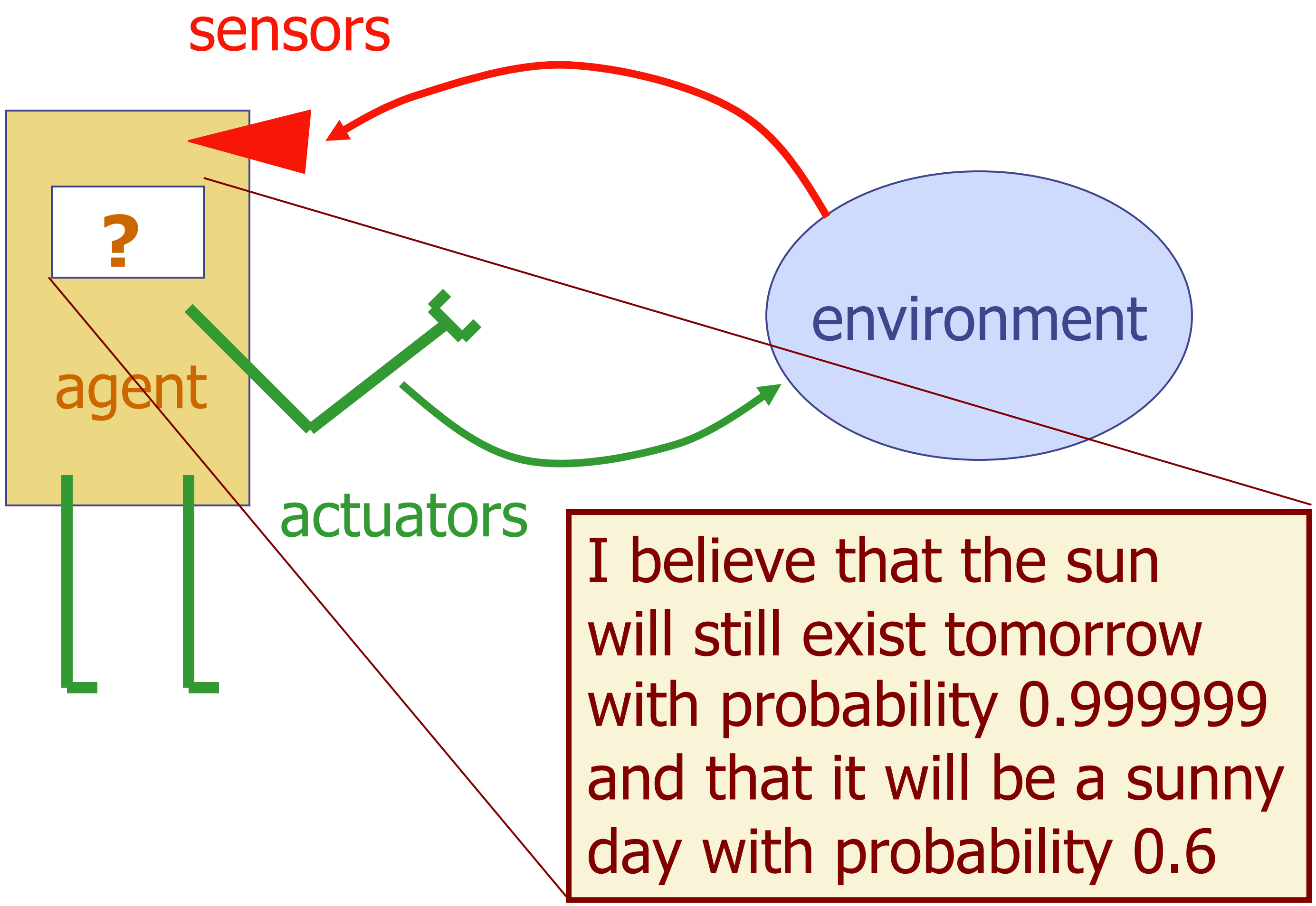
Bayesian Networks

Probabilistic Agent



Problem

- At a certain time t, the KB of an agent is some collection of beliefs
- At time t the agent's sensors make an

این observation خودش ممکن است تصادفی باشد

observation that changes the strength of one of its beliefs
- How should the agent update the strength of its other beliefs?

ممکن است blief ها با یک دیگر مرتبط
باشند (اطلاعات از یک اعتقاد موجب تغییر
اعتقاد به سایر blief ها می شود .
بنابراین agent باید ابزای داشته باشد که بتواند
اعتقادش رو به درست بودن سایر factها را upadat کند

Purpose of Bayesian Networks

- Facilitate the description of a collection of beliefs by making explicit causality relations and conditional independence among beliefs

یک سری پدیده احتمالاتی داریم که احتمال درست بودن یا نبودن آن را می دانیم
ولی فقط دانستن این احتمالات کافی نمی باشد زیرا ممکن است factها به یک دیگر مرتبط باشند

- Provide a more efficient way (than by using joint distribution tables) to update belief strengths when new evidence is observed

اطلاعات تصادفی را به صورت compact ذخیره بکنیم داخل حافظه
و هم خیلی efficient بتوانیم بر روی آن محاسبه انجام دهیم .

اگر بخواهیم تمام اطلاعات آماری داده های تصادفی را نگه داریم. :
باید یک توزیع احتمال توام از داده ها داشته باشیم
پس باید 2^n مقدار باید نگه داریم (چون باید جدول توزیع احتمال توام تمام داده ها باهم را داشته باشیم)

Other Names

این bayesian network ها گراف جهت دار هستند و بدون دور
یک DAG است .

- Belief networks
- Probabilistic networks
- Causal networks

در این DAG به ازای هر رویداد تصادفی یک node داریم
و یک یال از a به b می بریم وقتی a دلیل مستقیم b باشد
یعنی پدیده a به طور مستقیم تاثیر در پدیده b داشته باشد .
یعنی تغییری باشد که از طریق بقیه پدیده ها قابل بازنمایی نباشد

هر nodeیک سری parent دارد
توزیع احتمالاتی هر node ای به شرط مشاهده parentها رابايد در شبکه داشته باشیم
باید توزیع احتمال شرطی هر nodeای نسبت به parentهایش را باید داشته باشیم
به جای این که تابع توزیع احتمال توام همه متغیرها را داشته باشیم ، توزیع احتمال شرطی
هر node به شرط دیده شدن parent آن را نگه میداریم
واضح است اگر گراف ، گراف تنکی باشد این representation از قبلی خیلی بهتر است .

Bayesian Networks

- A simple, graphical notation for conditional independence assertions resulting in a compact representation for the full joint distribution

- Syntax:
  - a set of nodes, one per variable
  - a directed, acyclic graph (link = 'direct influences')
  - a conditional distribution (CPT) for each node given its parents:

$$P(X_i|Parents(X_i))$$

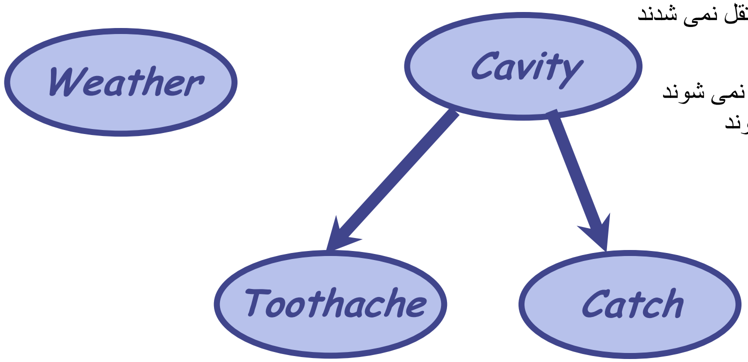
Example

مثال ۱

Topology of network encodes conditional independence assertions:

اگر cavity را داشته باشیم catch از toothache مستقل می شود

اگر جهت فلش ها بر عکس بود این دو از هم مستقل نمی شدند



دو تا cause اگر یک effect ببینیم از یک دیگر مستقل نمی شوند
با دیدن cause ها effect ها از یک دیگر مستقل می شوند

Weather is independent of other variables
Toothache and Catch are independent given Cavity

Example

مثال ۲

فعلا می خواهیم شبکه را طوری به وجود آوریم که cause ها را بالا بگذاریم و یال وصل کنیم به effectها

I'm at work, neighbor John calls to say my alarm is ringing, but neighbor Mary doesn't call. Sometime it's set off by a minor earthquake. Is there a burglar?

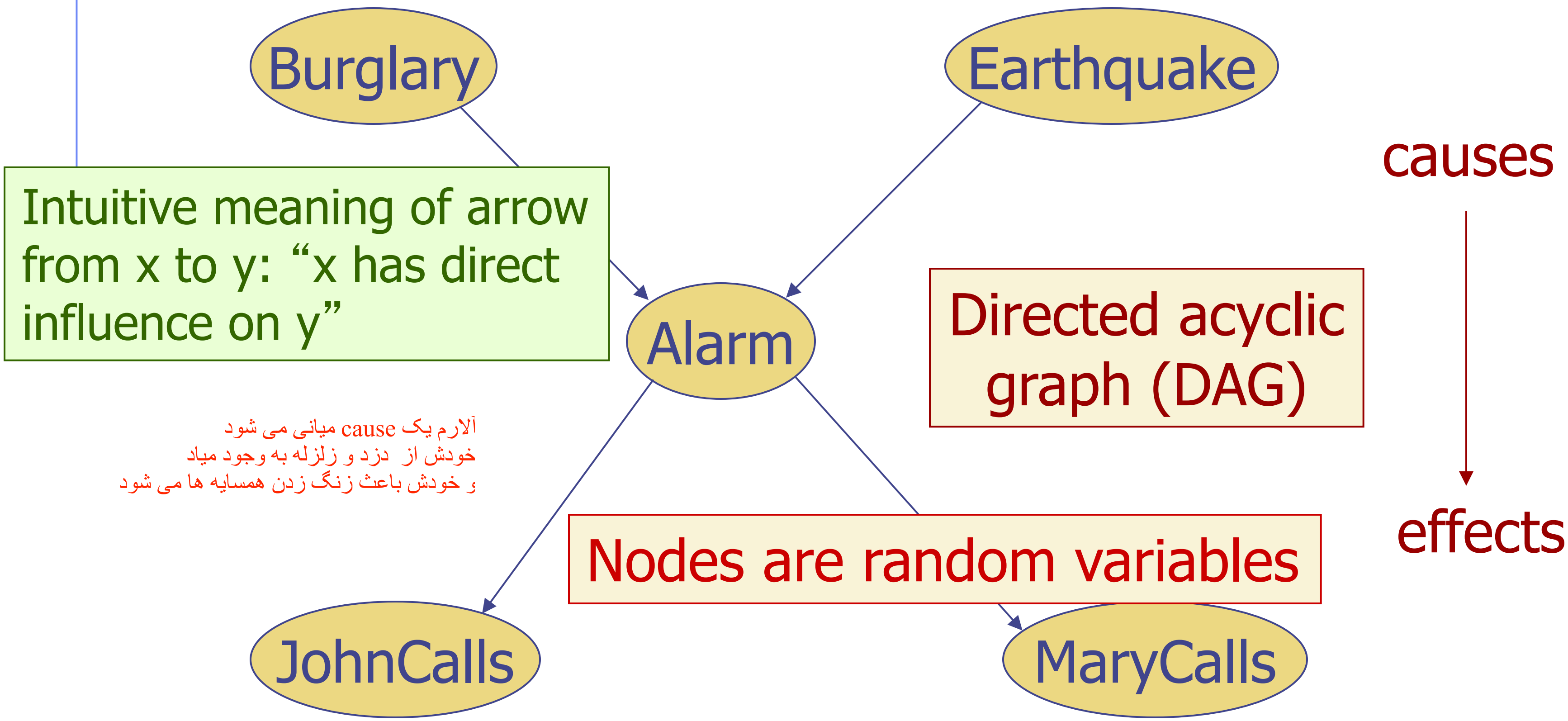
Variables: Burglar, Earthquake, Alarm, JohnCalls, MaryCalls

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

A Simple Belief Network

این که شخص سرکار باشد یا نباشد هم می تواند یک پدیده احتمالاتی باشد
ولی برای سادگی در نظر نمی گیریم

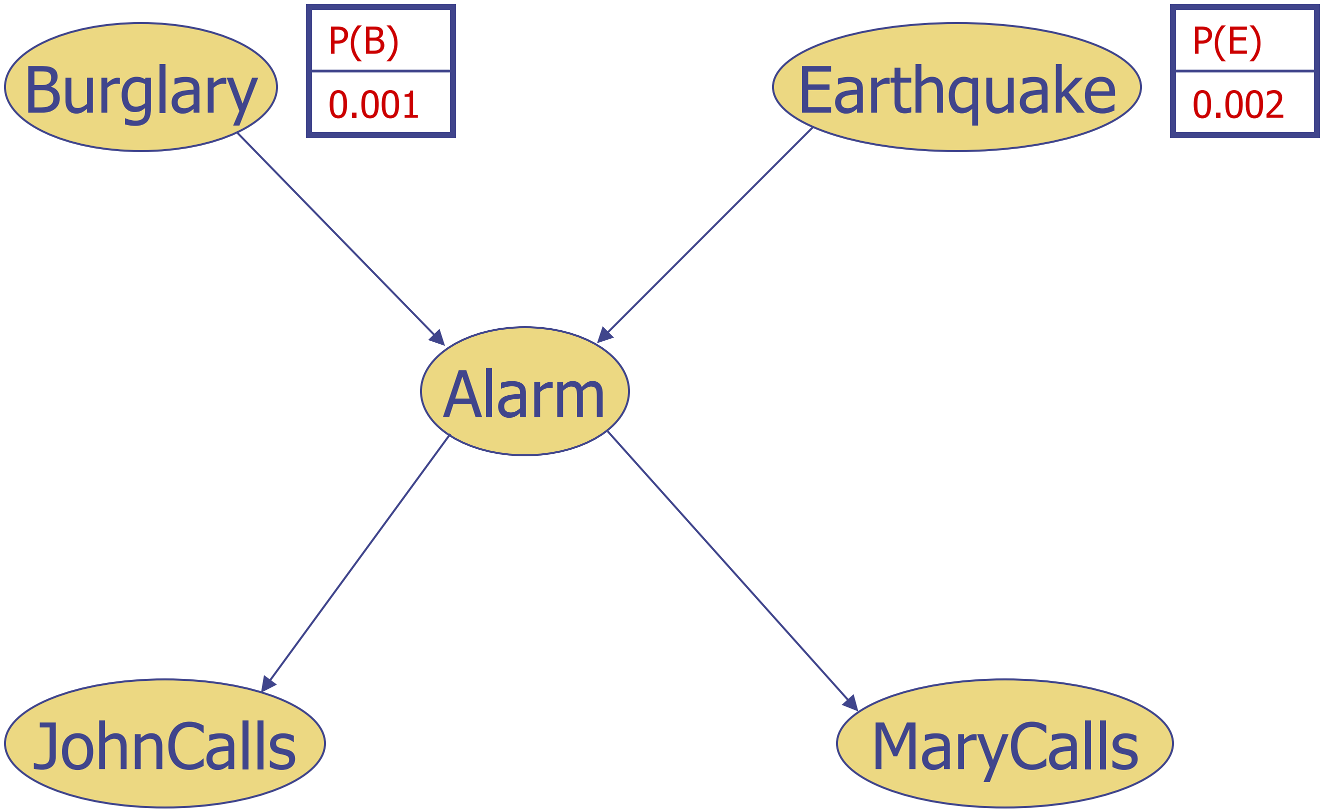
دزد و زلزله هر دو cause اصلی می باشند
می توانند زنجیره ای را به وجود آورند که باعث به صدا درآمدن آلارم و اتفاقات بعدی شود



