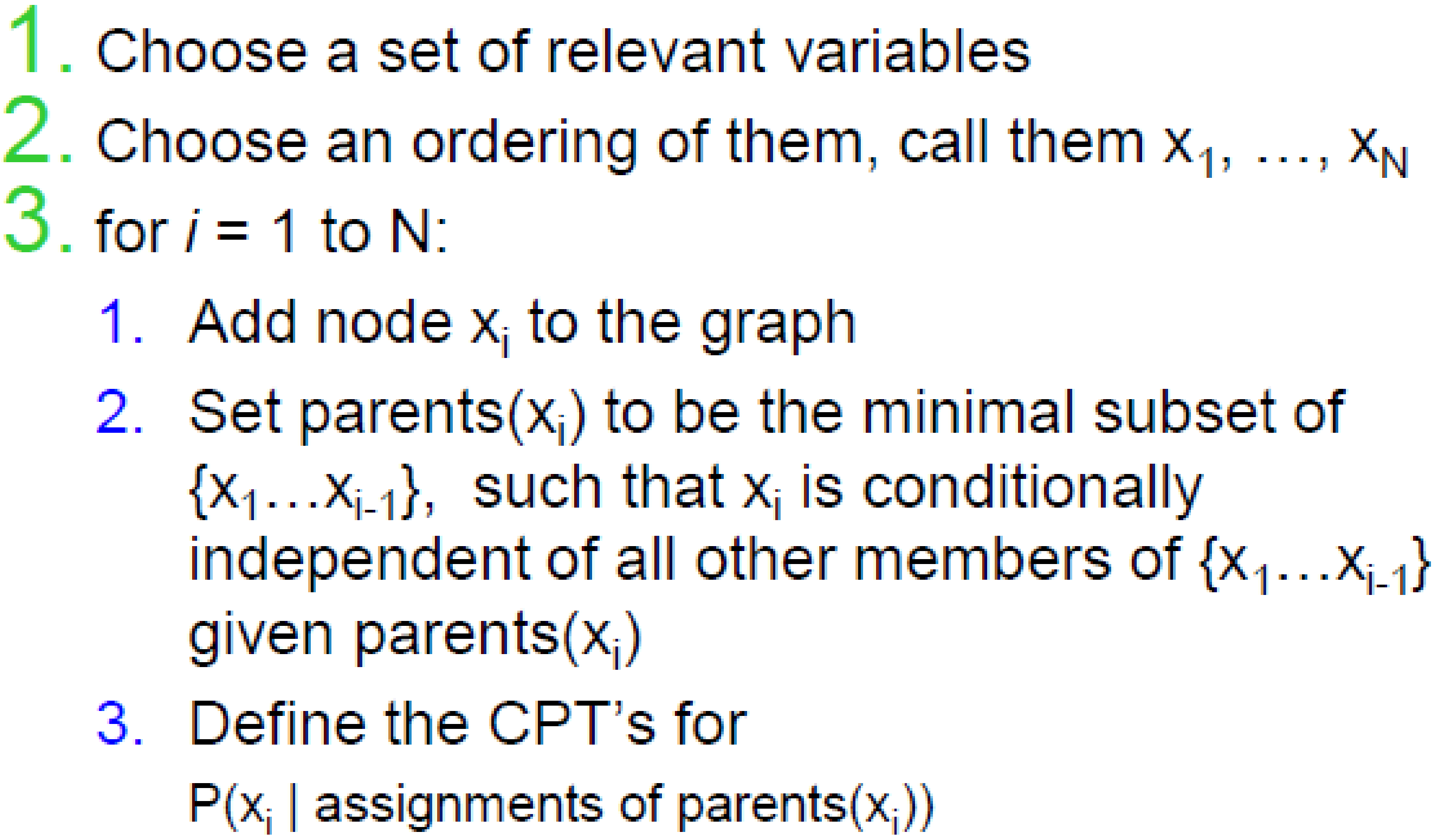
P(E)
.002

B	E	P(A|B,E)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J|A)
T	.90
F	.05

A	P(M|A)
T	.70
F	.01

Constructing Bayesian Networks - Summary



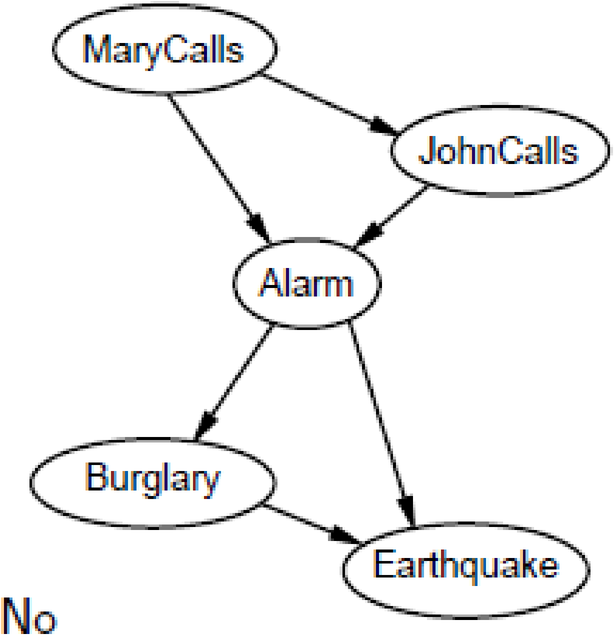
1. Choose a set of relevant variables
2. Choose an ordering of them, call them $x_1, \ldots, x_N$
3. for i = 1 to N:
    1. Add node x_i to the graph
    2. Set parents(x_i) to be the minimal subset of $\{x_1 \ldots x_{i-1}\}$, such that x_i is conditionally independent of all other members of $\{x_1 \ldots x_{i-1}\}$ given parents(x_i)
    3. Define the CPT's for

       P(x_i | assignments of parents(x_i))

From CS540 AI, X. Zhu, University of Wisconsin, Fall 2016

Constructing Bayesian Networks - Example

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



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

Where Are We Now?

