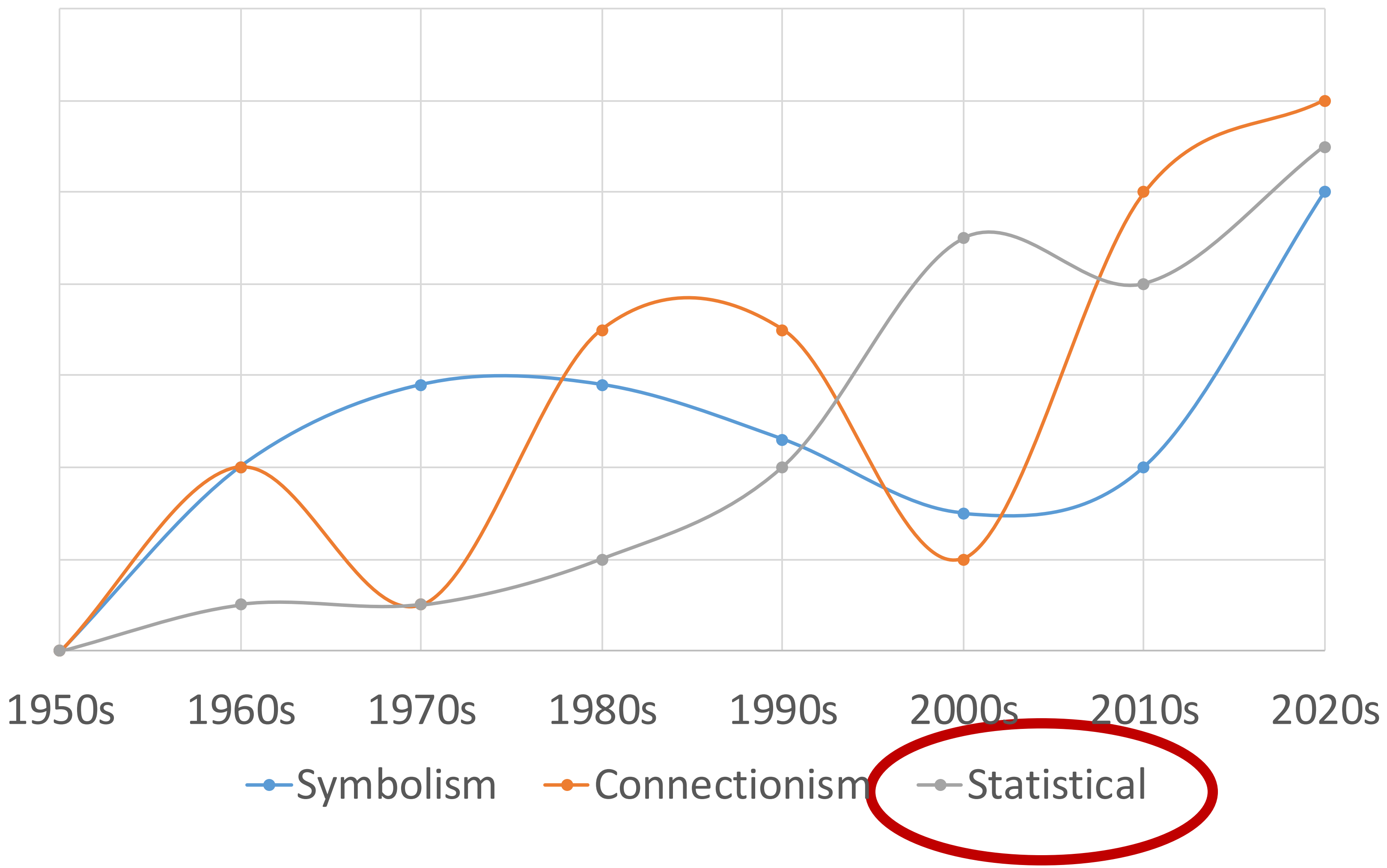
Three types of (strong) AI approaches



Probability

AIMA Chapter 13

[Adapted from slides by Dan Klein and Pieter Abbeel at UC Berkeley (ai.berkeley.edu).]

Uncertainty

- My flight to New York is scheduled to leave at 11:25
  - Let action A_t = leave home t minutes before flight and drive to the airport
  - Will A_t ensure that I catch the plane?
- Problems:
  - noisy sensors (radio traffic reports, Google maps)
  - uncertain action outcome (car breaking down, accident, etc.)
  - partial observability (other drivers' plans, etc.)
  - immense complexity of modelling and predicting traffic, security line, etc.

Probability

- Probability
  - Given the available evidence and the choice A_{120}, I will catch the plane with probability 0.92
- ***Subjective*** or ***Bayesian*** probability:
  - Probabilities relate propositions to one's own state of knowledge
    - ignorance: lack of relevant facts, initial conditions, etc.
    - laziness: failure to list all exceptions, compute detailed predictions, etc.
  - Not claiming a "probabilistic tendency" in the actual situation (traffic is not like quantum mechanics)

Decisions

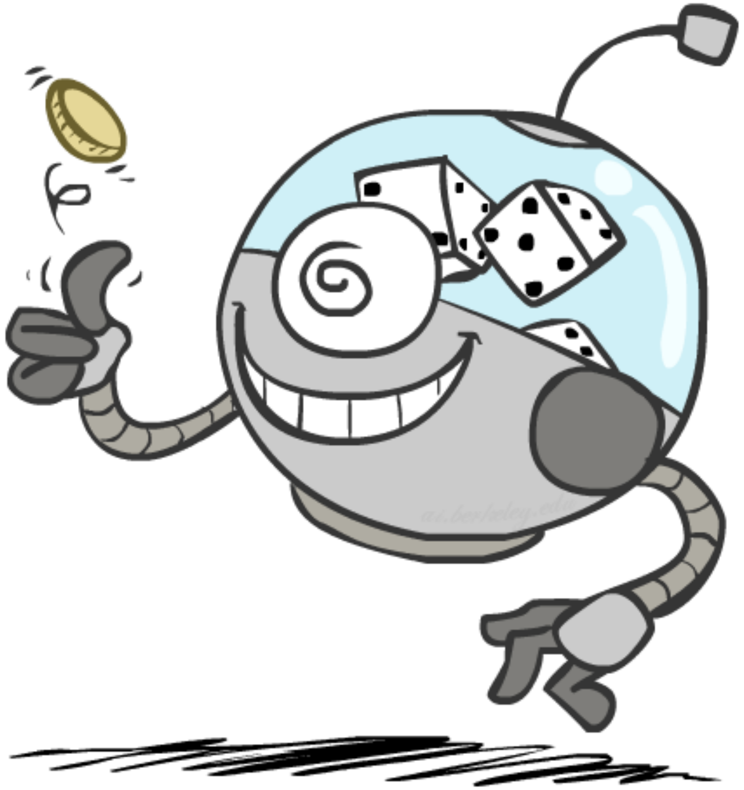
- Suppose I believe
  - $P(CatchPlane \mid A_{60}, all\ my\ evidence...) = 0.51$
  - $P(CatchPlane \mid A_{120}, all\ my\ evidence...) = 0.97$
  - $P(CatchPlane \mid A_{1440}, all\ my\ evidence...) = 0.9999$
- Which action should I choose?
- Depends on my ***preferences*** for, e.g., missing flight, airport food, etc.
- ***Utility theory*** is used to represent and infer preferences
- ***Decision theory*** = utility theory + probability theory
- ***Maximize expected utility*** : $a^* = argmax_a \sum_s P(s \mid a)\ U(s)$

Random Variables

- A random variable is some aspect of the world about which we (may) have uncertainty
  - R = Is it raining?
  - T = Is it hot or cold?
  - D = How long will it take to drive to work?
  - L = Where is the pacman?

- We denote random variables with capital letters

- Like variables in a CSP, random variables have domains
  - R in {true, false} (often write as {+r, -r})
  - T in {hot, cold}
  - D in [0, ∞)
  - L in possible locations, maybe {(0,0), (0,1), …}



Probability Distributions

- Associate a probability with each value of a random variable
  - Temperature:

$P(T)$

T	P
hot	0.5
cold	0.5

  - Weather:

$P(W)$

W	P
sun	0.6
rain	0.1
fog	0.3
meteor	0.0

  - A probability is a single number

$P(W = rain) = 0.1$ Shorthand notation: $P(rain) = P(W = rain),$

- Must have: $\forall x \;\; P(X = x) \geq 0$ and $\sum_x P(X = x) = 1$

Joint Distributions

- A *joint distribution* over a set of random variables: $X_1, X_2, \ldots X_n$ specifies a real number for each assignment (or *outcome*):

$$P(X_1 = x_1, X_2 = x_2, \ldots X_n = x_n)$$

$$P(x_1, x_2, \ldots x_n)$$

  - Must obey: $P(x_1, x_2, \ldots x_n) \geq 0$

$$\sum_{(x_1, x_2, \ldots x_n)} P(x_1, x_2, \ldots x_n) = 1$$

$P(T, W)$

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

- Size of distribution for n variables with domain size d? d^n
  - For all but the smallest distributions, cannot write out by hand!

Probabilistic Models

- A probabilistic model is a joint distribution over a set of random variables

- Probabilistic models:
  - (Random) variables with domains
  - Joint distributions: say whether assignments (outcomes) are likely
  - Ideally: only certain variables directly interact

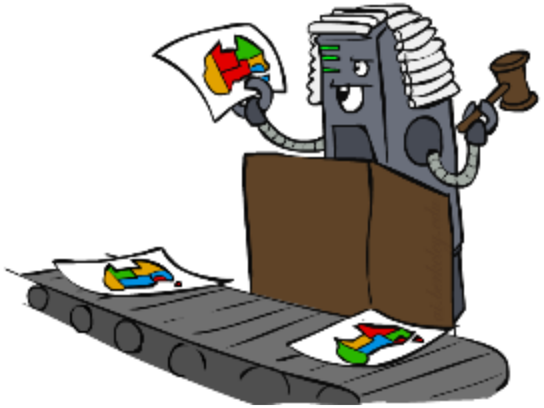
- Constraint satisfaction problems:
  - Variables with domains
  - Constraints: state whether assignments are possible
  - Ideally: only certain variables directly interact

Distribution over T,W

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

Constraint over T,W

T	W	P
hot	sun	T
hot	rain	F
cold	sun	F
cold	rain	T



Probabilities of events

- An *event* is a set E of outcomes

$$P(E) = \sum_{(x_1 \ldots x_n) \in E} P(x_1 \ldots x_n)$$

- Given a joint distribution over all variables, we can compute any event probability!
  - Probability that it’s hot AND sunny?
  - Probability that it’s hot?
  - Probability that it’s hot OR sunny?

$P(T, W)$

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

Marginal Distributions

- Marginal distributions are sub-tables which eliminate variables
- Marginalization (summing out): Combine collapsed rows by adding

$P(T, W)$

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

$$P(t) = \sum_{w} P(t, w)$$

$P(T)$

T	P
hot	0.5
cold	0.5

$$P(w) = \sum_{t} P(t, w)$$

$P(W)$

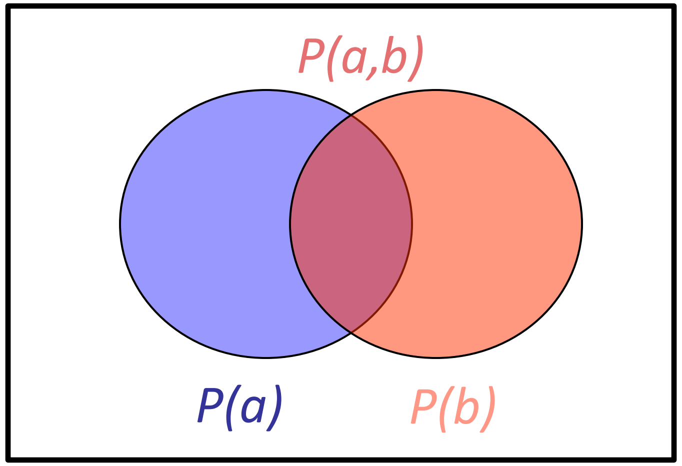
W	P
sun	0.6
rain	0.4

$$P(X_1 = x_1) = \sum_{x_2} P(X_1 = x_1, X_2 = x_2)$$

Conditional Probabilities

- The probability of an event given that another event has occurred

$$P(a|b) = \frac{P(a,b)}{P(b)}$$



$$P(T,W)$$

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

$$P(W=s|T=c) = \frac{P(W=s,T=c)}{P(T=c)} = 0.4$$

$$= P(W=s,T=c) + P(W=r,T=c)$$

$$= 0.2 + 0.3 \quad = 0.5$$

Conditional Distributions

- Conditional distributions are probability distributions over some variables given fixed values of others

Joint Distribution

$$P(T,W)$$

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

Conditional Distributions

$$P(W|T)$$

$$P(W|T=hot)$$

W	P
sun	0.8
rain	0.2

$$P(W|T=cold)$$

W	P
sun	0.4
rain	0.6

Probabilistic Inference

- Probabilistic inference
  - compute a desired probability from other known probabilities (e.g. conditional from joint)
- We generally compute conditional probabilities
  - These represent the agent’s beliefs given the evidence
  - P(on time | no reported accidents) = 0.90

Inference by Enumeration

- General case:
  - Evidence variables: $E_1 \dots E_k = e_1 \dots e_k$
  - Query variable: Q
  - Hidden variables: $H_1 \dots H_r$

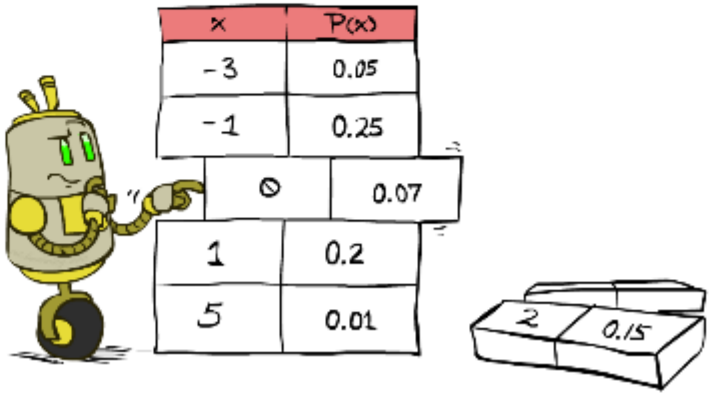
$\left.\right\} X_1, X_2, \dots X_n$

All variables

- We want:

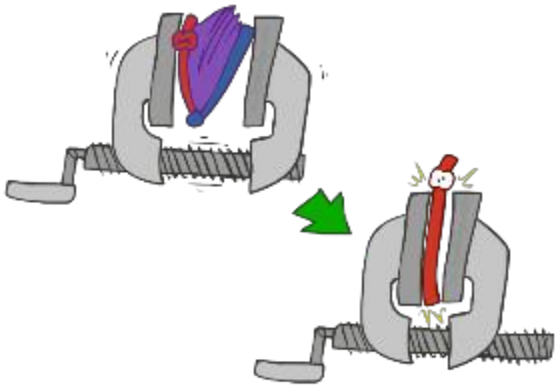
$$P(Q|e_1 \dots e_k)$$

- Step 1: Select the entries consistent with the evidence



x	P(x)
-3	0.05
-1	0.25
0	0.07
1	0.2
5	0.01
2	0.15

- Step 2: Sum out H to get joint of Query and evidence



$$P(Q, e_1 \dots e_k) = \sum_{h_1 \dots h_r} \underbrace{P(Q, h_1 \dots h_r, e_1 \dots e_k)}_{X_1, X_2, \dots X_n}$$

- Step 3: Normalize

$$\times \frac{1}{Z}$$

$$Z = \sum_q P(Q, e_1 \cdots e_k)$$

$$P(Q|e_1 \cdots e_k) = \frac{1}{Z} P(Q, e_1 \cdots e_k)$$

Inference by Enumeration

1. Select the entries consistent with the evidence
2. Sum out H to get joint of Query and evidence
3. Normalize

- P(W | winter)? sun: 0.5, rain: 0.5
- P(W | winter, hot)? sun: 0.67, rain: 0.33

S	T	W	P
summer	hot	sun	0.30
summer	hot	rain	0.05
summer	cold	sun	0.10
summer	cold	rain	0.05
winter	hot	sun	0.10
winter	hot	rain	0.05
winter	cold	sun	0.15
winter	cold	rain	0.20

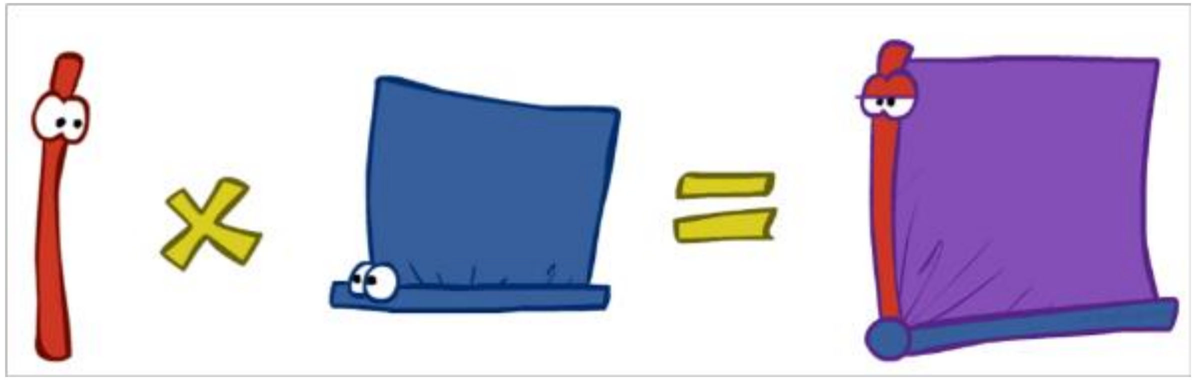
Inference by Enumeration

- Obvious problems:
  - Worst-case time complexity O(d^n)
  - Space complexity O(d^n) to store the joint distribution

The Product Rule

- Sometimes have conditional distributions but want the joint

$$P(y)P(x|y) = P(x,y) \quad \Longleftrightarrow \quad P(x|y) = \frac{P(x,y)}{P(y)}$$



The Product Rule

$$P(y)P(x|y) = P(x, y)$$

- Example:

$P(W)$

W	P
sun	0.8
rain	0.2

$P(D|W)$

D	W	P
wet	sun	0.1
dry	sun	0.9
wet	rain	0.7
dry	rain	0.3

⟺

$P(D, W)$

D	W	P
wet	sun	
dry	sun	
wet	rain	
dry	rain	

The Chain Rule

- More generally, can always write any joint distribution as an incremental product of conditional distributions

$$P(x_1, x_2, x_3) = P(x_1)P(x_2|x_1)P(x_3|x_1, x_2)$$

$$P(x_1, x_2, \ldots x_n) = \prod_i P(x_i|x_1 \ldots x_{i-1})$$

Bayes' Rule

- Two ways to factor a joint distribution over two variables:

$$P(x,y) = P(x|y)P(y) = P(y|x)P(x)$$

- Dividing, we get:

$$P(x|y) = \frac{P(y|x)}{P(y)}P(x)$$

- Why is this at all helpful?
  - Lets us build one conditional from its reverse
  - Often one conditional is tricky but the other one is simple
  - Foundation of many systems we'll see later

- In the running for most important AI equation!

