# Lecture 21: 10 April, 2025

Madhavan Mukund

https://www.cmi.ac.in/~madhavan

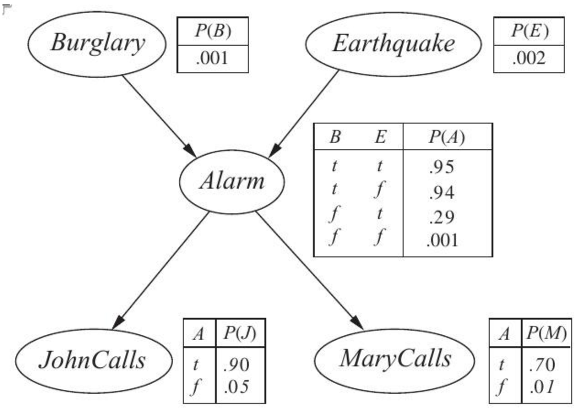
Data Mining and Machine Learning
January–April 2025

# Conditional probabilities

- Boolean variables $x_1, x_2, \ldots, x_n$
- Joint probabilities $P(v_1, v_2, \ldots, v_n)$
  - $2^n$ combinations of $x_1, x_2, \ldots, x_n$
  - $2^n - 1$ parameters
- Naïve Bayes assumption — complete independence
  - $P(x_i = 1)$ for each $x_i$
  - $n$ parameters
- Can we strive for something in between?
  - "Local" dependencies between some variables

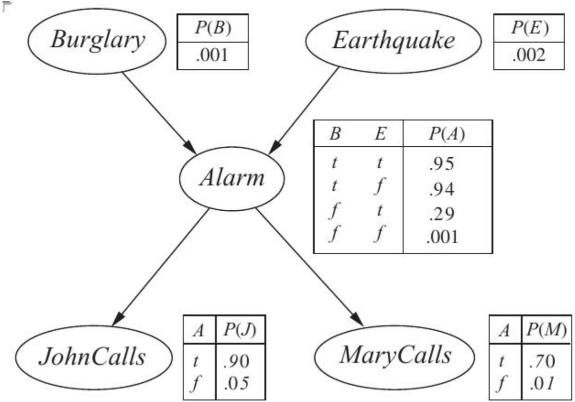
# Probabilistic graphical models — Judea Pearl, Turing Award 2011

- Represent local dependencies using directed graph
- Each node has a local (conditional) probability table
- Example: Burglar alarm
  - Pearl's house has a burglar alarm
  - Neighbours John and Mary call if they hear the alarm
  - John is prone to mistaking ambulances etc for the alarm
  - Mary listens to loud music and sometimes fails to hear the alarm
  - The alarm may also be triggered by an earthquake (California!)



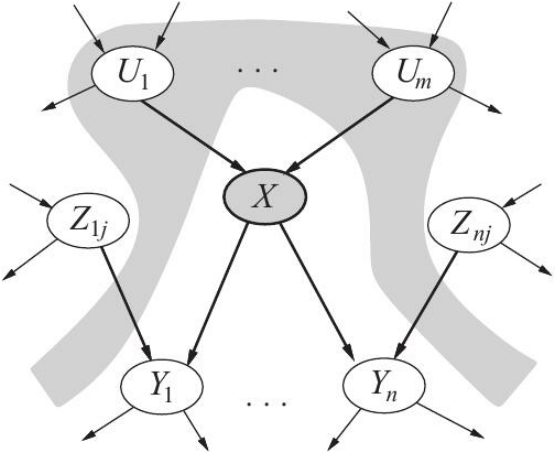
# Probabilistic graphical models

- Graph is a DAG, no cyclic dependencies
- Fundamental assumption:
  A node is conditionally independent of non-descendants, given its parents



