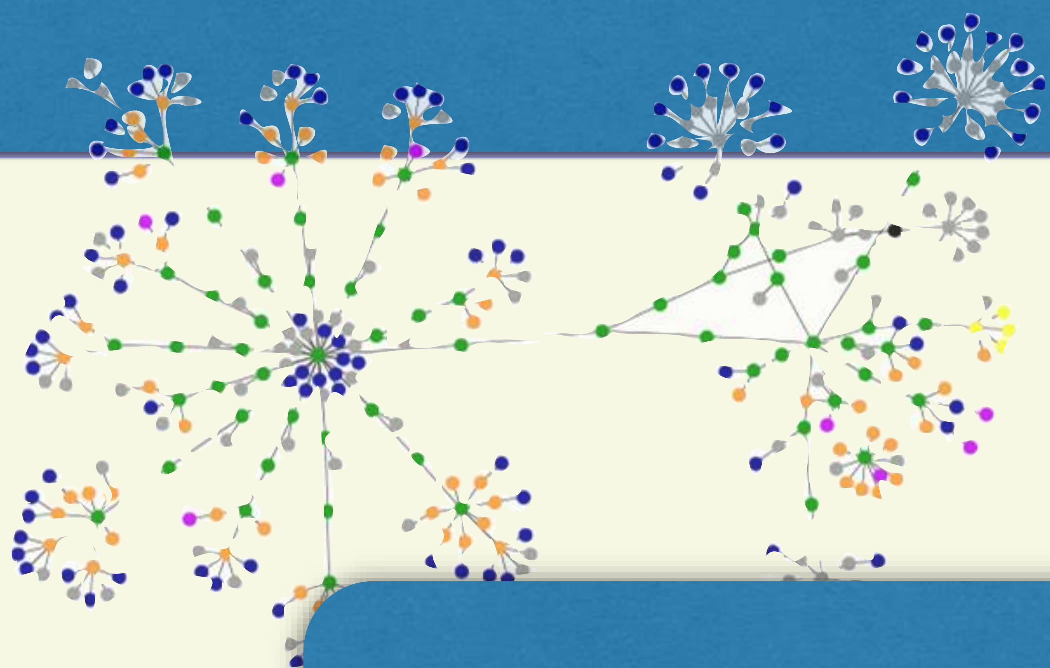
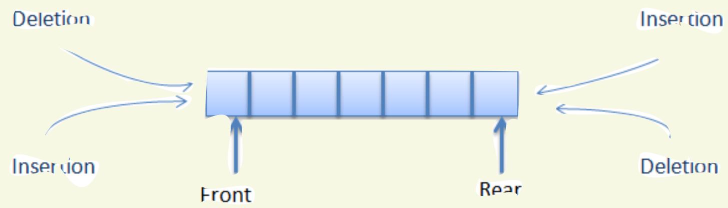
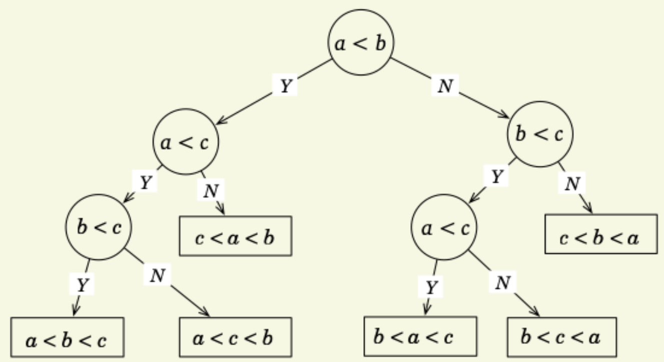
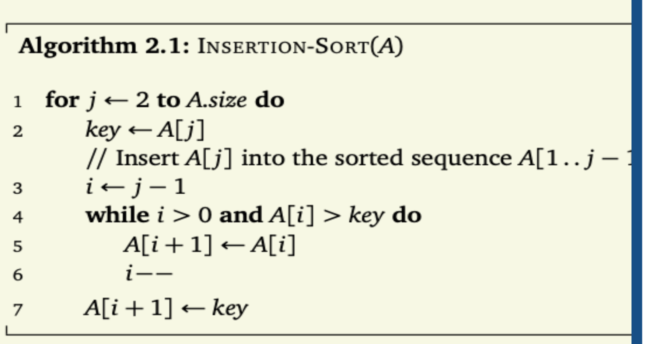






COURSE
“Técnicas Matemáticas para Big Data”
University of Aveiro


Algorithm 2.1: INSERTION-SORT(A)

```
for j ← 2 to A.size do
    key ← A[j]
    // Insert A[j] into the sorted sequence A[1..j − 1
    i ← j − 1
    while i > 0 and A[i] > key do
        A[i + 1] ← A[i]
        i−−
    A[i + 1] ← key
```

- Veracity:
  - Bayesian Networks;
  - Monty Hall Problem;
  - Examples;
  - Practical notebook

SUMMARY: Inference in graphical models, Bayesian Networks.

[T] Theoretical concepts;

[P] Exploring BNs.

Roots

Graphical models were proposed in the 1st half of the 20th century in several fields e.g., in genetics and later in AI.

Recently they began to attract the attention of the electric engineering community as well as the attention of statisticians.

Mile stones:

- Wright (1921) geneticist – proposed a graphical representations for probabilities (severely criticized by statisticians).
- Howard e Matheson (1981) – developed influence diagrams for decision analysis.
- J. Pearl (1982) proposed an algorithm for the propagation of beliefs in trees as a way to model human reasoning. Later he extended this algorithm to Bayesian networks without multiple paths.

Review: Probability Theory

- Sum rule (marginal distributions)

$$p(\mathbf{x}) = \sum_{\mathbf{y}} p(\mathbf{x}, \mathbf{y})$$

- Product rule

$$p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x}|\mathbf{y})p(\mathbf{y})$$

- From these we have Bayes' theorem

$$p(\mathbf{y}|\mathbf{x}) = \frac{p(\mathbf{x}|\mathbf{y})p(\mathbf{y})}{p(\mathbf{x})}$$