Inference with Bayes’ Rule

- Example: Diagnostic probability from causal probability:

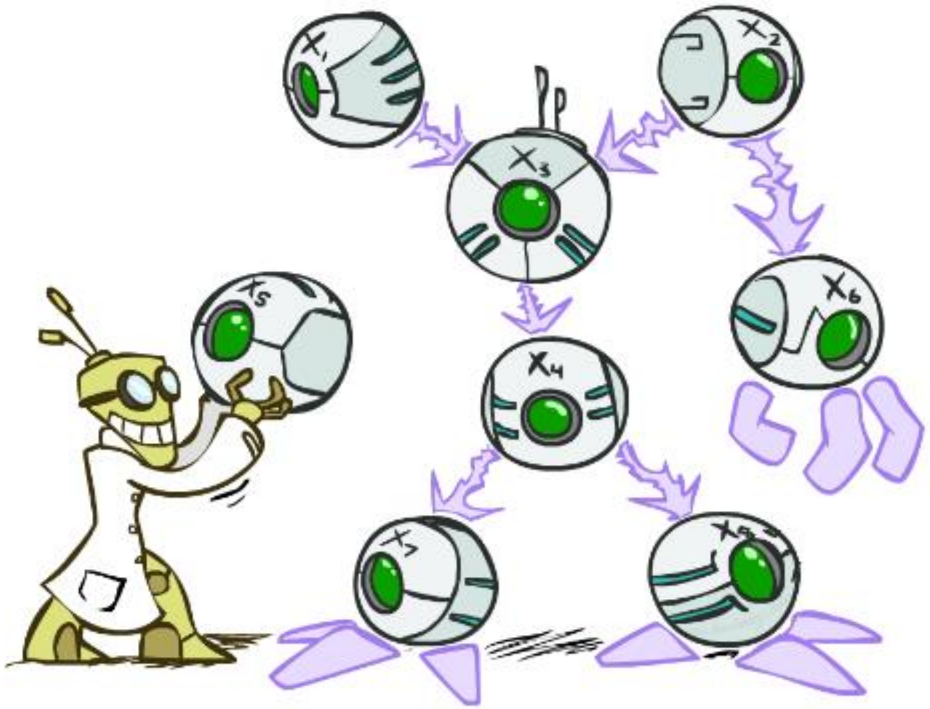
$$P(\text{cause}|\text{effect}) = \frac{P(\text{effect}|\text{cause})P(\text{cause})}{P(\text{effect})}$$

- Example:
  - M: meningitis, S: stiff neck

$$\left.\begin{array}{r} P(+m) = 0.0001 \\ P(+s|+m) = 0.8 \\ P(+s|-m) = 0.01 \end{array}\right\} \text{Example givens}$$

$$P(+m|+s) = \frac{P(+s|+m)P(+m)}{P(+s)} = \frac{P(+s|+m)P(+m)}{P(+s|+m)P(+m) + P(+s|-m)P(-m)} = \frac{0.8 \times 0.0001}{0.8 \times 0.0001 + 0.01 \times 0.9999}$$

Bayesian Networks



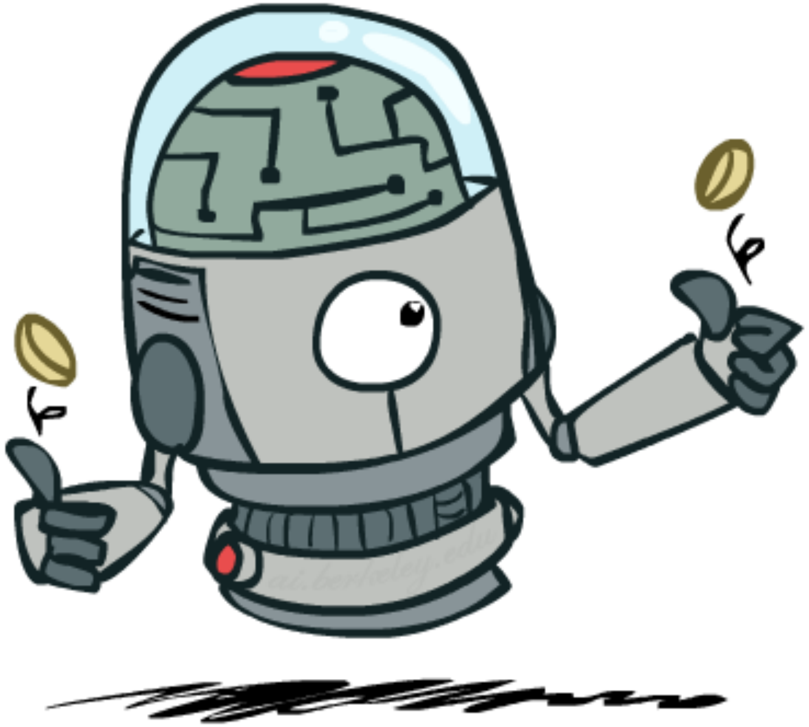
AIMA Chapter 14.1, 14.2

[Adapted from slides by Dan Klein and Pieter Abbeel at UC Berkeley (ai.berkeley.edu)
and from slides by Markus Svensén & Christopher M Bishop]

Additional Reference

- [PRML] Pattern Recognition and Machine Learning, Christopher Bishop, Springer 2006.
    - Chapter 8.1 - 8.3

Independence



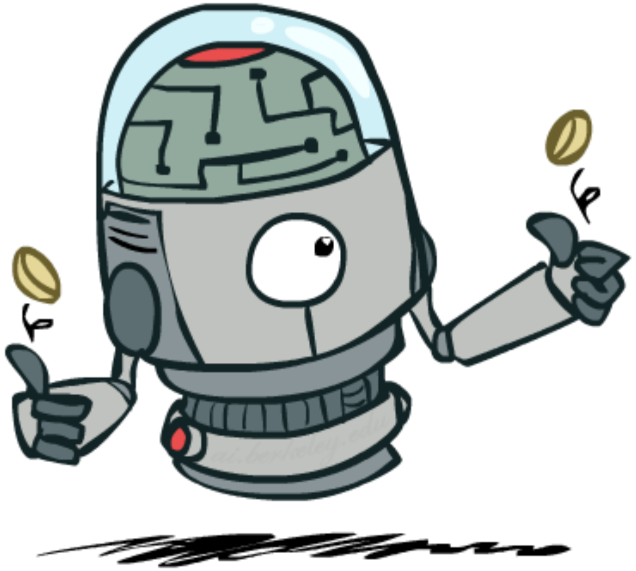
Independence

- Two variables X and Y are (absolutely) ***independent*** if

$$\forall x,y \qquad P(x,y) = P(x)P(y)$$

  - This says that their joint distribution ***factors*** into a product of two simpler distributions
  - Combine with product rule $P(x,y) = P(x|y)P(y)$ we obtain another form:

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



- Example: two dice rolls $Roll_1$ and $Roll_2$
  - $P(Roll_1=5, Roll_2=5) = P(Roll_1=5)P(Roll_2=5) = 1/6 \text{ x } 1/6 = 1/36$
  - $P(Roll_2=5 \mid Roll_1=5) = P(Roll_2=5)$

Conditional Independence

- Unconditional (absolute) independence is rare

- *Conditional independence* is our most basic and robust form of knowledge about uncertain environments.

- X is conditionally independent of Y given Z $X \perp\!\!\!\perp Y | Z$

  if and only if:

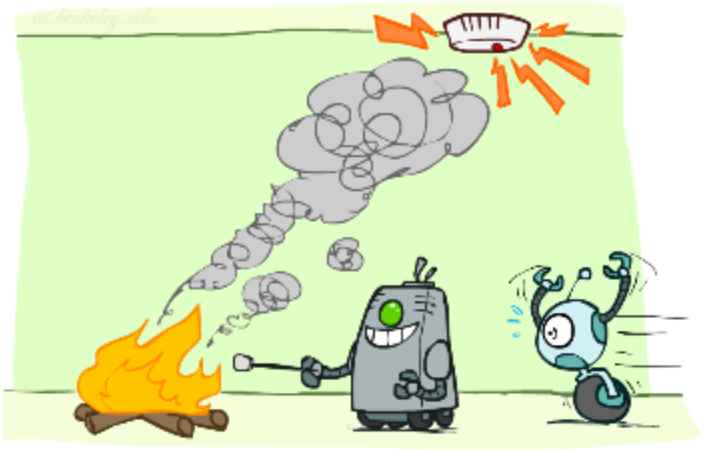
  $\forall x,y,z \quad P(x \mid y,z) = P(x|z)$

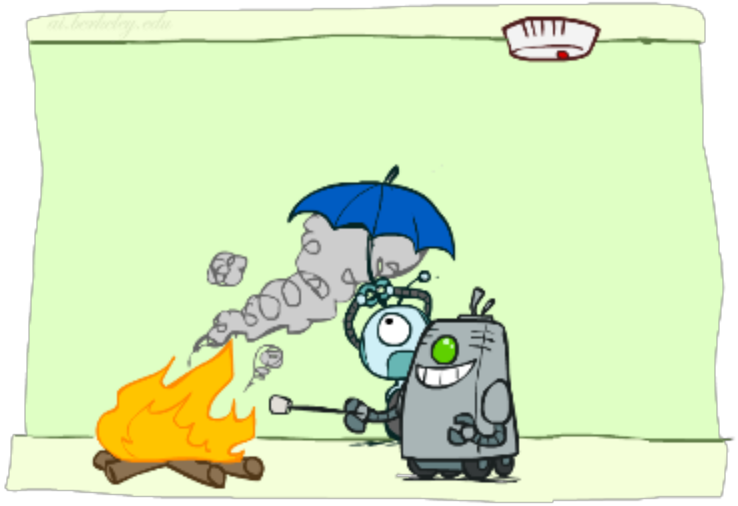
  or, equivalently, if and only if

  $\forall x,y,z \quad P(x,y \mid z) = P(x|z)P(y|z)$

Conditional Independence

- What about this domain:
  - Fire
  - Smoke
  - Alarm (smoke detector)





Conditional Independence

- What about this domain:
  - Traffic
  - Umbrella
  - Raining

Conditional Independence and the Chain Rule

- Chain rule: $P(X_1, X_2, \ldots X_n) = P(X_1)P(X_2|X_1)P(X_3|X_1, X_2) \ldots$

- Trivial decomposition:

$$P(\text{Traffic}, \text{Rain}, \text{Umbrella}) =$$
$$P(\text{Rain})P(\text{Traffic}|\text{Rain})P(\text{Umbrella}|\text{Rain}, \text{Traffic})$$

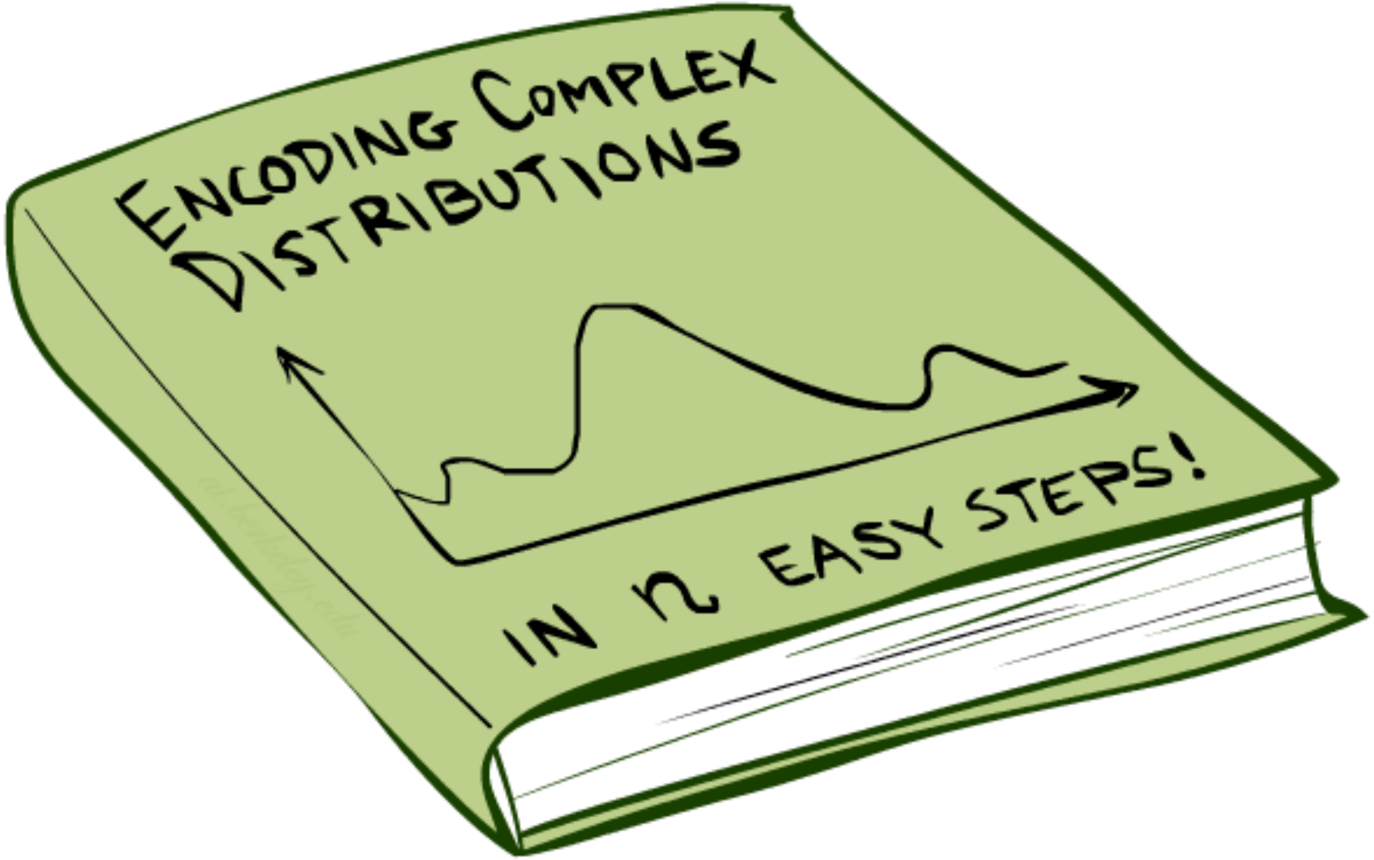
- With assumption of conditional independence:

$$P(\text{Traffic}, \text{Rain}, \text{Umbrella}) =$$
$$P(\text{Rain})P(\text{Traffic}|\text{Rain})\boxed{P(\text{Umbrella}|\text{Rain})}$$

Requires less space to encode!

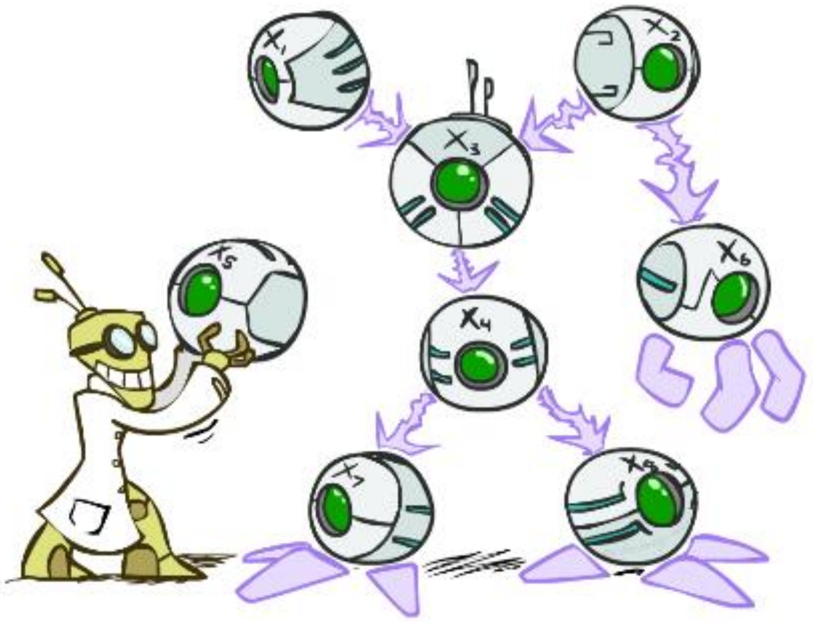
- BayesNets / graphical models help us express conditional independence assumptions, leading to more compact joint distribution representation

Bayesian Networks: Big Picture



Bayesian Networks: Big Picture

- Full joint distribution tables answer every question, but:
  - Size is exponential in the number of variables
  - Need gazillions of examples to learn the probabilities
  - Inference by enumeration (summing out hiddens) is too slow
- Bayesian networks:
  - Express all the conditional independence relationships in a domain
  - Factor the joint distribution into a product of small conditionals
  - Often reduce size from exponential to linear
  - Faster learning from fewer examples
  - Faster inference (linear time in some important cases)



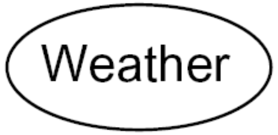
Bayesian Networks Syntax

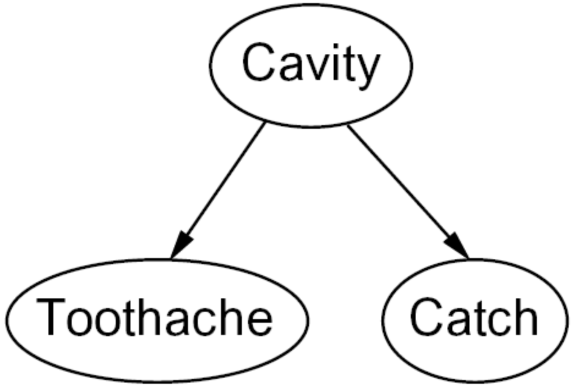
Bayesian Networks Syntax

- Nodes: variables (with domains)

- Arcs: interactions
  - Indicate “direct influence” between variables
  - For now: imagine that arrows mean direct causation (in general, they may not!)
  - Formally: encode conditional independence (more later)

- No cycle is allowed!

