CS 4260 and CS 5260
Vanderbilt University

Lecture on Uncertainty (Belief Networks)

This lecture assumes that you have

- Read Section 8.1 through 8.3 of ArtInt (though there is some repetition, as well as additional material)

ArtInt: Poole and Mackworth, Artificial Intelligence 2E
at http://artint.info/2e/html/ArtInt2e.html
to include slides at http://artint.info/2e/slides/ch04/lect1.pdf

Douglas H. Fisher

Feedback for Quiz Q-w9

Q1. Consider the binary-valued variable, W, with a domain of {w, ~w}. What is the minimum number of probabilities that need to be stored so that the probability of any assignment of W can be obtained (i.e., P(w), P(~w)).

Options:

0

1 <—

2

4

1 is the correct answer. Only P(w) need be stored, and P(~w) can be computed as 1-P(w). Alternatively, P(~w) could be stored, and P(w) computed.

Q2. Consider binary-valued variables W and X, where the domain of W is {w, ~w} and the domain of X is {x, ~x}. What is the minimum number of probabilities that need to be stored so that the probability of each assignment of values to W and X (i.e., P(w,x), P(w,~x), P(~w,x), P(~w,~x)) can be obtained?

Options:

1

2

3 <—

4

3 is the correct answer. P(~w, ~x) can be computed as 1 minus the sum of the other three, for example.

Answers to both Q1 and Q2 are consistent with the text's statement that $2^n - 1$ probabilities must be specified explicitly for value assignments of n binary variables (see beginning of section 8.2).

Q3. Consider binary-valued variables W and X, where the domain of W is {w, ~w} and the domain of X is {x, ~x}. Further, assume that W and X are independent of each other.

What is the minimum number of probabilities that need to be stored so that the probability of each assignment of values of W and X (i.e., P(w,x), P(w,~x), P(~w,x), P(~w,~x)) can be obtained?

Options:

1

2 <—

3

4

The answer is 2. If W and X are independent, then the joint probability of any pair of W,X values can be computed as the product of the individual probabilities of those values. For example, P(w,~x) = P(w) * P(~x). We only need store P(w) and P(x), from which P(~w) and P(~x), for example.

Douglas H. Fisher

Q4: Consider variables W, X, Y, and Z. All four of these variables are binary valued, so that W has a domain of w and $\sim w$, for example.

The joint probability distribution, $P(W, X, Y, Z)$, is specified by assigning values to probabilities to each combination of values. There are 16 such assignments necessary to specify the joint distribution:

$P(w, x, y, z)$
$P(w, x, y, \sim z)$
$P(w, x, \sim y, z)$
…
$P(\sim w, \sim x, \sim y, \sim z)$

Actually, there are only 15 assignments that need to be explicitly made because the sum of all assignments must sum to 1.0, so the last of the 16, say $P(\sim w, \sim x, \sim y, \sim z)$, can be computed by 1.0 - (sum of the other 15 probabilities).

Consider the following assumptions.

X is independent of W.
Y is conditionally independent of X given W.
Z is conditionally independent of W given X and Y.

Under these assumptions, how many probabilities need to be stored to compute the value of any assignment in $P(W, X, Y, Z)$ (e.g., $P(w, \sim x, \sim y, z)$). It may be helpful to recall the Chain Rule (e.g., $P(w, \sim x, \sim y, z) = P(w)*P(\sim x|w)*P(\sim y|w,\sim x)*P(z|w,\sim x,\sim y)$).

Options:

4

8 <—

15

16

8 is the correct answer. If there were NO (conditional) independencies, then the text's guideline of requiring 2^4 - 1 (or 15) probabilities would be correct. But with application of the chain rule, some factorizations can lead to reduced numbers of probabilities that need to be satisfied.

For example, $P(w, \sim x, \sim y, z) = P(w)*P(\sim x|w)*P(\sim y|w,\sim x)*P(z|w,\sim x,\sim y)$.

$P(w)$ needs to be specified. (count of 1 so far)

$P(\sim x|w) = P(\sim x)$ because X is independent of W. $P(x)$ needs to be specified. (count of 2 so far)

$P(\sim y|w,\sim x) = P(\sim y|w)$ because Y is conditionally independent of X given W. $P(y|w)$ and $P(y|\sim w)$ need to be specified (and $P(\sim y|w)$ and $P(\sim y|\sim w)$ can be computed. (count of 4 so far)

$P(z|w,\sim x,\sim y) = P(z|\sim x,\sim y)$ because Z is conditionally independent of W given X and Y. $P(z|x,y)$, $P(z|x,\sim y)$, $P(z|\sim x,y)$, $P(z|\sim x,\sim y)$ need to be specified, for example, from which conditional probabilities of $\sim z$ can be computed. (count of 8 total)

Douglas H. Fisher

Belief (or Bayesian) Networks

Consider an ordering of variables to factor a joint probability distribution: W, X, Y, Z

e.g. P(w and x and ~y and z) factorization ordering

= P(w) * P(x|w) * P(~y|w, x) * P(z|w, x, ~y)

Assume the following (conditional) independencies:

P(W)

X independent of W

P(X|W) = P(X), i.e., P(x|w) = P(x) and P(x|~w) = P(x), P(~x|w) = P(~x), P(~x|~w) = P(~x)

Y independent of X conditioned on W

P(Y|W, X) = P(Y|W), i.e.,

1 number instead of 2 numbers

P(y|w, x) = P(y|w), P(y|w,~x) = P(y|w), P(y|~w, x) = P(y|~w), P(y|~w,~x) = P(y|~w)

P(~y|w, x) = P(~y|w), P(~y|w,~x) = P(~y|w), P(~y|~w, x) = P(~y|~w), P(~y|~w,~x) = P(~y|~w)