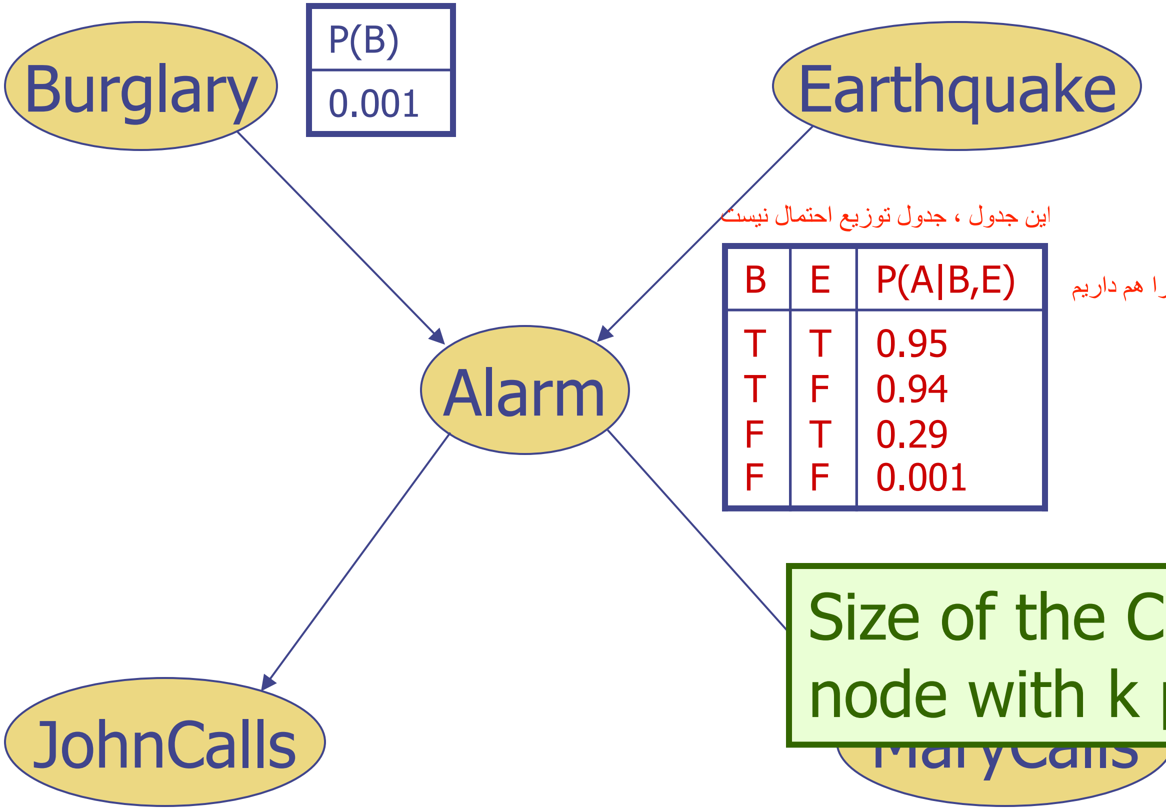
آلارم یک cause میانی می شود
خودش از دزد و زلزله به وجود میاد
و خودش باعث زنگ زدن همسایه ها می شود

اگر john و merry هر دوبا هم زندگی کنند می توانند یک رابطه causal با یک دیگر داشته باشند
بین زنگ زدن john و دزد هم می تواند رابطه causal داشته باشد مثلا زمانی که john خودش دزد باشد 😂

Assigning Probabilities to Roots



Conditional Probability Tables



P(B)
0.001

P(E)
0.002

این جدول ، جدول توزیع احتمال نیست

B	E	P(A|B,E)
T	T	0.95
T	F	0.94
F	T	0.29
F	F	0.001

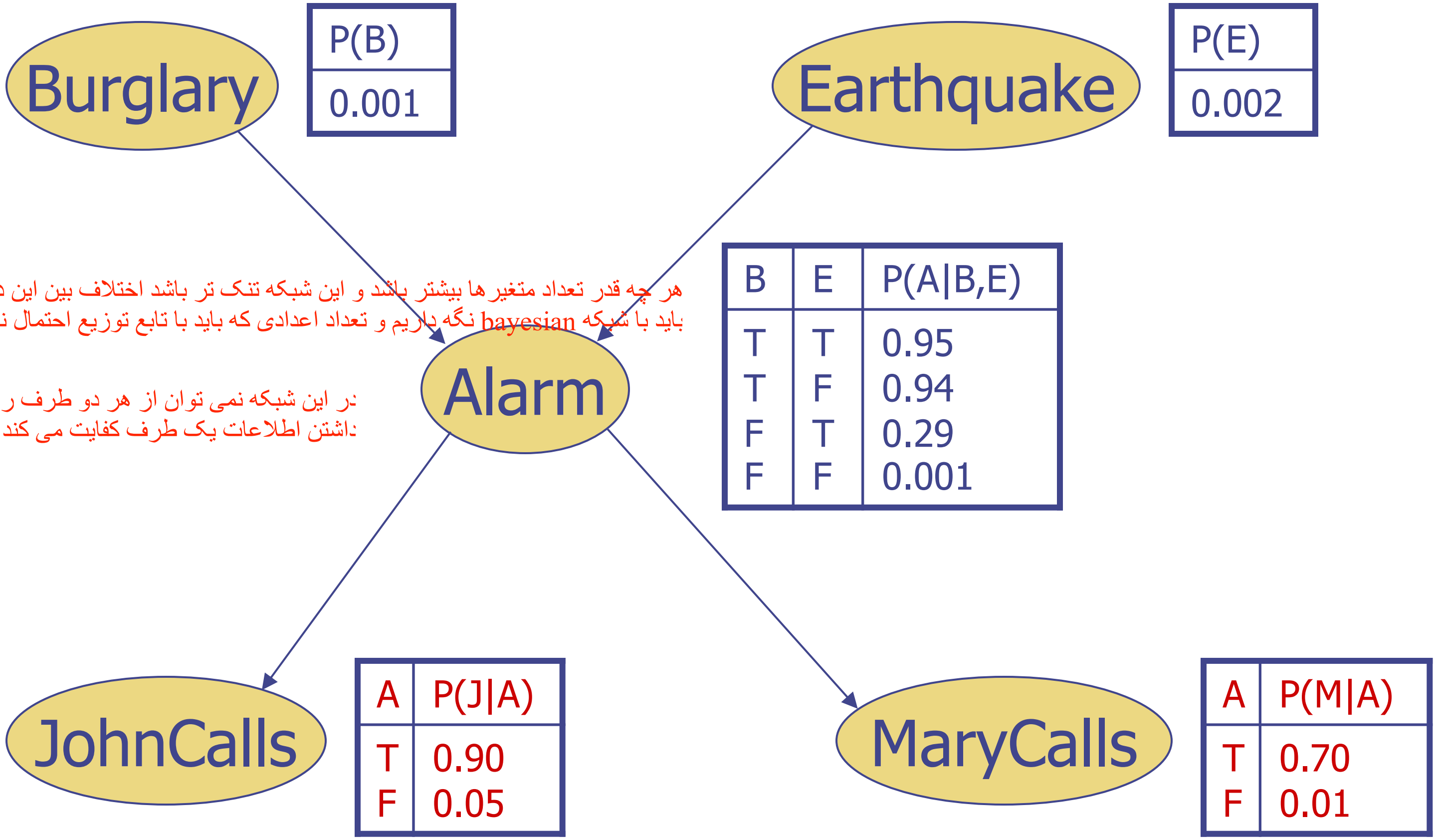
همچنین از این جدول احتمال false بودن را هم داریم

Size of the CPT for a node with k parents: ?

Conditional Probability Tables

هر چه قدر تعداد متغیر ها بیشتر باشد و این شبکه تنک تر باشد اختلاف بین این دو عدد (تعداد اعدادی که
باید با شبکه bayesian نگه داریم و تعداد اعدادی که باید با تابع توزیع احتمال نگه داریم) بیشتر می شود .

در این شبکه نمی توان از هر دو طرف راجب طرف دیگر اطلاعات گرفت
داشتن اطلاعات یک طرف کفایت می کند



P(B)
0.001

P(E)
0.002

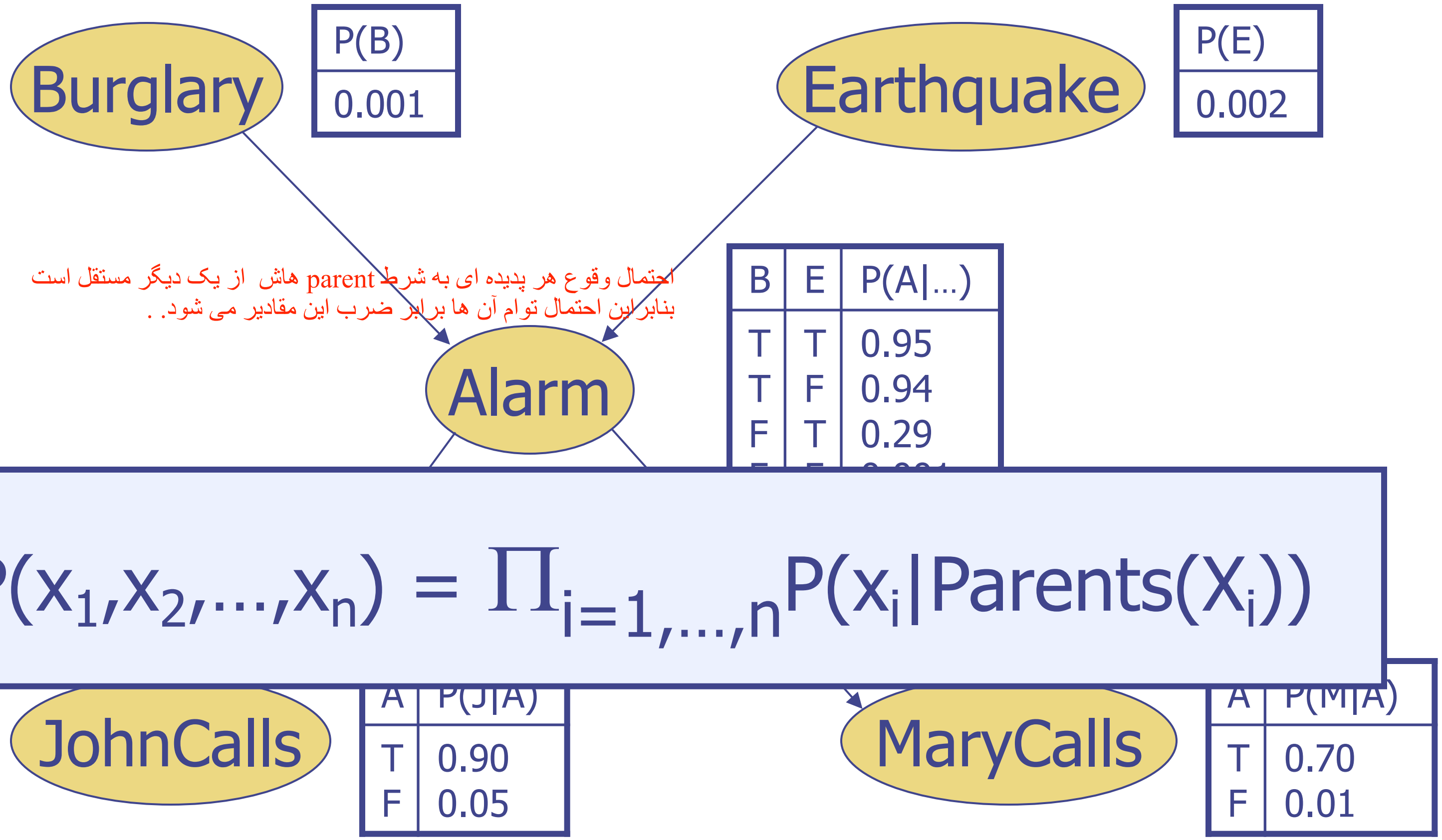
B	E	P(A|B,E)
T	T	0.95
T	F	0.94
F	T	0.29
F	F	0.001

A	P(J|A)
T	0.90
F	0.05

A	P(M|A)
T	0.70
F	0.01

What the BN Means

احتمال وقوع هر پدیده ای به شرط parent هاش از یک دیگر مستقل است
بنابراین احتمال توام آن ها برابر ضرب این مقادیر می شود . .