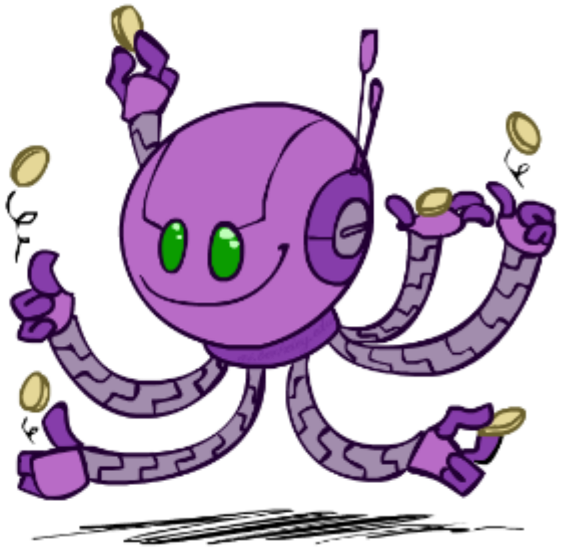




Example: Coin Flips

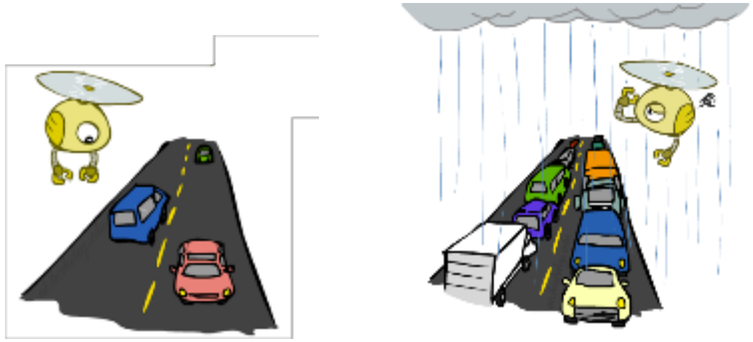
- N independent coin flips

X_1 X_2 $\cdots$ X_n



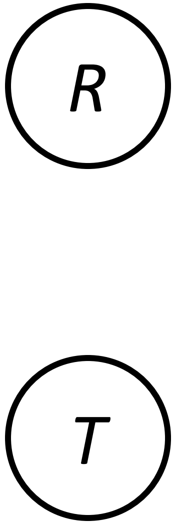
- No interactions between variables: absolute independence

Example: Traffic

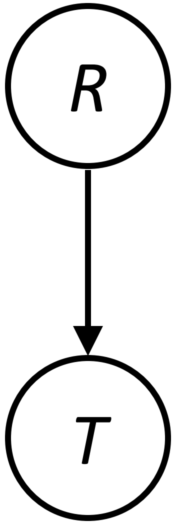


- Variables:
  - R: It rains
  - T: There is traffic

- Model 1: independence

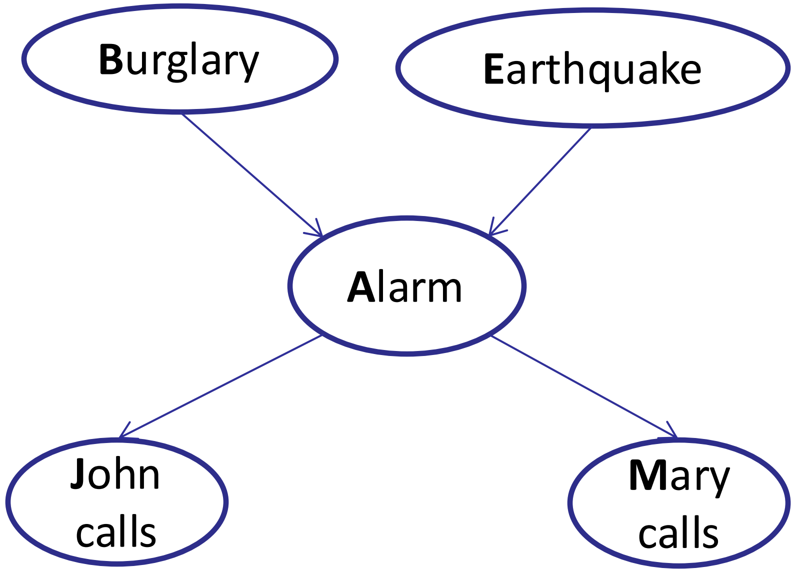


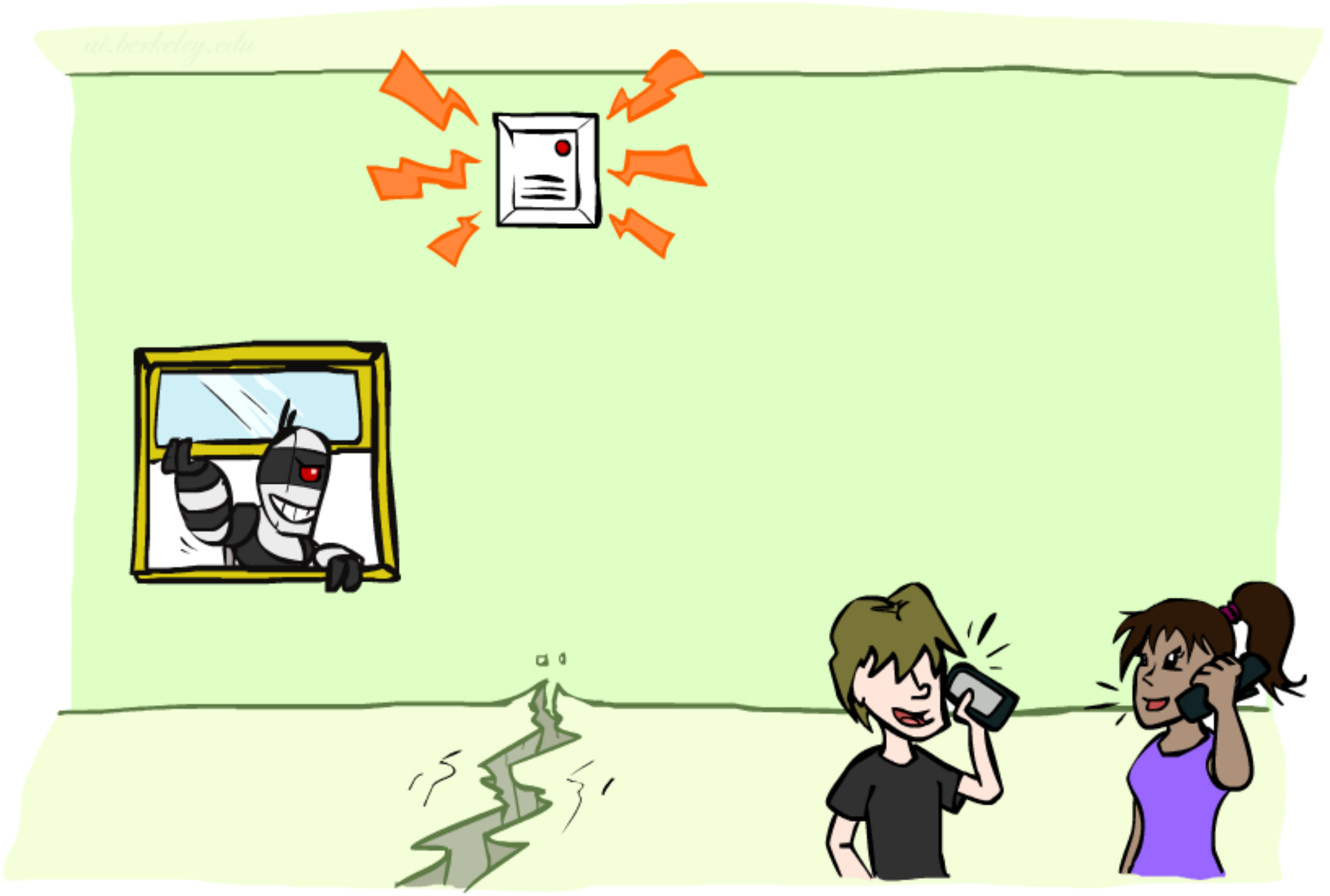
- Model 2: rain causes traffic



Example: Alarm Network

- Variables
  - B: Burglary
  - A: Alarm goes off
  - M: Mary calls
  - J: John calls
  - E: Earthquake!

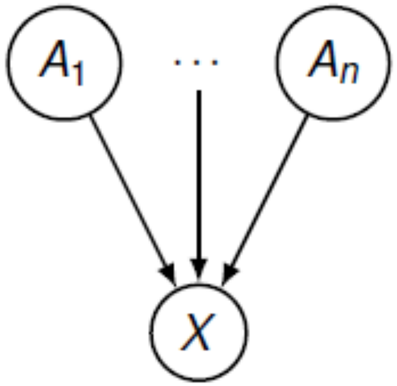




Bayesian Networks Syntax

- A directed, acyclic graph

- Conditional distributions for each node given its ***parent variables*** in the graph
  - ***CPT***: conditional probability table: each row is a distribution for child given a configuration of its parents
  - Description of a noisy "causal" process



$$P(X|A_1, \cdots, A_n)$$

A Bayes net = Topology (graph) + Local Conditional Probabilities

Example: Coin Flips

X_1 X_2 $\cdots$ X_n

$P(X_1)$

h	0.5
t	0.5

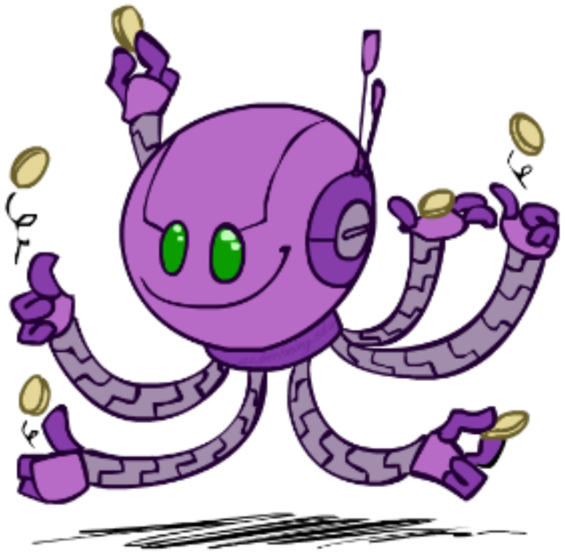
$P(X_2)$

h	0.5
t	0.5

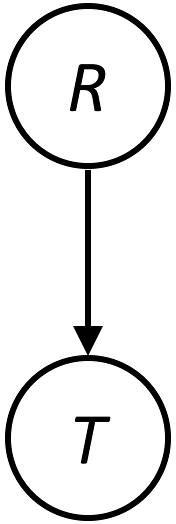
$\cdots$

$P(X_n)$

h	0.5
t	0.5



Example: Traffic

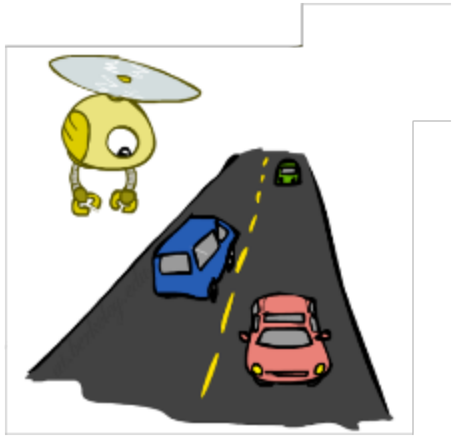


$P(R)$

+r	1/4
-r	3/4

$P(T|R)$

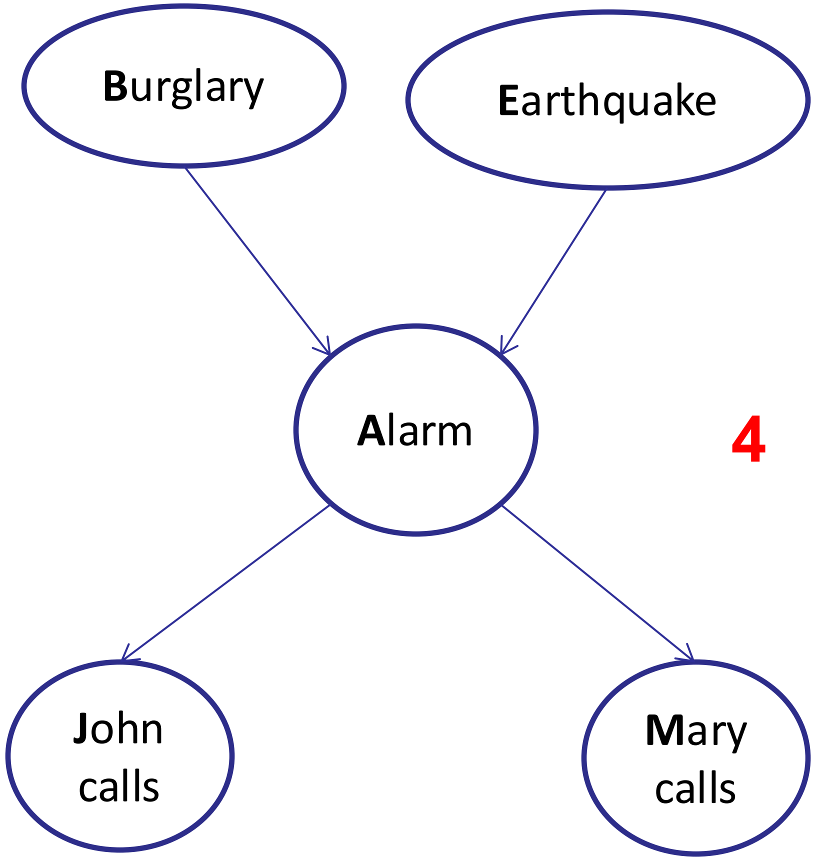
+r	+t	3/4
	-t	1/4
-r	+t	1/2
	-t	1/2



Example: Alarm Network

P(B)	
true	false
0.001	0.999

1



1

P(E)	
true	false
0.002	0.998

4

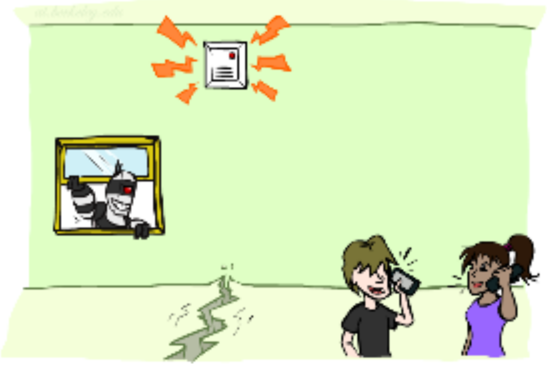
B	E	P(A|B,E)	
		true	false
true	true	0.95	0.05
true	false	0.94	0.06
false	true	0.29	0.71
false	false	0.001	0.999

A	P(J|A)	
	true	false
true	0.9	0.1
false	0.05	0.95

2

2

A	P(M|A)	
	true	false
true	0.7	0.3
false	0.01	0.99



Number of free parameters in each CPT:

- Parent domain sizes $d_1,\ldots,d_k$
- Child domain size d
- Each table row must sum to 1

$(d-1)\ \Pi_i\ d_i$

General formula for sparse BNs

- Suppose
  - n variables
  - Maximum domain size is d
  - Maximum number of parents is k
- Full joint distribution has size $O(d^n)$
- Bayes net has size $O(n \cdot d^{k+1})$
  - Linear scaling with n as long as causal structure is local

Bayesian Networks Semantics

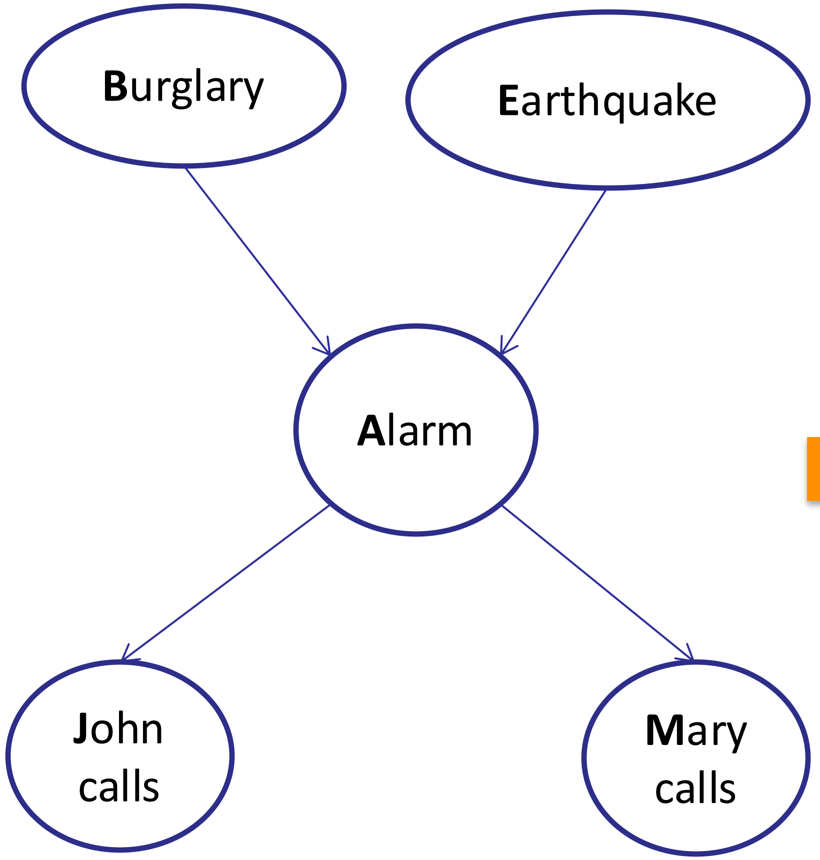
Bayesian networks global semantics

- Bayes nets encode joint distributions as product of conditional distributions on each variable:

$$P(X_1,..,X_n) \; = \; \prod_i P(X_i \mid Parents(X_i))$$

Example

P(B)	
true	false
0.001	0.999



P(E)	
true	false
0.002	0.998

$P(b, \neg e, a, \neg j, \neg m) =$

$P(b)\ P(\neg e)\ P(a|b,\neg e)\ P(\neg j|a)\ P(\neg m|a)$

=.001x.998x.94x.1x.3=.000028

B	E	P(A|B,E)	
		true	false
true	true	0.95	0.05
true	false	0.94	0.06
false	true	0.29	0.71
false	false	0.001	0.999