Z independent of W conditioned on X and Y

P(Z|W, X, Y) = P(Z|X, Y), i.e.,

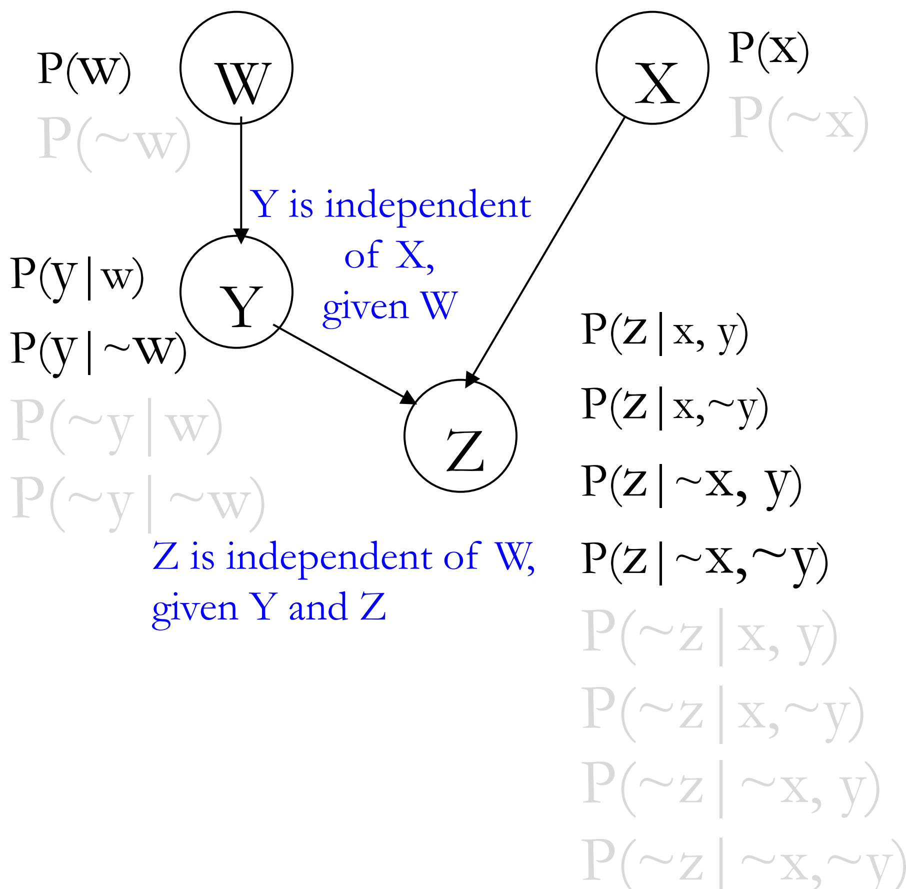
P(z|w, x, y) = P(z|x, y) and P(z|w, x,~y) = P(z|x, ~y) …… P(~z|~w, ~x, ~y) = P(~z|~x, ~y)

A Bayesian Network is a graphical representation of a joint probability distribution with (conditional) independence relationships made explicit

In particular, each variable (node) is (conditionally) independent of its non-descendants given its parents (i.e., given assigned values for each parent).

W has no parents – it is independent of X

X has no parents – it is independent of W (and Y)



Probabilities in light font, like this, can be computed rather than explicitly stored

Douglas H. Fisher

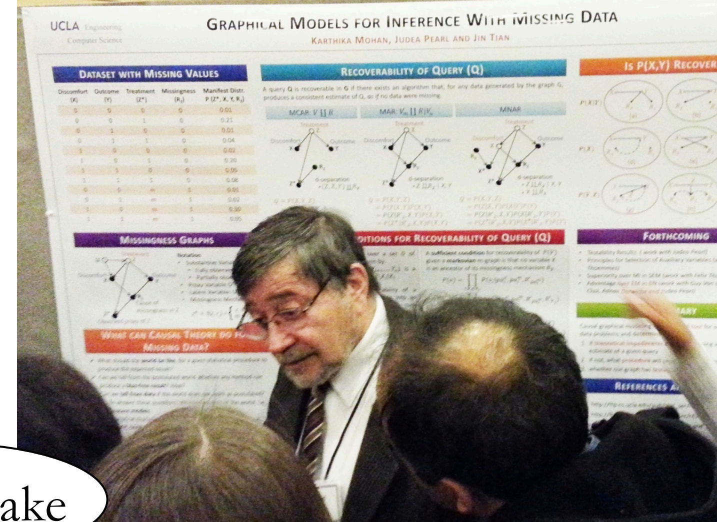
Space savings due to BNs and conditional independencies generally