P(B)
0.001

P(E)
0.002

B	E	P(A|…)
T	T	0.95
T	F	0.94
F	T	0.29
F	F	0.001

A	P(J|A)
T	0.90
F	0.05

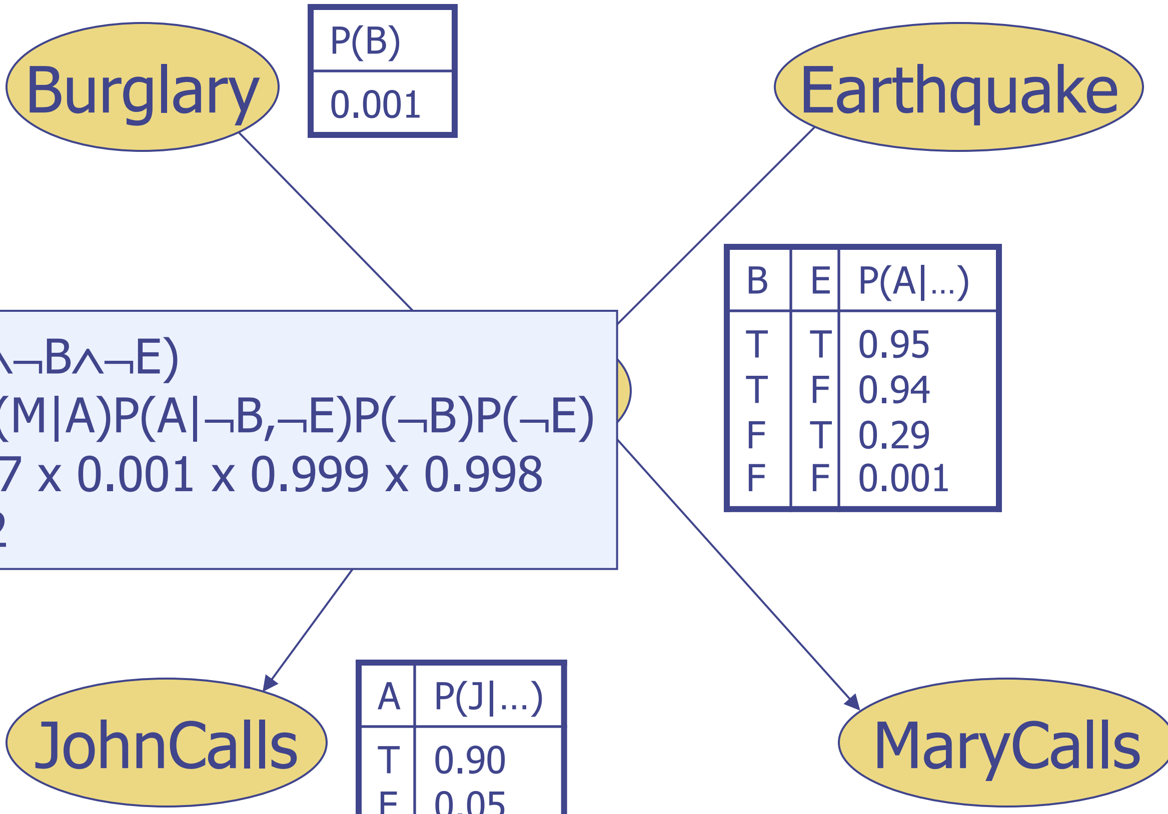
A	P(M|A)
T	0.70
F	0.01

$$P(x_1,x_2,\ldots,x_n) = \Pi_{i=1,\ldots,n} P(x_i|Parents(X_i))$$

P(john, merry,b,alarm) = p(j|m,a,b,e)*p(m,a,b,e)

Calculation of Joint Probability

یک node به شرط parentهاش از بچه های خودش هم مستقل می شود. ؟
چون گراف ما DAG است بنابراین یک پایینی دارد که ما از آن جا شروع می کنیم
جداکردن احتمال ها را از پایین به بالا انجام می دهیم



P(B)
0.001

P(E)
0.002

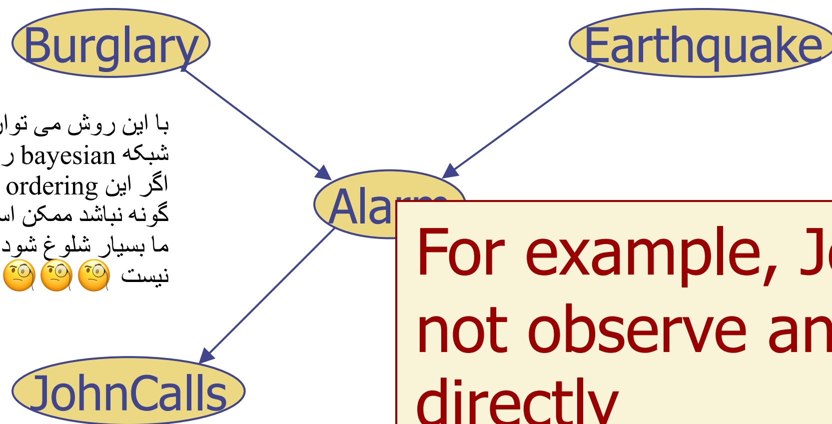
B	E	P(A|…)
T	T	0.95
T	F	0.94
F	T	0.29
F	F	0.001

$$
\begin{aligned}
&P(J \wedge M \wedge A \wedge \neg B \wedge \neg E) \\
&= P(J|A)P(M|A)P(A|\neg B,\neg E)P(\neg B)P(\neg E) \\
&= 0.9 \times 0.7 \times 0.001 \times 0.999 \times 0.998 \\
&= 0.00062
\end{aligned}
$$

A	P(J|…)
T	0.90
F	0.05

A	P(M|…)
T	0.70
F	0.01

What The BN Encodes



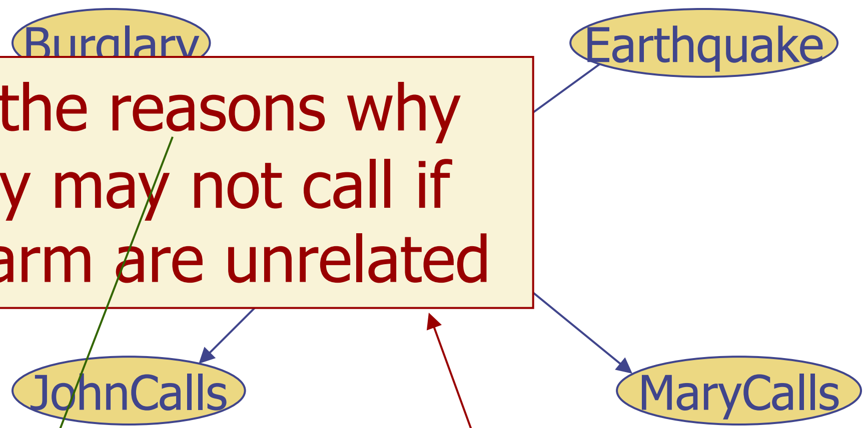
با این روش می توان از روی جدول توزیع احتمال توام هم می توان شبکه bayesian را ساخت
اگر این ordering که باید به ترتیب cause به effect باشد ،. این گونه نباشد ممکن است شبکه ما بسیار شلوغ شود تا حدی عامل انسانی هم قادر به تشخیص آن نیست 🧐🧐🧐

For example, John does not observe any burglaries directly

- Each of the beliefs JohnCalls and MaryCalls is independent of Burglary and Earthquake given Alarm or ¬Alarm
- The beliefs JohnCalls and MaryCalls are independent given Alarm or ¬Alarm

ممکن است اطلاعات causal محیط را نداشته باشیم
یعنی بتوانیم توزیع احتمال را از روی یک سری داده آماری بدست بیاوریم و بعد بتوانیم قضاوت کنیم که شبکه bayesianرا چه طور بکشیم
اگر این شرایط باشد : الان باید از خاصیت دوم (هر کسی به شرط parentهاش در شبکه مستقل باشد)استفاده کرد
بنابراین برای اینکه یک شبکه bayesian بسازیم یک ترتیب بین node ها به دست می آوریم (با در نظر گرفتن تمام اتفاقات تصادفی) بعد node ها را یکی یکی اضافه می کنیم و می بینیم که node اضافه شده چه parentای باشد داشته باشد تا از بقیه متغیر ها مستقل شود ؟

What The BN Encodes



For instance, the reasons why John and Mary may not call if there is an alarm are unrelated

- Each of the beliefs

  or ¬Alarm

- The beliefs JohnCalls and MaryCalls are independent given Alarm or ¬Alarm

Note that these reasons could be other beliefs in the network. The probabilities summarize these non-explicit beliefs

Structure of BN

- The relation:

E.g., JohnCalls is influenced by Burglary, but not directly. JohnCalls is directly influenced by Alarm

m

predecessors in the BN given its parents

- Said otherwise, the parents of a belief X_i are all the beliefs that "directly influence" X_i
- Usually (but not always) the parents of X_i are its causes and X_i is the effect of these causes

Construction of BN

- Choose the relevant sentences (random variables) that describe the domain
- Sel
  bel X_i
- The ordering guarantees that the BN will have no cycles
- For j=1,…,n do:
  - Add a node in the network labeled by X_j
  - Connect the node of its parents to X_j
  - Define the CPT of X_j

Example

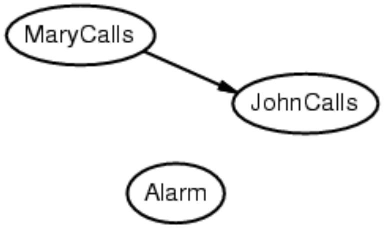
در این مثال ترتیب را به صورت برعکس در نظر می گیریم یعنی از effect ها به cause ها

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

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

Example

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

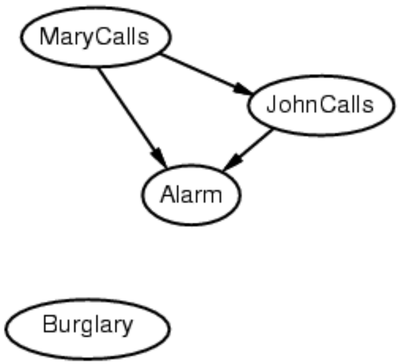


$\mathbf{P}(J \mid M) = \mathbf{P}(J)$? **No**

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

Example

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



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

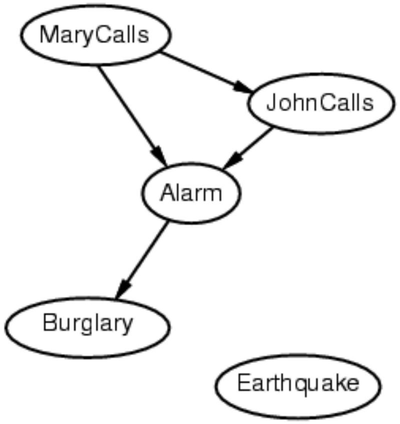
$\mathbf{P}(A \mid J, M) = \mathbf{P}(A \mid J)$? $\mathbf{P}(A \mid J, M) = \mathbf{P}(A)$? **No**

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

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

Example

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



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

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

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

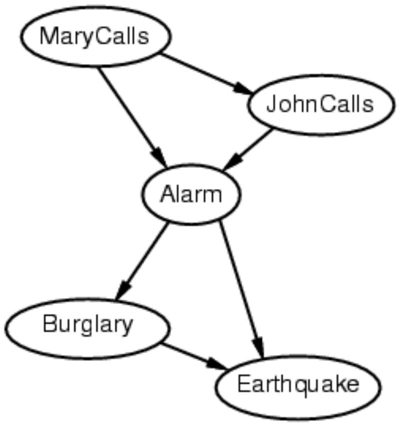
$\mathbf{P}(B \mid A, J, M) = \mathbf{P}(B)$? **No**

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

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

Example

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



همیشه node های پایینی را چک می کنیم که آیا مستقل می شود یا نه ؟

آیا میشه فقط به آلارم مربوط کرد یا به هر دوی.
اگر الارم را مشاهده کنیم و بدانیم دزد نبوده پس احتمال زلزله بیشتر می شود
پس زلزله به دزدی مربوط است .