  – with normalization factor

$$p(\mathbf{x}) = \sum_{\mathbf{y}} p(\mathbf{x}|\mathbf{y})p(\mathbf{y})$$

Review: Conditional Probabilty

- Conditional Probability (rewriting product rule)

$$P(A \mid B) = P(A, B) / P(B)$$

- Chain Rule

$$P(A,B,C,D) = P(A) \quad \frac{P(A,B)}{P(A)} \quad \frac{P(A,B,C)}{P(A,B)} \quad \frac{P(A,B,C,D)}{P(A,B,C)}$$

$$= P(A) \quad P(B \mid A) \quad P(C \mid A,B) \quad P(D \mid A,B,C)$$

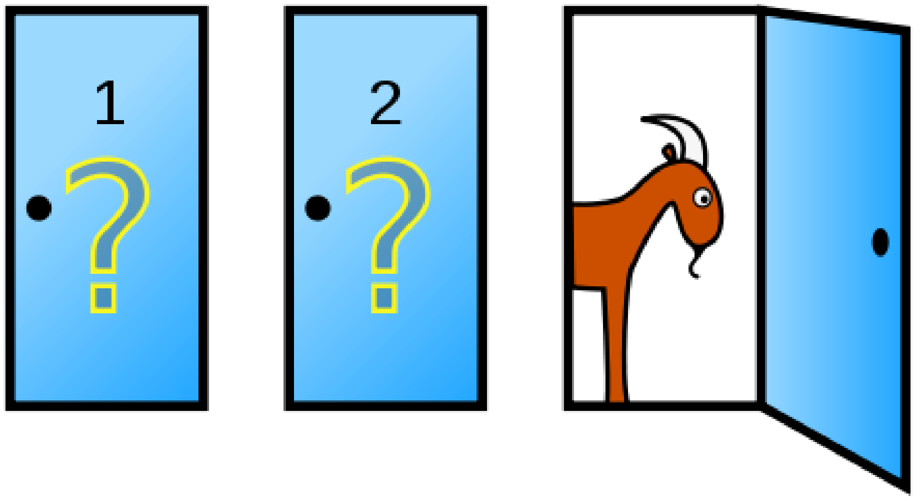
- Conditional Independence

$$P(A, B \mid C) = P(A \mid C) \; P(B \mid C)$$

  – statistical independence

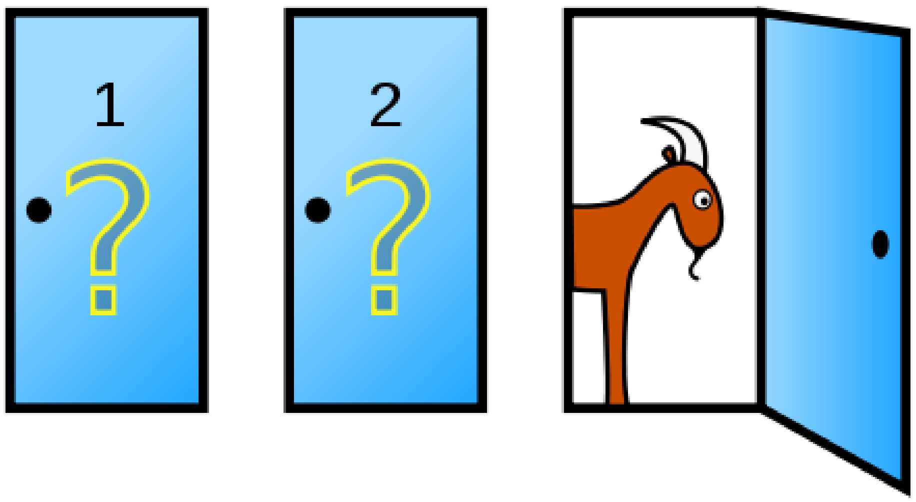
$$P(A, B) = P(A) \; P(B)$$

Monty Hall Problem

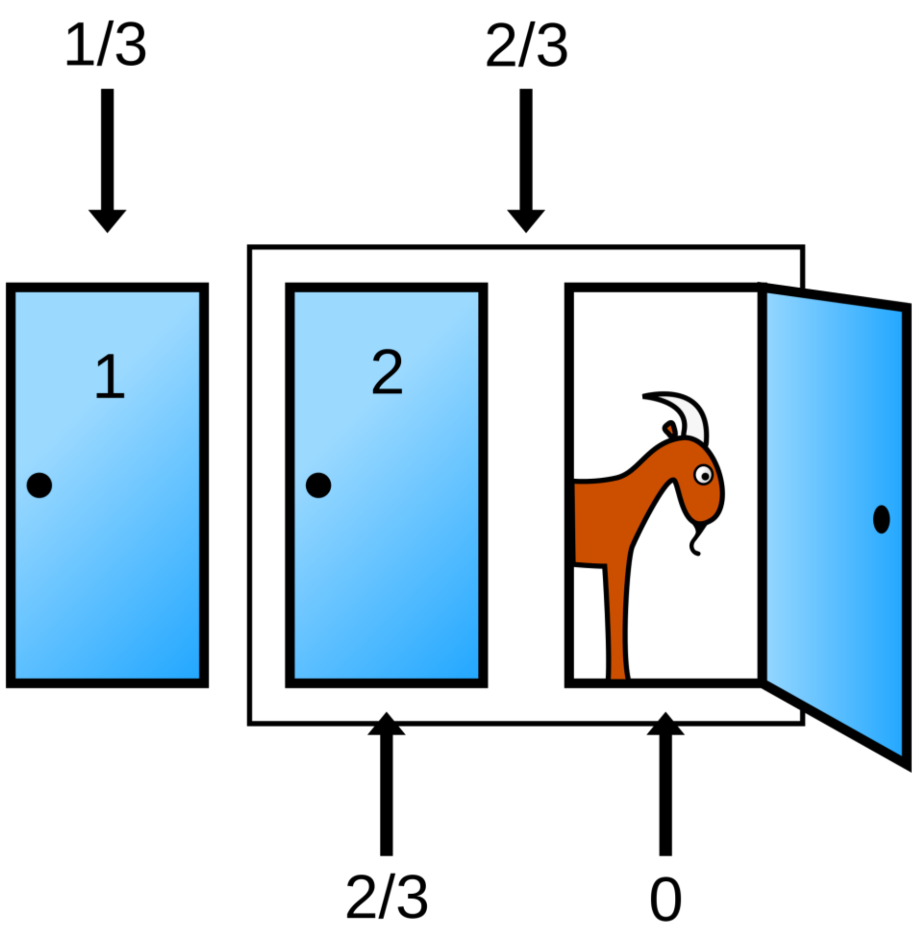


In search of a new car, the player picks a door, say *1*.
The game host then opens one of the other doors, say *3*,
to reveal a goat and offers to let the player switch from door *1*
to door *2*.
Should you switch?

Monty Hall Problem



In search of a new car, the player picks a door, say *1*.
The game host then opens one of the other doors, say *3*,
to reveal a goat and offers to let the player switch from door *1*
to door *2*.
Should you switch?



Should you switch?