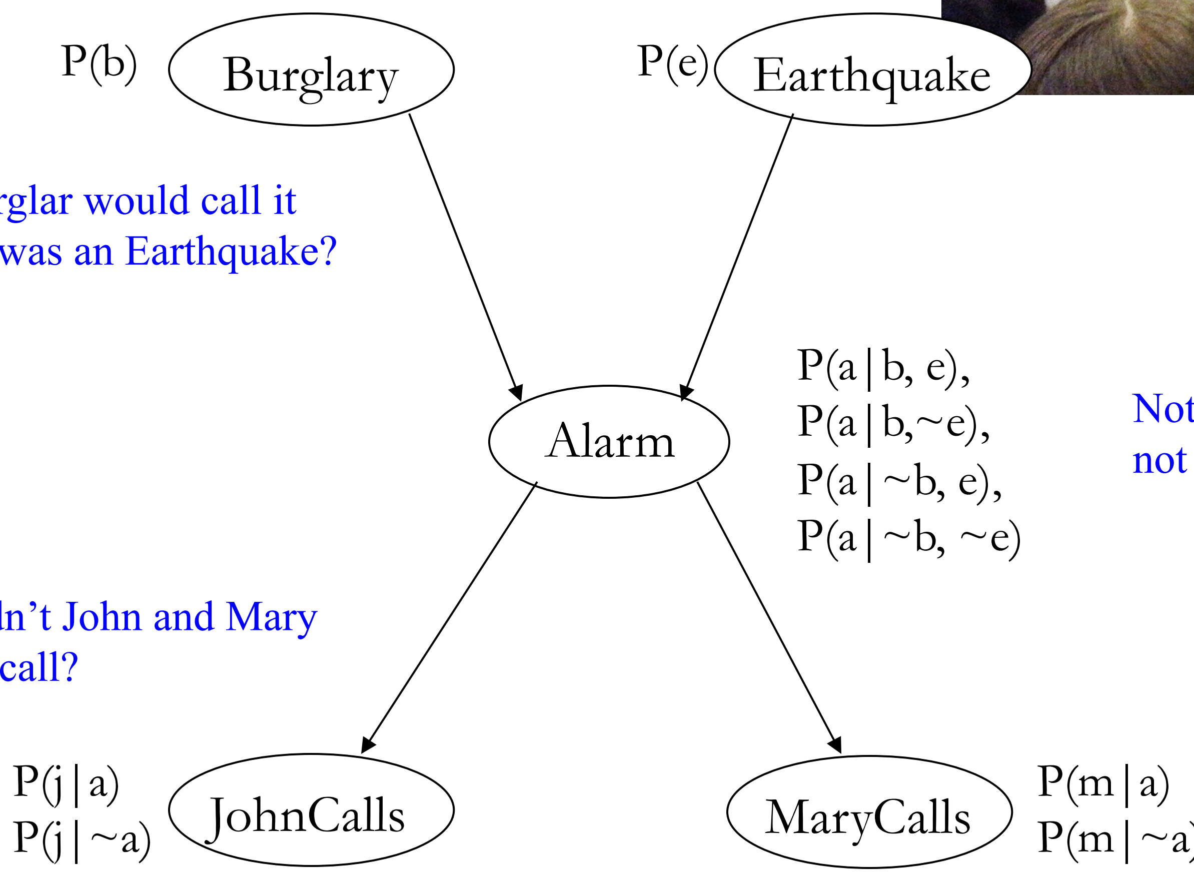
A bit more generally, assume n Boolean variables (for simplicity of analysis)

- 2^n joint probabilities that need be stored for n variables (actually, 2^n-1), in general
- In contrast, assume each variable directly influenced by at most k others (parents)
  - Then each probability table will be at most 2^k (actually 2^k-1) numbers
  - And complete network stores at most $n2^k$ numbers
- If n = 30 and k=5 then BN stores at most 960 numbers, compared to over 1,000,000,000 for full joint distribution

Illustration due to Russell and Norvig, Artificial Intelligence, 3rd edition

Example due to Judea Pearl





Perhaps burglar would call it off if there was an Earthquake?

Note state of the battery not explicitly stated

Why wouldn't John and Mary not always call?

Recall the chain rule:

Assume V_i a binary valued variable (T or F)

$$P(V_1 \text{ and } V_2 \text{ and } V_3 \text{ and } V_4 \text{ and } V_5)$$

A factorization ordering

$$= \underbrace{\underbrace{\underbrace{\underbrace{P(V_1)P(V_2 | V_1)}_{P(V_1, V_2)}P(V_3 | V_1, V_2)}_{P(V_1, V_2, V_3)}P(V_4 | V_1, V_2, V_3)}_{P(V_1, V_2, V_3, V_4)}P(V_5 | V_1, V_2, V_3, V_4)}_{P(V_1, V_2, V_3, V_4, V_5)}$$

$$P(V_1 \text{ and } V_2 \text{ and } V_3 \text{and } V_4 \text{ and } V_5)$$

An alternative ordering

$$= P(V_4)P(V_2 | V_4)P(V_3 | V_4, V_2)P(V_1 | V_4, V_2, V_3)P(V_5 | V_4, V_2, V_3, V_1)$$

Constructing a belief network

For a *particular factorization ordering*, construct a network as follows (Section 8.3.2 of text):

V_1 a "root"

$P(v_1)$, $P(\sim v_1)$

V_1

$P(v1) = 0.75$

$P(\sim v1) = 0.25 = 1 - P(v1)$

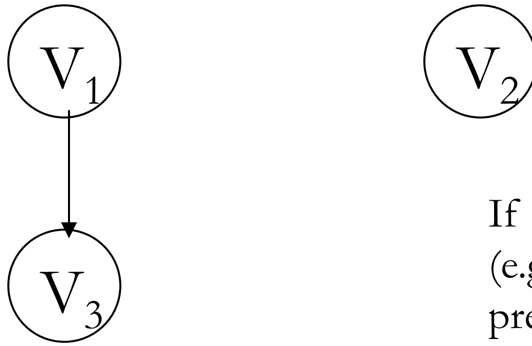
V_2 is second variable in ordering. If V_2 independent of a subset of its predecessors (possibly the empty set), conditioned on a disjoint subset of predecessors (including possibly all its predecessors), then the latter subset is its parents, else V_2 is a "root"

Suppose $P(V_2 | V_1) = P(V_2)$

$P(v1)$ V_1 V_2 $P(v2)$

V_3 is third variable in ordering.

$P(v1)$



$P(v2)$

$P(v3 | v1)$

$P(v3 | \sim v1)$

$P(\sim v3 | v1) = 1 – P(v3 | v1)$

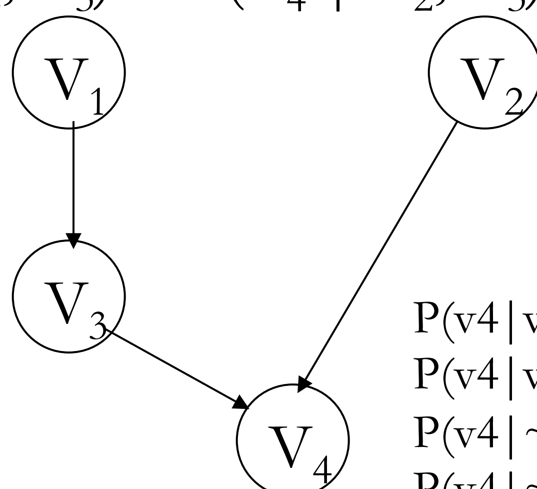
$P(\sim v3 | \sim v1) = 1 – P(v3 | \sim v1)$

If V_3 independent of a subset of its predecessors (e.g., $\{V_2\}$), conditioned on a disjoint subset of predecessors (e.g., $\{V_1\}$) , then the latter subset is its parents.

