UNIVERSITY OF ENGINEERING & MANAGEMENT, KOLKATA

Course Name : AI & ML

Bayesian Learning

Learning based on Bayes' Theorem

Brief recap of Probability

- Probability is a measure of the likelihood of an event to occur.
- **Probability of event to happen P(E) = $\frac{\text{Number of favourable outcomes}}{\text{Total Number of outcomes}}$**

Continued..

- **Joint Probability:** It tells the Probability of simultaneously occurring two random events.
- $P(A \cap B) = P(A).\ P(B)$
- Where;
- $P(A \cap B)$ = Probability of occurring events A and B both.
- P (A) = Probability of event A
- P (B) = Probability of event B

Continued..

- **Conditional Probability:** It is given by the Probability of event A given that event B occurred.
- The Probability of an event A conditioned on an event B is denoted and defined as;
- $P(A|B) = P(A \cap B)/P(B)$
- Similarly, $P(B|A) = P(A \cap B)/ P(A)$.
- $P(A \cap B) = p(A).P(B|A)$
- "The chance of both things happening is the chance that the first one happens, and then the second one is given when the first thing happened."

Continued..

- **Marginal Probability:**
- Marginal probability is defined as the probability of an event A occurring independent of any other event B. Further, it is considered as the probability of evidence under any consideration.
- $P(A) = P(A|B)*P(B) + P(A|\sim B)*P(\sim B)$

What is Bayes Theorem?

- Bayes theorem is one of the most popular machine learning concepts that help to calculate the probability of occurring one event with uncertain knowledge while another one has already occurred.
- Mathematically, Bayes theorem can be expressed by combining both equations on the right-hand side. We will get:
-

$$P(X|Y) = \frac{P(Y|X).P(X)}{P(Y)}$$

Details of Bayes Theorem

- Here, both events X and Y are independent events which means probability of the outcome of both events do not depend on one another.
- P(X|Y) is called as **posterior**, which we need to calculate. It is defined as updated probability after considering the evidence.
- P(Y|X) is called the likelihood. It is the probability of evidence when the hypothesis is true.
- P(X) is called the **prior probability**, the probability of the hypothesis before considering the evidence
- P(Y) is called marginal probability. It is defined as the probability of evidence under any consideration.
- Hence, Bayes Theorem can be written as:
- **posterior = likelihood * prior / evidence**

What is called Bayesian learning in the domain of Machine learning?

- Basically in terms of hypothesis (h) and given data, theorem can be re-written as
- P(h | D) = . $\frac{P(D|h)P(h)}{P(D)}$
- P(h) is also known as the prior probability of h, and it might reflect whatever prior knowledge we have about the likelihood that h is right.
- Similarly, P(D) is the prior probability of observing training data D. (i.e., the probability of D given no knowledge about which hypothesis holds).
- The likelihood of seeing data D in some environment where hypothesis h holds is denoted by P(D/h).
- The posterior probability of h is designated P(h/D) because it represents our confidence that h holds after seeing the training data D.
- In contrast to the prior probability P(h), which is independent of D, the posterior probability P(h/D) indicates the influence of the training data D.
- From the prior probability P(h), as well as P(D) and P(D/h), the Bayes theorem can be used to compute the posterior probability P(h/D).

MAP – Maximum A Posteriori hypothesis

- In ML, it is required to find out maximum probable hypothesis ‘h’ from a set of hypotheses ‘H’ given the observed training data. This maximally probable hypothesis is called the maximum a posteriori hypothesis (MAP). This can be identified from posterior probability of each candidate hypothesis:
- h MAP = argmax (h) P(h|D)