Recall Joint Distribution

The joint distribution of n variables is described by 2^n combinations and we have:

$$P(h_i|d_1, d_2, d_3, .., d_n) = \frac{P(d_1, d_2, d_3, ..., d_n|h_i) \cdot P(h_i)}{P(d_1, d_2, d_3, ..., d_n)}$$

where all 2^n probabilities must be known. To deal with this we have (at least) two solutions:

Recall Joint Distribution

The joint distribution of n variables is described by 2^n combinations and we have:

$$P(h_i|d_1, d_2, d_3, .., d_n) = \frac{P(d_1, d_2, d_3, ..., d_n|h_i) \cdot P(h_i)}{P(d_1, d_2, d_3, ..., d_n)}$$

where all 2^n probabilities must be known. To deal with this we have (at least) two solutions:

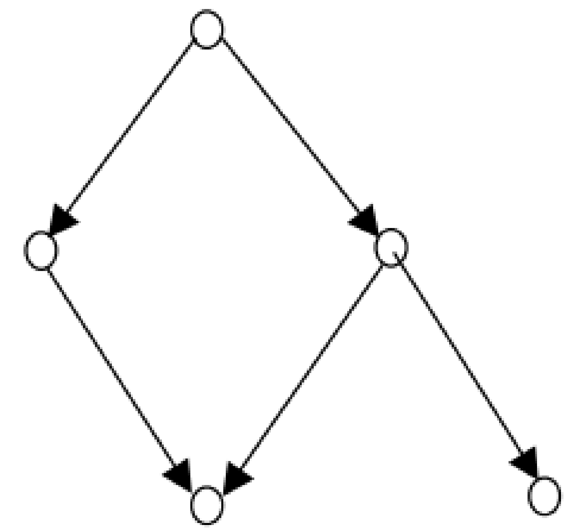
(1st) Approach is the **Naive Bayes Classifier**. We assume that all events are **conditional independent (Naive Bayes assumption)**, i.e. a single cause directly affects a number of events but all of them are conditional independent:

$$P(d_1, d_2, d_3, ..., d_n|h_i) = \prod_{j=1}^{n} P(d_j|h_i).$$

The NB Classifier is very restrictive in real situations as it is seen as a one-level graph and the probability of occurence of an event only depends on its parent (Markov property).

Recall Joint Distribution

(2nd) Approach is to describe the dependence of events by some mathematical structure (e.g. a DAG), which is the case of **Bayesian Networks**. BNs describe the **probability distribution** of a set of (random) variables **by combining conditional independence assumptions with conditional probability**.



Let's start with the NB Classifier:

- Along with decision trees, neural networks, nearest neighbor, one of the most practical learning methods
- When to use:
  - Moderate or large training set available
  - Attributes that describe instances are conditionally independent given classification
- Successful applications:
  - Diagnosis
  - Classifying text documents

Naive Bayes Classifier

Assume a target function f: X —> V where each instance x described by attributes $< a_1, \ldots, a_n >$.

The problem to solve is

$$v_{MAP} = \arg\max_{v_j \in V} P(v_j | a_1, a_2 \ldots a_n)$$

$$v_{MAP} = \arg\max_{v_j \in V} \frac{P(a_1, a_2 \ldots a_n | v_j) P(v_j)}{P(a_1, a_2 \ldots a_n)}$$

$$= \arg\max_{v_j \in V} P(a_1, a_2 \ldots a_n | v_j) P(v_j)$$

Naive Bayes assumption

$$P(a_1, a_2 \ldots a_n | v_j) = \prod_i P(a_i | v_j)$$

$$= \arg\max_{v_j \in V} P(v_j) \prod_i P(a_i | v_j)$$

Example 1 (of A.Wichert):

age	income	student	credit_rating	buys_computer
<=30	high	no	fair	no
<=30	high	no	excellent	no
30…40	high	no	fair	yes
>40	medium	no	fair	yes
>40	low	yes	fair	yes
>40	low	yes	excellent	no
31…40	low	yes	excellent	yes
<=30	medium	no	fair	no
<=30	low	yes	fair	yes
>40	medium	yes	fair	yes
<=30	medium	yes	excellent	yes
31…40	medium	no	excellent	yes
31…40	high	yes	fair	yes
>40	medium	no	excellent	no

<u>Setting</u>

C1: buys_computer=yes
C2: buys_computer=no

X has the conditions:
age <= 30
income = medium
student = yes
credit_rating = fair

We want to infere about
P(X|C1) and P(X|C2)

Example 1 (of A.Wichert):

- Compute $P(X|C_i)$ for each class

P(age="<30" | buys_computer="yes") = 2/9=0.222

P(age="<30" | buys_computer="no") = 3/5 =0.6

P(income="medium" | buys_computer="yes")= 4/9 =0.444

P(income="medium" | buys_computer="no") = 2/5 = 0.4

P(student="yes" | buys_computer="yes)= 6/9 =0.667

P(student="yes" | buys_computer="no")= 1/5=0.2

P(credit_rating="fair" | buys_computer="yes")=6/9=0.667

P(credit_rating="fair" | buys_computer="no")=2/5=0.4

P(buys_computer=„yes")=9/14

P(buys_computer=„no")=5/14

Example 1 (of A.Wichert):

- X=(age<=30 ,income =medium, student=yes,credit_rating=fair)

P(X|C_i) : P(X|buys_computer="yes")= 0.222 x 0.444 x 0.667 x 0.0.667 =0.044

P(X|buys_computer="no")= 0.6 x 0.4 x 0.2 x 0.4 =0.019

P(X|C_i)*P(C_i) : P(X|buys_computer="yes") * P(buys_computer="yes")=0.028

P(X|buys_computer="no") * P(buys_computer="no")=0.007

- X belongs to class "buys_computer=yes"

Example 1 (of A.Wichert):

- X=(age<=30 ,income =medium, student=yes,credit_rating=fair)

P(X|C$_i$) : P(X|buys_computer="yes")= 0.222 x 0.444 x 0.667 x 0.0.667 =0.044

P(X|buys_computer="no")= 0.6 x 0.4 x 0.2 x 0.4 =0.019

P(X|C$_i$)*P(C$_i$) : P(X|buys_computer="yes") * P(buys_computer="yes")=0.028

P(X|buys_computer="no") * P(buys_computer="no")=0.007

- X belongs to class "buys_computer=yes"

Remarks:

- We have estimated probabilities by the fraction of times the event is observed to n_c occur over the total number of opportunities n
- It provides poor estimates when n_c is very small

- When n_c is very small:

$$\hat{P}(a_i|v_j) = \frac{n_c + mp}{n + m}$$

  - n is number of training examples for which $v=v_j$
  - n_c number of examples for which $v=v_j$ and $a=a_i$
  - p is **prior** estimate
  - m is weight given to prior (i.e. number of ``virtual'' examples)

- When n_c is very small:

$$\hat{P}(a_i|v_j) = \frac{n_c + mp}{n + m}$$

  - n is number of training examples for which $v=v_j$
  - n_c number of examples for which $v=v_j$ and $a=a_i$
  - p is **prior** estimate
  - m is weight given to prior (i.e. number of ``virtual'' examples)