A	P(J|A)	
	true	false
true	0.9	0.1
false	0.05	0.95

A	P(M|A)	
	true	false
true	0.7	0.3
false	0.01	0.99

Probabilities in BNs

- Global semantics: $P(X_1,..,X_n) = \prod_i P(X_i \mid Parents(X_i))$

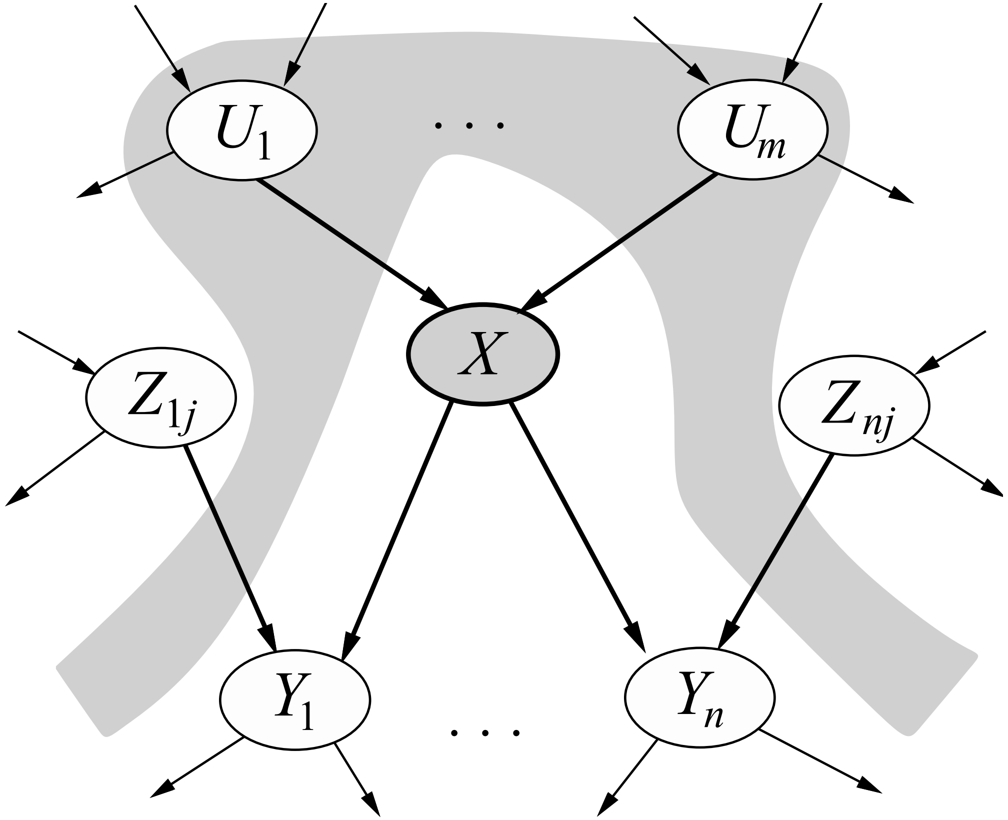
- Chain rule (valid for all distributions): $P(X_1,..,X_n) = \prod_i P(X_i \mid X_1,\ldots,X_{i-1})$

- So for any *i*, we have: $P(X_i \mid X_1,\ldots,X_{i-1}) = P(X_i \mid Parents(X_i))$
  - Conditional independence: parents “shield” node X_i from the other predecessors

- So the network topology implies that certain conditional independencies hold

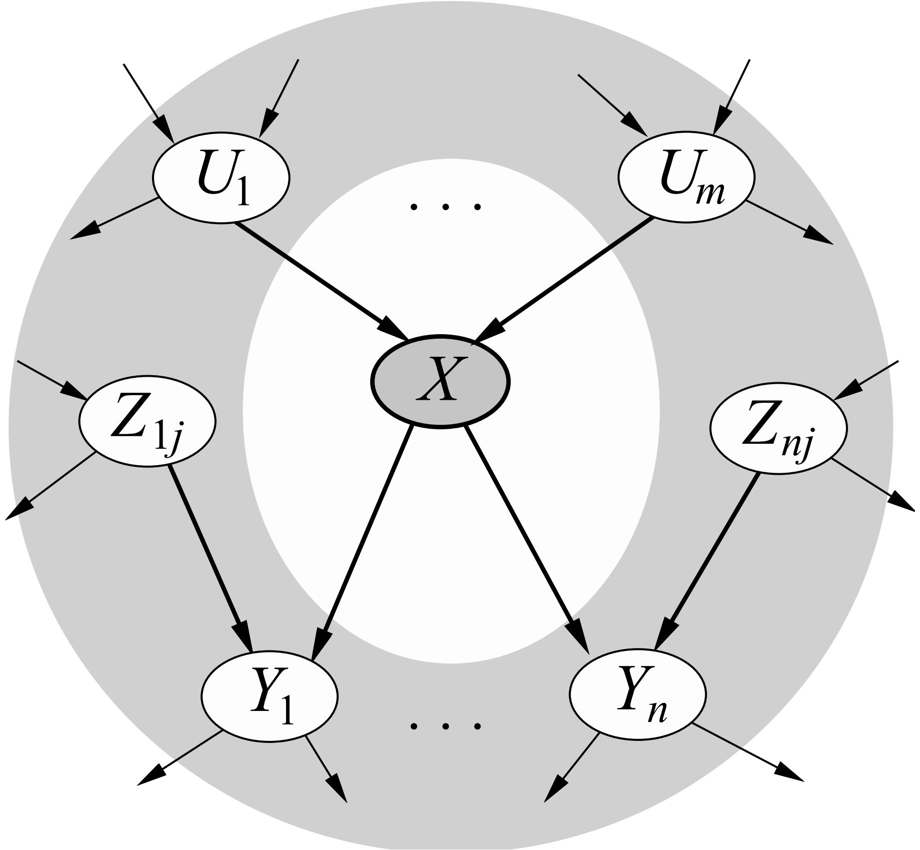
Conditional independence semantics

- ***Every variable is conditionally independent of its non-descendants given its parents***
- Conditional independence semantics <=> global semantics



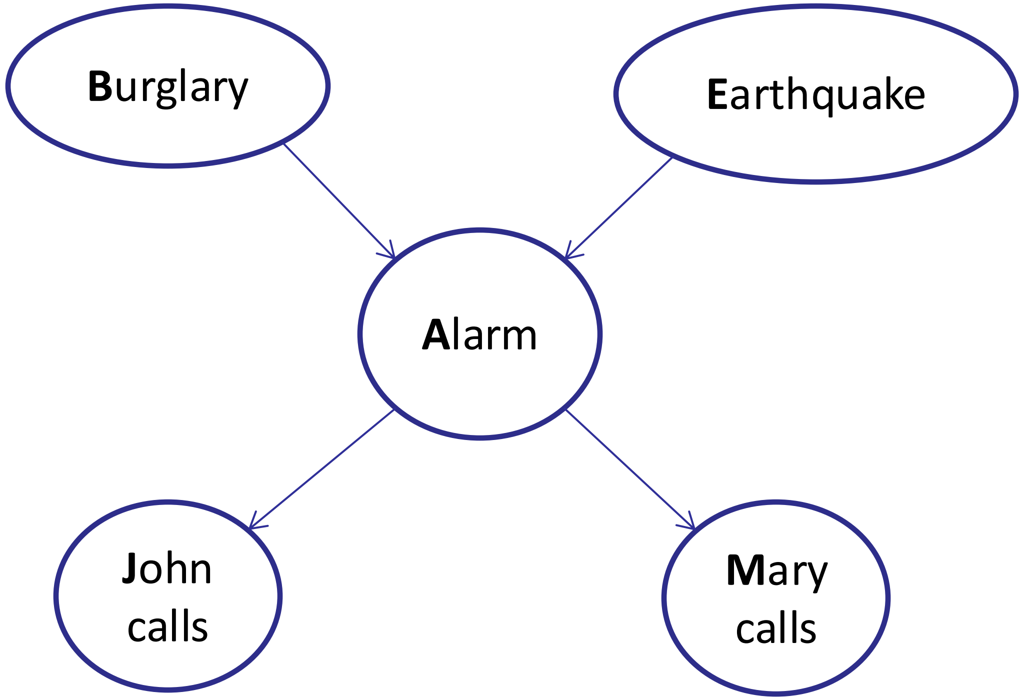
Markov blanket

- A variable's Markov blanket consists of parents, children, children's other parents
- ***Every variable is conditionally independent of all other variables given its Markov blanket***



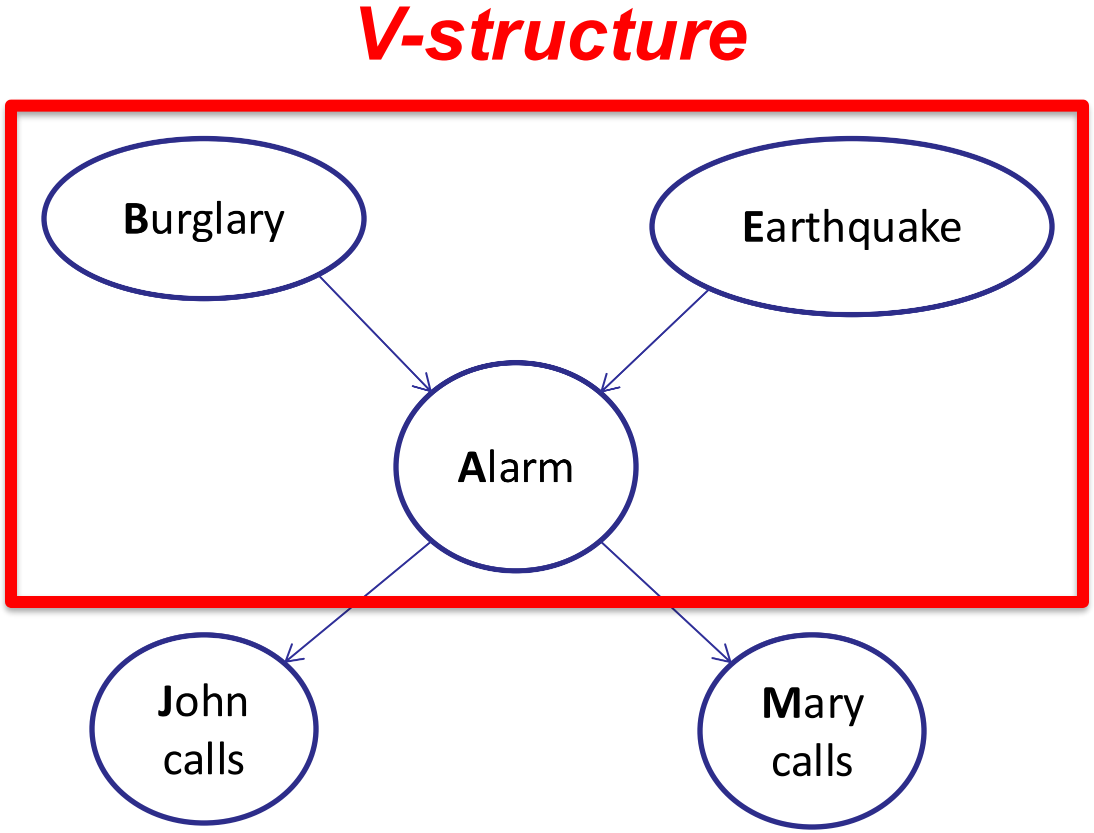
Example

- JohnCalls independent of Burglary given Alarm?
  - Yes
- JohnCalls independent of MaryCalls given Alarm?
  - Yes
- Burglary independent of Earthquake?
  - Yes

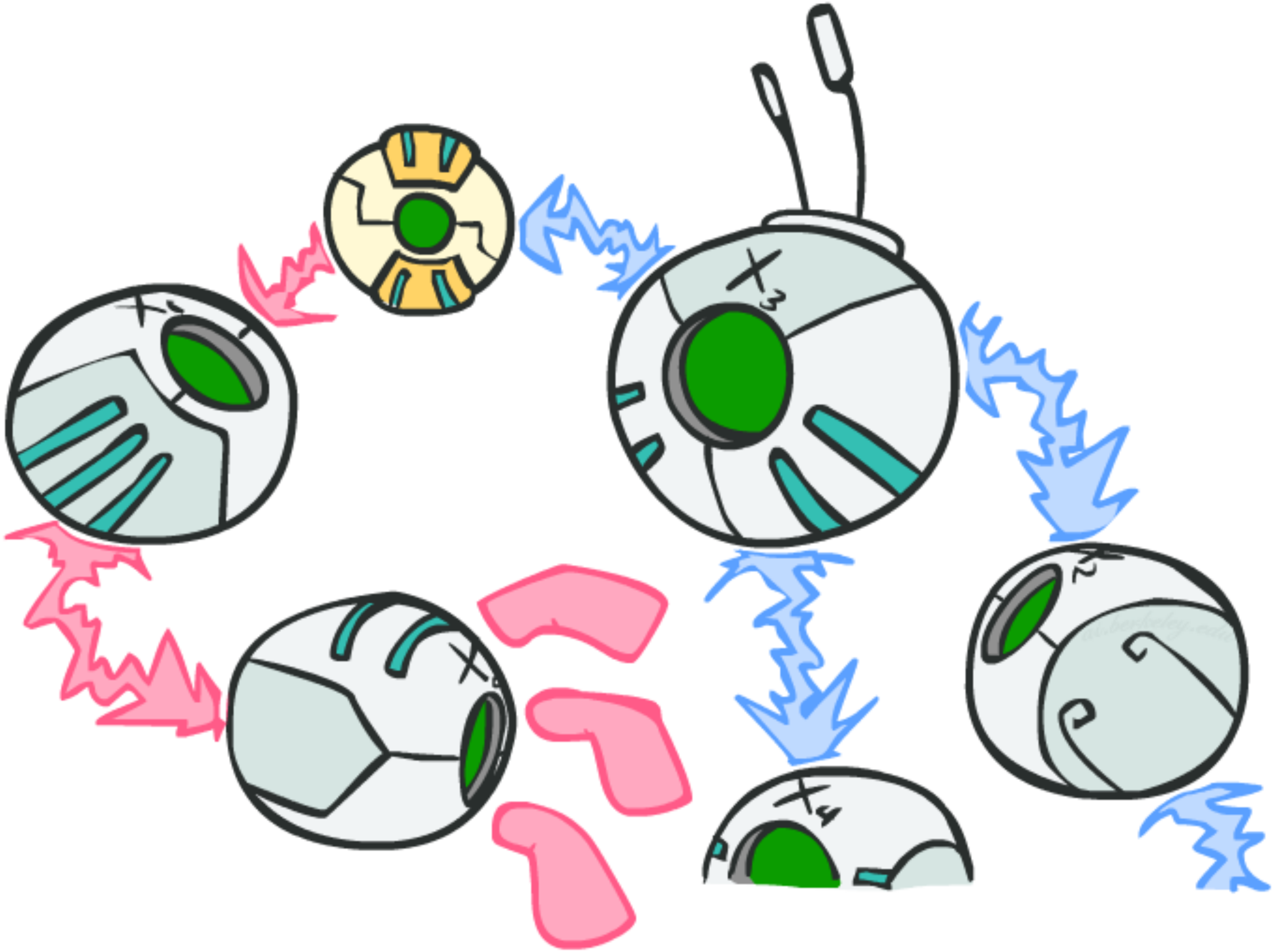


Example

- Burglary independent of Earthquake given Alarm?
  - NO!
  - Given that the alarm has sounded, both burglary and earthquake become more likely
  - But if we then learn that a burglary has happened, the alarm is ***explained away*** and the probability of earthquake drops back
- Burglary independent of Earthquake given JohnCalls?
- Any simple algorithm to determine conditional independence?

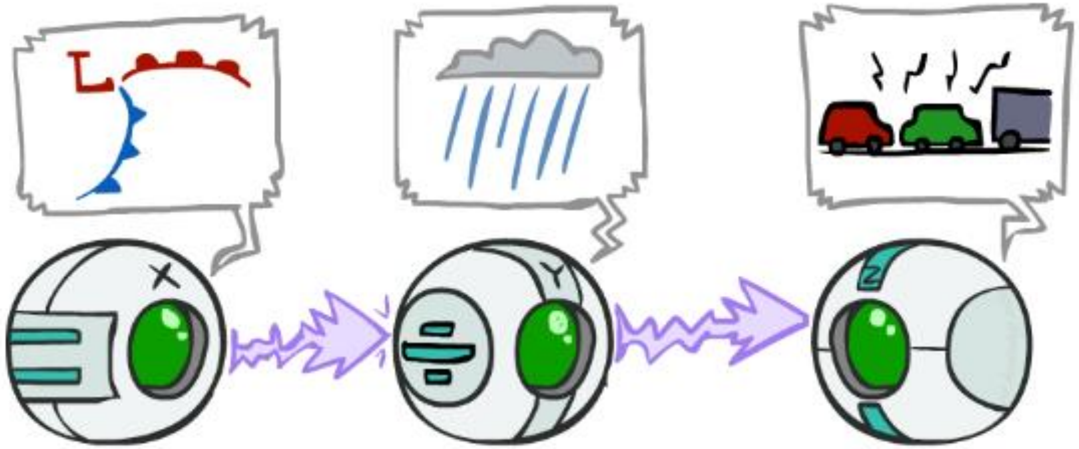


D-separation



Causal Chains

- This configuration is a “causal chain”



Global semantics:

$$P(x,y,z) = P(x)P(y|x)P(z|y)$$

- Guaranteed X independent of Z ? *No!*
- Guaranteed X independent of Z given Y?

$$P(z|x,y) = \frac{P(x,y,z)}{P(x,y)}$$

$$= \frac{P(x)P(y|x)P(z|y)}{P(x)P(y|x)}$$

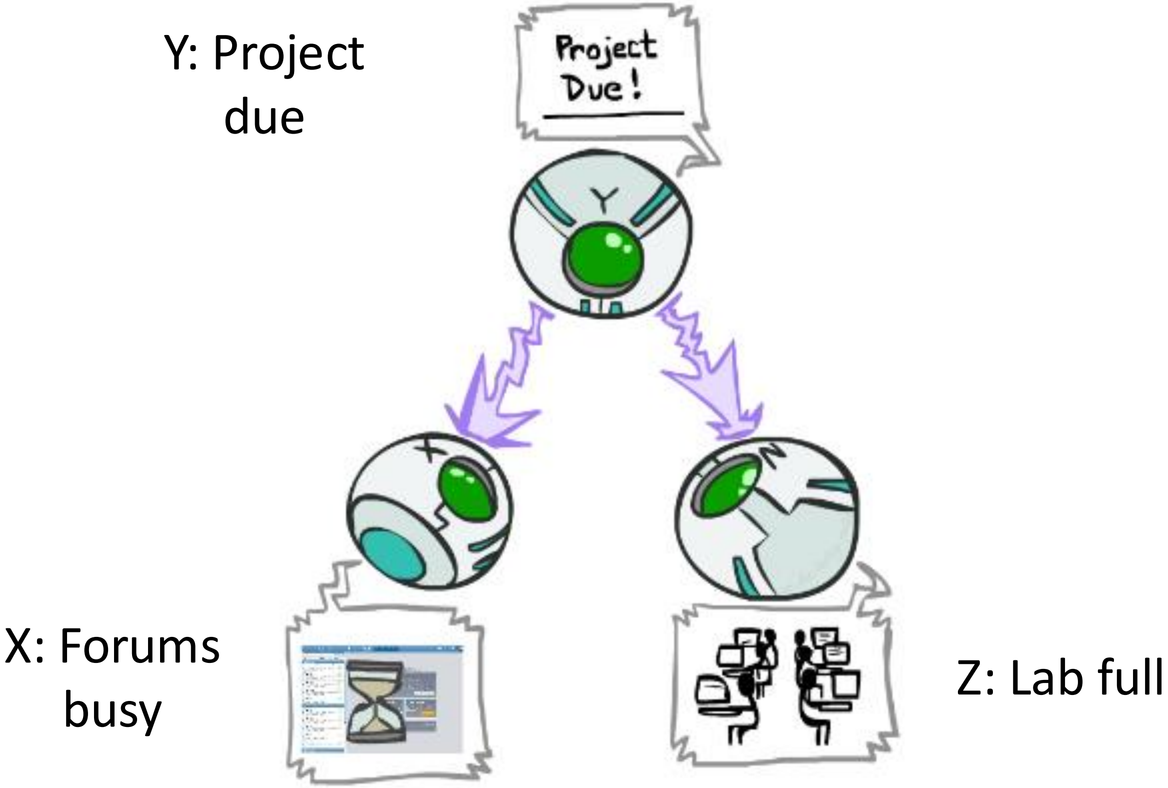
$$= P(z|y)$$

Yes!