Derive the Maximum Posteriori

- Let D be a training set of tuples and their associated class labels, and each tuple is represented by an n-D attribute vector $\mathbf{X} = (x_1, x_2, \ldots, x_n)$
- Suppose there are *m* classes $C_1, C_2, \ldots, C_m$.
- Classification is to derive the maximum posteriori, i.e., the maximal $P(C_i|\mathbf{X})$
- This can be derived from Bayes' theorem

$$P(C_i|\mathbf{X}) = \frac{P(\mathbf{X}|C_i)P(C_i)}{P(\mathbf{X})}$$

- Since P(X) is constant for all classes, only

$$P(C_i|\mathbf{X}) = P(\mathbf{X}|C_i)P(C_i)$$

  needs to be maximized

An Example Problem

- Consider a medical diagnosis problem in which there are two alternative hypotheses:
- (1) that the patient; has a- particular form of cancer. and
- (2) that the patient does not.
- The available data is from a particular laboratory test with two possible outcomes: + (positive) and - (negative). We have prior knowledge that over the entire population of people only .008 have this disease. Furthermore, the lab test is only an imperfect indicator of the disease. The test returns a correct positive result in only 98% of the cases in which the disease is actually present and a correct negative result in only 97% of the cases in which the disease is not present.
- Check whether a person is suffering from cancer or not.

Solution

- P(cancer)=0.008
- P(NOT cancer)= 0.992
- P(+ | cancer) = 0.98, P (- |cancer) = 0.02
- P(- | NOT cancer) = 0.97, P (+ | NOT cancer) = 0.03
- Then,
- $P(\text{cancer} \mid +) = \frac{P(+\mid cancer) * P(cancer)}{P(+)}$
- $P(\text{NOT cancer} \mid +) = \frac{P(+\mid NOT\ cancer) * P(NOT\ cancer)}{P(+)}$
- Calculate both, whichever is higher that will be considered as final conclusion.

Naïve Bayes Classifier

- Naïve Bayes algorithm is a supervised learning algorithm, which is based on **Bayes theorem** and used for solving classification problems.
- **It is a probabilistic classifier, which means it predicts on the basis of the probability of an object**.
- The Naïve Bayes algorithm is comprised of two words Naïve and Bayes, Which can be described as:
- **Naïve**: It is called Naïve because it assumes that the occurrence of a certain feature is independent of the occurrence of other features.
- **Bayes**: It is called Bayes because it depends on the principle of Bayes' Theorem.

Naïve Bayes Classifier

- A simplified assumption: attributes are conditionally independent (i.e., no dependence relation between attributes):

$$P(\mathbf{X}|C_i) = \prod_{k=1}^{n} P(x_k|C_i) = P(x_1|C_i) \times P(x_2|C_i) \times ... \times P(x_n|C_i)$$

- This greatly reduces the computation cost: Only counts the class distribution
- If A_k is categorical, $P(x_k|C_i)$ is the # of tuples in C_i having value x_k for A_k divided by $|C_{i,D}|$ (# of tuples of C_i in D)
- If A_k is continous-valued, $P(x_k|C_i)$ is usually computed based on Gaussian distribution with a mean μ and standard deviation σ

and $P(x_k|C_i)$ is

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

$$P(\mathbf{X}|C_i) = g(x_k, \mu_{C_i}, \sigma_{C_i})$$

Example

Problem: If the weather is sunny, then the Player should play or not?

	Outlook	Play
0	Rainy	Yes
1	Sunny	Yes
2	Overcast	Yes
3	Overcast	Yes
4	Sunny	No
5	Rainy	Yes
6	Sunny	Yes
7	Overcast	Yes
8	Rainy	No
9	Sunny	No
10	Sunny	Yes
11	Rainy	No
12	Overcast	Yes
13	Overcast	Yes

Solution:-

Steps:

- Convert the given dataset into frequency tables.
- Generate Likelihood table by finding the probabilities of given features.
- Now, use Bayes theorem to calculate the posterior probability.

Frequency table for weather conditions

Weather	Yes	No
Overcast	5	0
Rainy	2	2
Sunny	3	2
Total	10	4

Likelihood table for weather conditions

Weather	**No**	**Yes**	
Overcast	**0**	**5**	**5/14= 0.35**
Rainy	**2**	**2**	**4/14=0.29**
Sunny	**2**	**3**	**5/14=0.35**
All	**4/14=0.29**	**10/14=0.71**	

Problem: If the weather is sunny, then the Player should play or not?