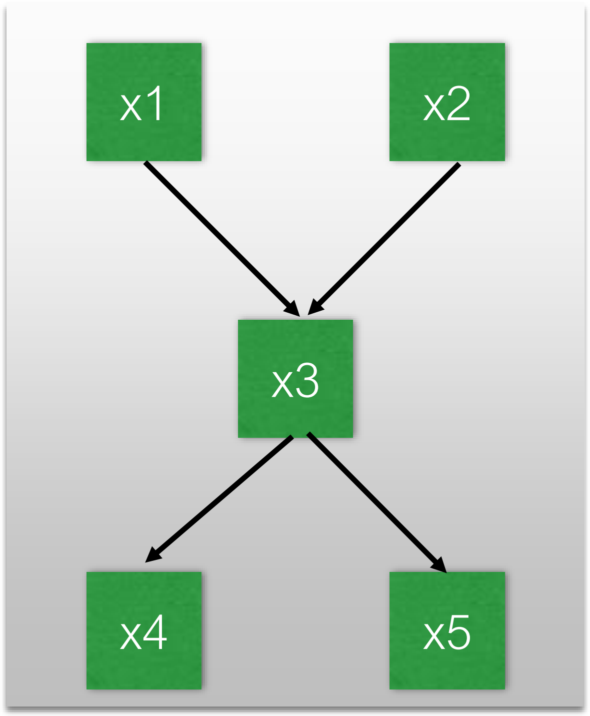
Bayesian (Belief) Networks

Bayesian Belief Networks (BBN) describe conditional independence among **subsets** of variables, allowing to combine prior knowledge about (in)dependencies among variables with observed training data

Bayesian (Belief) Networks

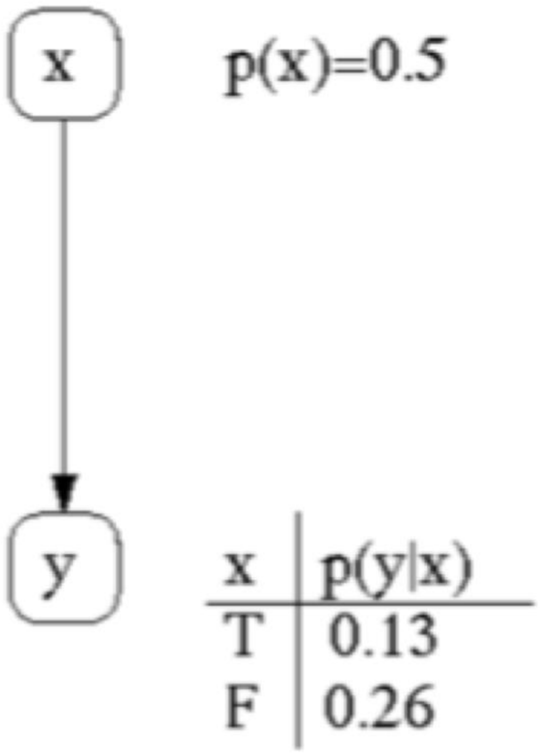
A Bayesian network is a correct representation of the domain only if each node is conditionally independent of its predecessors in the ordering, given its parents.

x4 is independent of x1 and x2 given x3



x1 is independent of x4 and x5 given x3 and x2

Law of Total Probability



- If two events x and y are independent, then the probability that events x and y both occur is

$$p(x, y) = p(x \wedge y) = p(x) \cdot p(y).$$

In this case the conditional probability is

$$p(x|y) = p(x).$$

If all N possible variables are independent, then

$$p(x_1, x_2, \cdots, x_N) = p(x_1) \cdot p(x_2) \cdot \cdots \cdot p(x_N) = \prod_{i=1}^{N} p(x_i)$$

Law of Total Probability

p(x)=0.5

x	p(y|x)
T	0.13
F	0.26

- If two events x and y are independent, then the probability that events x and y both occur is

$$p(x, y) = p(x \wedge y) = p(x) \cdot p(y).$$

In this case the conditional probability is

$$p(x|y) = p(x).$$

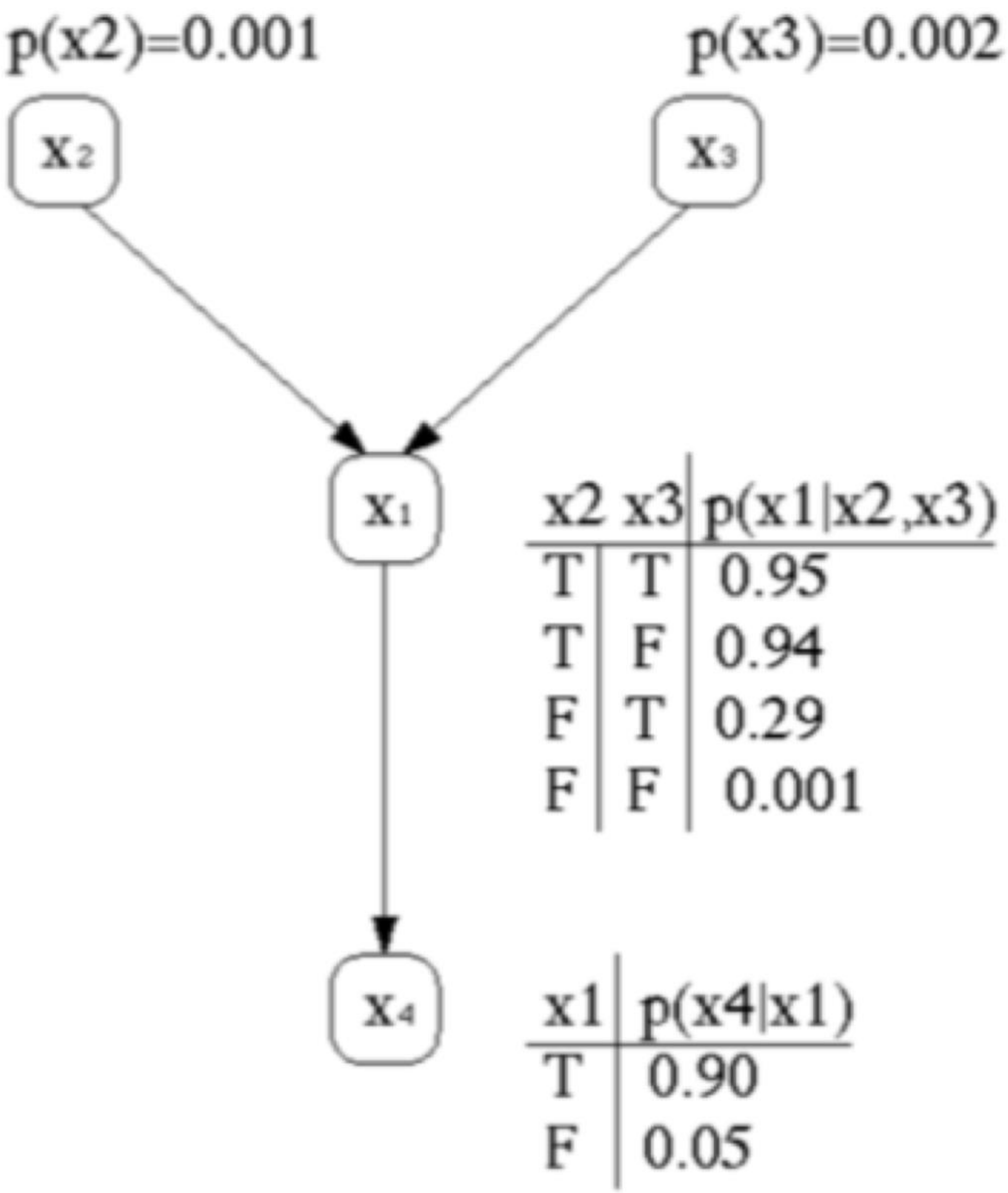
If all N possible variables are independent, then

$$p(x_1, x_2, \cdots, x_N) = p(x_1) \cdot p(x_2) \cdot \cdots \cdot p(x_N) = \prod_{i=1}^{N} p(x_i)$$