We learned that:

- **Bayesian networks can be used to encode relationships between variables such as conditional independence, and hence provide a more compact representation than the full joint probability distribution**
- **Any joint probability can be computed as:**

$$P(x_1,\ldots,x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(x_i))$$

- **In addition to joint probability, we can also compute any *conditional probability* P(X|E), thus we can perform inference**

- **Let's see how we can do this!**

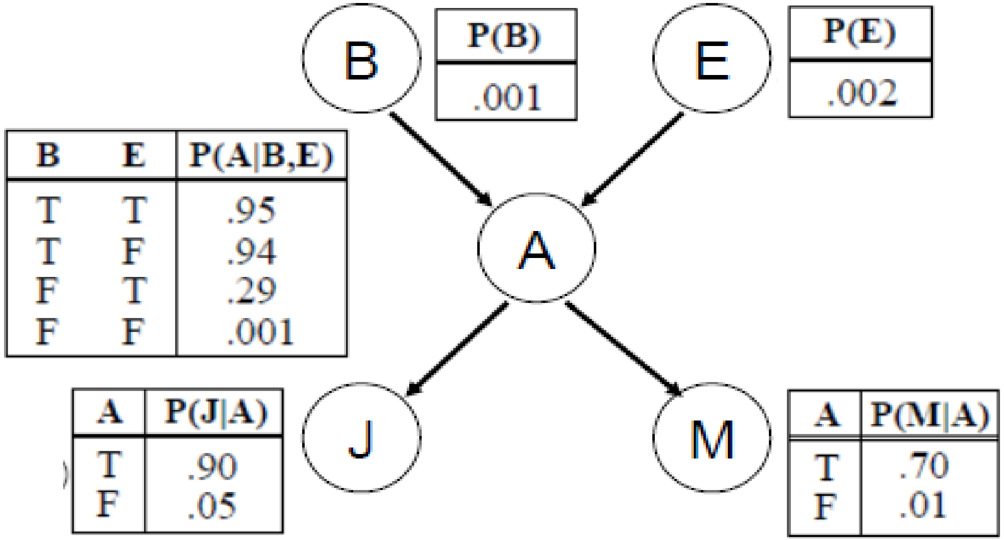
Inference by Enumeration

- **Example: P(b| j, m)=?**

(remember this is a shorthand for: P(B=T| J=T, M=T))

- **Step 1: We use the rule that relates joint and conditional probability:**

$$P(b \mid j, m) = \frac{P(b, j, m)}{P(j, m)}$$ **(i.e. the definition of conditional probability)**



P(B)
.001

P(E)
.002

B	E	P(A|B,E)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J|A)
T	.90
F	.05

A	P(M|A)
T	.70
F	.01

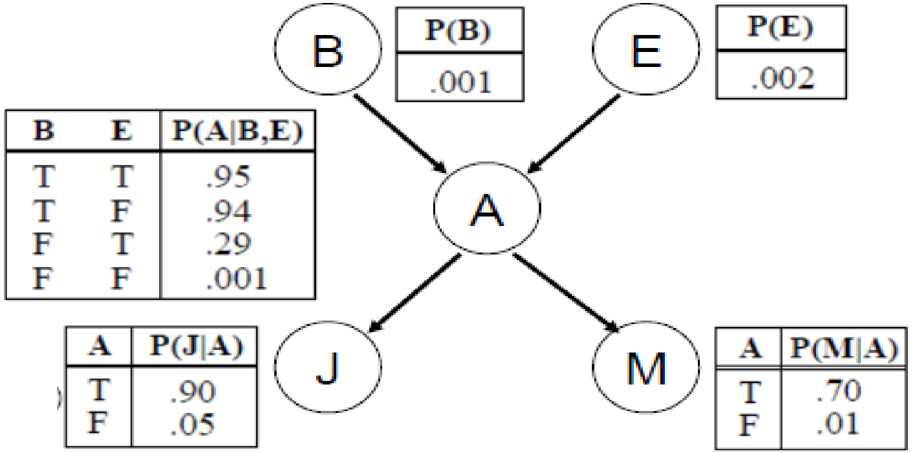
Inference by Enumeration (2)

- **Step 2: We add the hidden variables and sum the terms from the joint distribution**

$$P(b \mid j,m) = \frac{P(b,j,m)}{P(j,m)} = \alpha P(b,j,m) =$$

$$= \alpha \sum_{ea} P(b,j,m,E_e,A_a) =$$

$\sum_{e} P(E_e)$ – sum over all possible values of E, i.e. over e and $\neg e$



P(B)
.001

P(E)
.002

B	E	P(A|B,E)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

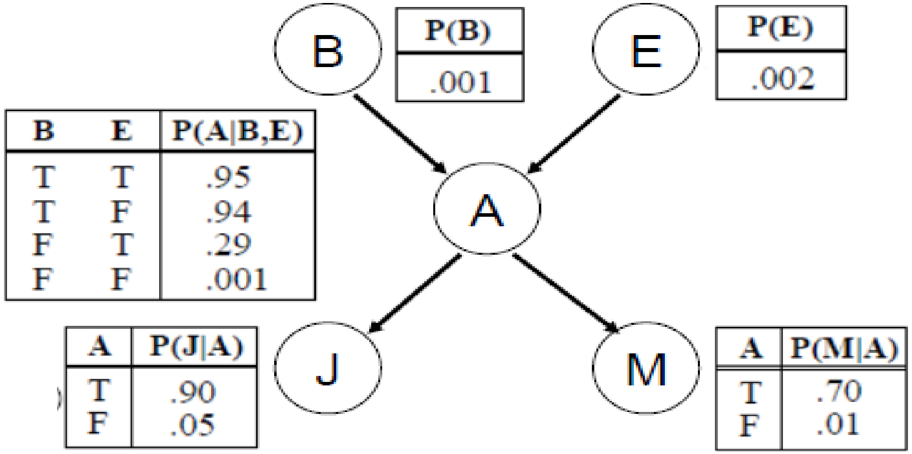
A	P(J|A)
T	.90
F	.05

A	P(M|A)
T	.70
F	.01

Inference by Enumeration (3)

- **Step 3: To compute the joint probability, we will use the Bayes network rule:**

$$P(x_1,\dots,x_n) = \prod_{i=1}^{n} P(x_i \mid Parents(x_i))$$



P(B)
.001

P(E)
.002

B	E	P(A|B,E)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J|A)
T	.90
F	.05