- Evidence along the chain “blocks” the influence

Common Cause

- This configuration is a "common cause"



Global semantics:

$$P(x,y,z) = P(y)P(x|y)P(z|y)$$

- Guaranteed X independent of Z ? *No!*
- Guaranteed X and Z independent given Y?

$$P(z|x,y) = \frac{P(x,y,z)}{P(x,y)}$$

$$= \frac{P(y)P(x|y)P(z|y)}{P(y)P(x|y)}$$

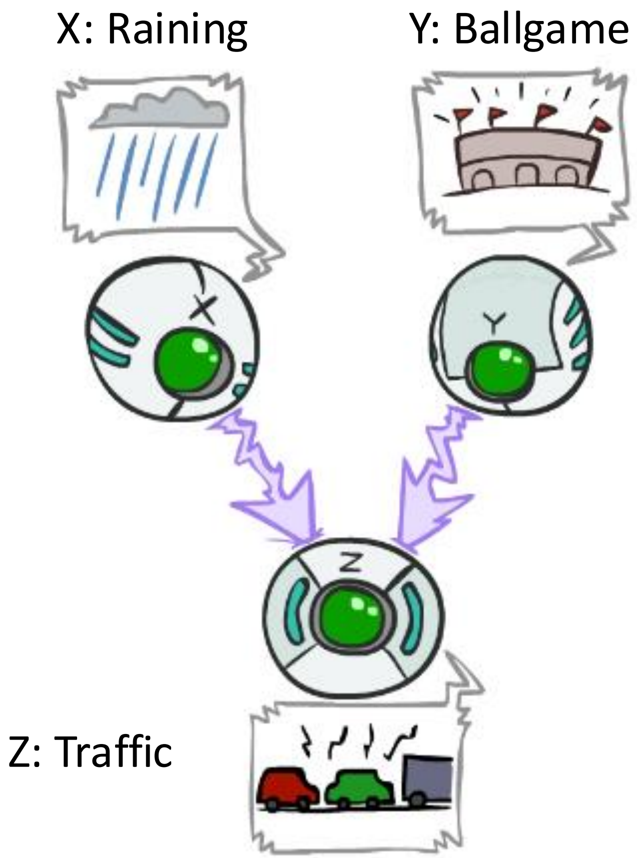
$$= P(z|y)$$

Yes!

- Observing the cause blocks influence between effects.

Common Effect

- Last configuration: two causes of one effect (v-structures)



- Are X and Y independent?
  - *Yes*: the ballgame and the rain cause traffic, but they are not correlated
  - Still need to prove they must be (try it!)
- Are X and Y independent given Z?
  - *No*: seeing traffic puts the rain and the ballgame in competition as explanation.
- This is backwards from the other cases
  - Observing an effect activates influence between possible causes.

D-separation - the General Case

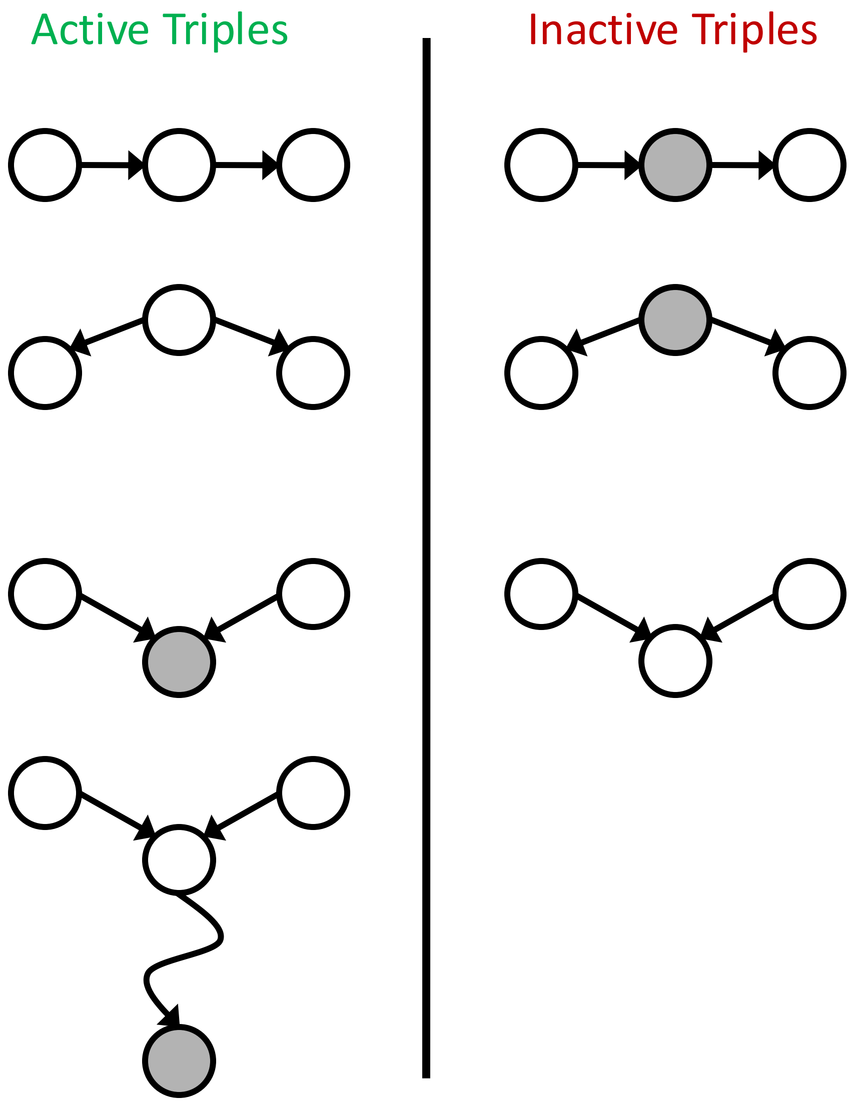
Active / Inactive Paths

- Question: X, Y, Z are non-intersecting subsets of nodes. Are X and Y conditionally independent given Z?

- A triple is active in the following three cases
  - Causal chain A → B → C where B is unobserved (either direction)
  - Common cause A ← B → C where B is unobserved
  - Common effect (aka v-structure)
    A → B ← C where B *or one of its descendents* is observed

- A path is active if each triple along the path is active
- A path is blocked if it contains a single inactive triple

- If all paths from X to Y are blocked, then X is said to be "**d-separated**" from Y by Z
- If d-separated, then X and Y are conditionally independent given Z

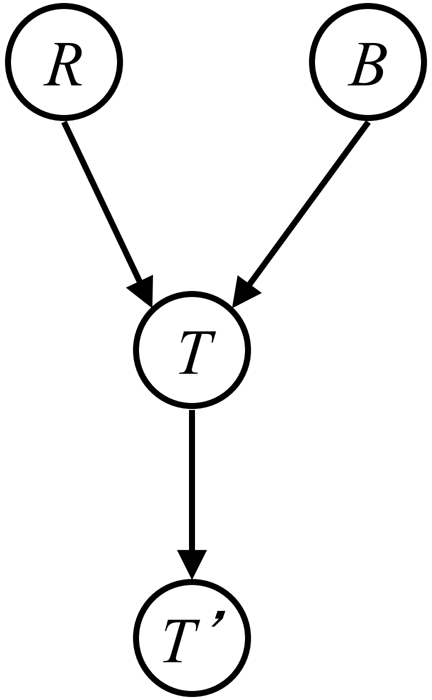


Example

$R \perp\!\!\!\perp B$ *Yes*

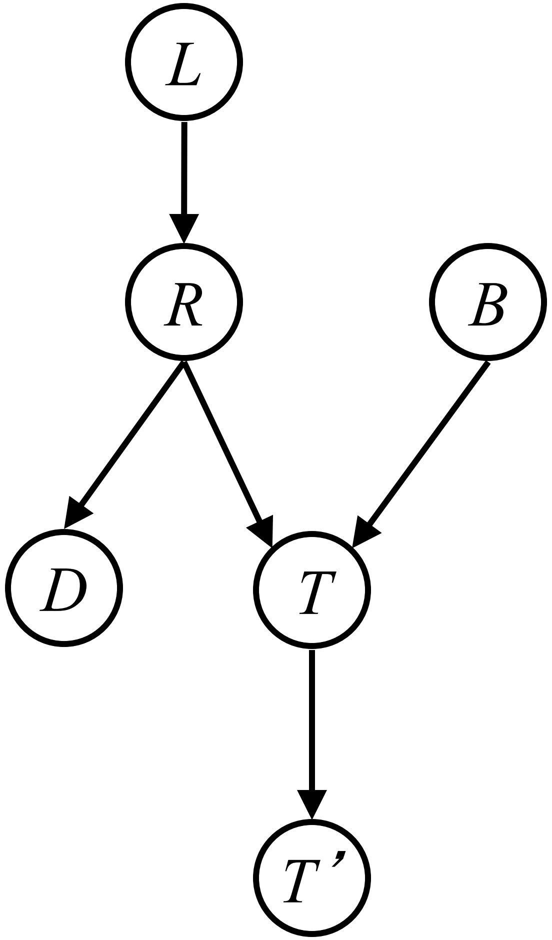
$R \perp\!\!\!\perp B | T$

$R \perp\!\!\!\perp B | T'$



Example

$L \perp\!\!\!\perp T' \mid T$	*Yes*
$L \perp\!\!\!\perp B$	*Yes*
$L \perp\!\!\!\perp B \mid T$	
$L \perp\!\!\!\perp B \mid T'$	
$L \perp\!\!\!\perp B \mid T, R$	*Yes*



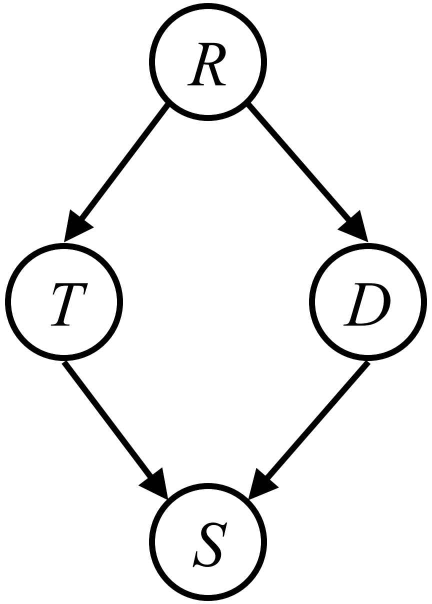
Example

- Variables:
  - R: Raining
  - T: Traffic
  - D: Roof drips
  - S: I'm sad
- Questions:

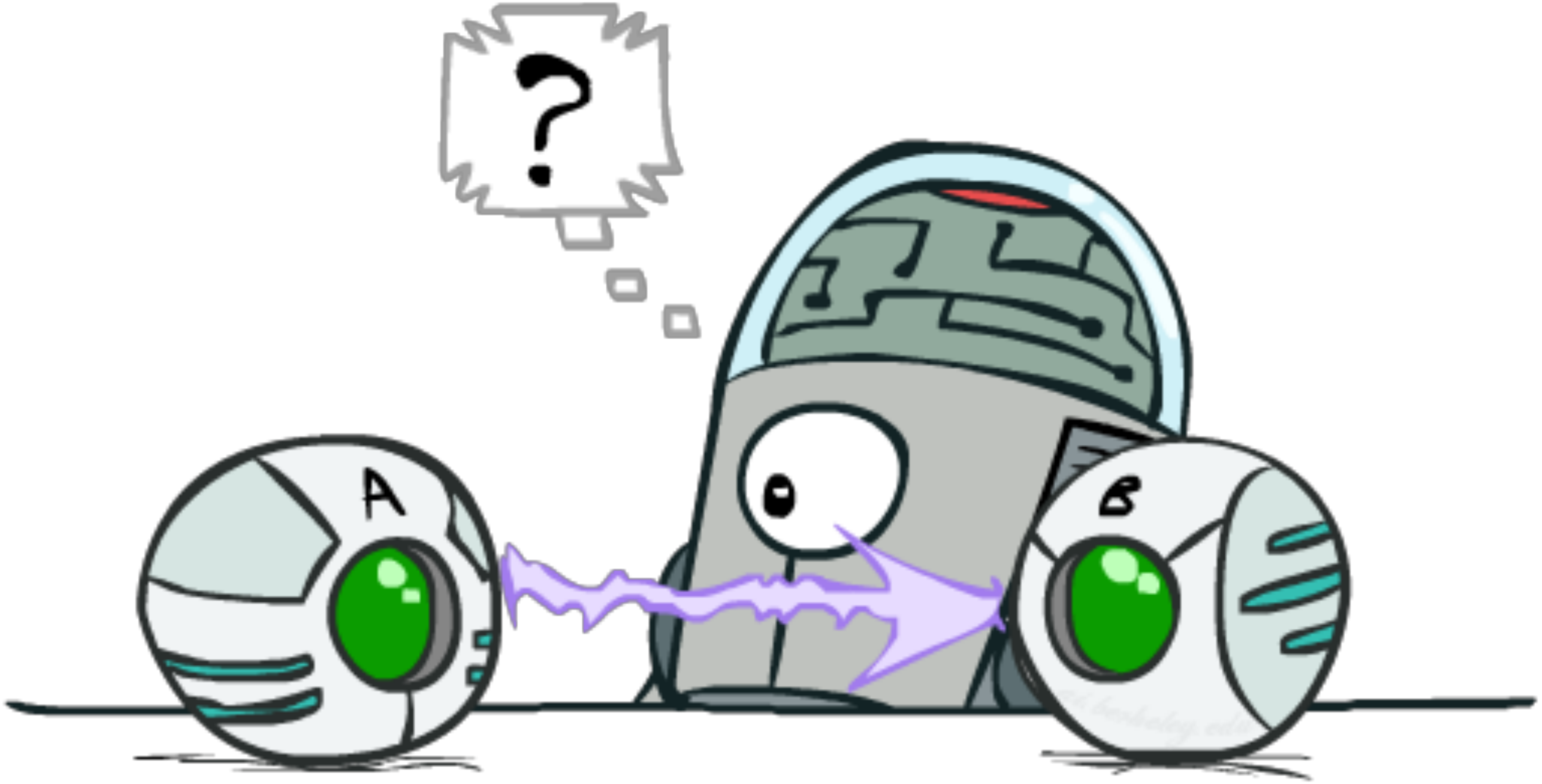
$$T \perp\!\!\!\perp D$$

$$T \perp\!\!\!\perp D | R$$ *Yes*

$$T \perp\!\!\!\perp D | R, S$$

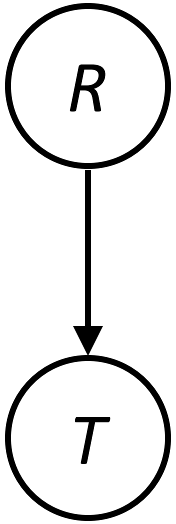


Node Ordering



Example: Traffic

- Causal direction



$P(R)$

+r	1/4
-r	3/4

$P(T|R)$

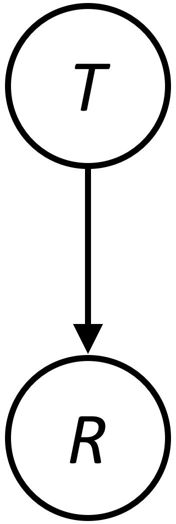
+r	+t	3/4
	-t	1/4
-r	+t	1/2
	-t	1/2

$P(T, R)$

+r	+t	3/16
+r	-t	1/16
-r	+t	6/16
-r	-t	6/16

Example: Reverse Traffic

- Reverse causality?