Assume $P(V_3 | V_1, V_2) = P(V_3 | V_1)$

Assume $P(V_4 | V_1, V_2, V_3) = P(V_4 | V_2, V_3)$

$P(v1)$



$P(v2)$

$P(v3 | v1)$

$P(v3 | \sim v1)$

$P(v4 | v2, v3)$, $P(\sim v4 | v2,v3) = 1-P(v4 | v2,v3)$

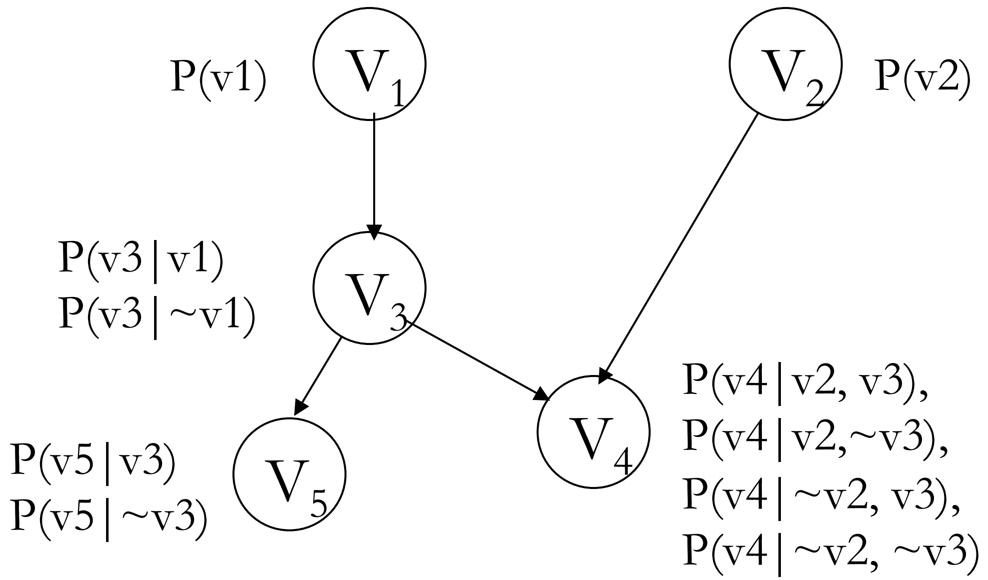
$P(v4 | v2,\sim v3)$, …

$P(v4 | \sim v2, v3)$, …

$P(v4 | \sim v2, \sim v3)$, …

Douglas H. Fisher

Assume $P(V_5 | V_1, V_2, V_3, V_4) = P(V_5 | V_3)$ (and $P(V_5 | V_1, V_2, V_3, V_4) = P(V_5 | V_1, V_4)$)



Components of a Bayesian Network: a topology (graph) that qualitatively indicates displays the conditional independencies, and probability tables at each node

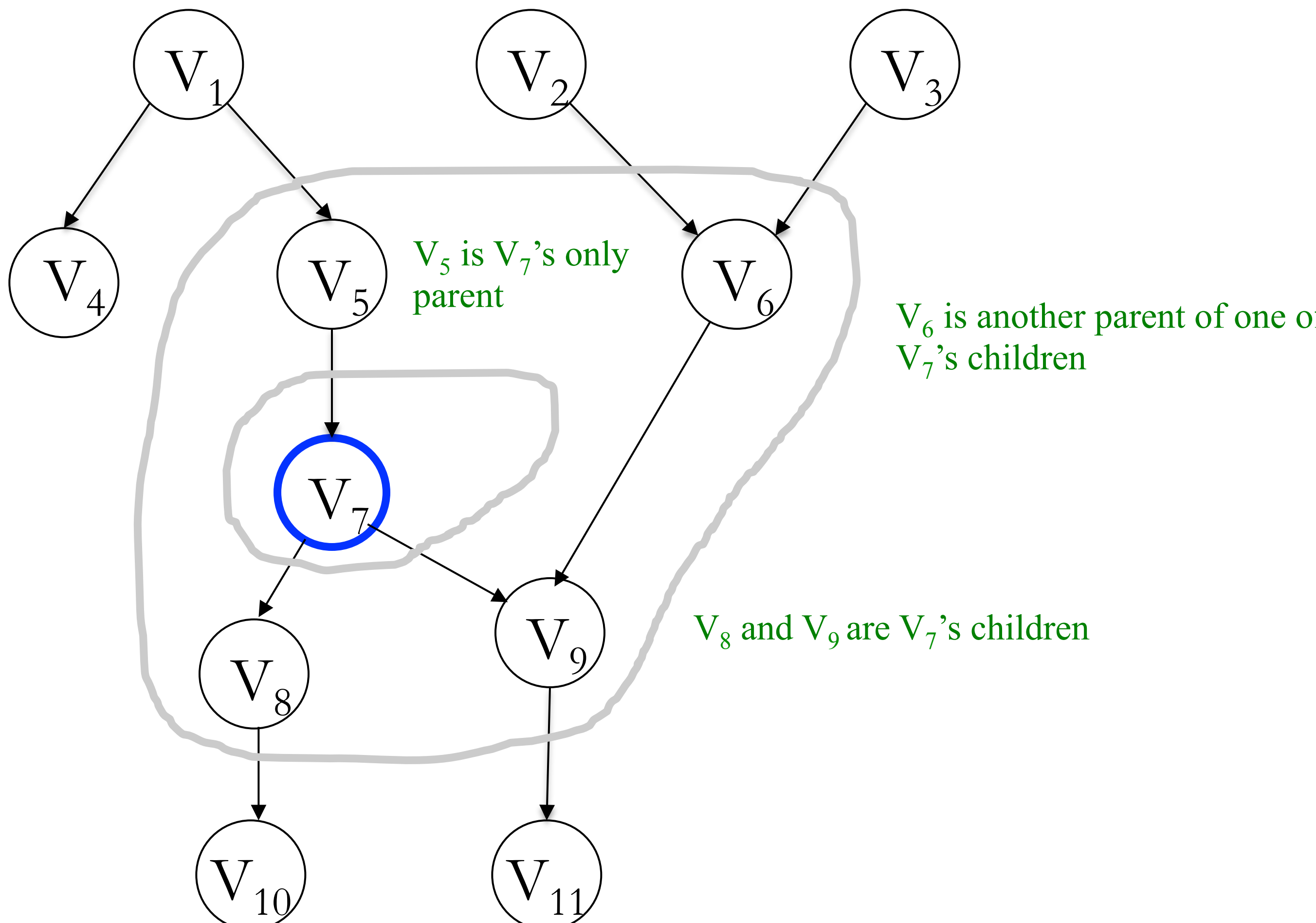
Semantics of graphical component: for each variable, V_i , then V_i is independent of all of its non-descendants conditioned on its parents