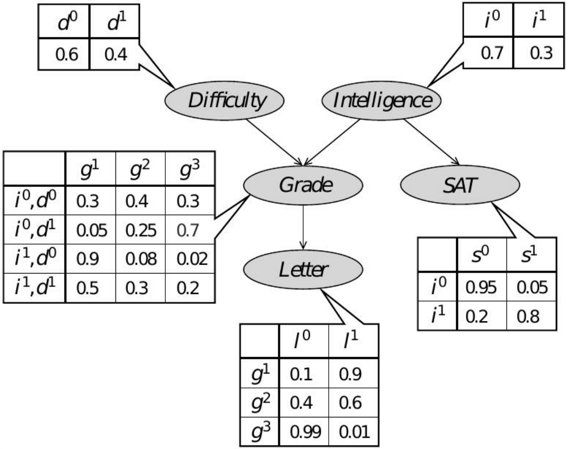
# Probabilistic graphical models

- Graph is a DAG, no cyclic dependencies
- Fundamental assumption:
  A node is conditionally independent of non-descendants, given its parents



# Student example

- Example due to Nir Friedman and Daphne Koller
- Student asks teacher for a reference letter
- Teacher has forgotten the student, so letter is entirely based on student's grade in the course



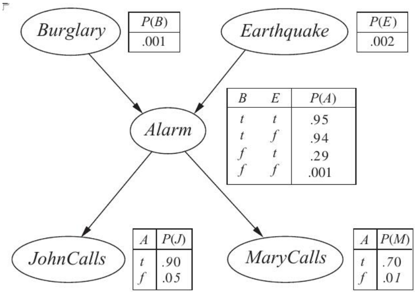
# Evaluating a network

- John and Mary call Pearl. What is the probability that there has been a burglary?
- $P(b, m, j)$, where $b$: burglary, $j$: John calls, $m$: Mary calls
- $P(b, m, j) = \sum_{a=0}^{1} \sum_{e=0}^{1} P(b, j, m, a, e)$, where $a$: alarm rings, $e$: earthquake
- Bayes Rule: $P(A, B) = P(A \mid B)P(B)$
- $P(x_1, x_2, \ldots, x_n) = P(x_1 \mid x_2, x_3, \ldots, x_n)P(x_2, x_3, \ldots, x_n)$
- Applied recursively, this gives us the chain rule

  $P(x_1, x_2, \ldots, x_n) = P(x_1 \mid x_2, \ldots, x_n)P(x_2 \mid x_3, \ldots, x_n) \cdots P(x_{n-1} \mid x_n)P(x_n)$

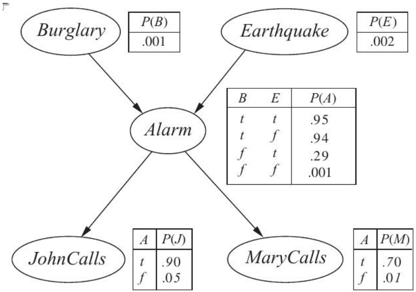
# Evaluating a network

- $P(x_1, x_2, \ldots, x_n) = P(x_1 \mid x_2, \ldots, x_n) P(x_2 \mid x_3, \ldots, x_n) \cdots P(x_{n-1} \mid x_n) P(x_n)$
- Can choose any ordering of $x_1, x_2, \ldots, x_n$
- Use topological ordering in a Bayesian network
- $P(m, j, a, b, e) =$
  $P(m \mid j, a, b, e) P(j \mid a, b, e) P(a \mid b, e) P(b \mid e) P(e)$
  $= P(m \mid a) P(j \mid a) P(a \mid b, e) P(b) P(e)$
- $P(m, j, b) =$
  $\sum_{a=0}^{1} \sum_{e=0}^{1} P(m \mid a) P(j \mid a) P(a \mid b, e) P(b) P(e)$