Applying Bayes‘ theorem:

- **P(Yes|Sunny)= P(Sunny|Yes)*P(Yes) / P(Sunny)**
- P(Sunny|Yes)= 3/10= 0.3
- P(Sunny)= 0.35
- P(Yes)=0.71
- So P(Yes|Sunny) = 0.3*0.71/0.35= **0.60**

Continued…

- **P(No|Sunny)= P(Sunny|No)*P(No)/P(Sunny)**
- P(Sunny|NO)= 2/4=0.5
- P(No)= 0.29
- P(Sunny)= 0.35
- So P(No|Sunny)= 0.5*0.29/0.35 = **0.41**
- So as we can see from the above calculation that **P(Yes|Sunny)>P(No|Sunny)**
- **Hence on a Sunny day, Player can play the game.**

Another example:-

X = (age = youth, income = medium, student = yes, credit rating = fair)

Class-labeled training tuples from the *AllElectronics* customer database.

RID	age	income	student	credit_rating	Class: buys_computer
1	youth	high	no	fair	no
2	youth	high	no	excellent	no
3	middle_aged	high	no	fair	yes
4	senior	medium	no	fair	yes
5	senior	low	yes	fair	yes
6	senior	low	yes	excellent	no
7	middle_aged	low	yes	excellent	yes
8	youth	medium	no	fair	no
9	youth	low	yes	fair	yes
10	senior	medium	yes	fair	yes
11	youth	medium	yes	excellent	yes
12	middle_aged	medium	no	excellent	yes
13	middle_aged	high	yes	fair	yes
14	senior	medium	no	excellent	no

Solution

- The class label attribute, buys computer, has two distinct values (namely, {yes, no}). Let C1 correspond to the class buys computer = yes and C2 correspond to buys computer = no.
- P(buys computer = yes) = 9/14 = 0.643
- P(buys computer = no) = 5/14 = 0.357
- P(age = youth | buys computer = yes) = 2/9 = 0.222
- P(age = youth | buys computer = no) = 3/5 = 0.600
- P(income = medium | buys computer = yes) = 4/9 = 0.444

Solution..

- P(income = medium | buys computer = no) = 2/5 = 0.400
- P(student = yes | buys computer = yes) = 6/9 = 0.667
- P(student = yes | buys computer = no) = 1/5 = 0.200
- P(credit rating = fair | buys computer = yes) = 6/9 = 0.667
- P(credit rating = fair | buys computer = no) = 2/5 = 0.400

Solution..

- Using the above probabilities, we obtain
- P(X|buys computer = yes) = P(age = youth | buys computer = yes) × P(income = medium | buys computer = yes) × P(student = yes | buys computer = yes) × P(credit rating = fair | buys computer = yes) = 0.222×0.444×0.667×0.667 = 0.044.
- Similarly, P(X|buys computer = no) = 0.600×0.400×0.200×0.400 = 0.019.
- To find the class, Ci , that maximizes P(X|Ci)P(Ci), we compute
- P(X|buys computer = yes)P(buys computer = yes) = 0.044×0.643 = 0.028
  P(X|buys computer = no)P(buys computer = no) = 0.019×0.357 = 0.007
- Therefore, the naïve Bayesian classifier predicts buys computer = yes for tuple X

Bayesian networks

- A simple, graphical notation for conditional independence assertions and hence for compact specification of full joint distributions

- Syntax:
  - a set of nodes, one per variable
  - a directed, acyclic graph (link ≈ "directly influences")
  - a conditional distribution for each node given its parents:

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

- In the simplest case, conditional distribution represented as a conditional probability table (CPT) giving the distribution over X_i for each combination of parent values
- A node is independent of its non-descendants given its parents.

Bayesian belief networks

Bayesian belief networks (BBNs)

Bayesian belief networks.

- Represent the full joint distribution over the variables more compactly with a **smaller number of parameters**.
- Take advantage of **conditional and marginal independences** among random variables