A	P(M|A)
T	.70
F	.01

- **Thus:**

$$P(b \mid j,m) = \frac{P(b,j,m)}{P(j,m)} = \alpha P(b,j,m) =$$

$$= \alpha \sum_{ea} P(b,j,m,E_e,A_a) =$$

$$= \alpha \sum_{ea} P(b)P(j \mid A_a)P(m \mid A_a)P(E_e)P(A_a \mid b,E_e)$$

Inference by Enumeration - All Together

$$P(b \mid j,m) = \frac{P(b,j,m)}{P(j,m)} = \alpha P(b,j,m) =$$

$$= \alpha \sum_{ea} P(b,j,m,E_e,A_a) =$$

$$= \alpha \sum_{ea} P(b)P(j \mid A_a)P(m \mid A_a)P(E_e)P(A_a \mid b,E_e) =$$

$$\begin{aligned}
&= \alpha P(b)[P(j \mid a)P(m \mid a)P(e)P(a \mid b,e) + \\
&\qquad P(j \mid \neg a)P(m \mid \neg a)P(e)P(\neg a \mid b,e) + \\
&\qquad P(j \mid a)P(m \mid a)P(\neg e)P(a \mid b,\neg e) + \\
&\qquad P(j \mid \neg a)P(m \mid \neg a)P(\neg e)P(\neg a \mid b,\neg e) =
\end{aligned}$$

For:

e,a

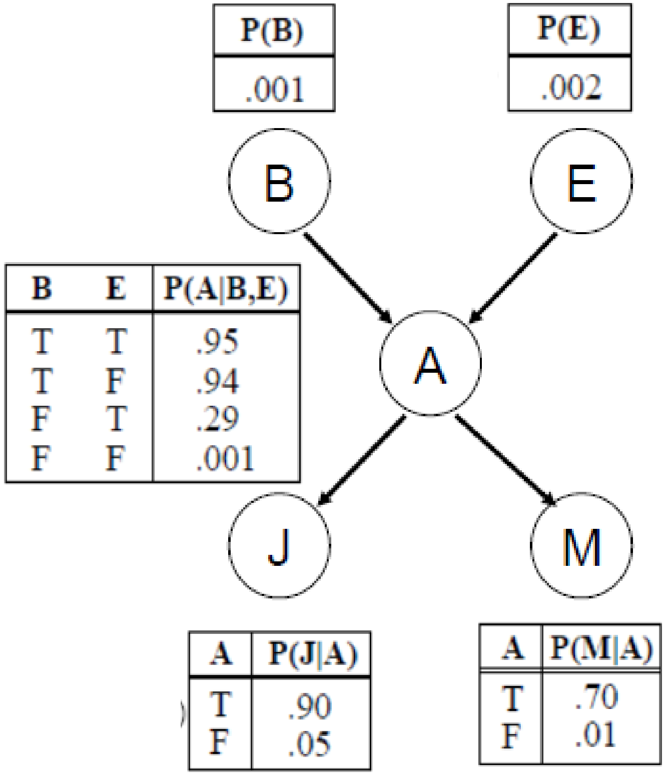
e,~a

~e,a

~e,~a

$$\begin{aligned}
&= \alpha * 0.001 * [0.9 * 0.7 * 0.002 * 0.95 + \\
&\qquad 0.05 * 0.01 * 0.002 * 0.05 + \\
&\qquad 0.9 * 0.7 * 0.998 * 0.94 + \\
&\qquad 0.05 * 0.01 * 0.998 * 0.06] =
\end{aligned}$$

$$= \alpha * 0.00059224$$



P(B)
.001

P(E)
.002

B	E	P(A|B,E)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J|A)
T	.90
F	.05

A	P(M|A)
T	.70
F	.01

Inference by Enumeration - All Together (2)

- **So we have:**

$$P(b \mid j,m) = \alpha P(b, j,m) = \alpha * 0.00059224$$

- **Similarly, we also compute the corresponding probability for ~b:**

$$P(\neg b \mid j,m) = \alpha P(\neg b, j,m) = \alpha * 0.0014919$$

- **Then the normalization constant α is:**

$$\alpha = \frac{1}{0.00059224 + 0.0014919}$$

- **=>**

$$\mathbf{P}(B \mid j,m) = \langle 0.284, 0.716 \rangle$$

for B=b (i.e. B=T)   for B=~b (i.e. B=F)

- **=> The probability of a burglary to be true, given that both John and Mary called is 28%**

A Small Improvement

- **To compute this, we had to add 4 terms, each computed by multiplying 4 numbers (or 5 if we had not taken *P(b)* out of the summations)**
- **In the worst case, when we have to sum out all variables, the complexity of the algorithm for a network with *n* Boolean variables will be *O(n2^n)***
- **A small improvement: we can also move *P(E$_e$)* out of the summation over *a*:**

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

$$=\alpha\sum_{ea}P(b,j,m,E_e,A_a)=$$

$$=\alpha P(b)\sum_{e}P(E_e)\sum_{a}P(j\,|\,A_a)P(m\,|\,A_a)P(A_a\,|\,b,E_e)$$

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

$$=\alpha\sum_{ea}P(b,j,m,E_e,A_a)=$$

$$=\alpha\sum_{ea}P(b)P(j\,|\,A_a)P(m\,|\,A_a)P(E_e)P(A_a\,|\,b,E_e)=$$

$$\begin{aligned}=\alpha P(b)[&P(j\,|\,a)P(m\,|\,a)P(e)P(a\,|\,b,e)+\\&P(j\,|\,\neg a)P(m\,|\,\neg a)P(e)P(\neg a\,|\,b,e)+\\&P(j\,|\,a)P(m\,|\,a)P(\neg e)P(a\,|\,b,\neg e)+\\&P(j\,|\,\neg a)P(m\,|\,\neg a)P(\neg e)P(\neg a\,|\,b,\neg e)=\end{aligned}$$

$$\begin{aligned}=\alpha*0.001*[&0.9*0.7*0.002*0.95+\\&0.05*0.01*0.002*0.05+\\&0.9*0.7*0.998*0.94+\\&0.05*0.01*0.998*0.06]=\end{aligned}$$

$$=\alpha*0.00059224$$

A Small Improvement (2)

$$P(b \mid j,m) = \alpha \sum_{ea} P(b)P(j \mid A_a)P(m \mid A_a)P(E_e)P(A_a \mid b,E_e) =$$

$$= \alpha P(b) \sum_{e} P(E_e) \sum_{a} P(j \mid A_a)P(m \mid A_a)P(A_a \mid b,E_e) =$$

$$\begin{aligned}
= \alpha P(b)[P(e)[&P(j \mid a)P(m \mid a)P(a \mid b,e) + \\
&P(j \mid \neg a)P(m \mid \neg a)P(\neg a \mid b,e)] + \\
[P(\neg e)[&P(j \mid a)P(m \mid a)P(a \mid b,\neg e) + \\
&P(j \mid \neg a)P(m \mid \neg a)P(\neg a \mid b,\neg e)]] =
\end{aligned}$$

$$\begin{aligned}
= \alpha * 0.001*[&0.002*(0.9*0.7*0.95 + 0.05*0.01*0.05) + \\
&0.998*(0.9*0.7*0.94 + 0.05*0.01*0.06)] =
\end{aligned}$$