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

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

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

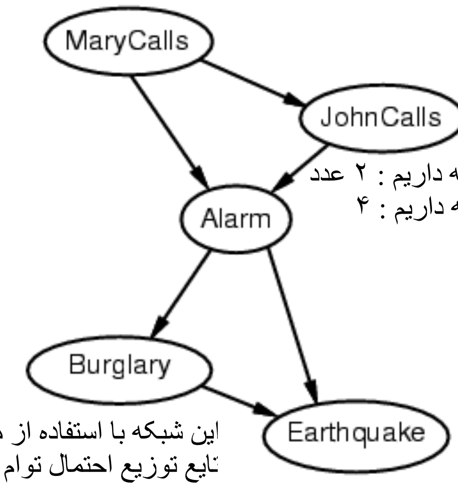
$\mathbf{P}(B \mid A, J, M) = \mathbf{P}(B)$? **No**

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

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

Example summary

وقتی داریم شبکه را می سازیم node ها را
که هر کدام نماینده یک پدیده تصادفی هستند را می گذاریم
و به هر node جدید باید یال هایی از node های قبلی بکشیم
یال ها را طوری اضافه می کنیم که به شرط مشاهده parentها یک node از بقیه گره ها
مستقل شوند
یعنی فقط parent ها درباره احتمال پدیده اطلاعات بدهند

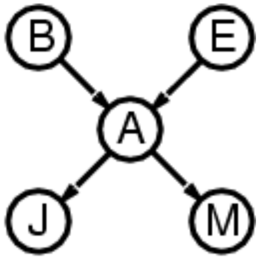


توزیع احتمال johncalls را به ازای تمام حالت های merrycalls باید نگه داریم : ۲ عدد
توزیع احتمال alartm به شرط مشاهده تمام حالت های parentهایش را نگه داریم : ۴

این شبکه با استفاده از مفهوم استقلال شرطی یک representation خلاصه تری از
تابع توزیع احتمال توام می دهد

- Deciding conditional independence is hard in noncausal directions
- (Causal models and conditional independence seem hardwired for humans!)
- Network is less compact: 1 + 2 + 4 + 2 + 4 = 13 numbers needed

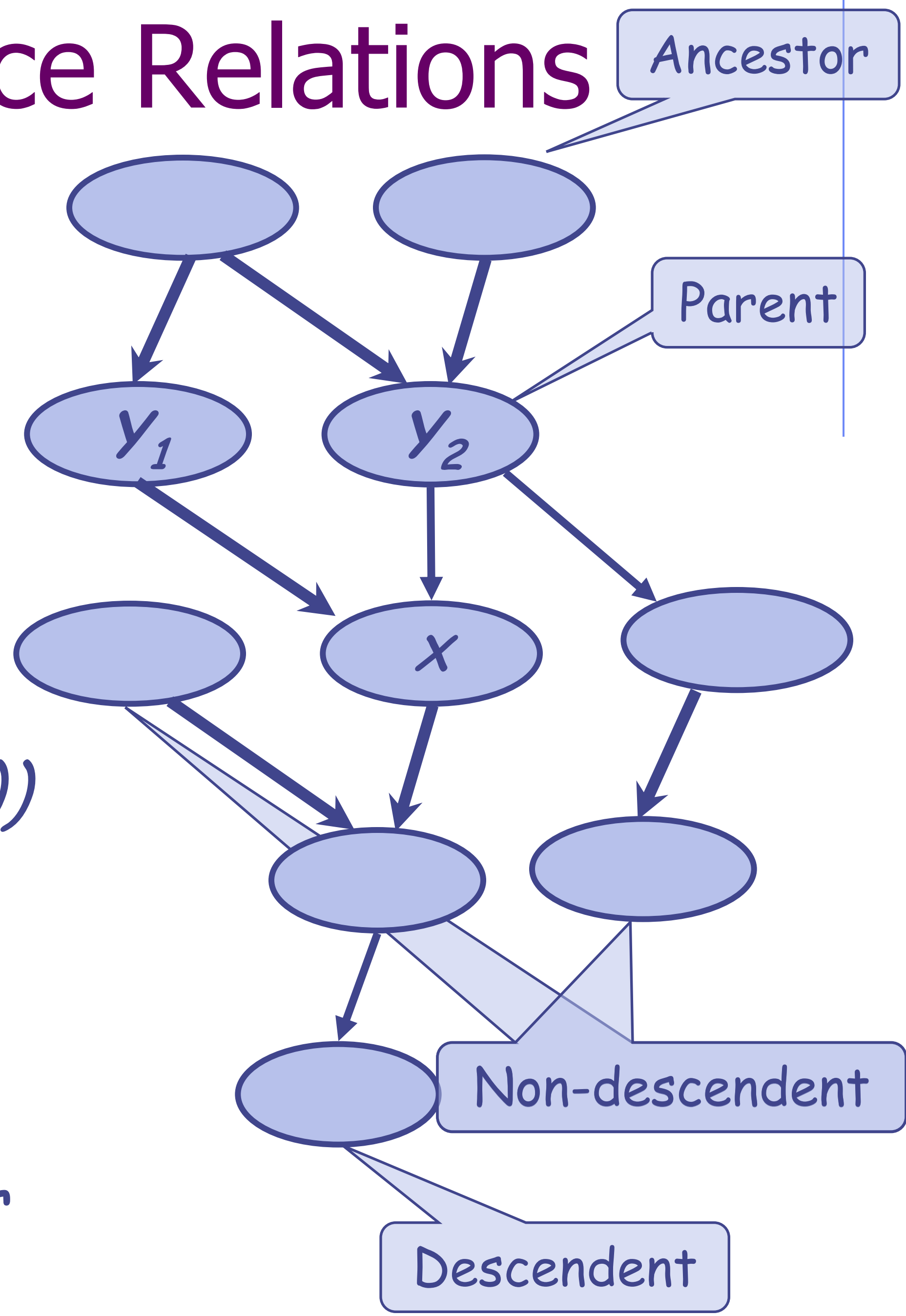
Compactness



- A CPT for Boolean *Xi* with *k* Boolean parents has *2k* rows for the combinations of parent values
- Each row requires one number *p* for *Xi = true*
(the number for *Xi = false* is just *1-p*)
- If each variable has no more than *k* parents, the complete network requires $O(n \cdot 2k)$ numbers
- I.e., grows linearly with *n*, vs. $O(2n)$ for the full joint distribution
- For burglary net, 1 + 1 + 4 + 2 + 2 = 10 numbers (vs. 25-1 = 31)

Cond. Independence Relations

- 1. Each random variable *X*, is conditionally independent of its non-descendents, given its parents *Pa(X)*
- Formally, *I(X; NonDesc(X) | Pa(X))*
- 2. Each random variable is conditionally independent of all the other nodes in the graph, given its neighbor



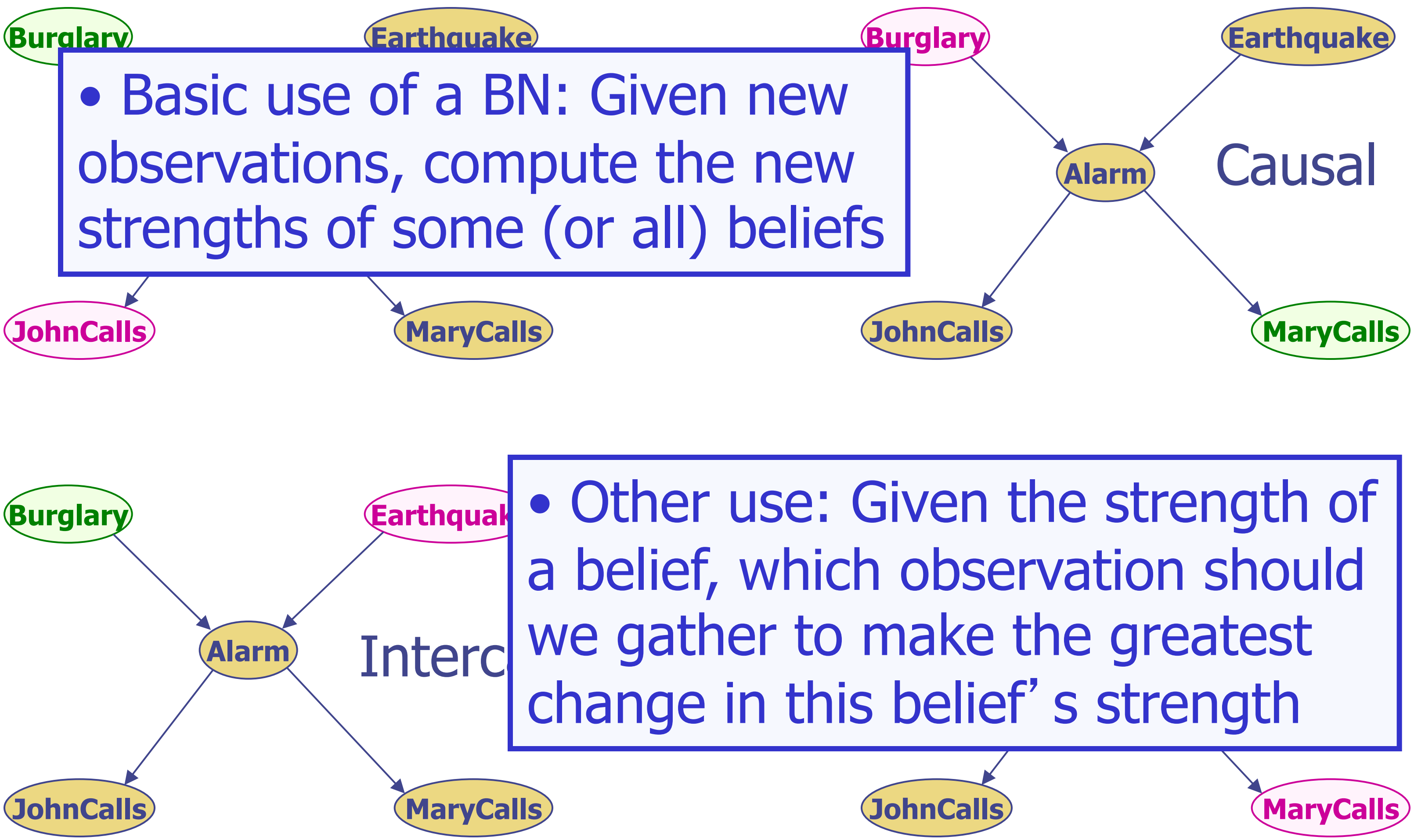
Inference in BNs

- Set E of evidence variables that are observed, e.g., {JohnCalls,MaryCalls}
- Query variable X, e.g., Burglary, for which we would like to know the posterior probability distribution P(X|E)

J	M	P(B|…)
T	T	?