- **A and B are independent**

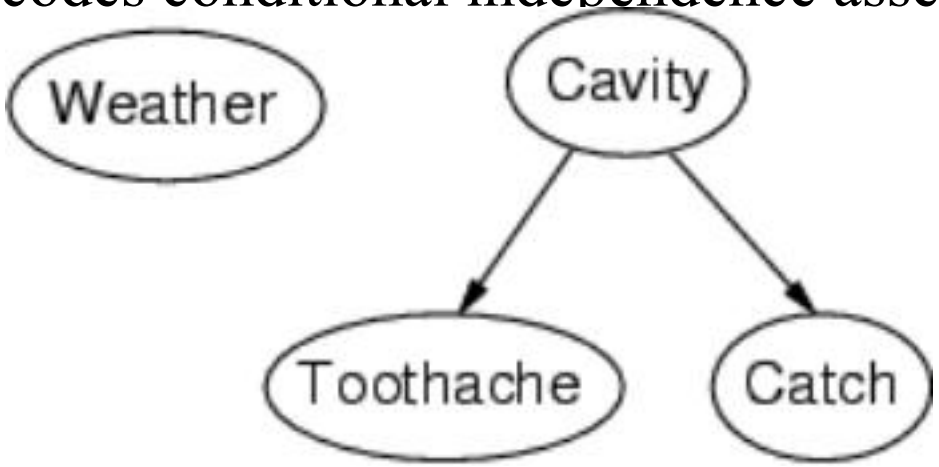
$$P(A, B) = P(A)P(B)$$

- **A and B are conditionally independent given C**

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

Example

- Topology of network encodes conditional independence assertions:



- *Weather* is independent of the other variables
- *Toothache* and *Catch* are conditionally independent given *Cavity*

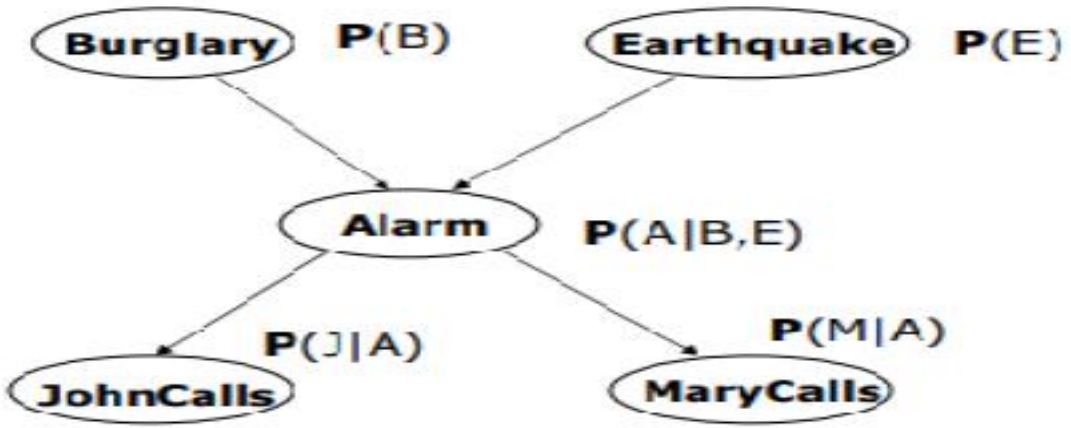
Example

- Assume your house has an alarm system against burglary. You live in the seismically active area and the alarm system can get occasionally set off by an earthquake. You have two neighbors, Mary and John, who do not know each other. If they hear the alarm they call you, but this is not guaranteed.
- We want to represent the probability distribution of events: – Burglary, Earthquake, Alarm, Mary calls and John calls
- 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