$P(T)$

+t	9/16
-t	7/16

$P(R|T)$

+t	+r	1/3
	-r	2/3

-t	+r	1/7
	-r	6/7

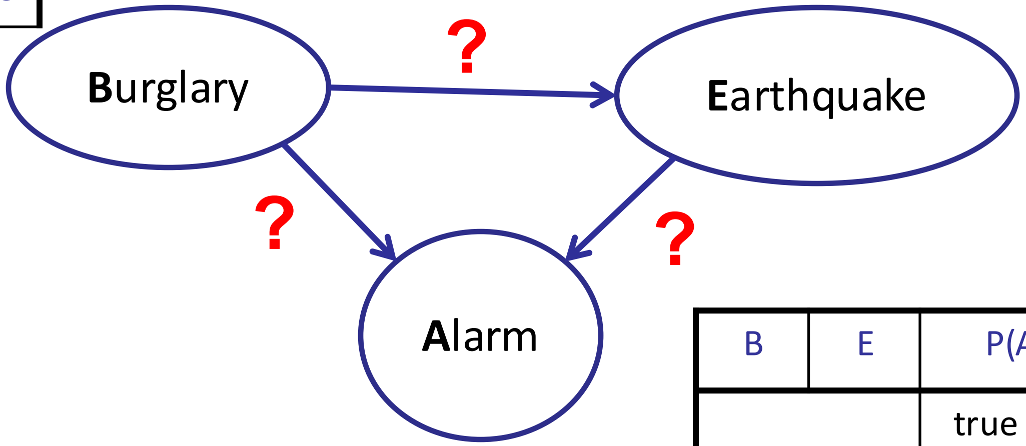
$P(T, R)$

+r	+t	3/16
+r	-t	1/16
-r	+t	6/16
-r	-t	6/16

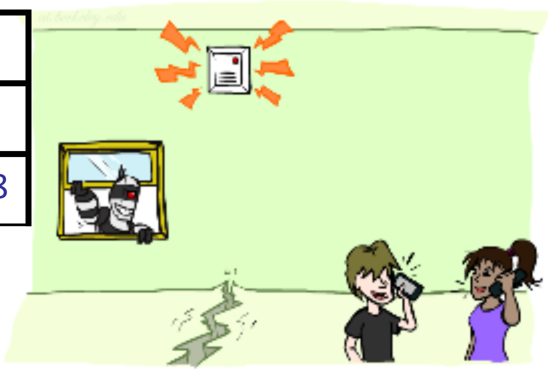
Example: Burglary

- Burglary
- Earthquake
- Alarm

P(B)	
true	false
0.001	0.999



P(E)	
true	false
0.002	0.998



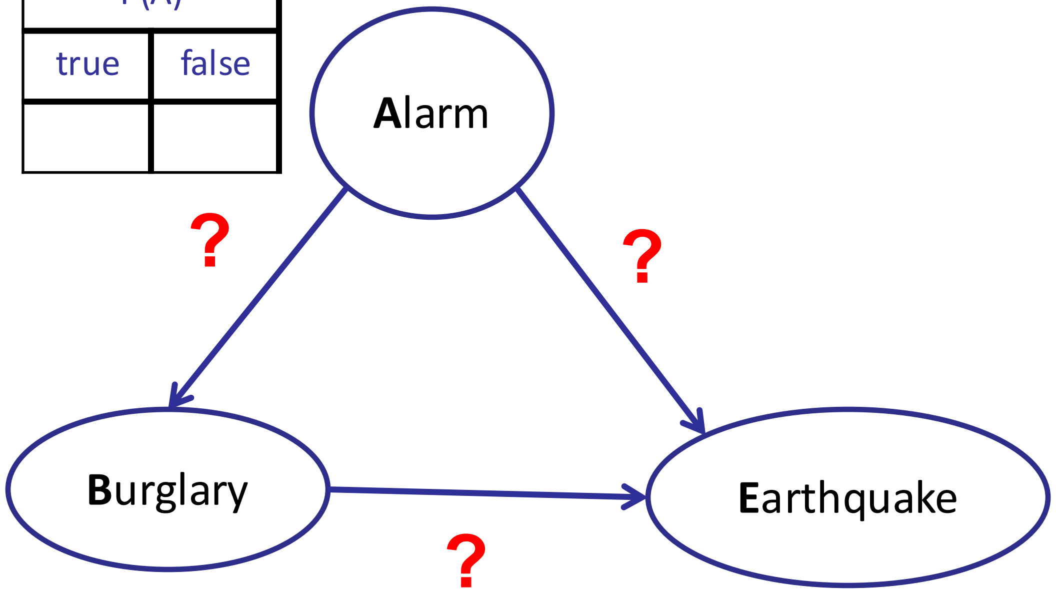
B	E	P(A|B,E)	
		true	false
true	true	0.95	0.05
true	false	0.94	0.06
false	true	0.29	0.71
false	false	0.001	0.999

2 edges, 6 free parameters

Example: Burglary

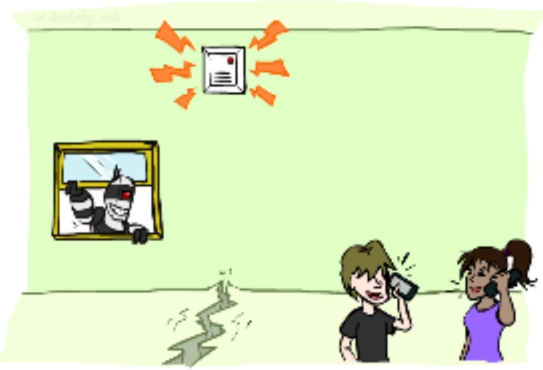
- Alarm
- Burglary
- Earthquake

P(A)	
true	false



A	P(B|A)	
	true	false
true		
false		

A	B	P(E|A,B)	
		true	false
true	true		
true	false		
false	true		
false	false		



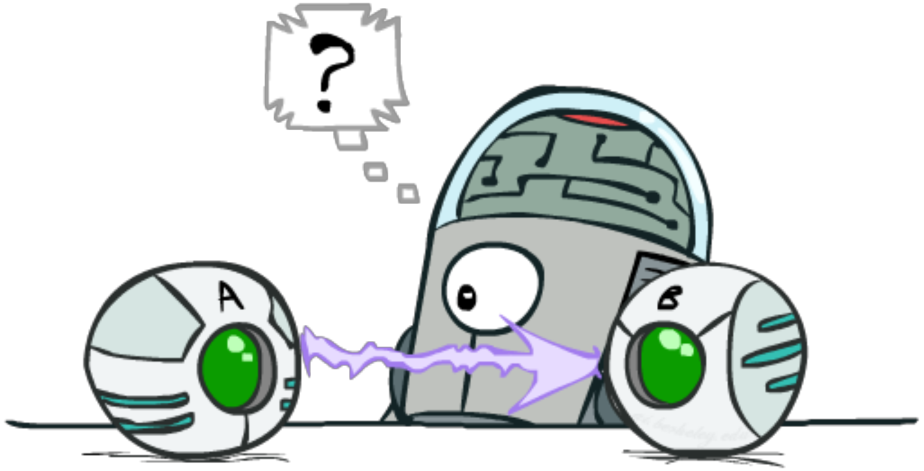
3 edges, 7 free parameters

Causality?

- When Bayes nets reflect the true causal patterns:
  (e.g., Burglary, Earthquake, Alarm)
  - Often simpler (fewer parents, fewer parameters)
  - Often easier to assess probabilities
  - Often more robust: e.g., changes in frequency of burglaries should not affect the rest of the model!

- BNs need not actually be causal
  - Sometimes no causal net exists over the domain (especially if variables are missing)
  - E.g. consider the variables *Traffic* and *Umbrella*
  - End up with arrows that reflect correlation, not causation

- What do the arrows really mean?
  - Topology may happen to encode causal structure
  - Topology really encodes conditional independence:
    $P(X_i \mid X_1,\ldots,X_{i-1}) = P(X_i \mid Parents(X_i))$



Example Application: Topic Modeling

[Containing some materials from slides by David Blei at Princeton and by Hongning Wang at UVa]

Introduction

- A large body of text available online
  - It is difficult to find and discover what we need.
- Topic models
  - Approaches to discovering the main themes of a large unstructured collection of documents
  - Can be used to automatically organize, understand, search, and summarize large electronic archives
  - Latent Dirichlet Allocation (LDA) is the most popular

Plate Notation

- Representation of repeated subgraphs in a Bayesian network

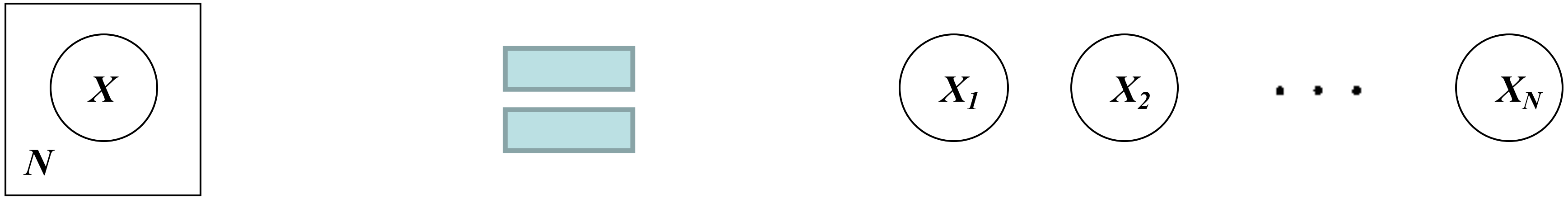


Plate Notation

- Representation of repeated subgraphs in a Bayesian network

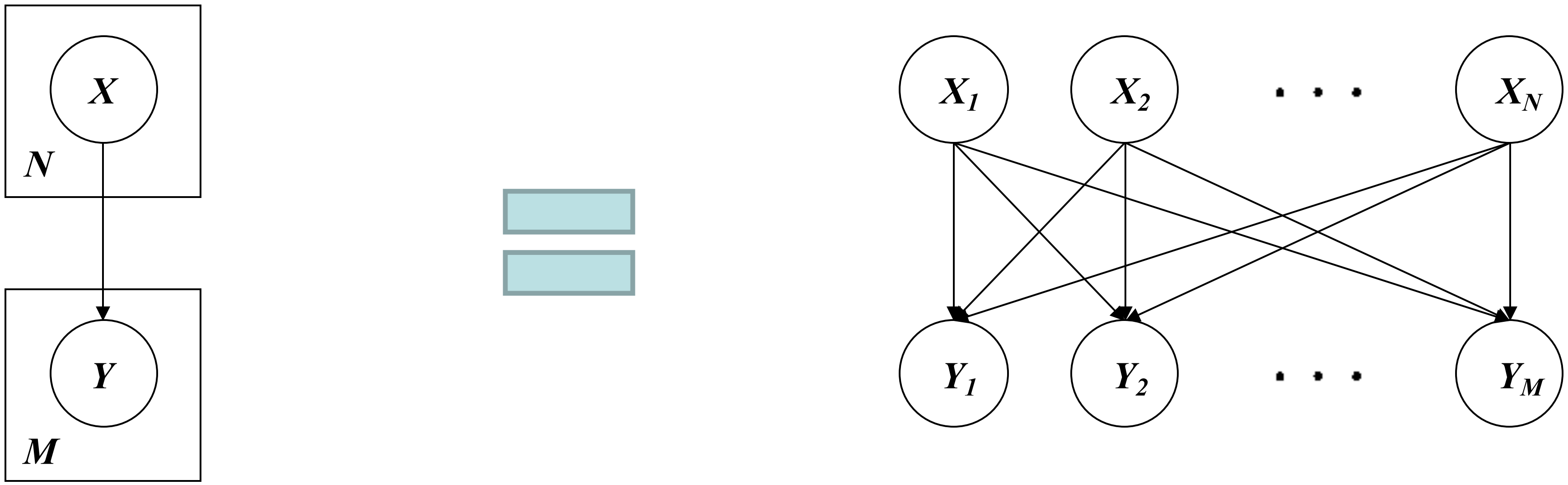
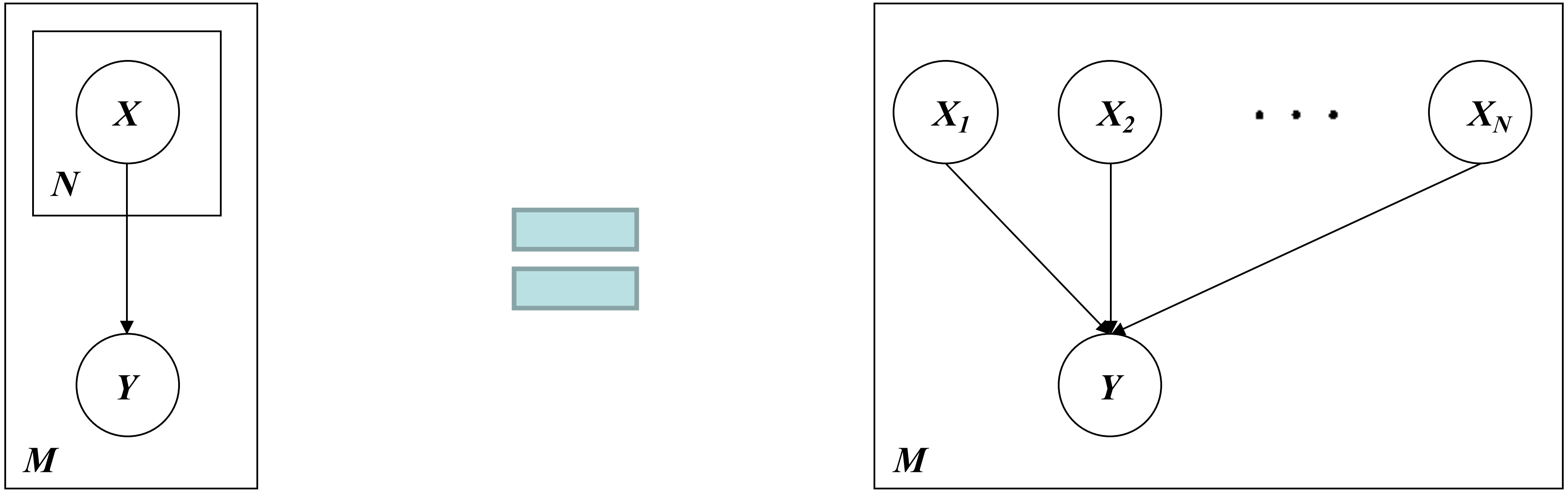
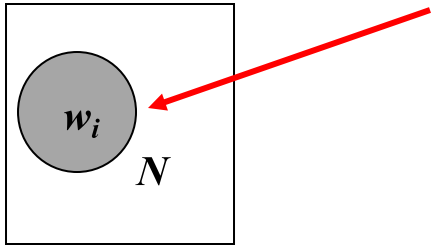


Plate Notation

- Representation of repeated subgraphs in a Bayesian network



How to generate a document



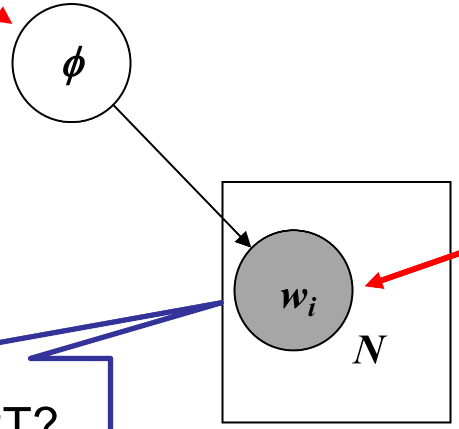
word *e.g.*, “how”, “to”, “get”, …

How to generate a document

distribution over words

e.g., [0.1, 0.3, 0.2, …]

how	0.1
to	0.3
get	0.2
…	…

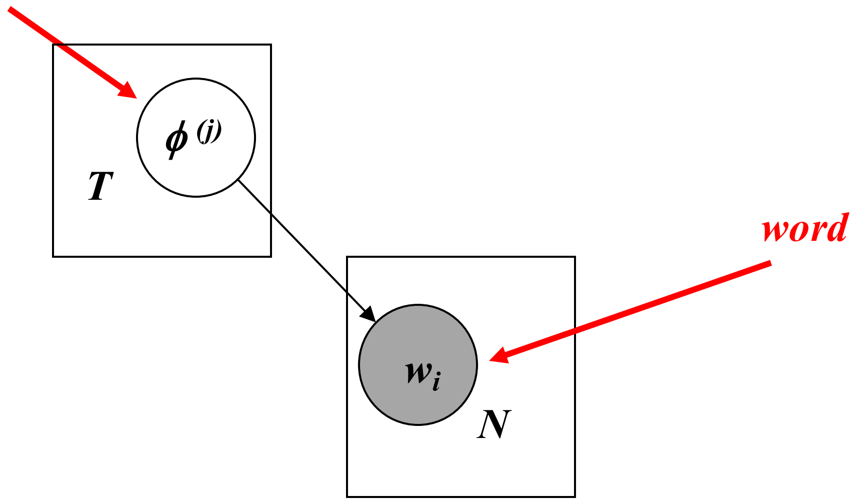


word

What is the CPT?