Distribution conditional to the observations made

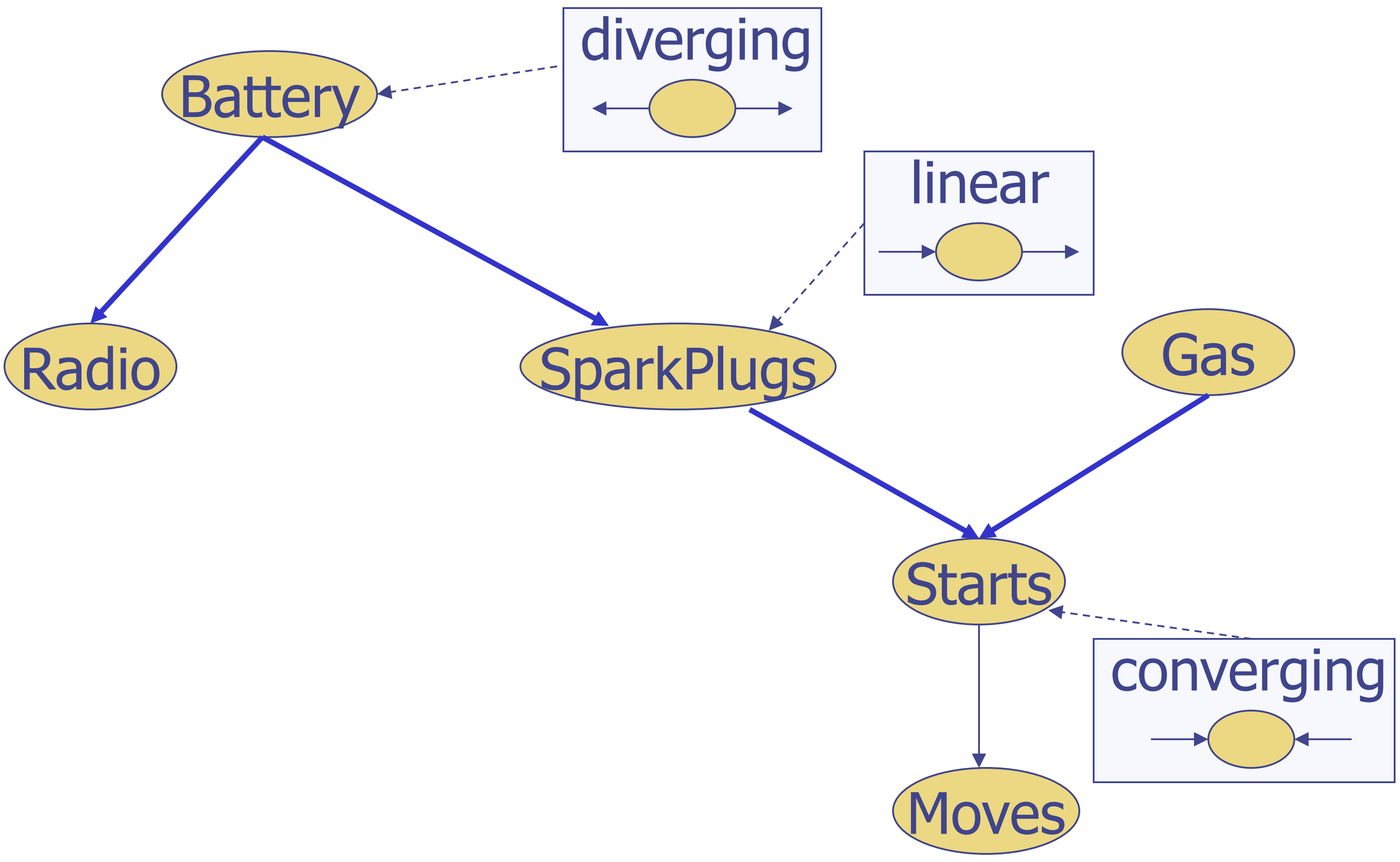
Inference Patterns



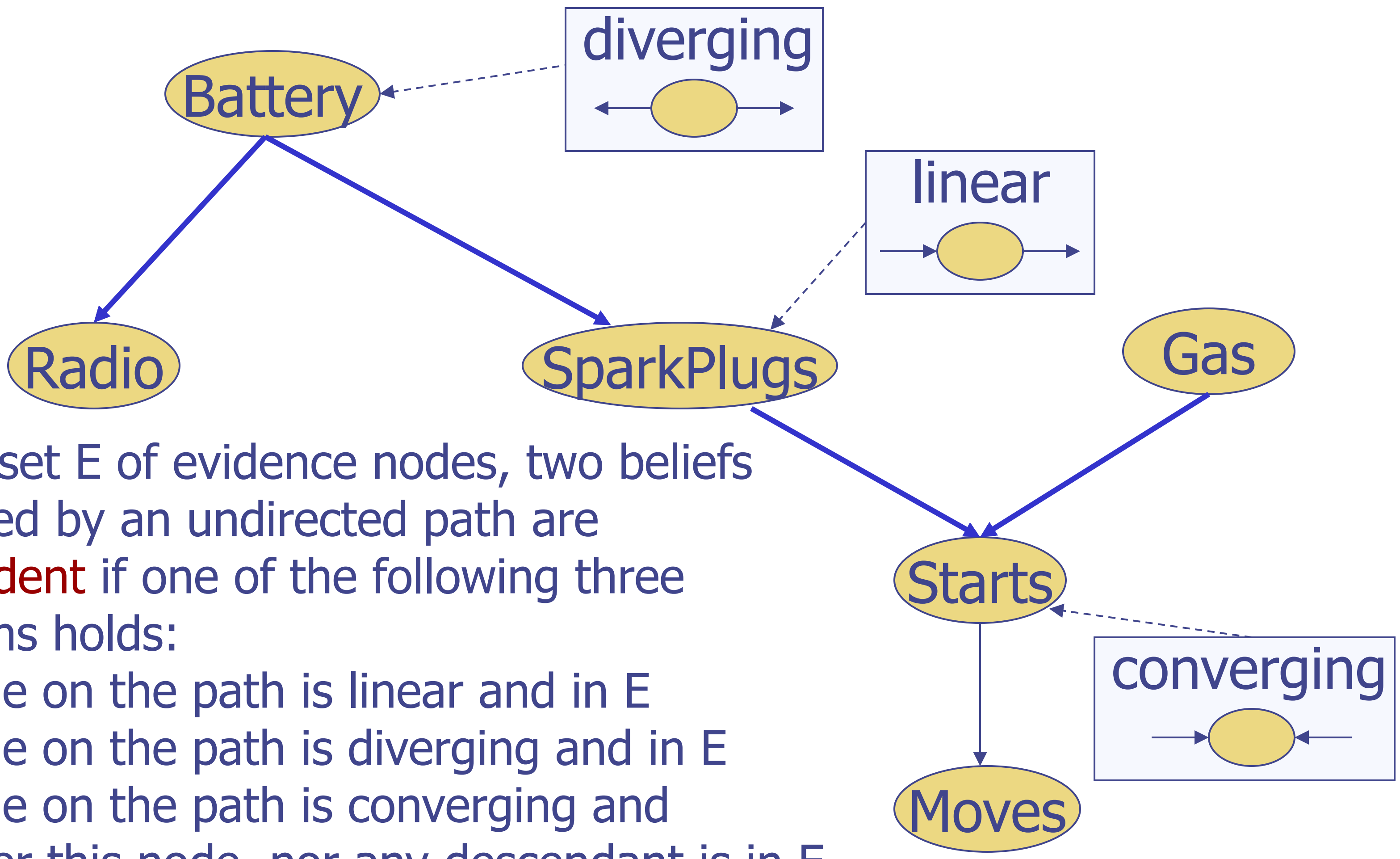
- Basic use of a BN: Given new observations, compute the new strengths of some (or all) beliefs

- Other use: Given the strength of a belief, which observation should we gather to make the greatest change in this belief's strength

Types Of Nodes On A Path



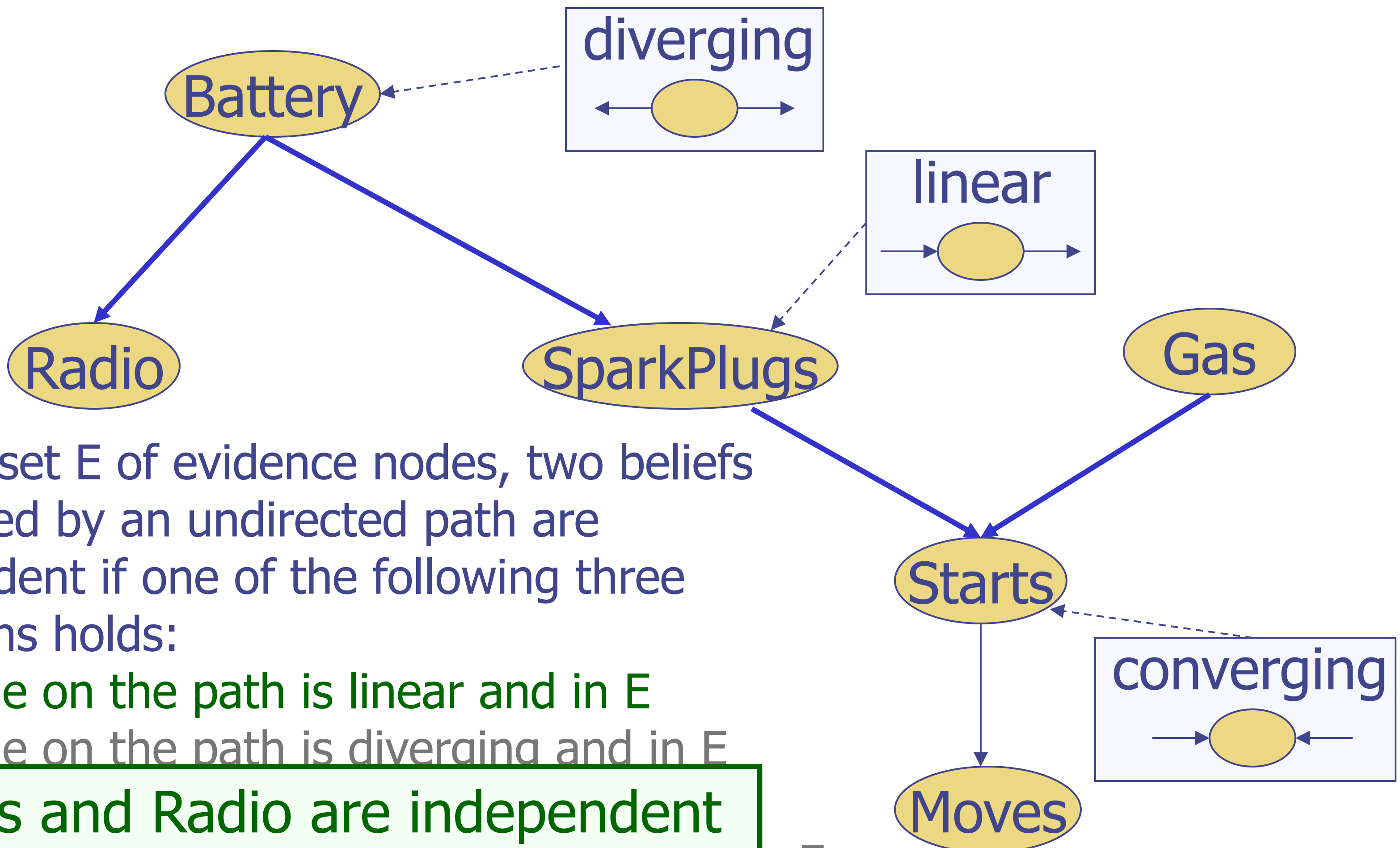
Independence Relations In BN



Given a set E of evidence nodes, two beliefs connected by an undirected path are independent if one of the following three conditions holds:

1. A node on the path is linear and in E
2. A node on the path is diverging and in E
3. A node on the path is converging and neither this node, nor any descendant is in E

Independence Relations In BN



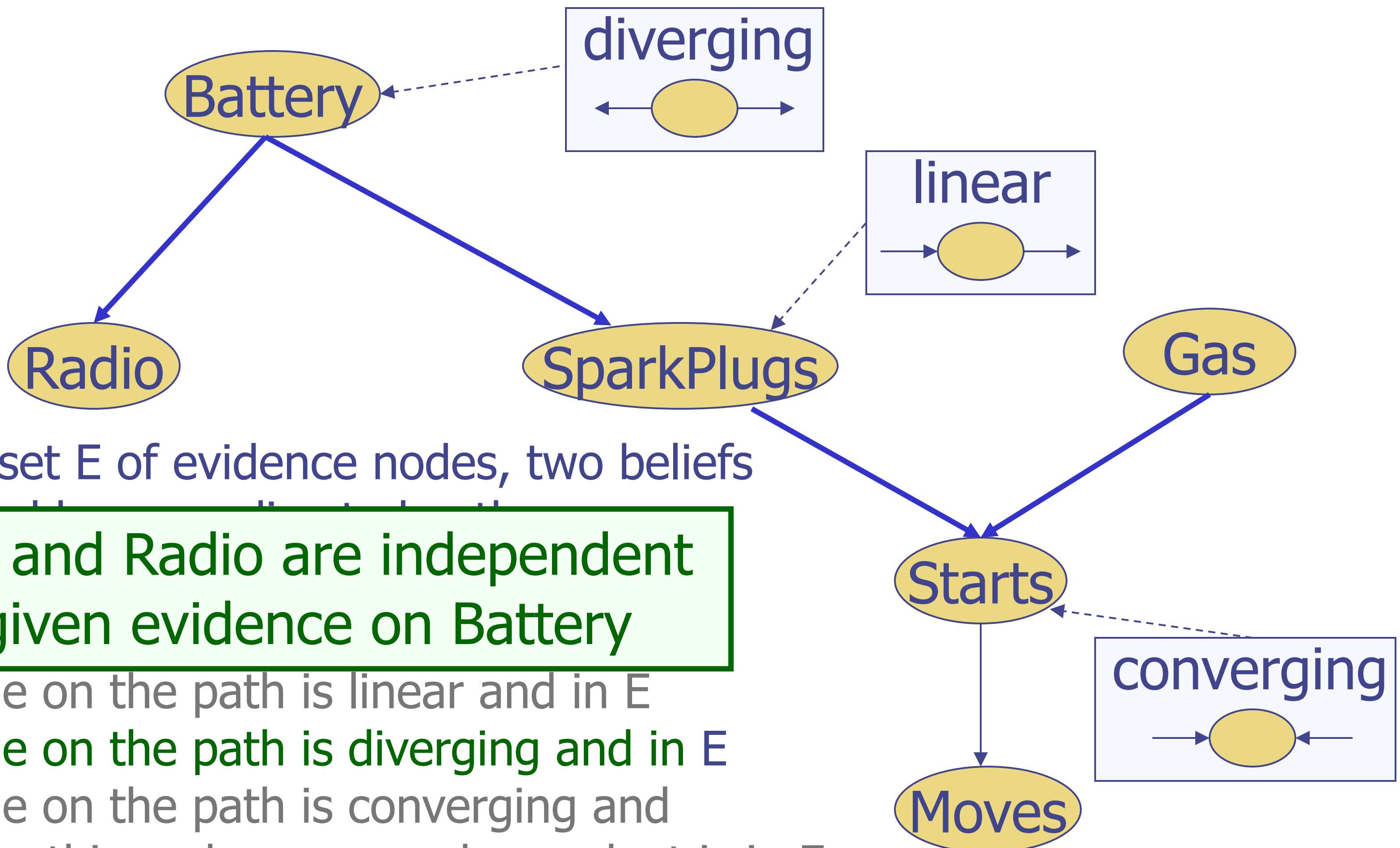
Given a set E of evidence nodes, two beliefs connected by an undirected path are independent if one of the following three conditions holds:

1. A node on the path is linear and in E
2. A node on the path is diverging and in E
3. A

n n E

Gas and Radio are independent given evidence on SparkPlugs

Independence Relations In BN

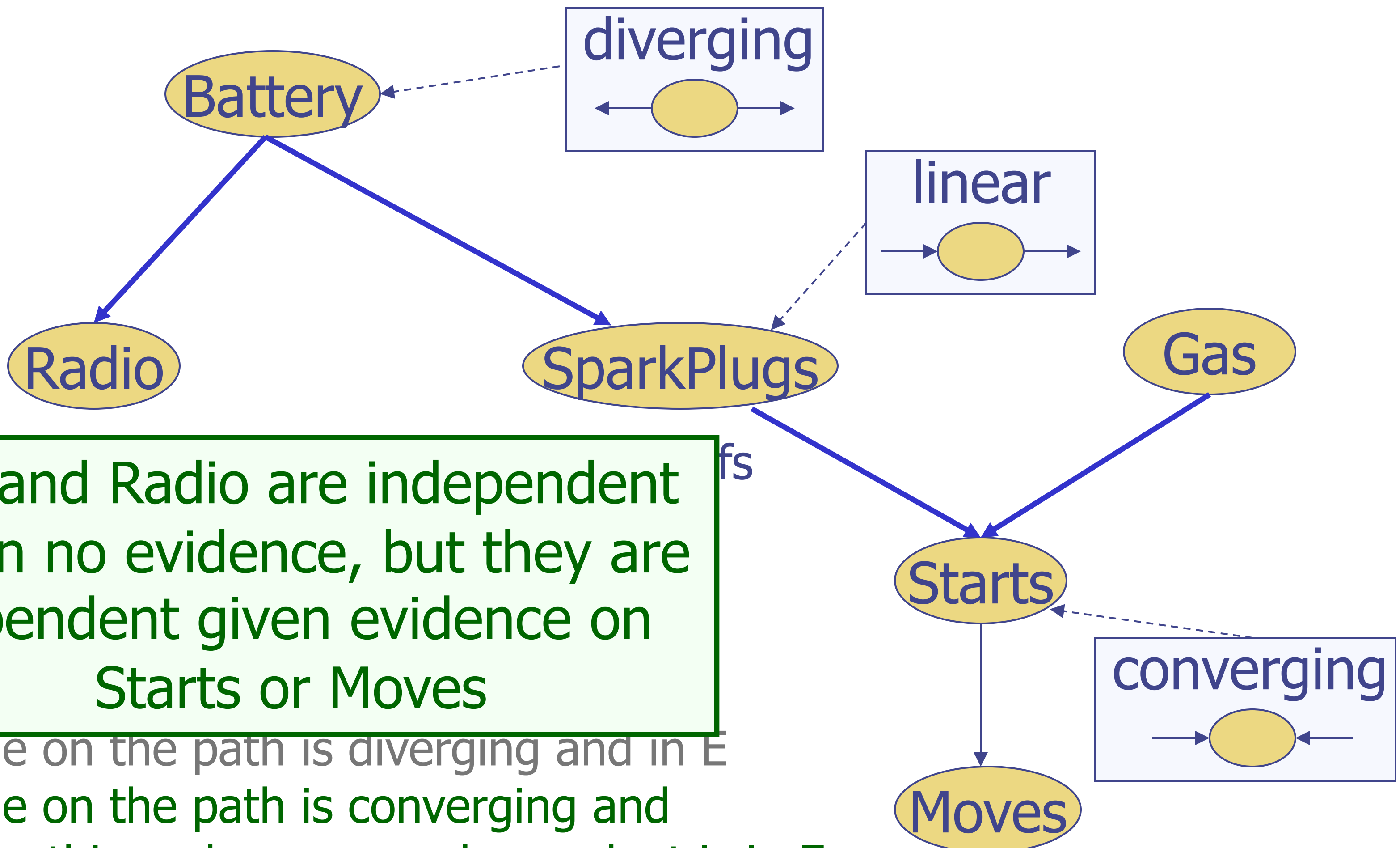


Given a set E of evidence nodes, two beliefs
co
inc
co

Gas and Radio are independent given evidence on Battery

1. A node on the path is linear and in E
2. A node on the path is diverging and in E
3. A node on the path is converging and neither this node, nor any descendant is in E

Independence Relations In BN



Gi ... fs
co
in
co
1.