- For a subset of variables conditionally related we have:

$$p(x,y) = p(x|y)\ p(y) \quad \text{or} \quad p(x,y) = p(y|x)\ p(x)$$

Example 2 - Law of Total Probability

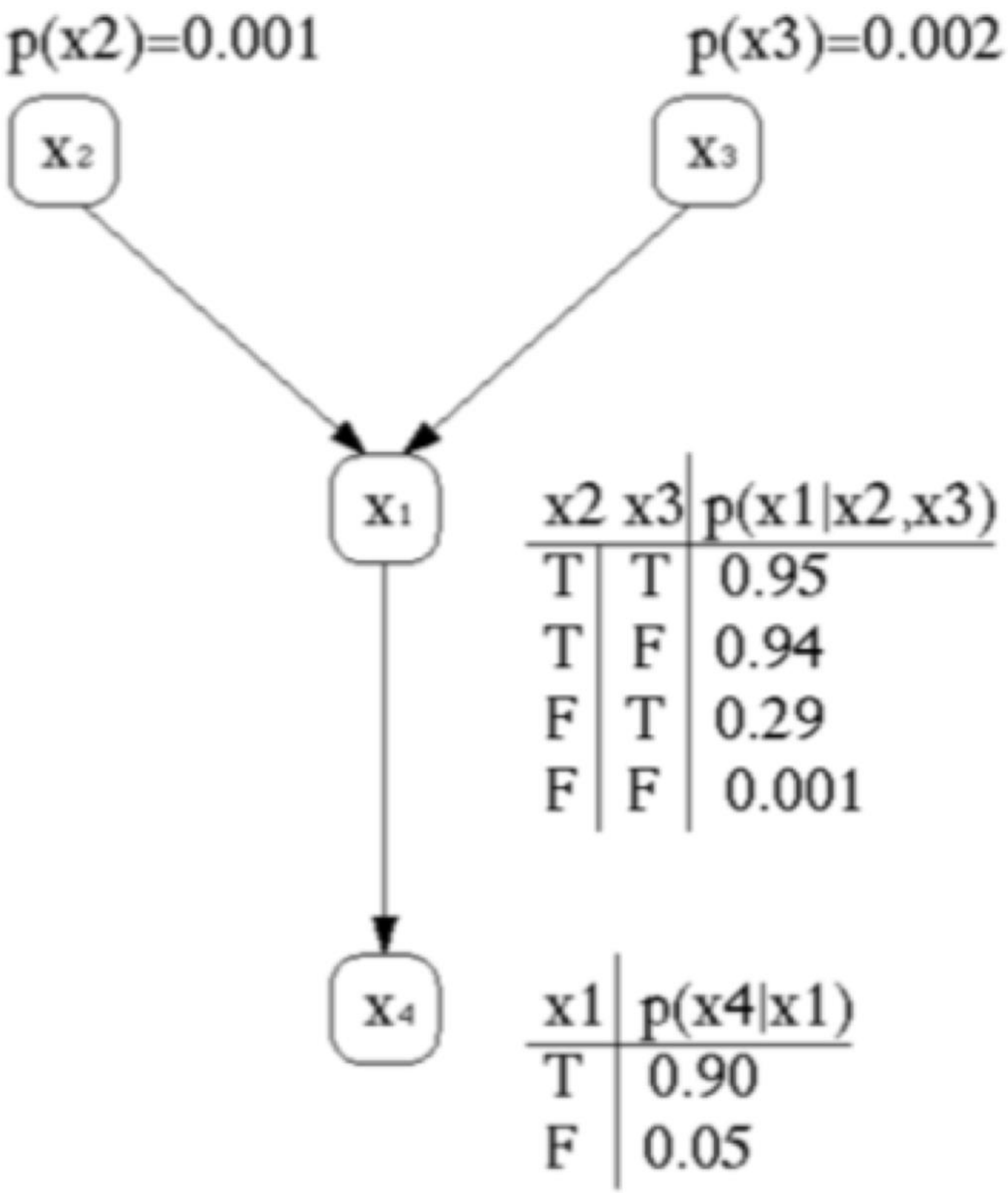


x2	x3	p(x1|x2,x3)
T	T	0.95
T	F	0.94
F	T	0.29
F	F	0.001

x1	p(x4|x1)
T	0.90
F	0.05

Task: Determine p(x1,x2,x3,x4) for (x1,x2,x3,x4) = (T,T,F,T) …

Example 2 - Law of Total Probability



Task: Determine p(x1,x2,x3,x4) for (x1,x2,x3,x4) = (T,T,F,T) …

Answer:

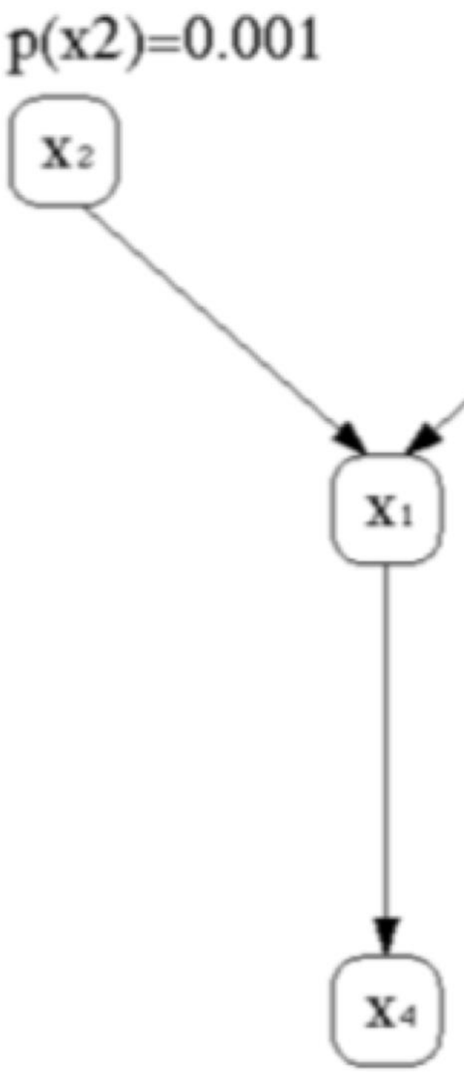
P = p(x1=T,x2=T,x3=F,x4=T) = A B C D

where

A = p(x1|x2=T,x3=F) = 0.94
B = p(x2=T) = 0.001
C = p(x3=F) = 1 - 0.002 = 0.998
D = p(x4|x1=T) = 0.90