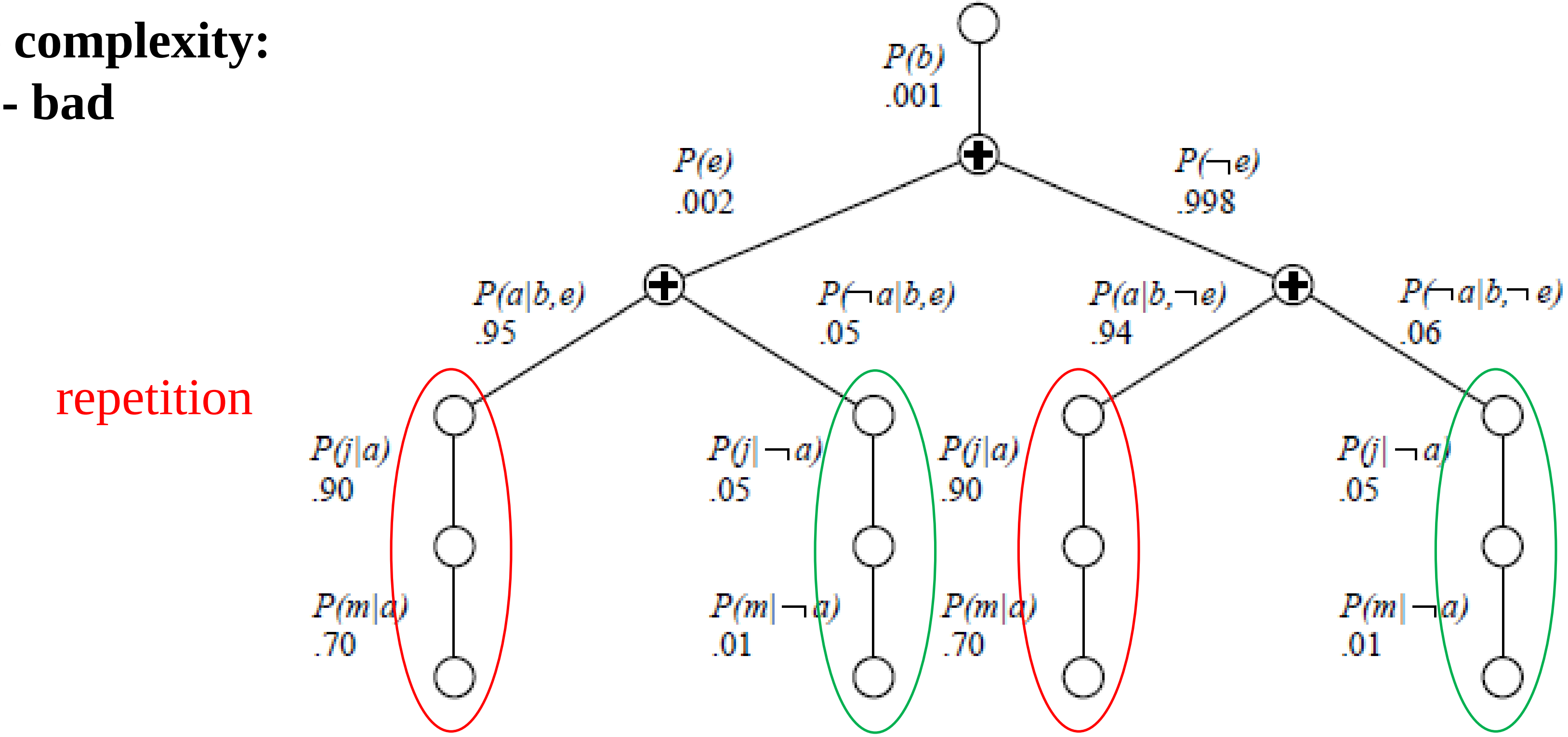
$$= \alpha * 0.00059224$$

- **This reduces the number of multiplications, but still there is a redundancy, e.g. *P(j|a)P(m|a)* and *P(j|~a)P(m|~a)* are computed twice, once for *e* and then for *~e***

Structure of the Computation

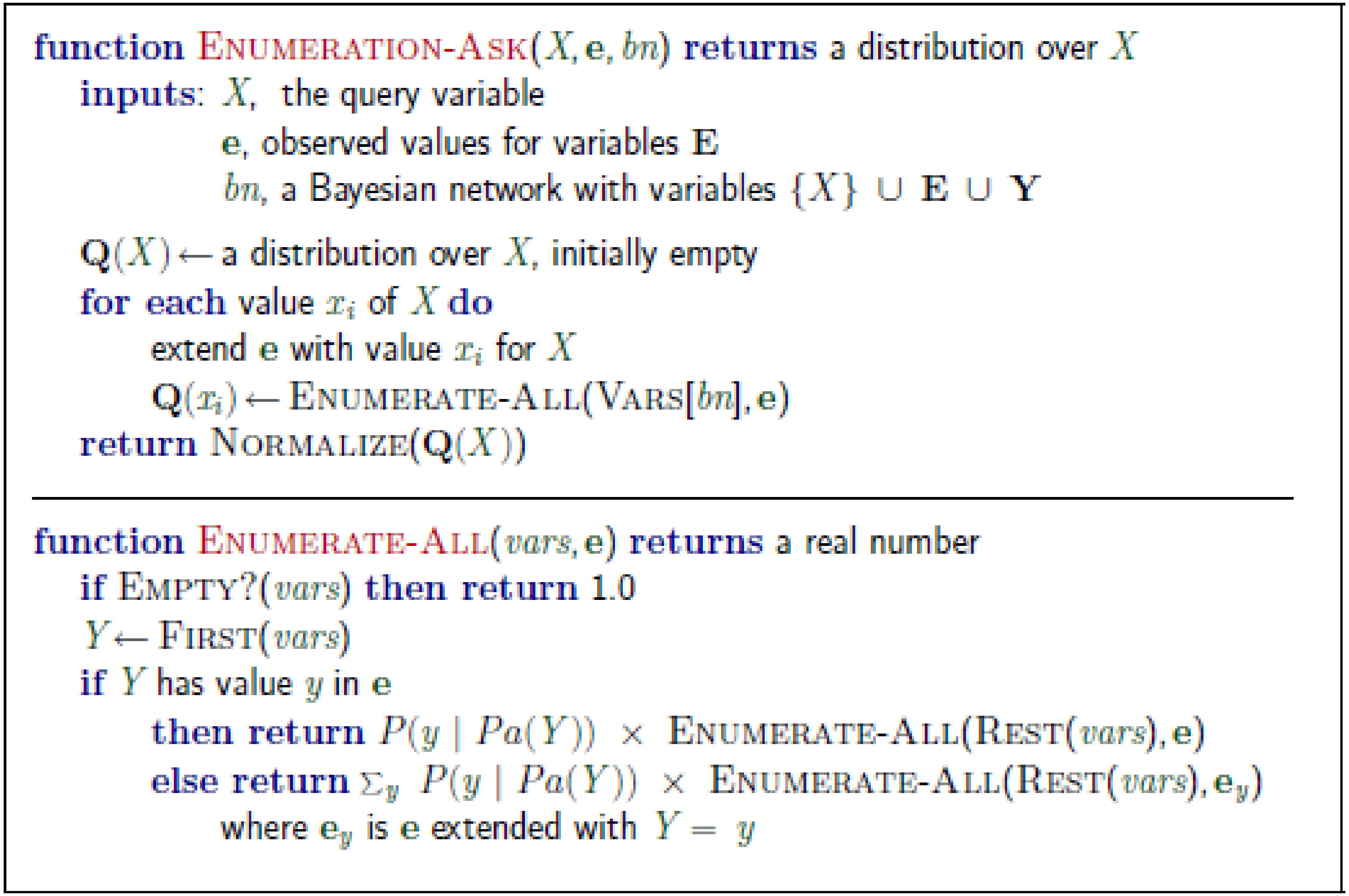
- This tree is evaluated using the Enumeration-Ask algorithm (next slide) using depth-first recursion – looping through the variables, summing and multiplying CPT entries as we go