~~I will add another slide concerned with conditional indepencies conditioned on a node's Markov Blanket~~ (next slide)

Markov Blanket

Additional conditional independence property: for each variable, V_i, then V_i is independent of *all other nodes* conditioned on its Markov Blanket

The Markov Blanket of a node, is all the node's parents, all the node's children, and all the other parent's of the node's children



The Markov Blanket of V_7 is surrounded in grey: V_5, V_6, V_8, V_9

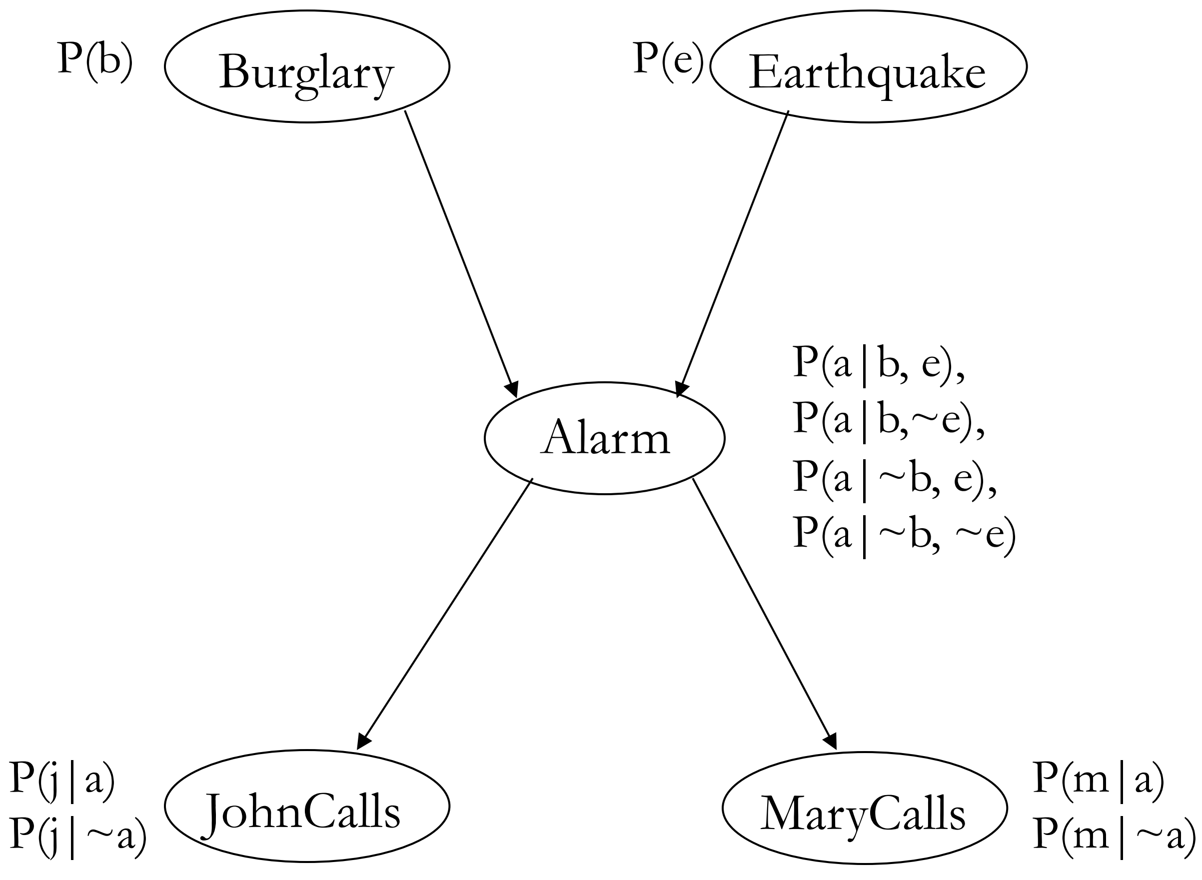
So, V_7 is conditionally independent of V_1, V_2, V_3, V_4, V_{10}, V_{11}, conditioned on V_5, V_6, V_8, V_9

Douglas H. Fisher

Any order of the variables can lead to a "correct" BN, but
the order that the variables are considered can yield BNs of very different complexity

This BN might have been constructed with ordering of

Burglary, Earthquake, Alarm, JohnCalls, MaryCalls



P(Burglary)