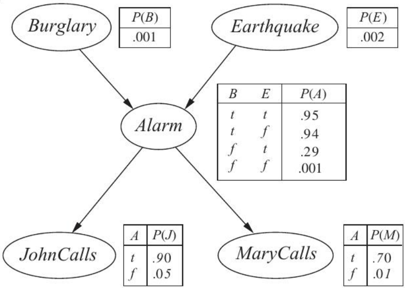
# Evaluating a network

- $P(x_1, x_2, \ldots, x_n) = P(x_1 \mid x_2, \ldots, x_n)P(x_2 \mid x_3, \ldots, x_n) \cdots P(x_{n-1} \mid x_n)P(x_n)$
- Can choose any ordering of $x_1, x_2, \ldots, x_n$
- Use topological ordering in a Bayesian network
- $P(m, j, a, b, e) =$
  $P(m \mid j, a, b, e)P(j \mid a, b, e)P(a \mid b, e)P(b \mid e)P(e)$
  $= P(m \mid a)P(j \mid a)P(a \mid b, e)P(b)P(e)$
- $P(m, j, b) =$
  $$\sum_{e=0}^{1} \sum_{a=0}^{1} P(m \mid a)P(j \mid a)P(a \mid b, e)P(b)P(e)$$



# Evaluating a network

- $P(x_1, x_2, \ldots, x_n) = P(x_1 \mid x_2, \ldots, x_n) P(x_2 \mid x_3, \ldots, x_n) \cdots P(x_{n-1} \mid x_n) P(x_n)$
- Can choose any ordering of $x_1, x_2, \ldots, x_n$
- Use topological ordering in a Bayesian network
- $P(m, j, a, b, e) =$
  $P(m \mid j, a, b, e) P(j \mid a, b, e) P(a \mid b, e) P(b \mid e) P(e)$
  $= P(m \mid a) P(j \mid a) P(a \mid b, e) P(b) P(e)$
- $P(m, j, b) =$
  $$\sum_{e=0}^{1} \sum_{a=0}^{1} P(m \mid a) P(j \mid a) P(a \mid b, e) P(b) P(e)$$
- $P(m, j, b) = P(b) \sum_{e=0}^{1} P(e) \sum_{a=0}^{1} P(m \mid a) P(j \mid a) P(a \mid b, e)$




| P(B) |
|---|
| .001 |

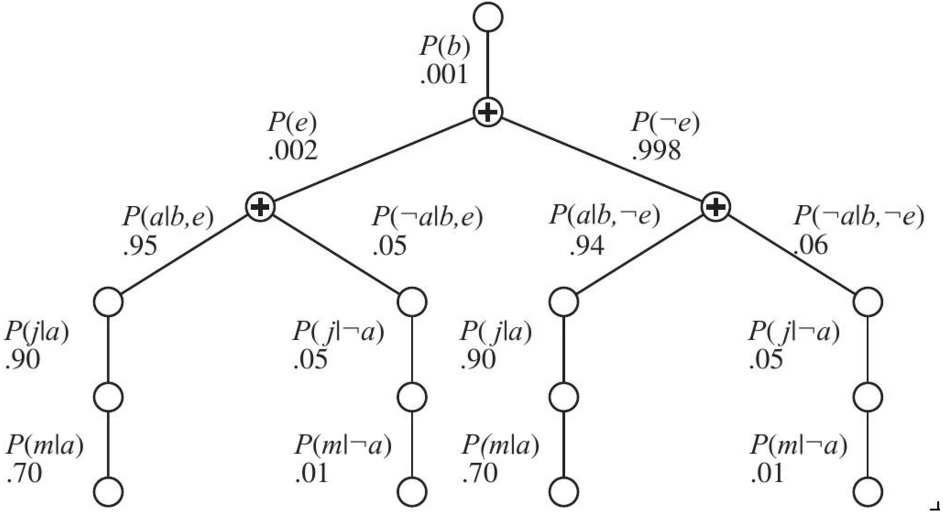
| P(E) |
|---|
| .002 |

| B | E | P(A) |
|---|---|---|
| t | t | .95 |
| t | f | .94 |
| f | t | .29 |
| f | f | .001 |

| A | P(J) |
|---|---|
| t | .90 |
| f | .05 |

| A | P(M) |
|---|---|
| t | .70 |
| f | .01 |

# Evaluation tree



# Designing the Bayesian network

- Need to choose node ordering wisely to get a compact Bayesian network
- Ordering *MaryCalls*, *JohnCalls*, *Alarm*, *Burglary*, *Earthquake* produces this network
- Ordering *MaryCalls*, *JohnCalls*, *Earthquake*, *Burglary*, *Alarm* is even worse
- Causal model (causes to effects) works better than diagnostic model (effects to causes)