Therefore, P = 0,084%

Example 2 - Law of Total Probability



x2	x3	p(x1|x2,x3)
T	T	0.95
T	F	0.94
F	T	0.29
F	F	0.001

x1	p(x4|x1)
T	0.90
F	0.05

Task: Determine p(x1,x2,x3,x4) for (x1,x2,x3,x4) = (T,T,F,T) …

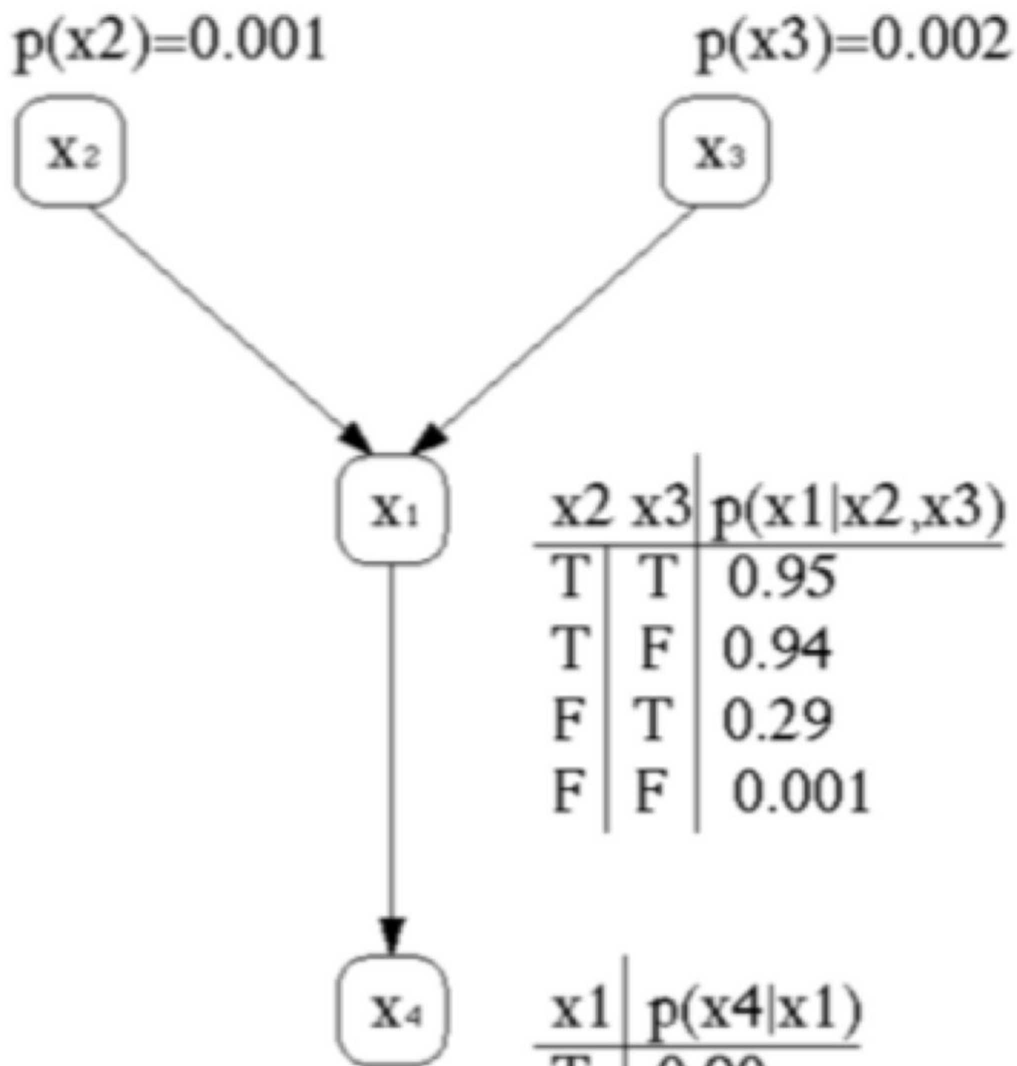
Answer:

P = p(x1=T,x2=T,x3=F,x4=T) = A B C D

where

- This relationship between occurrence of events called causality is represented by conditional dependency inducing *time*.
- In our example x_2 and x_3 cause x_1 and only then x_1 *causes* x_4.
- This kind of decomposition via conditional independence is modelled by Bayesian networks.
- Bayesian networks provide a natural representation for (causally induced) conditional independence.
- They represent a set of conditional independence assumptions, by the topology of an acyclic directed graph and sets of conditional probabilities.

Example 2 - Law of Total Probability



x2	x3	p(x1|x2,x3)
T	T	0.95
T	F	0.94
F	T	0.29
F	F	0.001

x1	p(x4|x1)
T	0.90
F	0.05

The classical example here is:

- x2: burglary
- x3: earthquake
- x1: alarm
- x4: security firm call

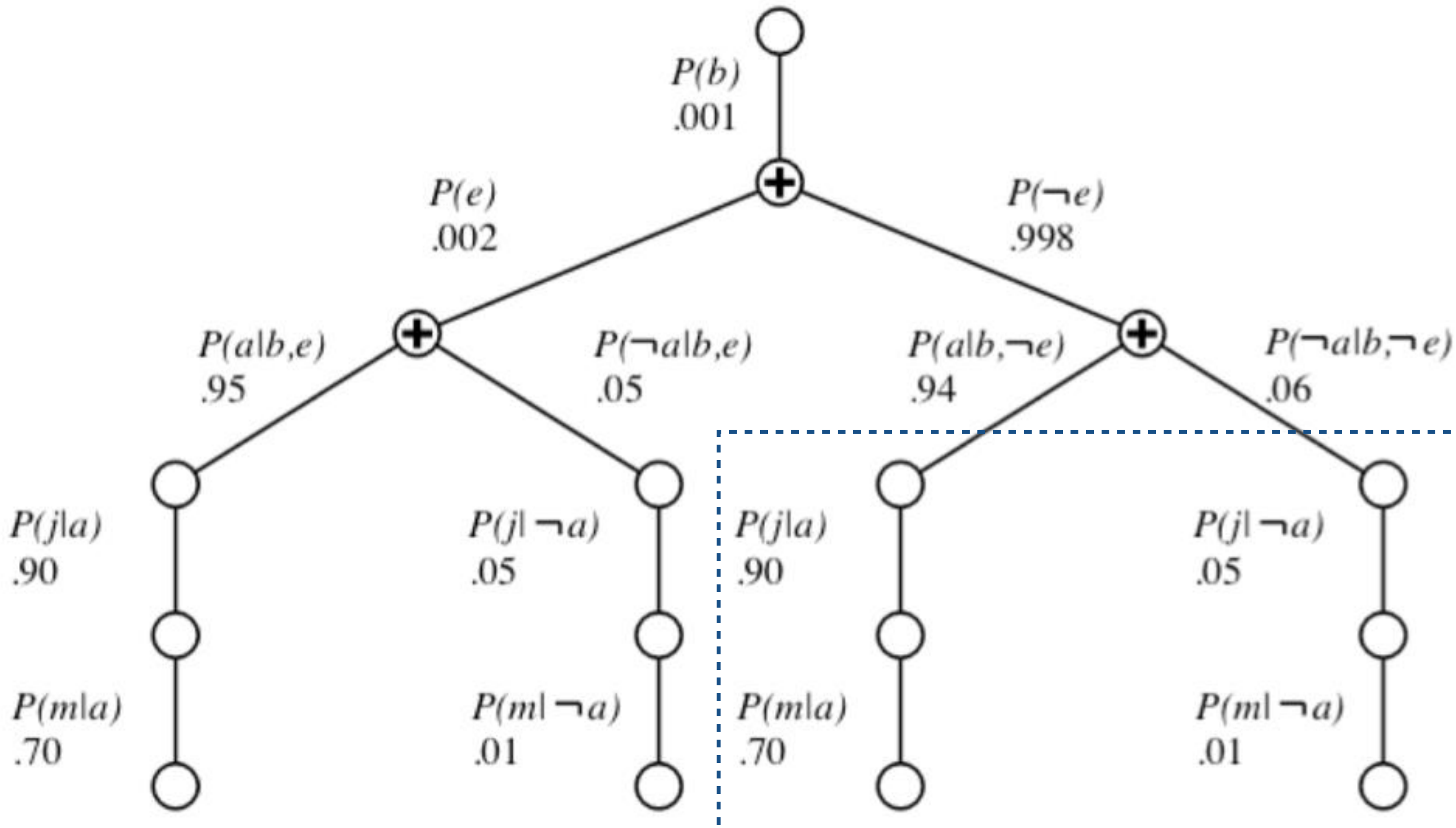
which may model:

- a burglar can set the alarm off
- an earthquake can set the alarm off

- the alarm may trigger and the security firm may call us

Bayesian Belief Networks

- Variable elimination (i.e. tree pruning)



Enumeration is inefficient: repeated computation

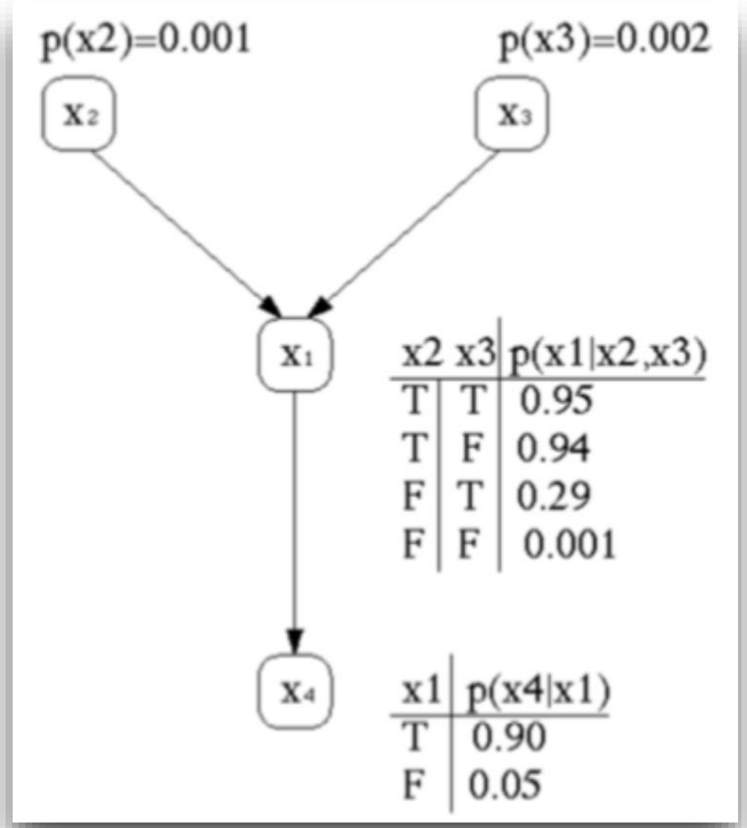
e.g., computes $P(j|a)P(m|a)$ for each value of e

Learning BN

- Four categories of learning problems
  - Graph structure may be known/unknown
  - Variable values may be fully observed / partly unobserved
- Easy case: learn parameters for graph structure is *known*, and data is *fully observed*
- Interesting case: graph *known*, data *partly known*
- Gruesome case: graph structure *unknown*, data *partly unobserved*

Learning BN

- **From Fully Observed Data**



Example: Consider learning the parameter $p(x_1|x_2, x_3)$

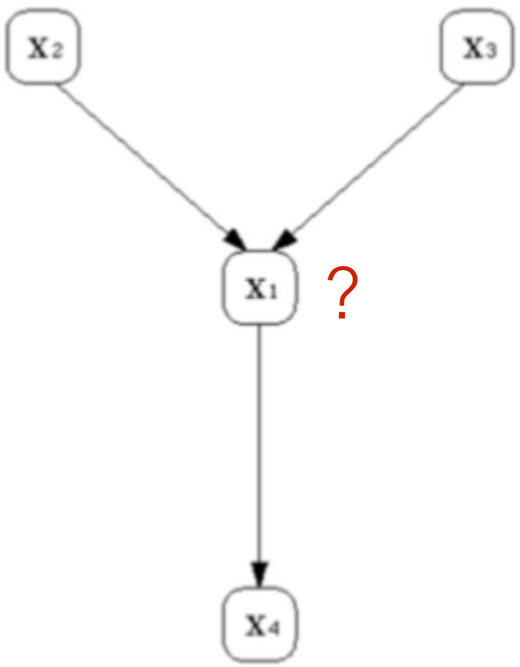
$$p(x_1|x_2,x_3) = \frac{p(x_1,x_2,x_3)}{p(x_2,x_3)} = \frac{card(x_1 \wedge x_2 \wedge x_3)}{card(x_2 \wedge x_3)} = \frac{\sum_{k=1}^{K} \delta(x_1 = 1, x_2 = 1, x_3 = 1)}{\sum_{k=1}^{K} \delta(x_2 = 1, x_3 = 1)}$$

one writes as well

$$\theta_{x_1|ij} = p(x_1 = 1|x_2 = i, x_3 = j) = \frac{\sum_{k=1}^{K} \delta(x_1 = 1, x_2 = i, x_3 = j)}{\sum_{k=1}^{K} \delta(x_2 = i, x_3 = j)}$$