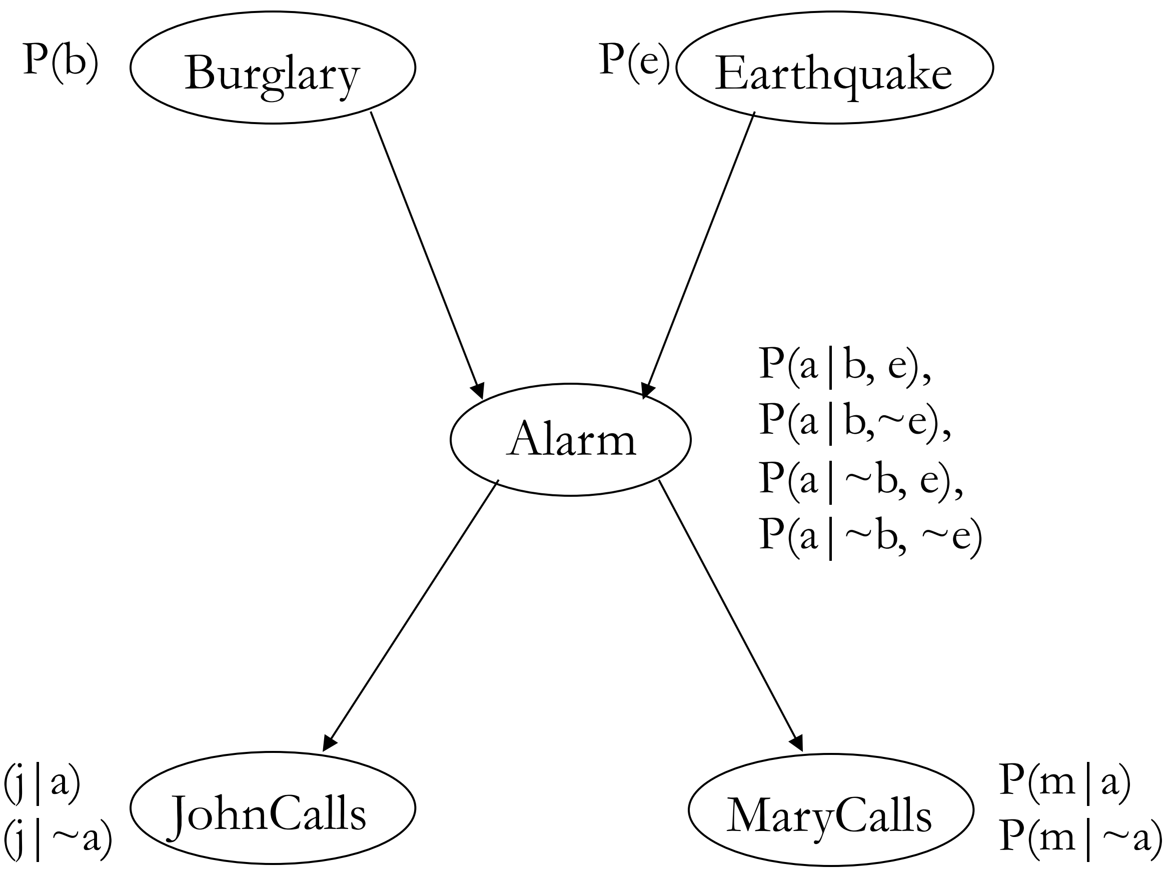
P(Earthquake | Burglary) = P(Earthquake)

P(Alarm | Burglary, Earthquake) no simplification

P(JohnCalls | Burglary, Earthquake, Alarm) = P(JohnCalls | Alarm)

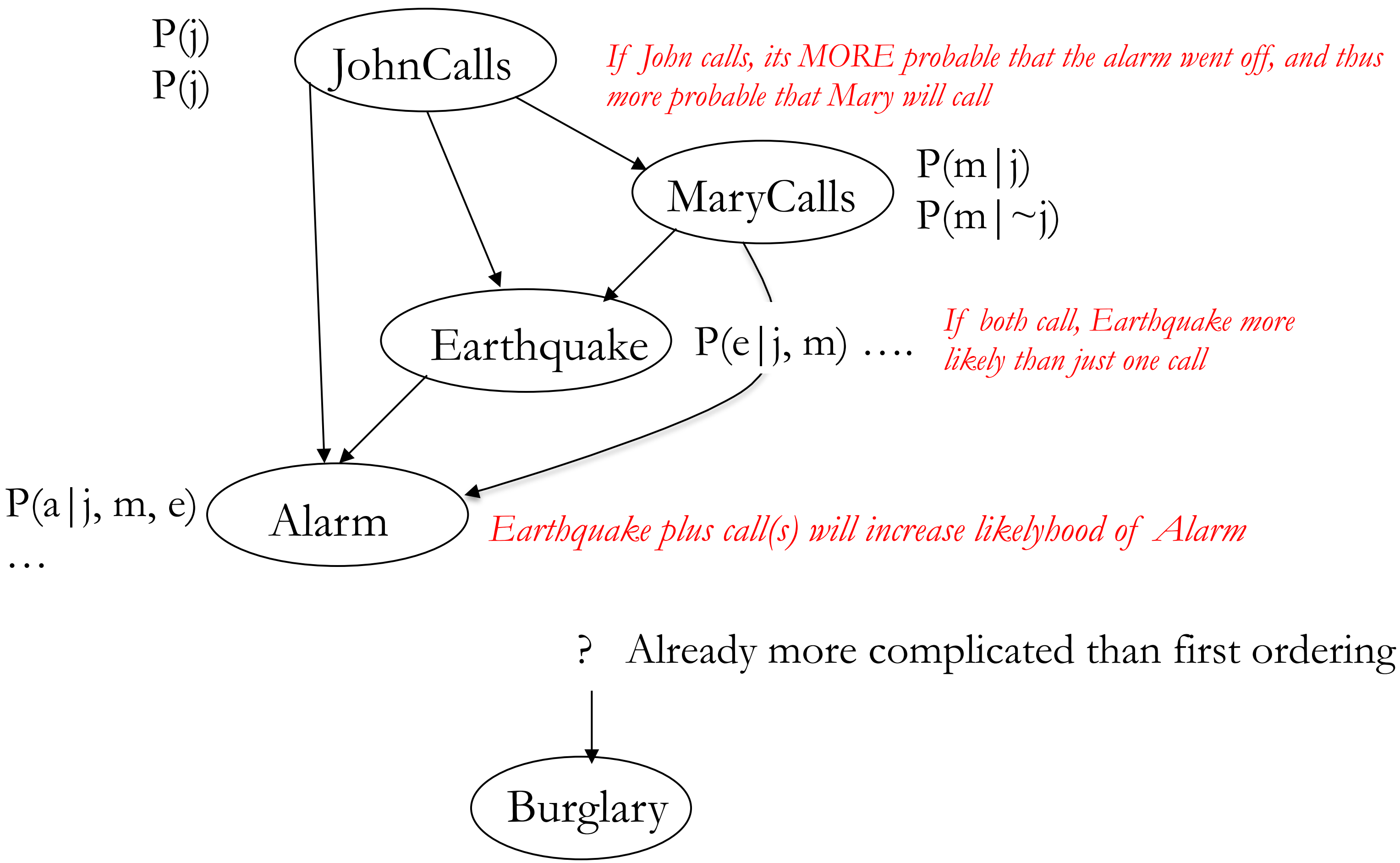
P(MaryCalls | Burglary, Earthquake, Alarm, JohnCalls) = P(MaryCalls | Alarm)