- Space complexity: *$O(2^n)$* - bad



Enumeration is inefficient: repeated computation
e.g., computes $P(j|a)P(m|a)$ for each value of e

Inference by Enumeration Algorithm



```
function ENUMERATION-ASK($X$, e, $bn$) returns a distribution over $X$
   inputs: $X$, the query variable
           e, observed values for variables E
           $bn$, a Bayesian network with variables $\{X\} \cup \mathbf{E} \cup \mathbf{Y}$

   $\mathbf{Q}(X) \leftarrow$ a distribution over $X$, initially empty
   for each value $x_i$ of $X$ do
       extend e with value $x_i$ for $X$
       $\mathbf{Q}(x_i) \leftarrow$ ENUMERATE-ALL(VARS[$bn$], e)
   return NORMALIZE($\mathbf{Q}(X)$)
```

```
function ENUMERATE-ALL($vars$, e) returns a real number
   if EMPTY?($vars$) then return 1.0
   $Y \leftarrow$ FIRST($vars$)
   if $Y$ has value $y$ in e
       then return $P(y \mid Pa(Y)) \times$ ENUMERATE-ALL(REST($vars$), e)
       else return $\Sigma_y$ $P(y \mid Pa(Y)) \times$ ENUMERATE-ALL(REST($vars$), $\mathbf{e}_y$)
           where $\mathbf{e}_y$ is e extended with $Y = y$
```

Inference by Variable Elimination

- **The enumeration algorithm can be improved by eliminating the repeated calculations -** ***variable elimination*** **algorithm**
- **Idea: Do the calculations once and save the results for later use**
- **The expression is evaluated** ***right-to left*** **(= the tree is evaluated** ***bottom up*****)**

$$P(b \mid j,m) = \alpha \sum_{ea} P(b)P(j \mid A_a)P(m \mid A_a)P(E_e)P(A_a \mid b, E_e)$$

- **The results are stored, the summations over each variable are done only for those portions of the expression that depend on the variable**

Variable Elimination - Example

- We evaluate this expression:

$$\mathbf{P}(B\mid j,m)=\alpha\underbrace{\mathbf{P}(B)}_{f_1(B)}\sum_{e}\underbrace{P(E_e)}_{f_2(E)}\sum_{a}\underbrace{P(A_a\mid B,E_e)}_{f_3(A,B,E)}\underbrace{P(j\mid A_a)}_{f_4(A)}\underbrace{P(m\mid A_a)}_{f_5(A)}$$

- We have annotated it with the name of the *factors*
  - e.g. factor $f_5(A)$ corresponds to $P(m|A_a)$ and it depends only on A as J and M are fixed by the query (e.g. we know that J=T and M=T)

A	B	E	$f_3(A,B,E)$
T	T	T	0.95
T	T	F	0.94
T	F	T	0.29
T	F	F	0.01
F	T	T	0.05
F	T	F	0.06
F	F	T	0.71
F	F	F	0.999

- Let's write all factors:

A	$f_5(A)$
T	0.7
F	0.1

A	$f_4(A)$
T	0.9
F	0.05

E	$f_2(E)$
T	0.002
F	0.998

B	$f_1(B)$
T	0.001
F	0.999

Operations on Factors

- There are 2 operations on factors that we need
  - Point-wise product of a pair of factors
  - Summing out a variable from a product of factors (i.e. eliminating a variable)
- We will show how these operations are defined and how to perform them using our example

$$\mathbf{P}(B \mid j,m) = \alpha \underbrace{\mathbf{P}(B)}_{f_1(B)} \sum_e \underbrace{P(E_e)}_{f_2(E)} \sum_a \underbrace{P(A_a \mid B, E_e)}_{f_3(A,B,E)} \underbrace{P(j \mid A_a)}_{f_4(A)} \underbrace{P(m \mid A_a)}_{f_5(A)}$$

Computing f_4*f_5

- **We are here:**

$$\mathbf{P}(B \mid j,m) = \alpha \underbrace{\mathbf{P}(B)}_{f_1(B)} \sum_{e} \underbrace{P(E_e)}_{f_2(E)} \sum_{a} \underbrace{P(A_a \mid B, E_e)}_{f_3(A,B,E)} \underbrace{P(j \mid A_a)}_{f_4(A)} \underbrace{P(m \mid A_a)}_{f_5(A)}$$

- **Firstly we compute f_4*f_5**
- **The multiplication of factors is point-wise**