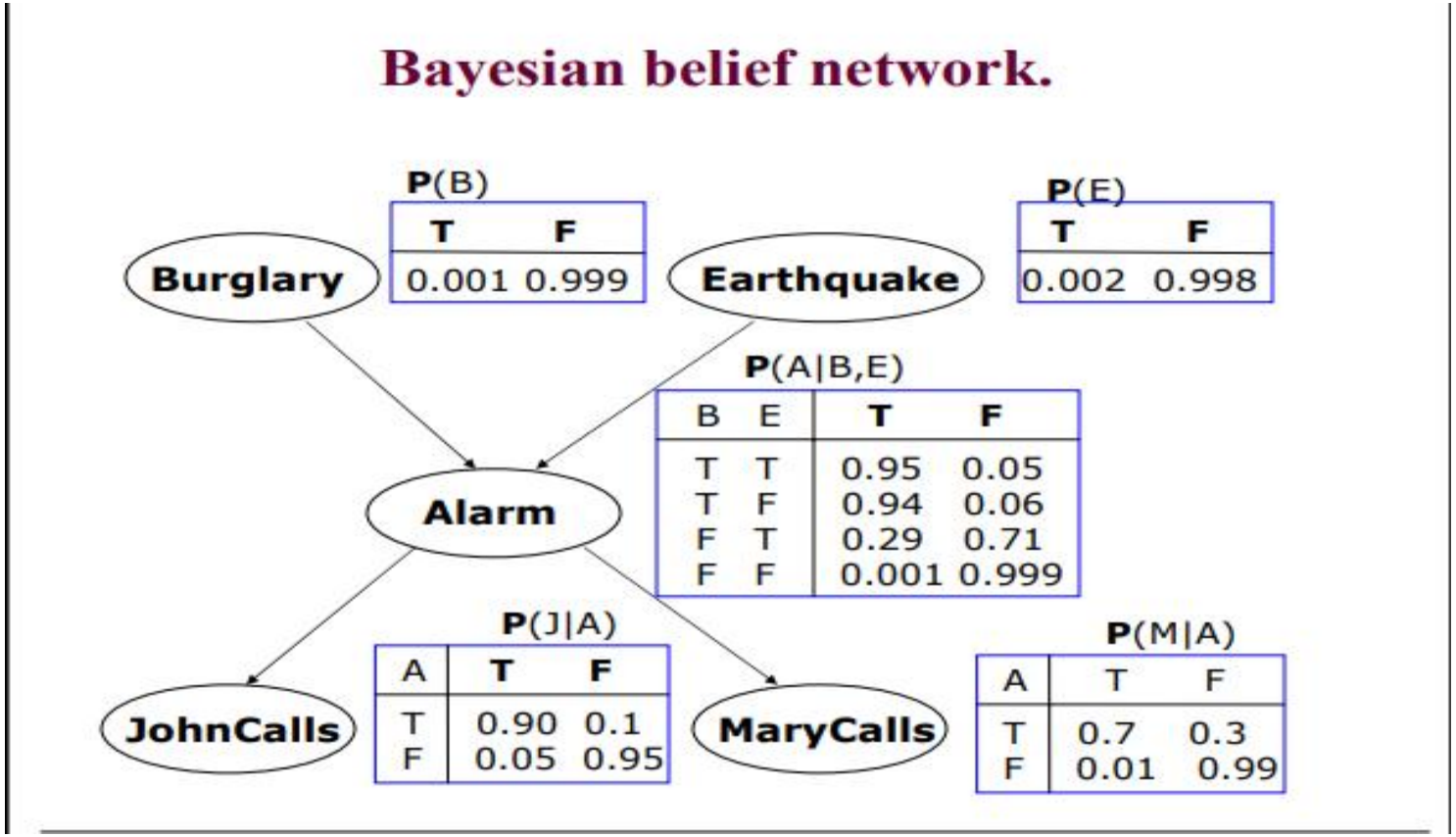
Network

Bayesian belief network.

- relate variables and their parents



Network with CPT



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

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

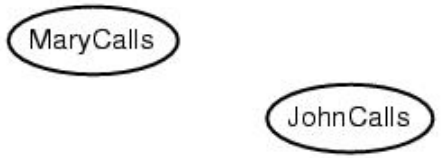
This choice of parents guarantees:

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

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

Example

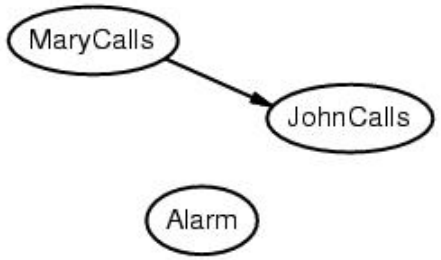
- Suppose we choose the ordering *M, J, A, B, E*



$\boldsymbol{P}(J \mid M) = \boldsymbol{P}(J)?$

Example

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

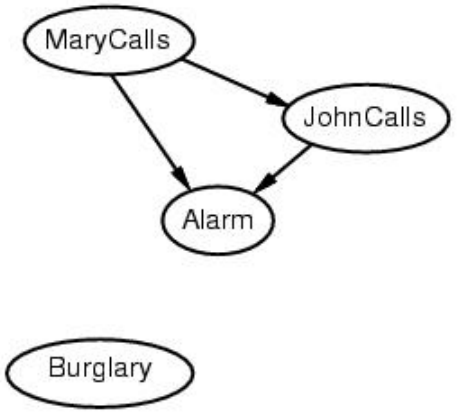


$\boldsymbol{P}(J \mid M) = \boldsymbol{P}(J)$ **No**

$\boldsymbol{P}(A \mid J, M) = \boldsymbol{P}(A \mid J)$? $\boldsymbol{P}(A \mid J, M) = \boldsymbol{P}(A)$?