Learning BN

- From Fully Observed Data
- **Partially Observed Data** via Maximum Likelihood Estimation



$$p(data|\theta) = \prod_{k=1}^{K} p(x_{(1,k)} x_{(2,k)}, x_{(3,k)}, x_{(4,k)})$$

$$p(data|\theta) = \prod_{k=1}^{K} p(x_{(4,k)}|x_{(1,k)}) \cdot p(x_{(1,k)}|x_{(2,k)}, x_{(3,k)}) \cdot p(x_{(2,k)}) \cdot p(x_{(3,k)})$$

$$\log p(data|\theta) = \sum_{k=1}^{K} \log p(x_{(4,k)}|x_{(1,k)}) + \log p(x_{(1,k)}|x_{(2,k)}, x_{(3,k)}) + \log p(x_{(2,k)}) + \log p(x_{(3,k)})$$

$$\frac{\partial \log p(data|\theta)}{\partial \theta_{x1|x2,x3}} = \sum_{k=1}^{K} \frac{\partial \log p(x_{(1,k)}|x_{(2,k)}, p(x_{(3,k)}))}{\partial \theta_{x1|x2,x3}}$$

$$\theta_{x_1|ij} = p(x_1 = 1|x_2 = i, x_3 = j) = \frac{\sum_{k=1}^{K} \delta(x_1 = 1, x_2 = i, x_3 = j)}{\sum_{k=1}^{K} \delta(x_2 = i, x_3 = j)}$$

Pomegranate

https://github.com/jmschrei/pomegranate/tree/master/examples

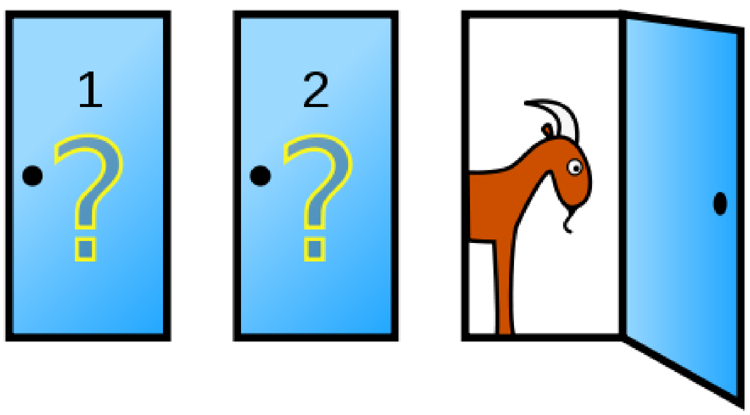
master ▾ pomegranate / examples / Go to file

ksachdeva and jmschrei Fix notebooks to work with python3 (#674) … × 0f412a8 on 12 Jan History

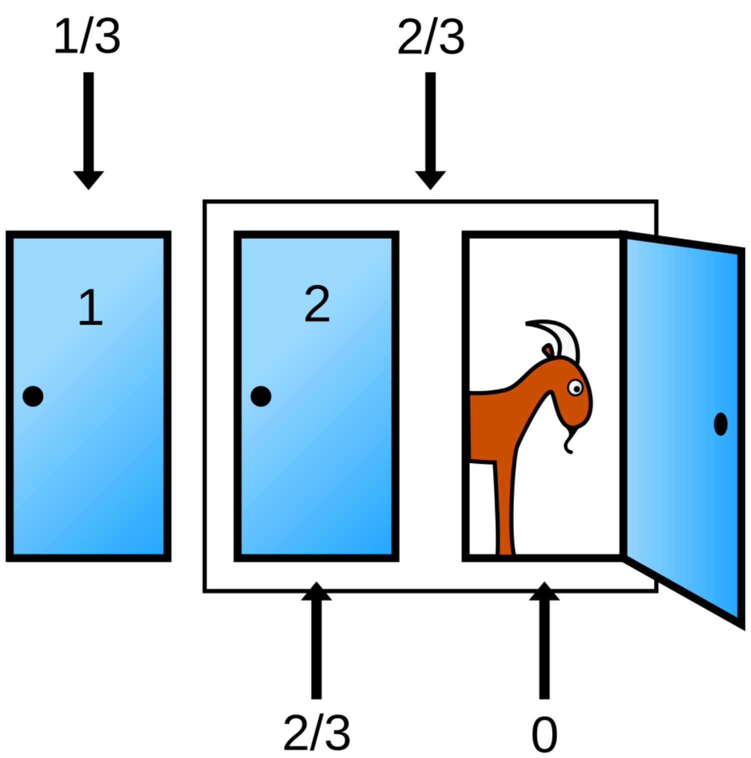
..

bayes_classifier_hmm_cheating_coin_toss.ipynb	ADD example	12 months ago
bayesnet_asia.ipynb	Fix notebooks to work with python3 (#674)	11 months ago
bayesnet_huge_monty_hall.ipynb	Fix notebooks to work with python3 (#674)	11 months ago
bayesnet_monty_hall_classic.ipynb	Fix notebooks to work with python3 (#674)	11 months ago
bayesnet_monty_hall_train.ipynb	Fix notebooks to work with python3 (#674)	11 months ago
distributions_conditionalupdate.ipynb	Fix notebooks to work with python3 (#674)	11 months ago
distributions_test_table.ipynb	Fix notebooks to work with python3 (#674)	11 months ago
hmm_example.ipynb	Fix notebooks to work with python3 (#674)	11 months ago
hmm_infinite.ipynb	Fix notebooks to work with python3 (#674)	11 months ago
hmm_rainy_sunny.ipynb	Fix notebooks to work with python3 (#674)	11 months ago
hmm_tied_states.ipynb	Fix notebooks to work with python3 (#674)	11 months ago
naivebayes_hmm_cheating_coin_toss.ipynb	Fix notebooks to work with python3 (#674)	11 months ago
naivebayes_multivariate_male_female.ipynb	Fix minor typos (#360)	3 years ago
naivebayes_simple_male_female.ipynb	Fix notebooks to work with python3 (#674)	11 months ago

Monty Hall Problem



In search of a new car, the player picks a door, say *1*.
The game host then opens one of the other doors, say *3*,
to reveal a goat and offers to let the player switch from door *1* to door *2*.
Should you switch?



Open the file:
C05_bayesnet_monty_hall_classic.ipynb

Team Challenge (Scope of Work Assignment 1)

Define a certain problem (it may be real or invented) and apply a Bayesian Network to it. You can either work with real data and find the probabilities by counting the events occurrence (take example 1 as reference), or, in case of not having the data, you can define the probabilities of the BN (take example 2 as reference) and then clearly explain why you decided to attribute those probabilities based on the problem in hand.