A	$f_4(A)$
T	0.9
F	0.05

*

A	$f_5(A)$
T	0.7
F	0.1

=

A	$f_4(A)$*$f_5(A)$
T	0.9*0.7=0.63
F	0.05*0.1=0.05

Computing f_3*f_4*f_5

$$\mathbf{P}(B \mid j,m) = \alpha \underbrace{\mathbf{P}(B)}_{f_1(B)} \sum_{e} \underbrace{P(E_e)}_{f_2(E)} \sum_{a} \underbrace{P(A_a \mid B, E_e)}_{f_3(A,B,E)} \underbrace{P(j \mid A_a)}_{f_4(A)} \underbrace{P(m \mid A_a)}_{f_5(A)}$$

- **Now we compute f_3*(f_4*f_5):**

Letters dropped for brevity

A	B	E	f_3(A,B,E)
T	T	T	0.95
T	T	F	0.94
T	F	T	0.29
T	F	F	0.01
F	T	T	0.05
F	T	F	0.06
F	F	T	0.71
F	F	F	0.999

*

A	f_4(A)*f_5(A)
T	0.9*0.7=0.63
F	0.05*0.1=0.005

=

A	B	E	f_3*f_4*f_5
T	T	T	0.63*0.95
T	T	F	0.63*0.94
T	F	T	0.63*0.29
T	F	F	0.63*0.01
F	T	T	0.005*0.05
F	T	F	0.005*0.06
F	F	T	0.005*0.71
F	F	F	0.005*0.999

Eliminating A

We are here:

$$\mathbf{P}(B\mid j,m)=\alpha\underbrace{\mathbf{P}(B)}_{f_1(B)}\sum_{e}\underbrace{P(E_e)}_{f_2(E)}\underset{\Uparrow}{\sum_{a}}\underbrace{P(A_a\mid B,E_e)}_{f_3(A,B,E)}\underbrace{P(j\mid A_a)}_{f_4(A)}\underbrace{P(m\mid A_a)}_{f_5(A)}$$

- **Now we need to sum out A , i.e. to eliminate A which is done by summing the matrix – B & E are fixed, we sum the values of A**

A	B	E	$f_3*f_4*f_5$
T	T	T	0.63*0.95
T	T	F	0.63*0.94
T	F	T	0.63*0.29
T	F	F	0.63*0.01
F	T	T	0.005*0.05
F	T	F	0.005*0.06
F	F	T	0.005*0.71
F	F	F	0.005*0.999

→

B	E	$\Sigma f_3*f_4*f_5=f_6$
T	T	0.63*0.95+0.005*0.005=0.5985
T	F	0.63*0.94+0.005*0.06=0.5922
F	T	0.63*0.29+0.005*0.71=0.1830
F	F	0.63*0.01+0.005*0.999=0.001129

Computing f_2*f_6

We are here: $$\mathbf{P}(B \mid j,m) = \alpha \underbrace{\mathbf{P}(B)}_{f_1(B)} \sum_e \underbrace{P(E_e)}_{f_2(E)} \underbrace{\underset{\uparrow}{\sum_a} P(A_a \mid B, E_e) P(j \mid A_a) P(m \mid A_a)}_{f_6(B,E)}$$

E	**f_2(E)**
T	0.002
F	0.998

*

B	**E**	**f_6(B,E)**
T	T	0.5985
T	F	0.5922
F	T	0.1830
F	F	0.001129

=

B	**E**	**f_2(E)$_*$$f_6$(B,E)**
T	T	0.5985*0.002
T	F	0.5922*0.998
F	T	0.1830*0.002
F	F	0.001129*0.998

Eliminating E

We are here: $$\mathbf{P}(B \mid j,m) = \alpha \underbrace{\mathbf{P}(B)}_{f_1(B)} \sum_{e} \underbrace{P(E_e)}_{f_2(E)} \underbrace{\sum_{a} P(A_a \mid B, E_e) P(j \mid A_a) P(m \mid A_a)}_{f_6(B,E)}$$

B	E	$f_2(E)_* f_6(B,E)$
T	T	0.5985*0.002
T	F	0.5922*0.998
F	T	0.1830*0.002
F	F	0.001129*0.998

→

B	$\Sigma f_2(E)_* f_6(B,E)=f_7(B)$
T	0.5985*0.002+0.5922*0.998=0.592213
F	0.1830*0.002+0.001129*0.998=0.001493

Computing $f_1 * f_7$ and Finishing

We are here: $$\mathbf{P}(B\mid j,m)=\alpha\underbrace{\mathbf{P}(B)}_{f_1(B)}\underbrace{\sum_e P(E_e)\sum_a P(A_a\mid B,E_e)P(j\mid A_a)P(m\mid A_a)}_{f_7(B)}$$

B	$f_1(B)$
T	0.001
F	0.999

*

B	$f_7(B)$
T	0.592213
F	0.001493

=

B	$f_1(B)*f_7(B)$
T	0.592213*0.001=0.000592
F	0.001493*0.999=0.001491

- **Finishing :**

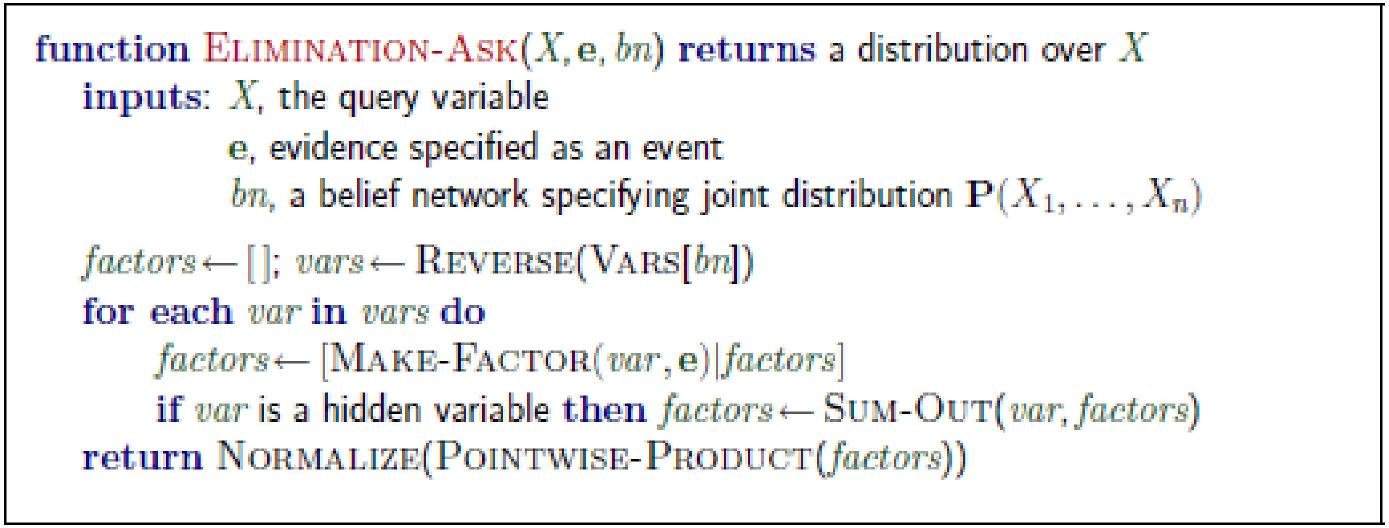
=> $\mathbf{P}(B\mid j,m)=\alpha<0.000592,0.001491>$