Gas and Radio are independent given no evidence, but they are dependent given evidence on Starts or Moves

2. A node on the path is diverging and in E
3. A node on the path is converging and neither this node, nor any descendant is in E

BN Inference

- Simplest Case:

$$A \rightarrow B$$

P(B) = P(a)P(B|a) + P(~a)P(B|~a)

$$P(B) = \sum_{A} P(A)P(B \mid A)$$

$$A \rightarrow B \rightarrow C$$

P(C) = ???

p(c) = SIGMA on (a,b) (p(a) * p(b|a) * p(c|b))

BN Inference

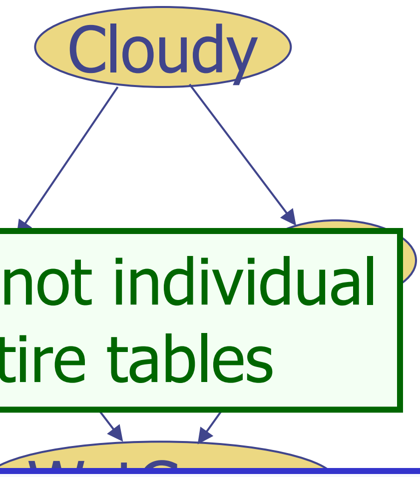
- Chain:

p(Xn) = SIGMA on (x1 ,…,Xn-1)(p(x1)*p(x2|x1)*p(x3|x2)*…..*p(xn|xn-1))

What is time complexity to compute $P(X_n)$?

What is time complexity if we computed the full joint?

Inference Ex. 2



میتوان فاکتور گیری کرد
in powerpoint

Algorithm is computing not individual probabilities, but entire tables

- Two ideas crucial to avoiding exponential blowup:
  - because of the structure of the BN, some subexpression in the joint depends only on a small number of variables
  - By computing them once and caching the result, we can avoid generating them exponentially many times

$$= \sum_{R,S} P(w \mid r,s) f_C(r,s) \qquad f_C(R,S)$$

Variable Elimination

General idea:

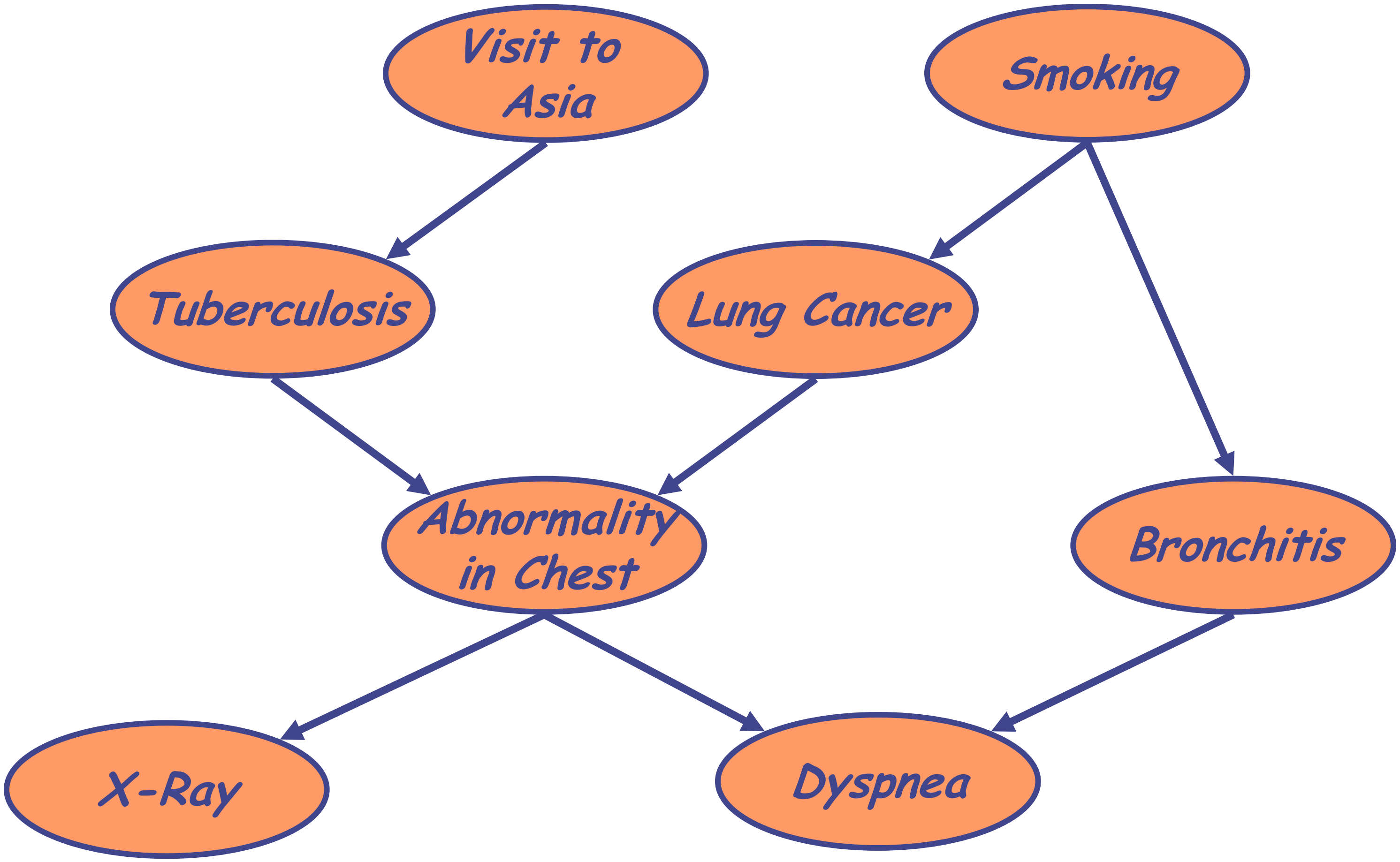
- Write query in the form

$$P(X_n, \mathbf{e}) = \sum_{x_k} \cdots \sum_{x_3} \sum_{x_2} \prod_i P(x_i \mid pa_i)$$

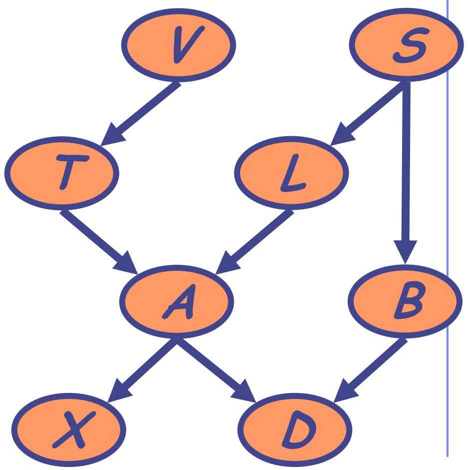
- Iteratively
  - Move all irrelevant terms outside of innermost sum
  - Perform innermost sum, getting a new term
  - Insert the new term into the product

A More Complex Example

- "Asia" network:



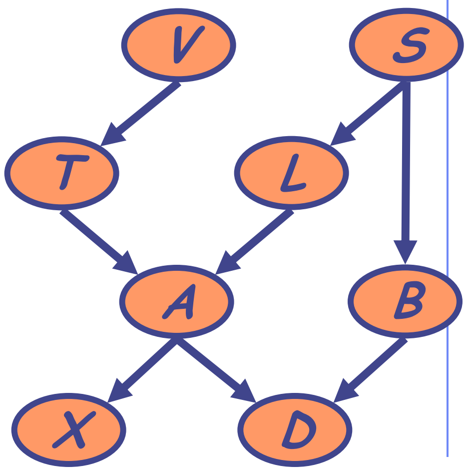
- We want to compute $P(d)$
- Need to eliminate: v,s,x,t,l,a,b



Initial factors

$$P(v)P(s)P(t|v)P(l|s)P(b|s)P(a|t,l)P(x|a)P(d|a,b)$$

- We want to compute *P(d)*
- Need to eliminate: *v,s,x,t,l,a,b*



Initial factors

$$P(v)P(s)P(t \mid v)P(l \mid s)P(b \mid s)P(a \mid t,l)P(x \mid a)P(d \mid a,b)$$

بعد از این فاکتور گیری تابع حاصل تابعی بر حسب t
می باشد زیرا در این مرحله فقط بر روی v جمع زدیم و تصمیمی
برای t نگرفته ایم

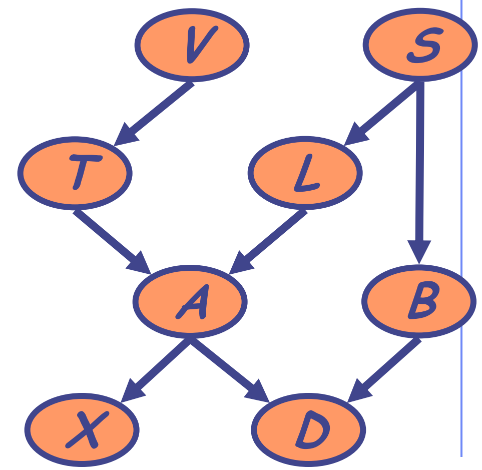
Eliminate: v

Compute: $$f_v(t) = \sum_v P(v)P(t \mid v)$$

$$\Rightarrow f_v(t)P(s)P(l \mid s)P(b \mid s)P(a \mid t,l)P(x \mid a)P(d \mid a,b)$$

Note: $f_v(t) = P(t)$

In general, result of elimination is not necessarily a probability term



- We want to compute $P(d)$
- Need to eliminate: s,x,t,l,a,b

- Initial factors

$$P(v)P(s)P(t \mid v)P(l \mid s)P(b \mid s)P(a \mid t,l)P(x \mid a)P(d \mid a,b)$$

$$\Rightarrow f_v(t)P(s)P(l \mid s)P(b \mid s)P(a \mid t,l)P(x \mid a)P(d \mid a,b)$$

Eliminate: s

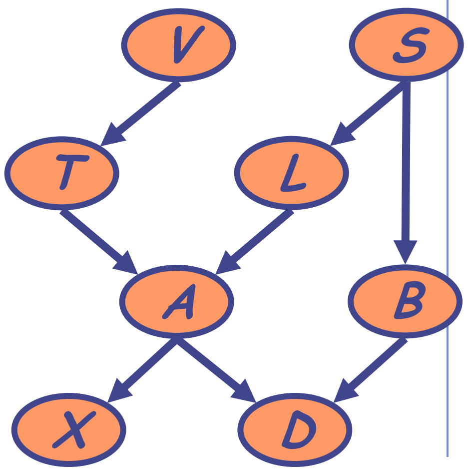
Compute: $$f_s(b,l) = \sum_s P(s)P(b \mid s)P(l \mid s)$$

$$\Rightarrow f_v(t)f_s(b,l)P(a \mid t,l)P(x \mid a)P(d \mid a,b)$$

برای ۴ حالت مختلف Lو B باید احتمال را نگه داریم

Summing on s results in a factor with two arguments $f_s(b,l)$
In general, result of elimination may be a function of several variables

- We want to compute $P(d)$
- Need to eliminate: x,t,l,a,b



- Initial factors

$$P(v)P(s)P(t \mid v)P(l \mid s)P(b \mid s)P(a \mid t,l)P(x \mid a)P(d \mid a,b)$$

$$\Rightarrow f_v(t)P(s)P(l \mid s)P(b \mid s)P(a \mid t,l)P(x \mid a)P(d \mid a,b)$$

$$\Rightarrow f_v(t)f_s(b,l)P(a \mid t,l)\underline{P(x \mid a)}P(d \mid a,b)$$

در این مثال خاص وقتی بر روی x سیگما می زنیم حاصل یک می شود چون که اگر بر روی متغیر اصلی یک تابع احتمال شرطی سیگما بزنیم یک می شود

در واقع برای محاسبه احتمال d اصلا نیازی به xنداریم به همین خاطر حاصل یک می شود

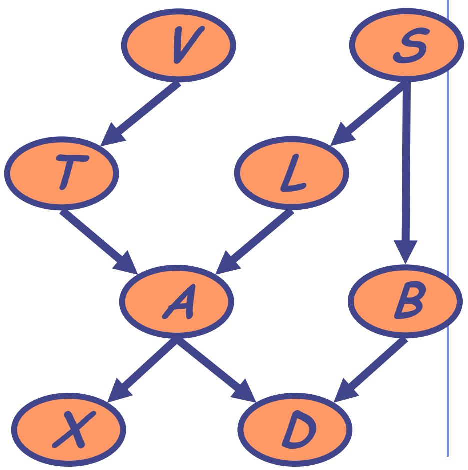
Eliminate: x

Compute: $$f_x(a) = \sum_x P(x \mid a)$$

$$\Rightarrow f_v(t)f_s(b,l)\underline{f_x(a)}P(a \mid t,l)P(d \mid a,b)$$

Note: $f_x(a) = 1$ for all values of a !!