Example

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



$\boldsymbol{P}(J \mid M) = \boldsymbol{P}(J)$ **No**

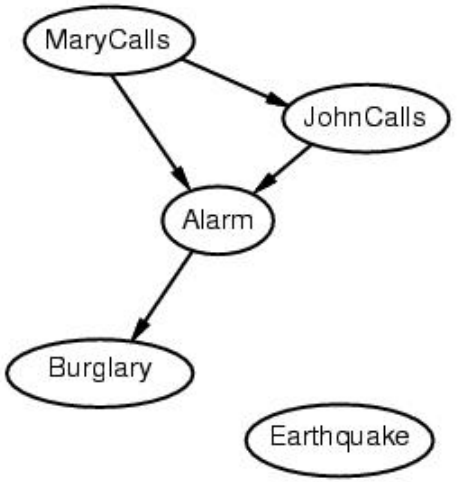
$\boldsymbol{P}(A \mid J, M) = \boldsymbol{P}(A \mid J)$? $\boldsymbol{P}(A \mid J, M) = \boldsymbol{P}(A)$? **No**

$\boldsymbol{P}(B \mid A, J, M) = \boldsymbol{P}(B \mid A)$?

$\boldsymbol{P}(B \mid A, J, M) = \boldsymbol{P}(B)$?

Example

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



$\boldsymbol{P}(J \mid M) = \boldsymbol{P}(J)$ **No**

$\boldsymbol{P}(A \mid J, M) = \boldsymbol{P}(A \mid J)$? $\boldsymbol{P}(A \mid J, M) = \boldsymbol{P}(A)$? **No**

$\boldsymbol{P}(B \mid A, J, M) = \boldsymbol{P}(B \mid A)$? **Yes**

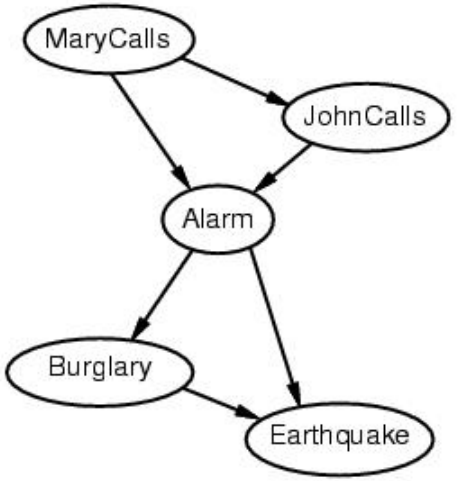
$\boldsymbol{P}(B \mid A, J, M) = \boldsymbol{P}(B)$? **No**

$\boldsymbol{P}(E \mid B, A, J, M) = \boldsymbol{P}(E \mid A)$?

$\boldsymbol{P}(E \mid B, A, J, M) = \boldsymbol{P}(E \mid A, B)$?

Example

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



$\boldsymbol{P}(J \mid M) = \boldsymbol{P}(J)$ **No**

$\boldsymbol{P}(A \mid J, M) = \boldsymbol{P}(A \mid J)$? $\boldsymbol{P}(A \mid J, M) = \boldsymbol{P}(A)$? **No**

$\boldsymbol{P}(B \mid A, J, M) = \boldsymbol{P}(B \mid A)$? **Yes**

$\boldsymbol{P}(B \mid A, J, M) = \boldsymbol{P}(B)$? **No**

$\boldsymbol{P}(E \mid B, A, J, M) = \boldsymbol{P}(E \mid A)$? **No**

$\boldsymbol{P}(E \mid B, A, J, M) = \boldsymbol{P}(E \mid A, B)$? **Yes**

Some Applications of BN

- Medical diagnosis
- Troubleshooting of hardware/software systems
- Fraud/uncollectible debt detection
- Data mining
- Analysis of genetic sequences
- Data interpretation, computer vision, image understanding

Thank You