Equalities we believe are true or "close enough" to justify BN construction as shown



Douglas H. Fisher

How about ordering of JohnCalls, MaryCalls, Earthquake, Alarm, Burglary?



In general, ordering from "causes" to "manifestations" leads to simpler networks

Illustration due to Russell and Norvig, Artificial Intelligence, 3rd edition, as well as Judea Pearl

Where does knowledge of conditional independence come from?

a) From data. Consider congressional voting records. Suppose that we have data on House votes (and political party). Suppose variables are ordered

Party, Immigration, StarWars, ….

Party P(Republican) = 0.52 (226/435 Republicans
209/435 Democrats)

To determine relationship between Party and Immigration, we count

Actual Counts

	Immigration Yes	Immigration No
Republican	17	209
Democrat	160	49

Predicted Counts (if Immigration and Party independent)

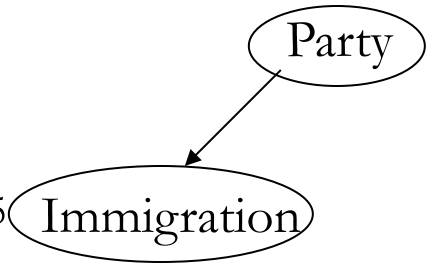
	Yes	No
Republican	92	134
Democrat	85	124

Very different distributions – conclude **dependent**

P(Rep)*P(Yes) * 435
= 0.52 * (17+160)/435 * 435

17/226

P(Yes | Rep) = 0.075
P(Yes | Dem) = 0.765



P(Republican) = 0.52 (226/435 Republicans
209/435 Democrats)

Actual Counts

	Immigration Yes	Immigration No
Republican	17	209
Democrat	160	49

Consider StarWars

- Is StarWars independent of Party and Immigration?
  - (i.e., is P(StarWars | Party, Immigration) approx equal P(StarWars) for all combinations of variable values?)
  - if yes, then stop and make StarWars a "root", else continue
- Is StarWars independent of Immigration conditioned on Party?
  - if yes, then stop and make StarWars a child of Party, else continue
- Is StarWars independent of Party conditioned on Immigration?
  - if yes, then stop and make StarWars a child of Immigration, else continue
- Make StarWars a child of both Party and Immigration

17/226

Party P(Republican) = 0.52 (226/435 Republicans
209/435 Democrats)

P(Yes | Rep) = 0.075
P(Yes | Dem) = 0.765 Immigration

	Actual Counts: Immigration Yes	Actual Counts: Immigration No	Actual Counts: StarWars Yes	Actual Counts: StarWars No
Republican	17	209	219	7
Democrat	160	49	24	185

Consider StarWars

Is StarWars independent of Party and Immigration?

Actual Counts

	Immigration Yes, StarWars Yes	Immigration Yes, StarWars No	Immigration No, StarWars Yes	Immigration No, StarWars No
Republican	14	3	205	4
Democrat	8	152	16	33

P(Rep & Imm=Y)P(SW=Y)435

Predicted Counts

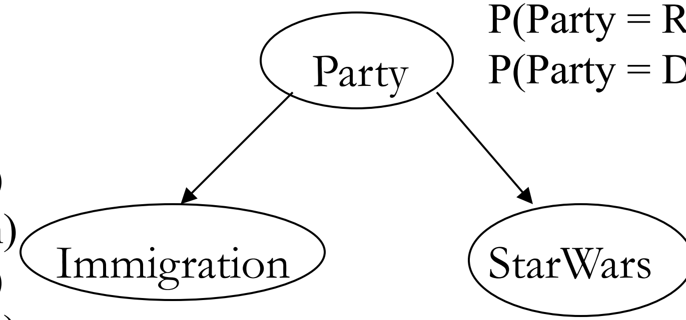
	Immigration Yes, StarWars Yes	Immigration Yes, StarWars No	Immigration No, StarWars Yes	Immigration No, StarWars No
Republican	9.5	7.5	117	92
Democrat	89	71	27	22

different – not independent

Douglas H. Fisher

Further tests might indicate

P(Party = Rep)
P(Party = Dem)



P(Immigration = y| Party = Rep)
P(Immigration = y| Party = Dem)
P(Immigration = n| Party = Rep)
P(Immigration = n| Party = Dem)