How to generate a document

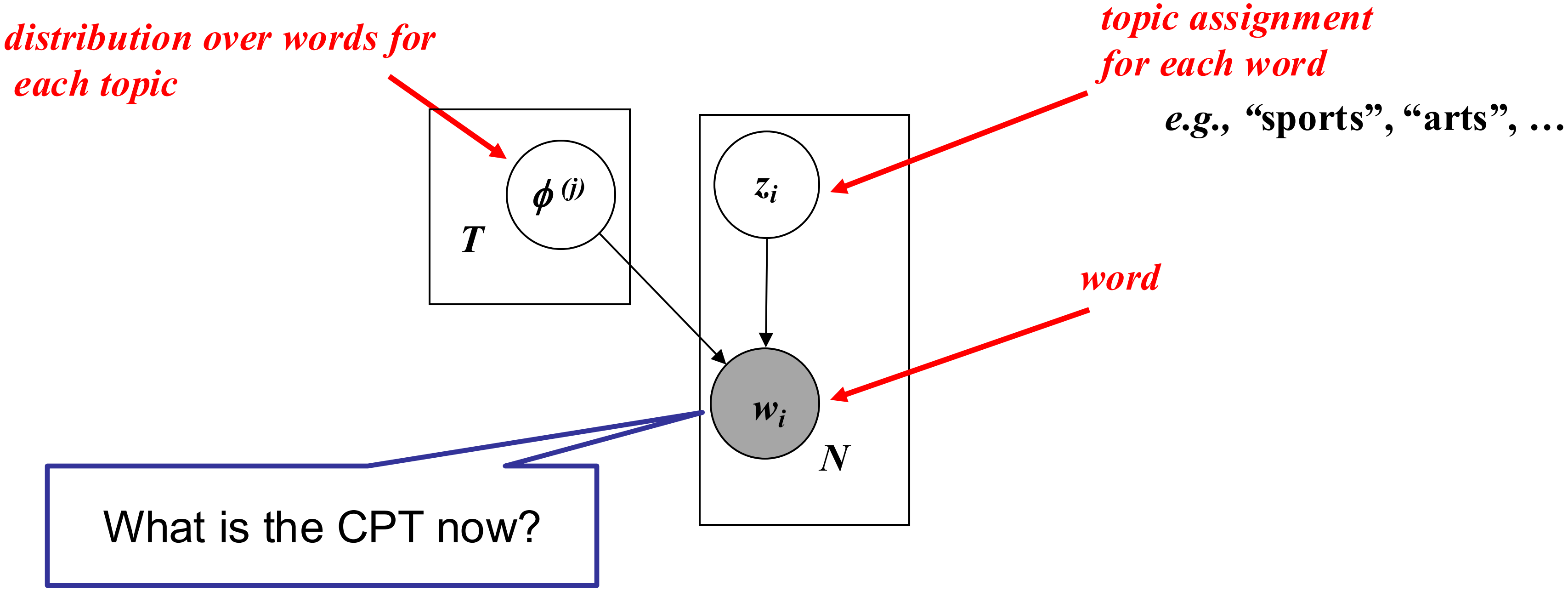
e.g., **sports: [0.1, 0.3, 0.2, …]**
arts: [0.2, 0.3, 0.1, …]
science: [0.3, 0.4, 0.1, …]
……



distribution over words for each topic

word

How to generate a document



distribution over words for each topic

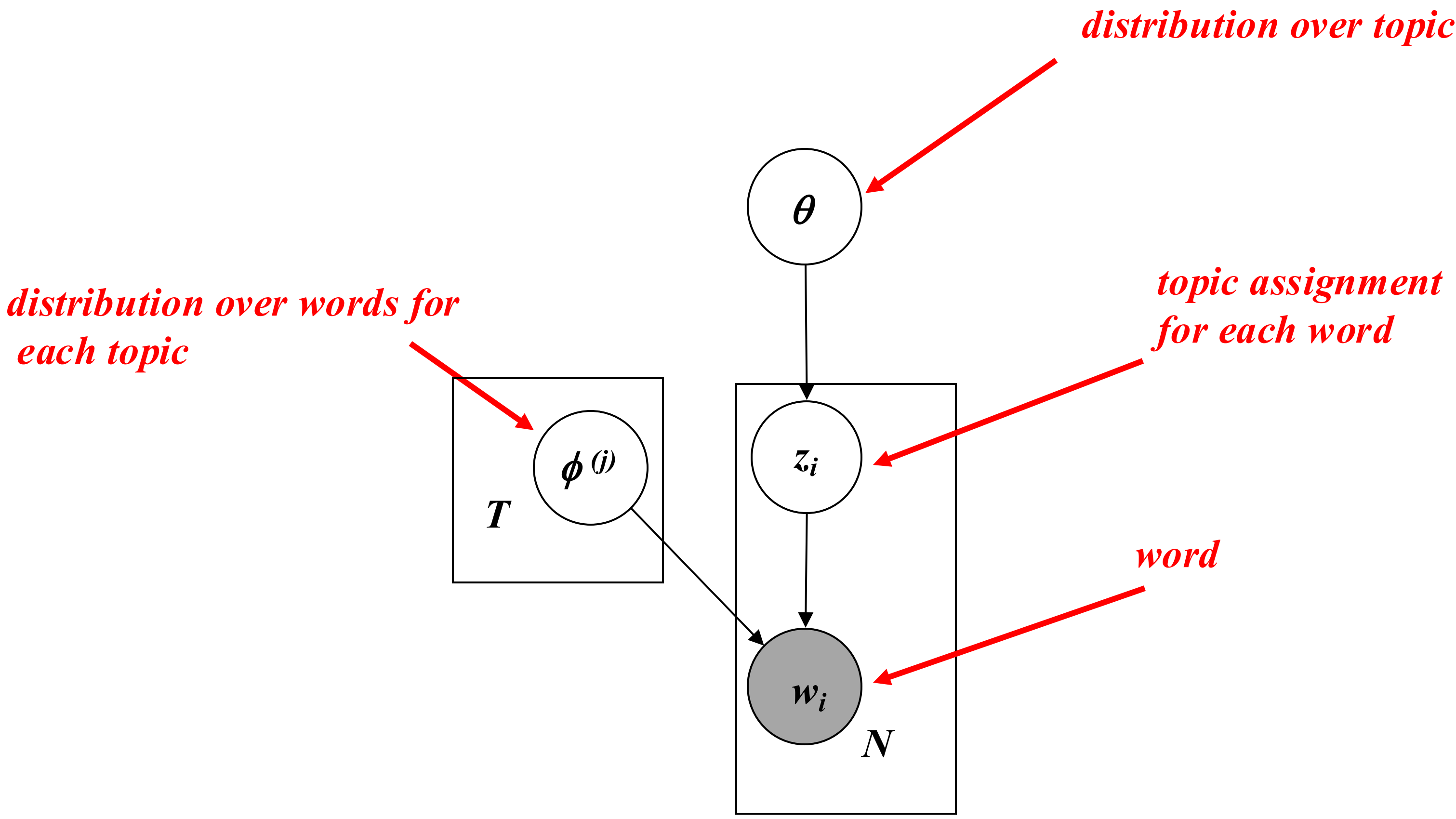
topic assignment for each word

***e.g.*, "sports", "arts", …**

word

What is the CPT now?

How to generate a document



distribution over topics ***e.g., [0.2, 0.3, 0.2, …]***

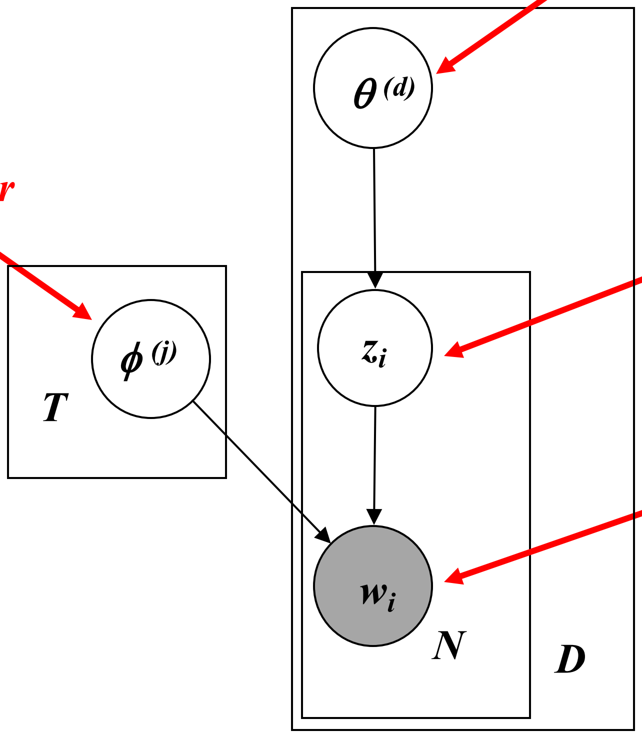
sports	**0.1**
arts	**0.3**
science	**0.2**
…	**…**

distribution over words for each topic

topic assignment for each word

word

How to generate documents



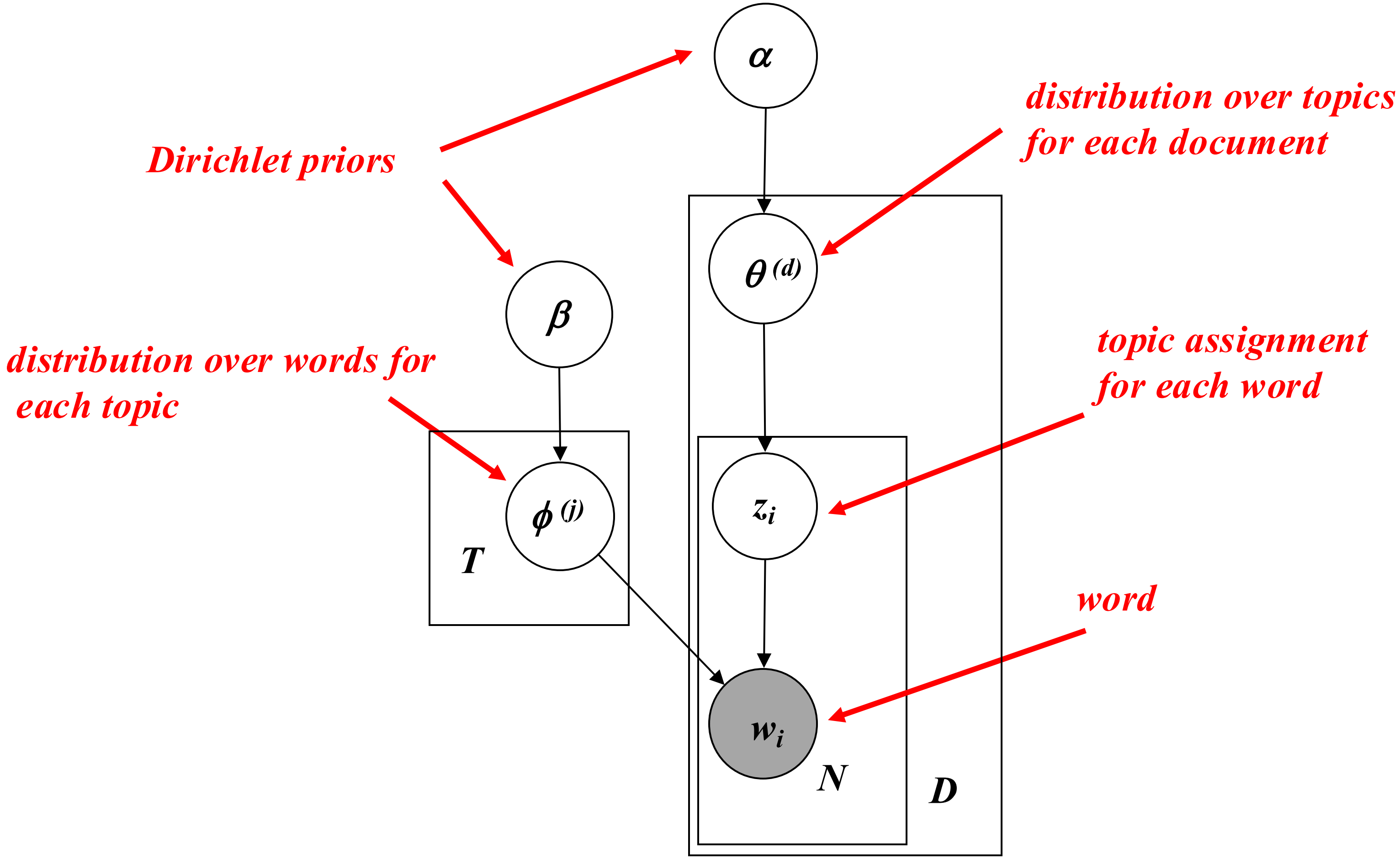
distribution over topics for each document

distribution over words for each topic

topic assignment for each word

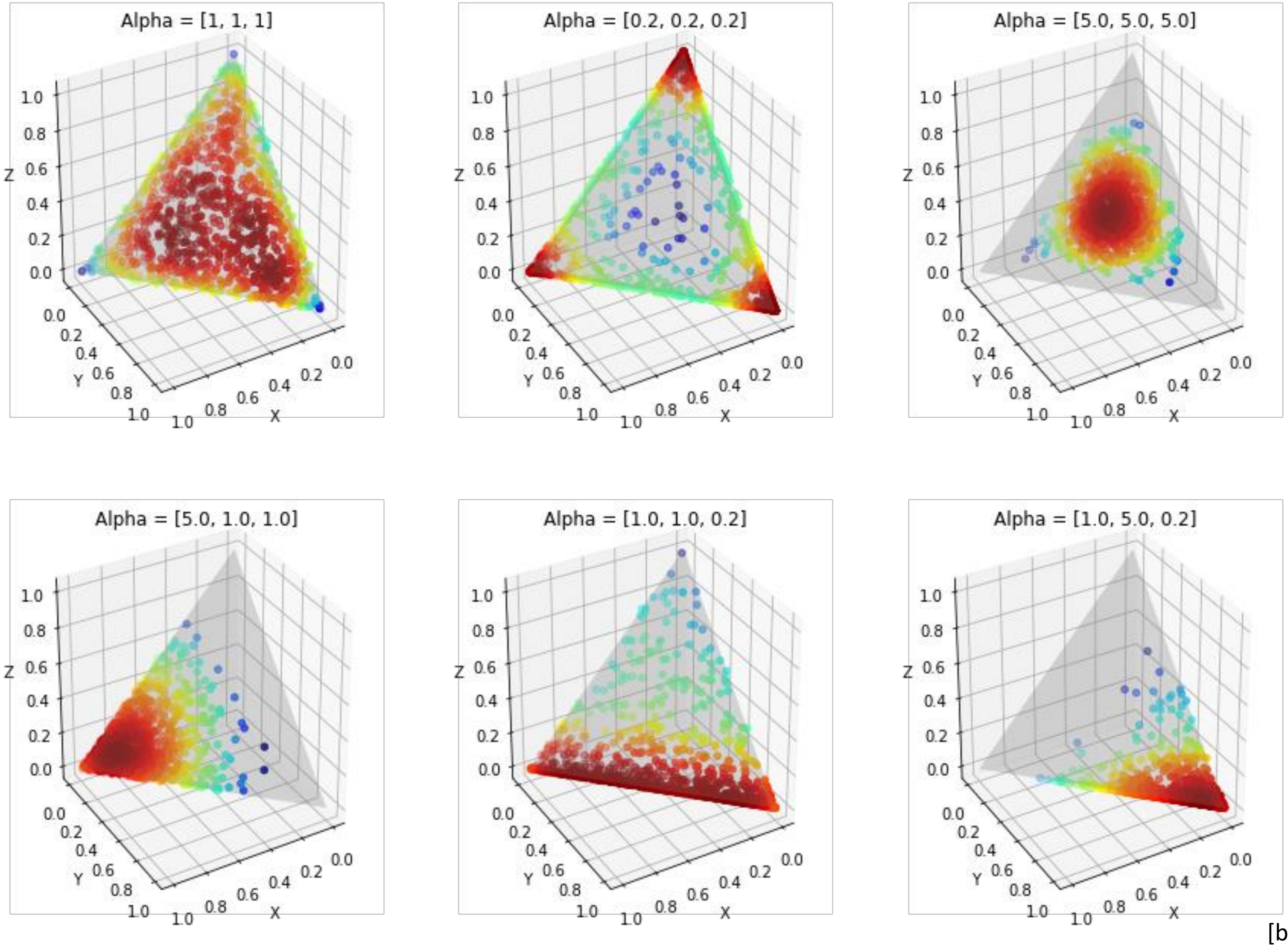
word

How to generate documents



α
Dirichlet priors
distribution over topics for each document
$\theta^{(d)}$
β
distribution over words for each topic
topic assignment for each word
$\phi^{(j)}$
T
z_i
word
w_i
N
D

Dirichlet Distribution



Alpha = [1, 1, 1]

Alpha = [0.2, 0.2, 0.2]

Alpha = [5.0, 5.0, 5.0]

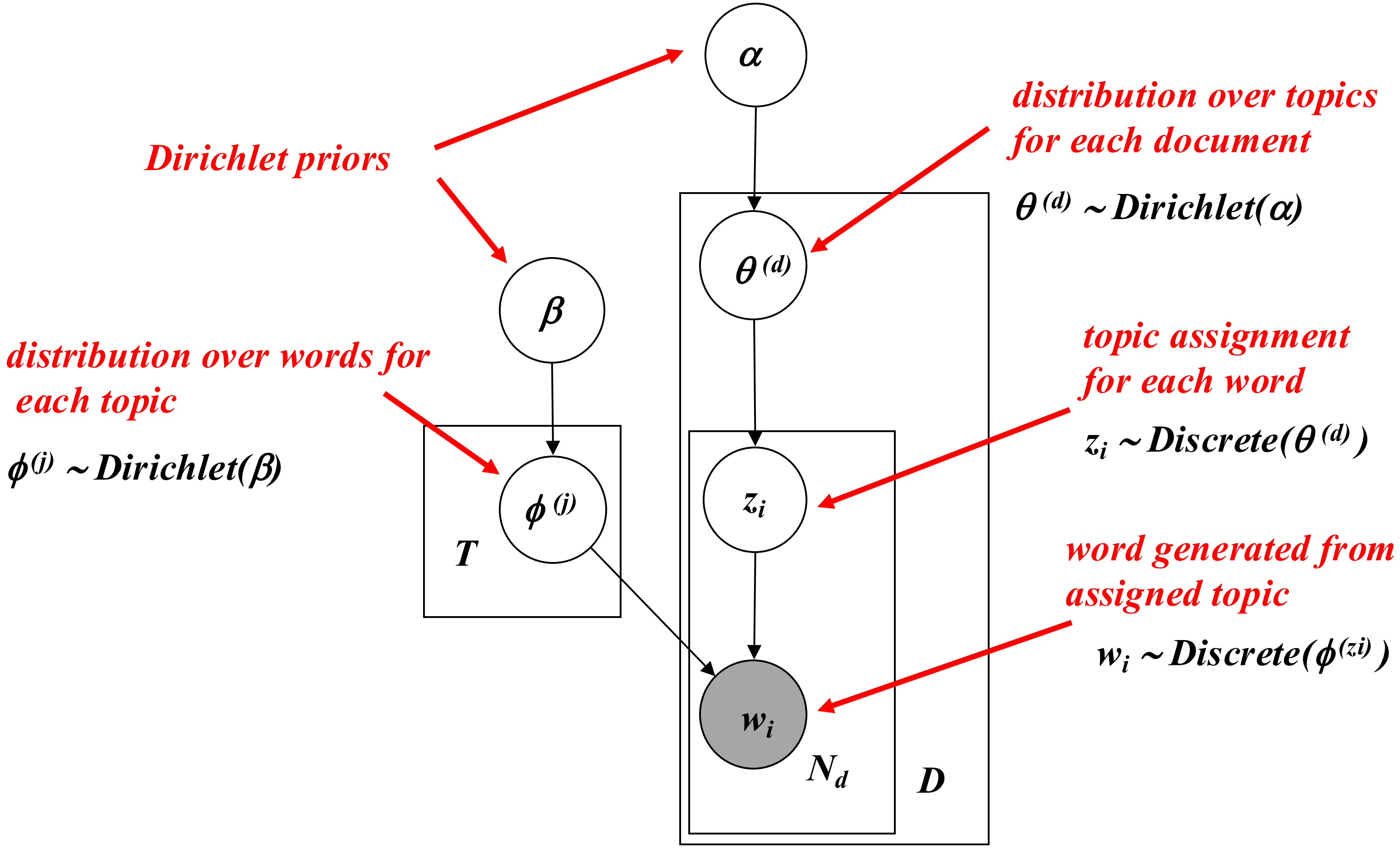
Alpha = [5.0, 1.0, 1.0]

Alpha = [1.0, 1.0, 0.2]

Alpha = [1.0, 5.0, 0.2]

[by Thushan Ganegedara at USydney]

Latent Dirichlet Allocation (LDA)