- **Normalizing:** $\mathbf{P}(B\mid j,m)=<0.284245,0.715755>$

- **=> The probability of a burglary to be true, given that both John and Mary called is 28% (the same answer as before ☺)**

Variable Elimination Algorithm - Pseudocode

- **If we have functions for pointwise-product and summing out, the algorithm can be written in a few lines:**



```
function ELIMINATION-ASK($X$, e, $bn$) returns a distribution over $X$
    inputs: $X$, the query variable
            e, evidence specified as an event
            $bn$, a belief network specifying joint distribution $\mathbf{P}(X_1, \ldots, X_n)$

    $factors \leftarrow [\,]$; $vars \leftarrow$ REVERSE(VARS[$bn$])
    for each $var$ in $vars$ do
        $factors \leftarrow$ [MAKE-FACTOR($var$, e)|$factors$]
        if $var$ is a hidden variable then $factors \leftarrow$ SUM-OUT($var$, $factors$)
    return NORMALIZE(POINTWISE-PRODUCT($factors$))
```

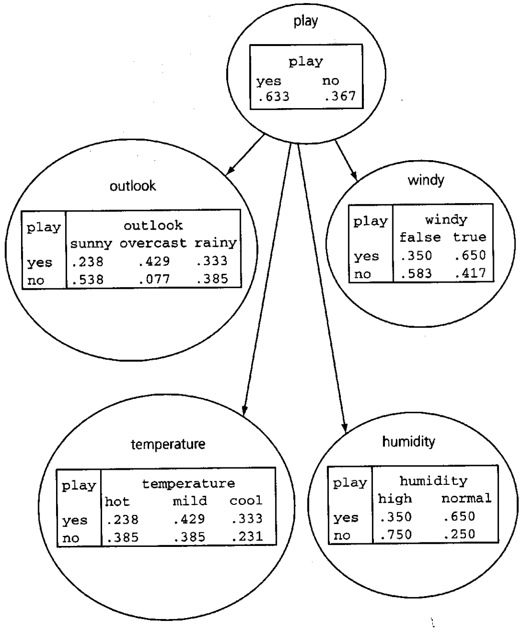
- **It is sensitive to the order of variables that are eliminated**
  - **Different order -> different intermediate factors**
- **Heuristic for choosing the order: eliminate whichever variable minimizes the size of the next factor to be computed**

Approximate Inference

- **Compared to enumeration, variable elimination allows us to save computation by re-using intermediate results but it is still computationally expensive**
- **Alternative: *approximate inference by sampling* – idea:**
  - **Generate many samples – 1 sample is a complete assignment of all variables**
  - **Count the fraction of samples matching the query and evidence**
  - **If the number of samples is big enough, the fractions converge to P(query | evidence)**
- **There are 3 main sampling algorithms**
  - **Simple (direct) sampling**
  - **Likelihood weighting**
  - **Gibbs sampling (an example of Markov Chain Monte Carlo method)**
- **If you are interested see: R&N, ch.14.5**

Naïve Bayes and Bayesian Networks

- **A Naïve Bayes classifier is a special case of Bayesian networks**
- **The topology represents the Naïve Bayes assumption of conditional independence - the values of the attributes are conditionally independent given the class value**
  - ***outlook*, *windy*, *temperature* and *humidity* are independent given *play***



play	
yes	no
.633	.367

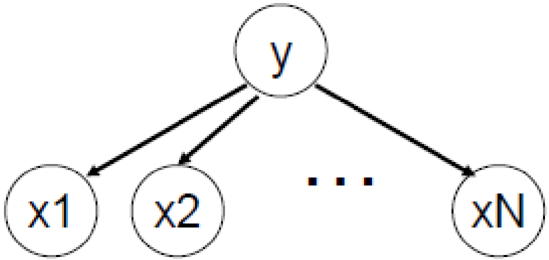
play	outlook		
	sunny	overcast	rainy
yes	.238	.429	.333
no	.538	.077	.385

play	windy	
	false	true
yes	.350	.650
no	.583	.417

play	temperature		
	hot	mild	cool
yes	.238	.429	.333
no	.385	.385	.231

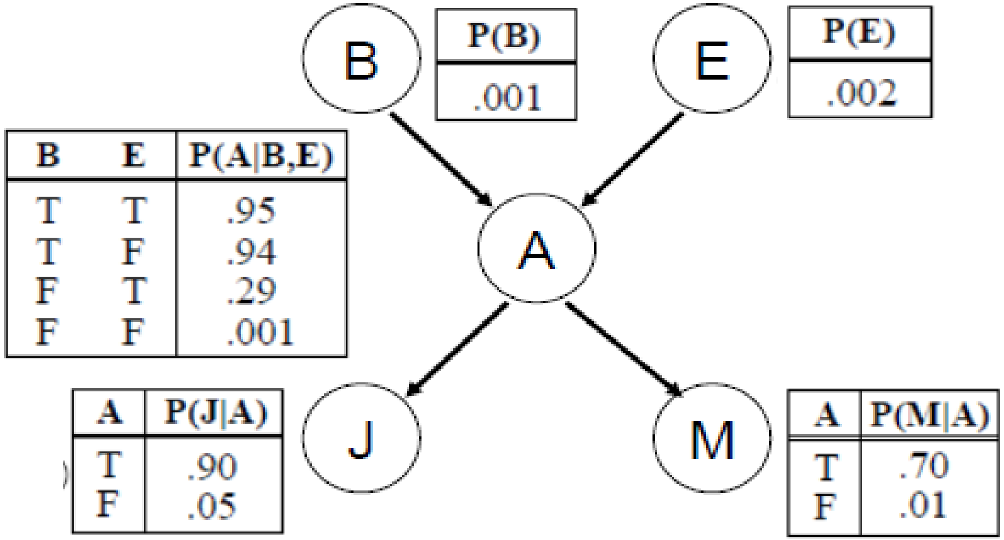
play	humidity	
	high	normal
yes	.350	.650
no	.750	.250

- **More generally:**
  - **Class node *y* as a root**
  - **Evidence nodes *x* as leaves (features)**
  - **Assume conditional independence between features, given the class**



Where Do We Get the CPT From?