- We want to compute $P(d)$
- Need to eliminate: t,l,a,b



- Initial factors

$$P(v)P(s)P(t \mid v)P(l \mid s)P(b \mid s)P(a \mid t,l)P(x \mid a)P(d \mid a,b)$$

$$\Rightarrow f_v(t)P(s)P(l \mid s)P(b \mid s)P(a \mid t,l)P(x \mid a)P(d \mid a,b)$$

$$\Rightarrow f_v(t)f_s(b,l)P(a \mid t,l)P(x \mid a)P(d \mid a,b)$$

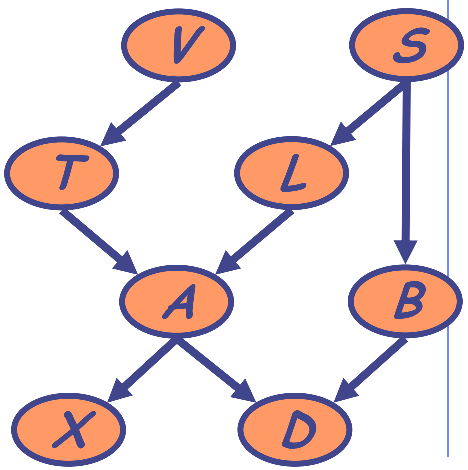
$$\Rightarrow \underline{f_v(t)}f_s(b,l)f_x(a)\underline{P(a \mid t,l)}P(d \mid a,b)$$

Eliminate: t

Compute: $$f_t(a,l) = \sum_t f_v(t)P(a \mid t,l)$$

$$\Rightarrow f_s(b,l)f_x(a)\underline{f_t(a,l)}P(d \mid a,b)$$

- We want to compute *P(d)*
- Need to eliminate: *l,a,b*



- Initial factors

$$P(v)P(s)P(t\,|\,v)P(l\,|\,s)P(b\,|\,s)P(a\,|\,t,l)P(x\,|\,a)P(d\,|\,a,b)$$

$$\Rightarrow f_v(t)P(s)P(l\,|\,s)P(b\,|\,s)P(a\,|\,t,l)P(x\,|\,a)P(d\,|\,a,b)$$

$$\Rightarrow f_v(t)f_s(b,l)P(a\,|\,t,l)P(x\,|\,a)P(d\,|\,a,b)$$

$$\Rightarrow f_v(t)f_s(b,l)f_x(a)P(a\,|\,t,l)P(d\,|\,a,b)$$

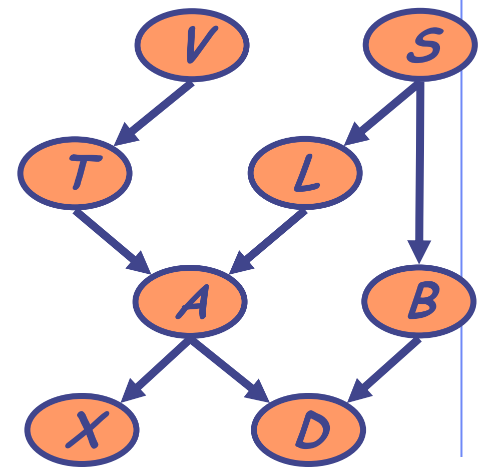
$$\Rightarrow \underline{f_s(b,l)}f_x(a)\underline{f_t(a,l)}P(d\,|\,a,b)$$

Eliminate: *l*

Compute: $$f_l(a,b) = \sum_l f_s(b,l)f_t(a,l)$$

$$\Rightarrow \underline{f_l(a,b)}f_x(a)P(d\,|\,a,b)$$

- We want to compute $P(d)$
- Need to eliminate: b



- Initial factors

$$P(v)P(s)P(t \mid v)P(l \mid s)P(b \mid s)P(a \mid t,l)P(x \mid a)P(d \mid a,b)$$

$$\Rightarrow f_v(t)P(s)P(l \mid s)P(b \mid s)P(a \mid t,l)P(x \mid a)P(d \mid a,b)$$

$$\Rightarrow f_v(t)f_s(b,l)P(a \mid t,l)P(x \mid a)P(d \mid a,b)$$

$$\Rightarrow f_v(t)f_s(b,l)f_x(a)P(a \mid t,l)P(d \mid a,b)$$

$$\Rightarrow f_s(b,l)f_x(a)f_t(a,l)P(d \mid a,b)$$

$$\Rightarrow \underline{f_l(a,b)}\,\underline{f_x(a)}\,\underline{P(d \mid a,b)} \Rightarrow \underline{f_a(b,d)} \Rightarrow \underline{f_b(d)}$$

Eliminate: a,b

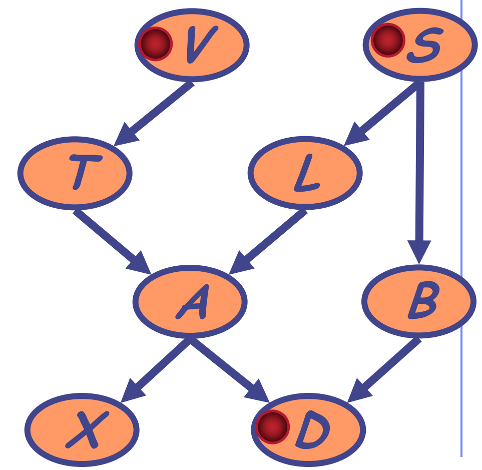
Compute:

$$f_a(b,d) = \sum_a f_l(a,b)f_x(a)p(d \mid a,b) \quad f_b(d) = \sum_b f_a(b,d)$$

Variable Elimination

- We now understand variable elimination as a sequence of **rewriting** operations
- Actual computation is done in elimination step
- Computation depends on order of elimination

Dealing with evidence

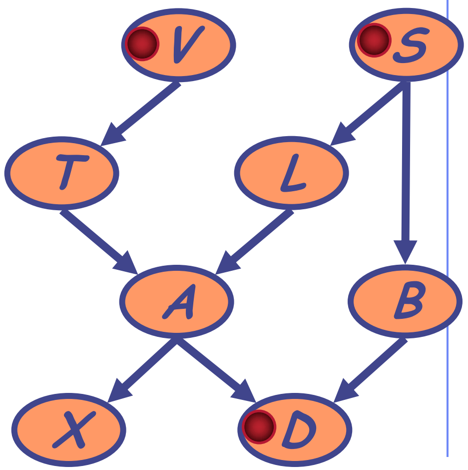


- How do we deal with evidence?

- Suppose get evidence $V = t, S = f, D = t$
- We want to compute $P(L, V = t, S = f, D = t)$

با داشتن اطلاعات از evidence ها بخواهیم احتمال یک پدیده را حساب کنیم :
یک تابع احتمال توام محاسبه کرد درباره متغیری که می خواهیم درباره آن قضاوت کنیم
با توجه به شواهدی که از دنیا مشاهده کردیم
وقتی این شواهد را می بینیم در رابطه توزیع احتمال توام به جای بعضی از متغیرها مقدار قرار می گیرد
باید خانه هایی را جمع بزنیم که شرایط در آن برقرار است .

Dealing with Evidence



- We start by writing the factors:

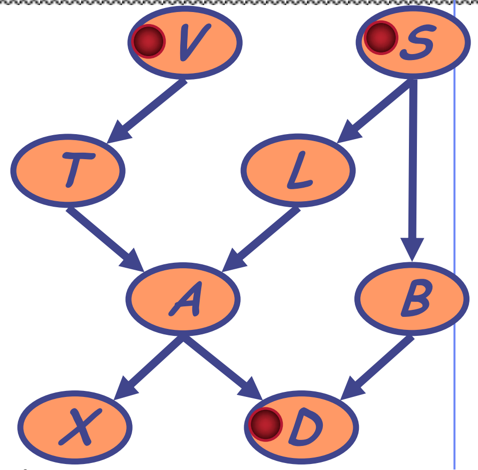
$$P(v)P(s)P(t \mid v)P(l \mid s)P(b \mid s)P(a \mid t,l)P(x \mid a)P(d \mid a,b)$$

- Since we know that $V = t$, we don't need to eliminate V
- Instead, we can replace the factors $P(V)$ and $P(T|V)$ with

$$f_{P(V)} = P(V = t) \quad f_{p(T|V)}(T) = P(T \mid V = t)$$

- These "select" the appropriate parts of the original factors given the evidence
- Note that $f_{p(V)}$ is a constant, and thus does not appear in elimination of other variables

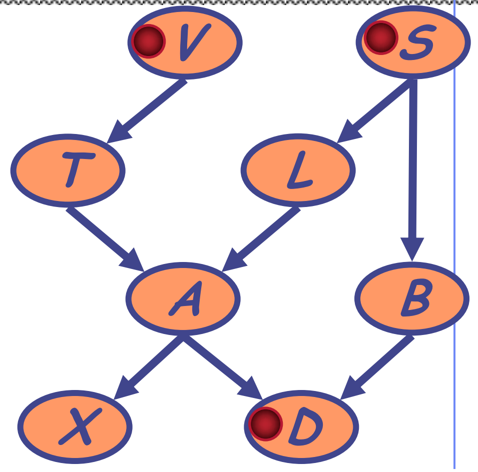
Dealing with Evidence



- Given evidence $V = t$, $S = f$, $D = t$
- Compute $P(L, V = t, S = f, D = t)$
- Initial factors, after setting evidence:

$$f_{P(v)} f_{P(s)} f_{P(t|v)}(t) f_{P(l|s)}(l) f_{P(b|s)}(b) P(a \mid t, l) P(x \mid a) f_{P(d|a,b)}(a,b)$$

Dealing with Evidence



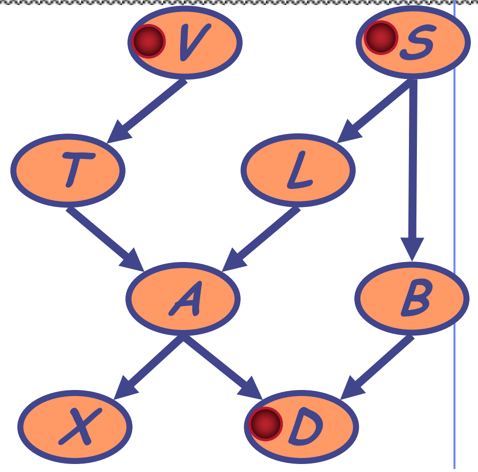
- Given evidence $V = t, S = f, D = t$
- Compute $P(L, V = t, S = f, D = t)$
- Initial factors, after setting evidence:

$$f_{P(v)} f_{P(s)} f_{P(t|v)}(t) f_{P(l|s)}(l) f_{P(b|s)}(b) P(a \mid t,l) P(x \mid a) f_{P(d|a,b)}(a,b)$$

- Eliminating x, we get

$$f_{P(v)} f_{P(s)} f_{P(t|v)}(t) f_{P(l|s)}(l) f_{P(b|s)}(b) P(a \mid t,l) f_x(a) f_{P(d|a,b)}(a,b)$$

Dealing with Evidence



- Given evidence $V = t, S = f, D = t$
- Compute $P(L, V = t, S = f, D = t)$
- Initial factors, after setting evidence:

$$f_{P(v)} f_{P(s)} f_{P(t|v)}(t) f_{P(l|s)}(l) f_{P(b|s)}(b) P(a \mid t,l) P(x \mid a) f_{P(d|a,b)}(a,b)$$

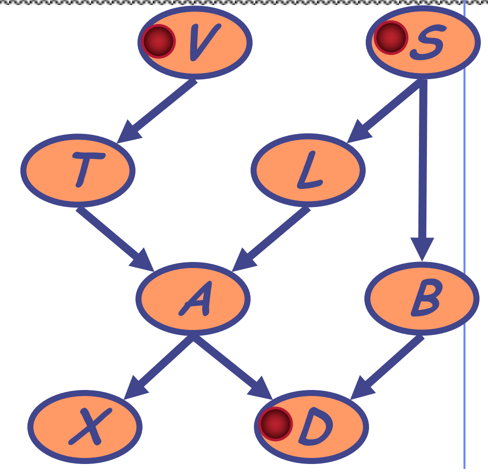
- Eliminating x, we get

$$f_{P(v)} f_{P(s)} f_{P(t|v)}(t) f_{P(l|s)}(l) f_{P(b|s)}(b) P(a \mid t,l) f_x(a) f_{P(d|a,b)}(a,b)$$

- Eliminating t, we get

$$f_{P(v)} f_{P(s)} f_{P(l|s)}(l) f_{P(b|s)}(b) f_t(a,l) f_x(a) f_{P(d|a,b)}(a,b)$$

Dealing with Evidence



- Given evidence $V = t, S = f, D = t$
- Compute $P(L, V = t, S = f, D = t)$
- Initial factors, after setting evidence:

$$f_{P(v)} f_{P(s)} f_{P(t|v)}(t) f_{P(l|s)}(l) f_{P(b|s)}(b) P(a \mid t, l) P(x \mid a) f_{P(d|a,b)}(a,b)$$

- Eliminating x, we get

$$f_{P(v)} f_{P(s)} f_{P(t|v)}(t) f_{P(l|s)}(l) f_{P(b|s)}(b) P(a \mid t, l) f_x(a) f_{P(d|a,b)}(a,b)$$

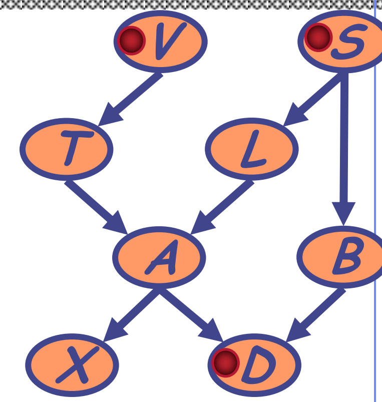
- Eliminating t, we get

$$f_{P(v)} f_{P(s)} f_{P(l|s)}(l) f_{P(b|s)}(b) f_t(a,l) f_x(a) f_{P(d|a,b)}(a,b)$$