P(StarWars = y| Party = Rep)
P(StarWars = y| Party = Dem)
P(StarWars = n| Party = Rep)
P(StarWars = n| Party = Dem)

i.e., Immigration and StarWars are independent conditioned on Party

This process of building a BN from data is a form of *unsupervised* machine learning

In this particular example, the BN above can be viewed as supporting the naïve Bayesian classifier (for predicting Party)

Suppose given I=y and SW=n, predict Party

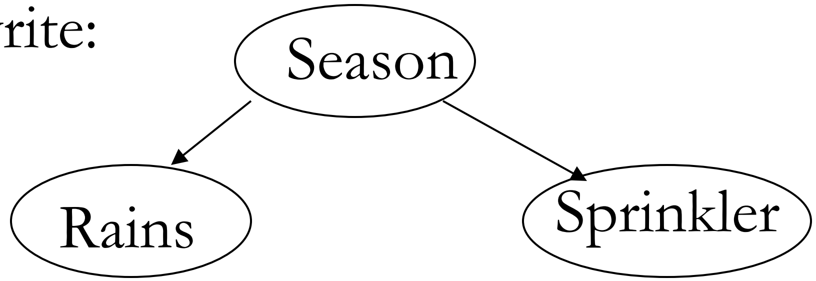
P(Party=Dem | I=y, SW=n)
= P(I=y, SW=n | Party=Dem)P(Dem)/P(I=y, SW=n)
α P(I=y, SW=n | Party=Dem)P(Dem)
= P(I=y | Party=Dem)P(SW=n | Party=Dem)P(Dem)

P(Party=Rep | I=y, SW=n)
= P(I=y, SW=n | Party=Rep)P(Rep)/P(I=y, SW=n)
α P(I=y, SW=n | Party=Rep)P(Rep)
= P(I=y | Party=Rep)P(SW=n | Party=Rep)P(Rep)

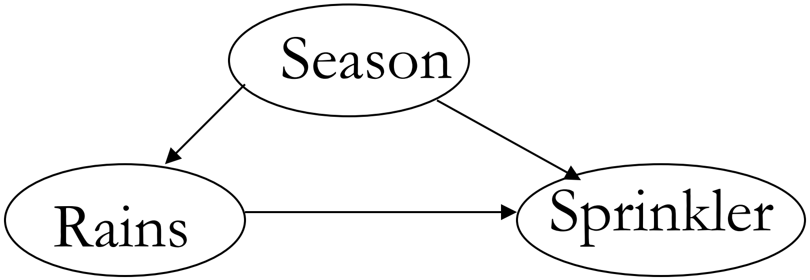
Where does knowledge of conditional independence come from?

b) "First principles"

For example, suppose that the grounds keeper sets sprinkler timers to a fixed schedule that depends on the season (Summer, Winter, Spring, Fall), and suppose that the probability that it rains or not is dependent on season. We might write:



This model might differ from one in which a homeowner manually turns on a sprinkler



More on building BNs from first principles

Consider CS courses in the Vanderbilt catalog

CS	4959
CS	1101
CS	1103
CS	1151
CS	2201
CS	2212
CS	2231
CS	3250
CS	3251
CS	3259
CS	3270
CS	3281
CS	3282
CS	4260
CS	4278
CS	4285
CS	4287
CS	2204
CS	3252
CS	3265
CS	3274
CS	4269
CS	4279
CS	4283
CS	4288
CS	3258
CS	3276
CS	4266

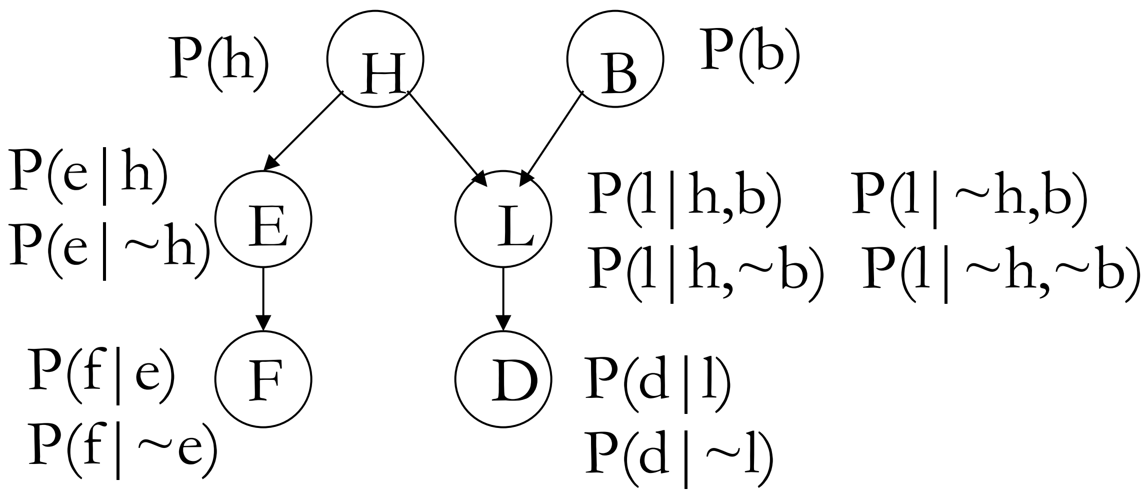
For the highlighted courses, construct a BN for predicting grades in query courses from known or assumed grades in evidence courses

Construct a network

Can you compute the probability of each grade (A,B,C,D,F) from one or more known or assumed grades in other courses?

In lecture, we barely started on inference with BNs – we will pick up here next lecture

Consider the following:



H: Hardware problems (h) or not (~h)
B: Bugs in code (b) or not (~b)
E: Editor running (e) or not (~e)
L: Lisp interpreter running (l) or not (~l)
F: Cursor flashing (f) or not (~f)
D: prompt displayed (d) or not (~d)