- Where do we get the CPT numbers from?
  - Ask domain expert
  - Learn them from data



P(B)
.001

P(E)
.002

B	E	P(A|B,E)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J|A)
T	.90
F	.05

A	P(M|A)
T	.70
F	.01

Learning CPT from Data

- **Learn from a dataset like this:**

~b, ~e, ~ a, j, m

~b, ~e, ~ a, ~j, ~m

~b, ~e, ~ a, ~j, ~m

b, e, ~ a, j, ~m

~b, ~e, ~ a, ~j, ~m

b, ~e, a, j, m

~b, ~e, ~ a, ~j, ~m

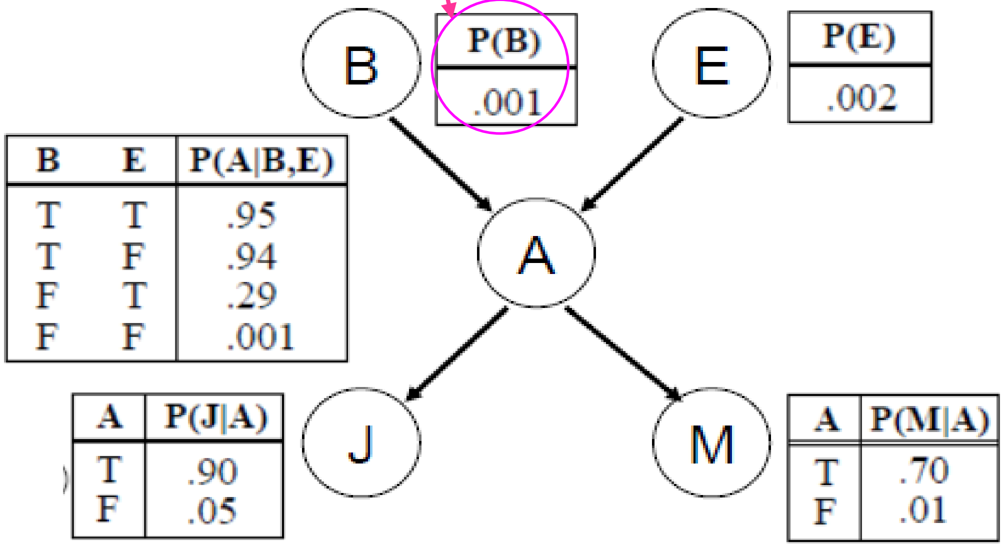
b, e, a, j, m

…

- **How to learn P(B) when B=T=?**

1) Count # b and # ~b in the dataset

2) P(B)= # b/ (# b + # ~b)



P(B)
.001

P(E)
.002

B	E	P(A|B,E)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J|A)
T	.90
F	.05

A	P(M|A)
T	.70
F	.01

Based on CS540 AI, X. Zhu, University of Wisconsin, Fall 2016

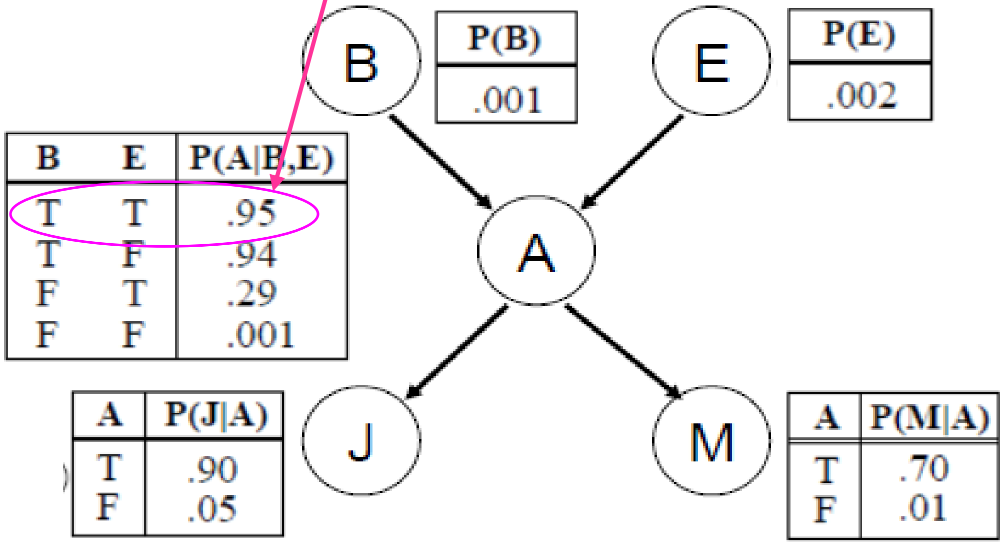
Learning CPT from Data (2)

- **Learn from a dataset like this:**

~b, ~e, ~ a, j, m
~b, ~e, ~ a, ~j, ~m
~b, ~e, ~ a, ~j, ~m
b, e, ~ a, j, ~m
~b, ~e, ~ a, ~j, ~m
b, ~e, a, j, m
~b, ~e, ~ a, ~j, ~m
b, e, a, j, m
…

- **How to learn P(A|B, E) when B=T and E=T?**
- **1) Count # a and # ~a in the dataset when B=T and E=T**

2) P(A|B,E) = # a/ (# a + # ~a)



P(B)
.001

P(E)
.002

B	E	P(A|B,E)
T	T	.95
T	F	.94
F	T	.29
F	F	.001

A	P(J|A)
T	.90
F	.05

A	P(M|A)
T	.70
F	.01

Summary

- **Bayesian networks are graphical models that provide a way to represent conditional independence relationships. Hence, they provide a more compact representation than the full joint probability distribution.**
- **They assume that a node is conditionally independent of its non-descendants, given its parents**
- **Inference in Bayesian networks means computing the probability distribution of a set of query variables, given a set of evidence variables**
- **We studied two algorithms for inference: inference by enumeration and inference by variable elimination**
- **A Naïve Bayes classifier is a special case of Bayesian networks**

Acknowledgements

- Some slides and examples are based on:
  - CS540 AI, Xiaojin Zhu, University of Wisconsin, Fall 2016
  - J. Kelleher, B. McNamee and A. D’Arcy, Fundamentals of Machine Learning for Predictive Data Analytics, MIT, 2015