- Eliminating a, we get

$$f_{P(v)} f_{P(s)} f_{P(l|s)}(l) f_{P(b|s)}(b) f_a(b,l)$$

Dealing with Evidence



- Given evidence *V = t, S = f, D = t*
- Compute *P(L, V = t, S = f, D = t)*
- Initial factors, after setting evidence:

$$f_{P(v)} f_{P(s)} f_{P(t|v)}(t) f_{P(l|s)}(l) f_{P(b|s)}(b) P(a \mid t,l) P(x \mid a) f_{P(d|a,b)}(a,b)$$

- Eliminating *x*, we get

$$f_{P(v)} f_{P(s)} f_{P(t|v)}(t) f_{P(l|s)}(l) f_{P(b|s)}(b) P(a \mid t,l) f_x(a) f_{P(d|a,b)}(a,b)$$

- Eliminating *t*, we get

$$f_{P(v)} f_{P(s)} f_{P(l|s)}(l) f_{P(b|s)}(b) \cancel{f_t(a,l) f_x(a) f_{P(d|a,b)}(a,b)}$$

- Eliminating *a*, we get

$$f_{P(v)} f_{P(s)} f_{P(l|s)}(l) f_{P(b|s)}(b) f_a(b,l)$$

- Eliminating *b*, we get

$$f_{P(v)} f_{P(s)} f_{P(l|s)}(l) f_b(l)$$

Variable Elimination Algorithm

- Let X1,…, Xm be an ordering on the non-query variables $$\sum_{X_1}\sum_{X_2}\ldots\sum_{X_m}\prod_{j} P(X_j \mid \text{Parents}(X_j))$$
- For I = m, …, 1
  - Leave in the summation for Xi only factors mentioning Xi
  - Multiply the factors, getting a factor that contains a number for each value of the variables mentioned, including Xi
  - Sum out Xi, getting a factor f that contains a number for each value of the variables mentioned, not including Xi
  - Replace the multiplied factor in the summation

Complexity of variable elimination

- Suppose in one elimination step we compute

$$f_x(y_1,\ldots,y_k)=\sum_x f'_x(x,y_1,\ldots,y_k)$$

$$f'_x(x,y_1,\ldots,y_k)=\prod_{i=1}^{m} f_i(x,y_{1,1},\ldots,y_{1,l_i})$$

This requires

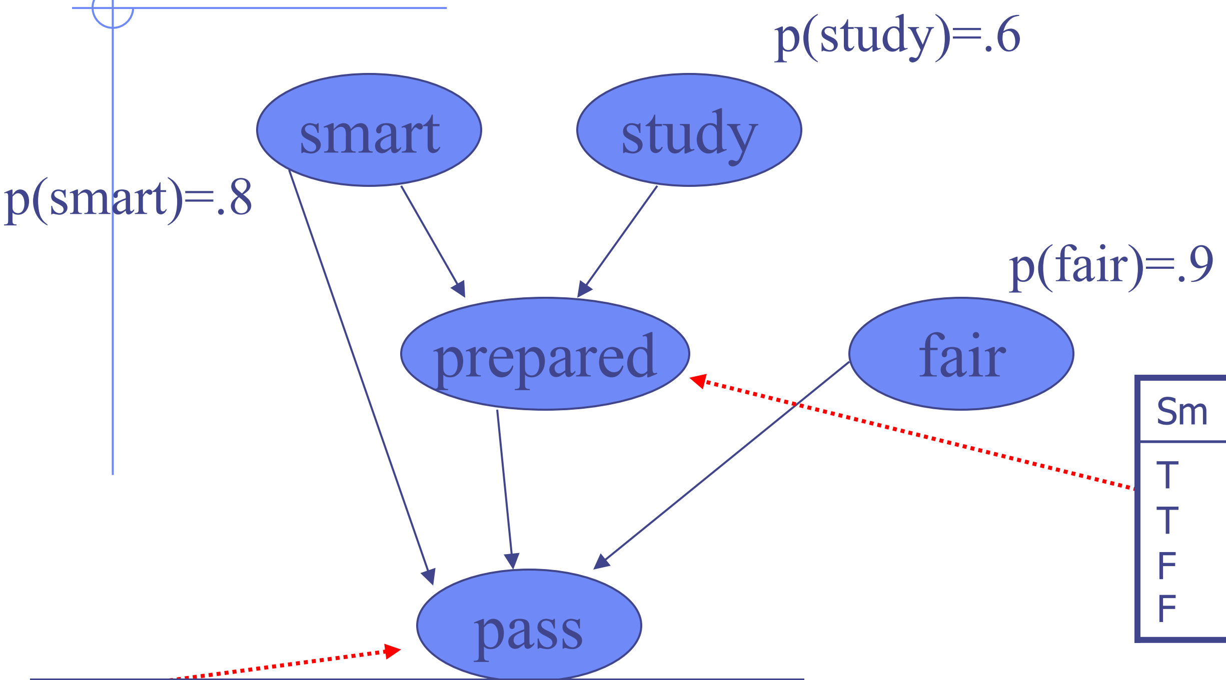
- $m\cdot|Val(X)|\cdot\prod_i|Val(Y_i)|$ multiplications
  - For each value for $x, y_1, \ldots, y_k$, we do m multiplications
- $|Val(X)|\cdot\prod_i|Val(Y_i)|$ additions
  - For each value of $y_1, \ldots, y_k$, we do $|Val(X)|$ additions

Complexity is exponential in number of variables in the intermediate factor!

Understanding Variable Elimination

- We want to select "good" elimination orderings that reduce complexity
- This can be done be examining a graph theoretic property of the "induced" graph; we will not cover this in class.
- This reduces the problem of finding good ordering to graph-theoretic operation that is well-understood—unfortunately computing it is NP-hard!

Exercise: Variable elimination



Sm	St	P(Pr|Sm,St)
T	T	.9
T	F	.5
F	T	.7
F	F	.1

Sm	Pr	F	P(Pa|Sm,Pr,F)
T	T	T	.9
T	T	F	.1
T	F	T	.7
T	F	F	.1
F	T	T	.7
F	T	F	.1
F	F	T	.2
F	F	F	.1

Query: What is the probability that a student is smart, given that they pass the exam?

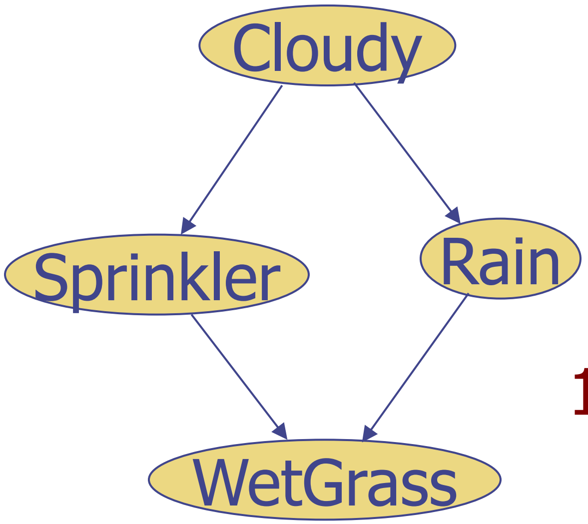
Approaches to inference

- Exact inference
  - Inference in Simple Chains
  - Variable elimination
  - Clustering / join tree algorithms
- **Approximate inference**
  - Stochastic simulation / sampling methods
  - Markov chain Monte Carlo methods

Stochastic simulation - direct

- Suppose you are given values for some subset of the variables, G, and want to infer values for unknown variables, U
- Randomly generate a very large number of instantiations from the BN
  - Generate instantiations for **all** variables – start at root variables and work your way "forward"
- Rejection Sampling: keep those instantiations that are consistent with the values for G
- Use the frequency of values for U to get estimated probabilities
- Accuracy of the results depends on the size of the sample (asymptotically approaches exact results)

Direct Stochastic Simulation



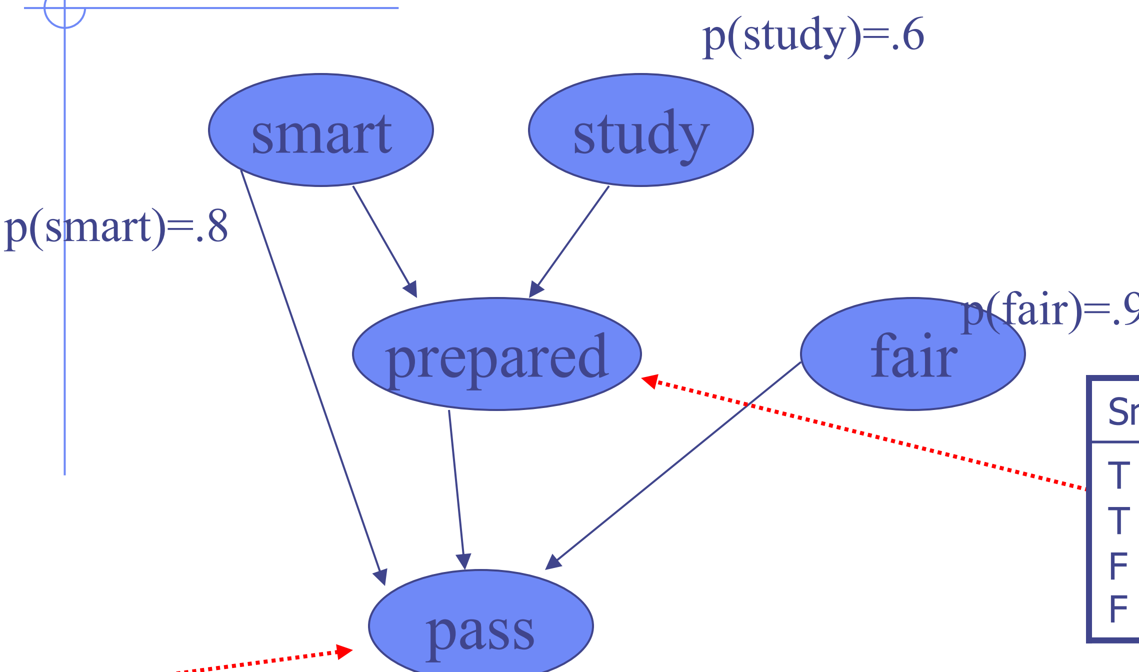
P(WetGrass|Cloudy)?

P(WetGrass|Cloudy)
= P(WetGrass $\wedge$ Cloudy) / P(Cloudy)

1. Repeat N times:
   1.1. Guess Cloudy at random
   1.2. For each guess of Cloudy, guess Sprinkler and Rain, then WetGrass

2. Compute the ratio of the # runs where WetGrass and Cloudy are True over the # runs where Cloudy is True

Exercise: Direct sampling



Sm	St	P(Pr|…)
T	T	.9
T	F	.5
F	T	.7
F	F	.1

Sm	Pr	F	P(Pa|…)
T	T	T	.9
T	T	F	.1
T	F	T	.7
T	F	F	.1
F	T	T	.7
F	T	F	.1
F	F	T	.2
F	F	F	.1

Topological order = …?

Random number generator: .35, .76, .51, .44, .08, .28, .03, .92, .02, .42

Likelihood weighting

- Idea: Don't generate samples that need to be rejected in the first place!
- Sample only from the unknown variables Z
- Weight each sample according to the likelihood that it would occur, given the evidence E

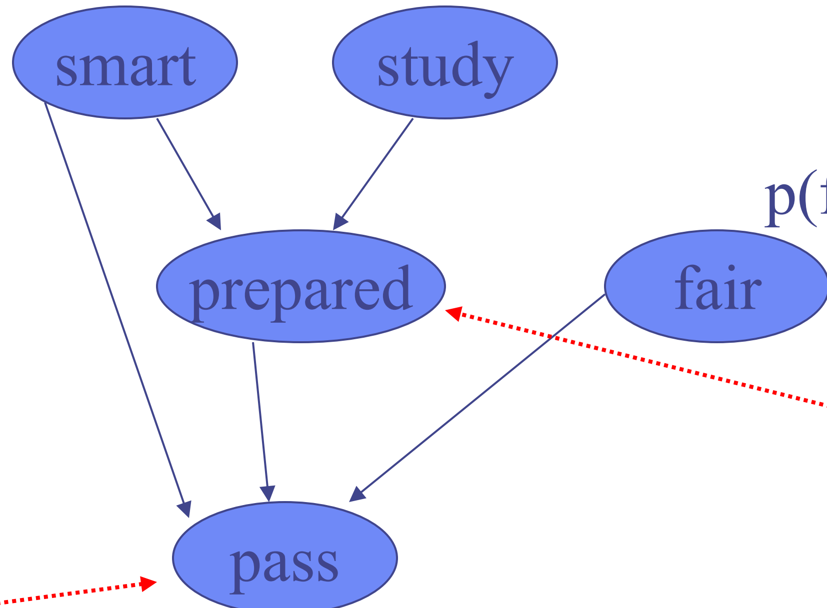
Markov chain Monte Carlo algorithm

- So called because
  - Markov chain – each instance generated in the sample is dependent on the previous instance
  - Monte Carlo – statistical sampling method
- Perform a random walk through variable assignment space, collecting statistics as you go
  - Start with a random instantiation, consistent with evidence variables
  - At each step, for some nonevidence variable, randomly sample its value, consistent with the other current assignments
- Given enough samples, MCMC gives an accurate estimate of the true distribution of values

Exercise: MCMC sampling

p(smart)=.8 p(study)=.6

p(fair)=.9



Sm	St	P(Pr|…)
T	T	.9
T	F	.5
F	T	.7
F	F	.1

Sm	Pr	F	P(Pa|…)
T	T	T	.9
T	T	F	.1
T	F	T	.7
T	F	F	.1
F	T	T	.7
F	T	F	.1
F	F	T	.2
F	F	F	.1

Topological order = …?
Random number generator: .35, .76, .51, .44, .08, .28, .03, .92, .02, .42

Summary

- **Bayes nets**
  - Structure
  - Parameters
  - Conditional independence
- **BN inference**
  - Exact Inference
    - Variable elimination
  - Sampling methods

Applications

- h ... ynet /r
- M ... e d
- F
- T ... are s