Dirichlet priors

distribution over topics for each document

$\theta^{(d)} \sim \text{Dirichlet}(\alpha)$

distribution over words for each topic

$\phi^{(j)} \sim \text{Dirichlet}(\beta)$

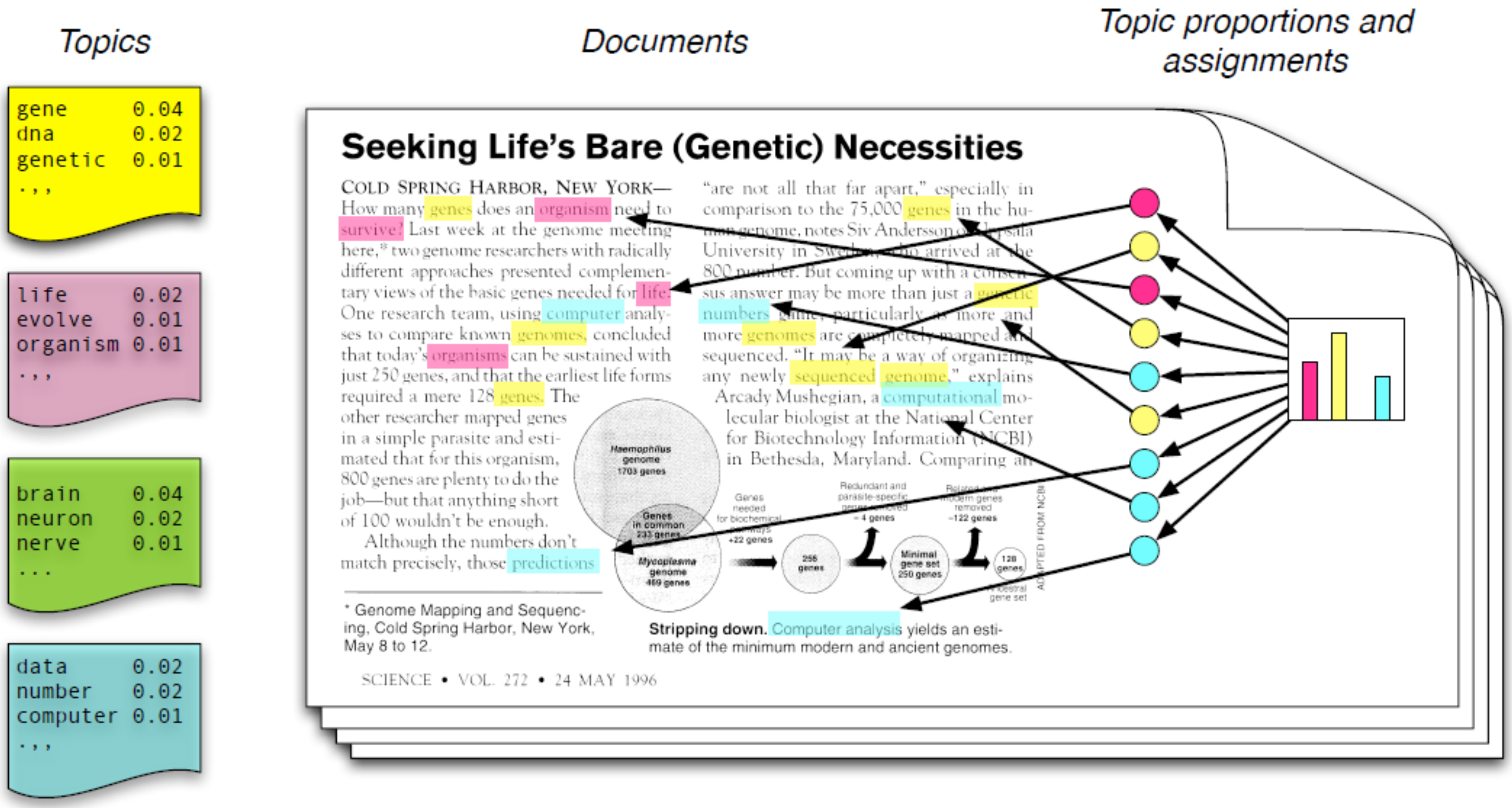
topic assignment for each word

$z_i \sim \text{Discrete}(\theta^{(d)})$

word generated from assigned topic

$w_i \sim \text{Discrete}(\phi^{(z_i)})$

Illustration



Topics

gene	0.04
dna	0.02
genetic	0.01
...	

life	0.02
evolve	0.01
organism	0.01
...	

brain	0.04
neuron	0.02
nerve	0.01
...	

data	0.02
number	0.02
computer	0.01
...	

Documents

Topic proportions and assignments

Seeking Life's Bare (Genetic) Necessities

COLD SPRING HARBOR, NEW YORK— How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

Although the numbers don't match precisely, those predictions "are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an

Haemophilus genome 1703 genes
Genes in common 233 genes
Mycoplasma genome 469 genes
Genes needed for biochemical pathways +22 genes
256 genes
Redundant and parasite-specific genes removed −4 genes
Minimal gene set 250 genes
Related and modern genes removed −122 genes
128 genes
Ancestral gene set
ADAPTED FROM NCBI

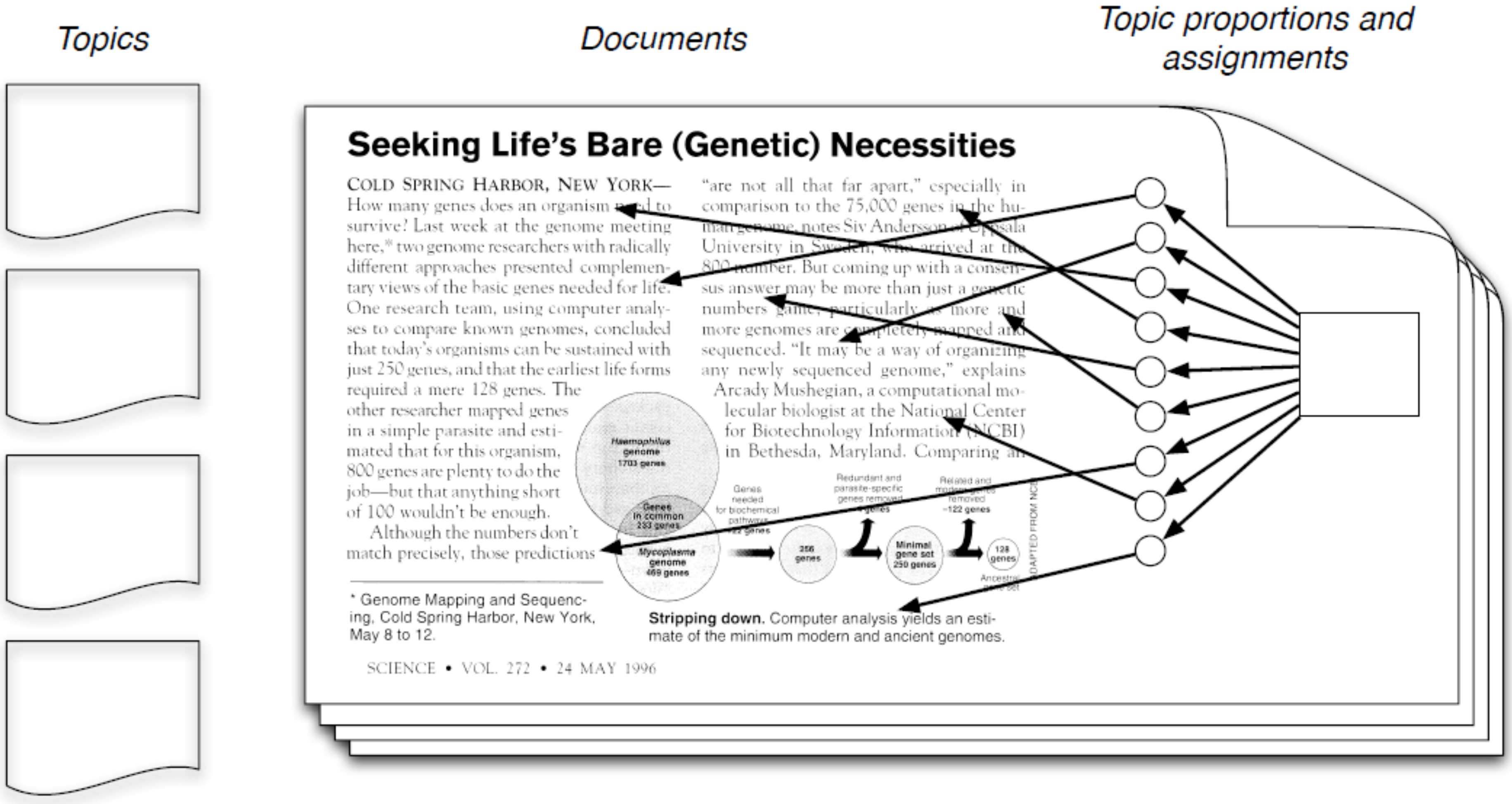
Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

SCIENCE • VOL. 272 • 24 MAY 1996

- Each **topic** is a distribution of words; each **document** is a mixture of corpus-wide topics; and each **word** is drawn from one of those topics.

Illustration



Topics
Documents
Topic proportions and assignments
Haemophilus genome 1703 genes
Genes in common 233 genes
Mycoplasma genome 469 genes
Genes needed for biochemical pathways +22 genes
256 genes
Redundant and parasite-specific genes removed -6 genes
Minimal gene set 250 genes
Related and modern genes removed -122 genes
128 genes
Ancestral gene set
ADAPTED FROM NCBI

Seeking Life's Bare (Genetic) Necessities

COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

Although the numbers don't match precisely, those predictions "are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an

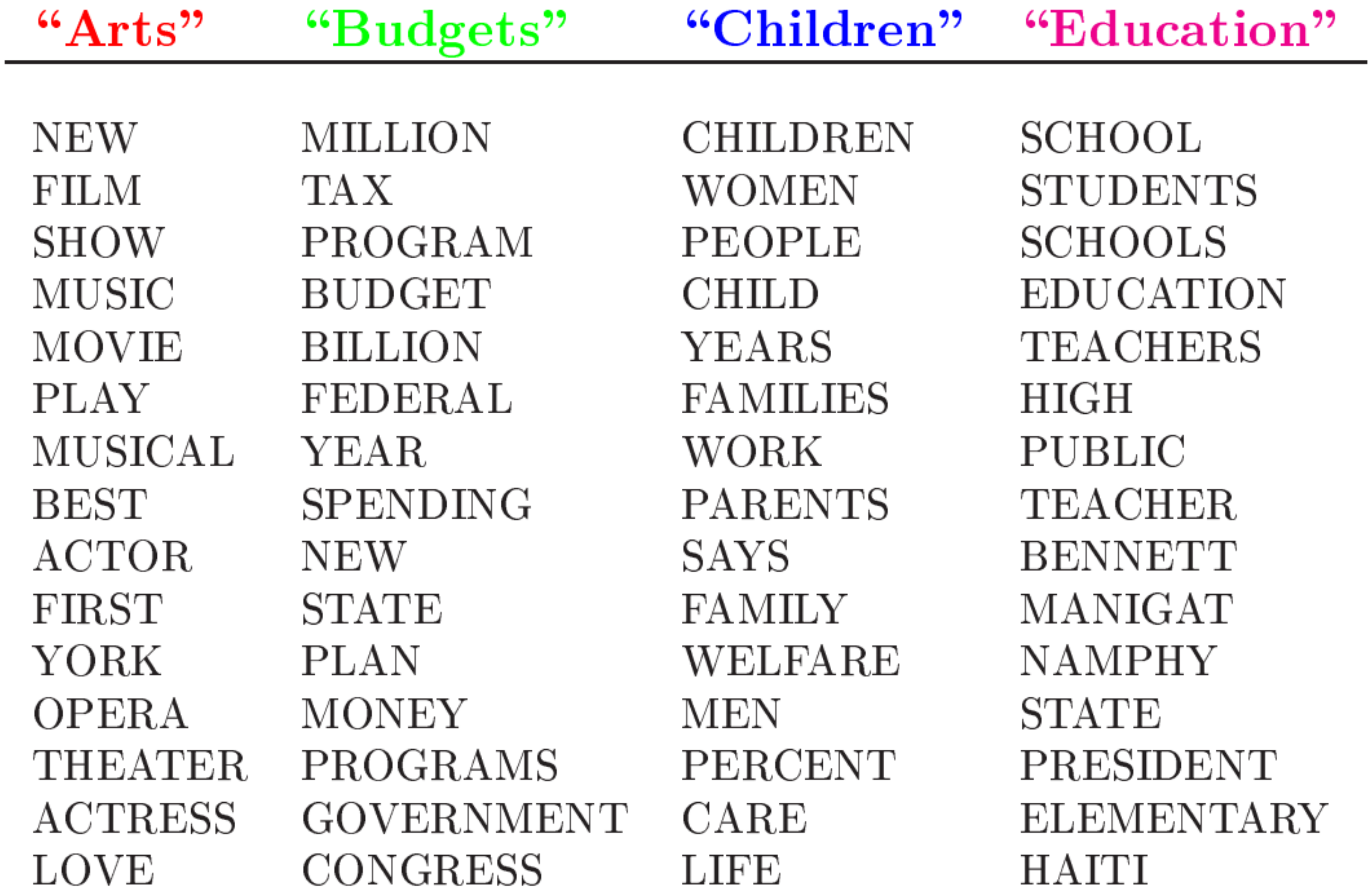
Stripping down. Computer analysis yields an estimate of the minimum modern and ancient genomes.

* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

SCIENCE • VOL. 272 • 24 MAY 1996

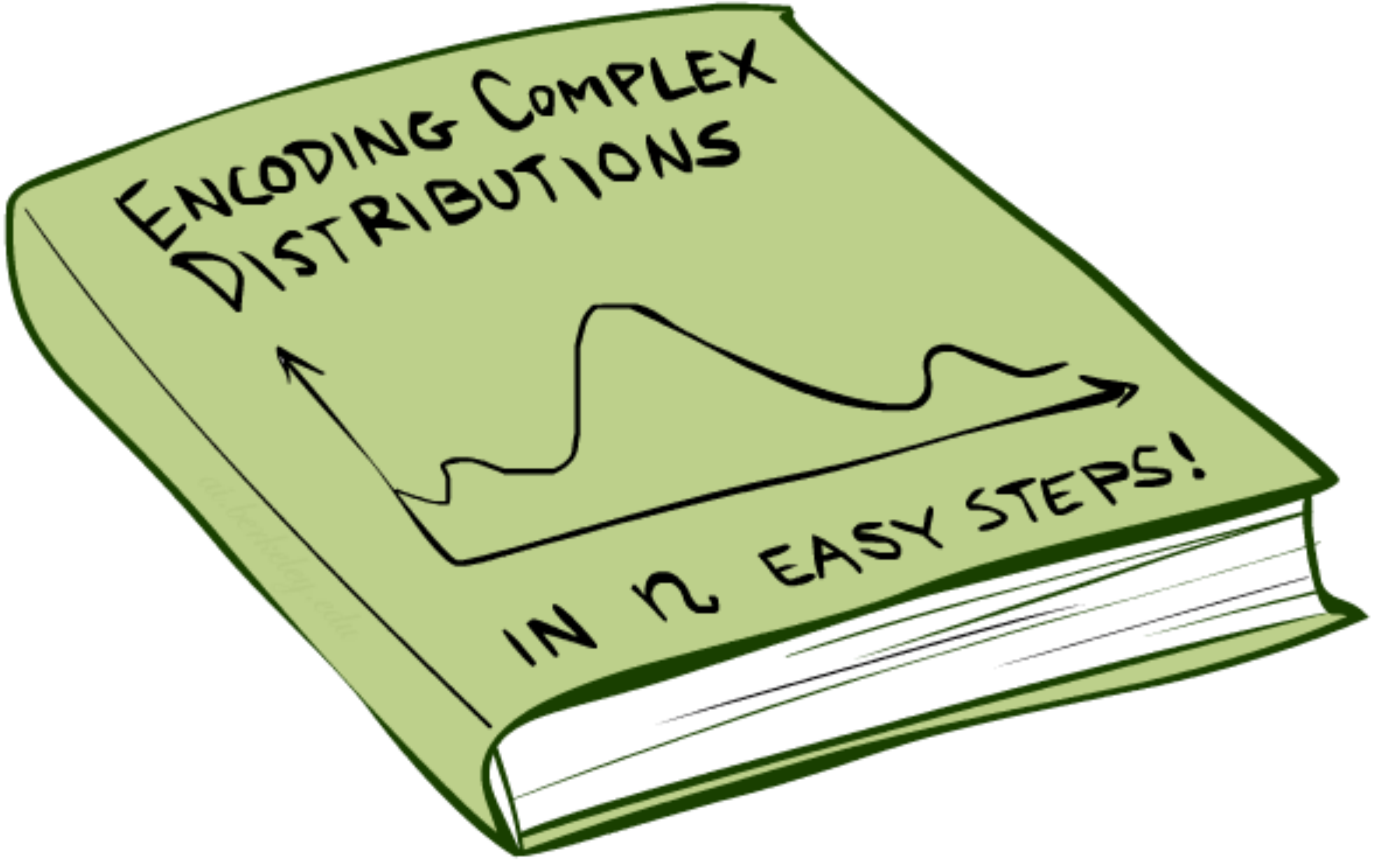
- In reality, we only observe documents. The other structures are hidden variables that must be inferred. (We will discuss inference later.)

Topics inferred by LDA



“Arts”	“Budgets”	“Children”	“Education”
NEW	MILLION	CHILDREN	SCHOOL
FILM	TAX	WOMEN	STUDENTS
SHOW	PROGRAM	PEOPLE	SCHOOLS
MUSIC	BUDGET	CHILD	EDUCATION
MOVIE	BILLION	YEARS	TEACHERS
PLAY	FEDERAL	FAMILIES	HIGH
MUSICAL	YEAR	WORK	PUBLIC
BEST	SPENDING	PARENTS	TEACHER
ACTOR	NEW	SAYS	BENNETT
FIRST	STATE	FAMILY	MANIGAT
YORK	PLAN	WELFARE	NAMPHY
OPERA	MONEY	MEN	STATE
THEATER	PROGRAMS	PERCENT	PRESIDENT
ACTRESS	GOVERNMENT	CARE	ELEMENTARY
LOVE	CONGRESS	LIFE	HAITI

Markov Networks



Markov Networks

- A Bayesian network encodes a joint distribution with a directed acyclic graph
  - A CPT captures uncertainty between a node and its parents

- A Markov network (or Markov random field) encodes a joint distribution with an undirected graph
  - A potential function captures uncertainty between a clique of nodes

Markov Networks

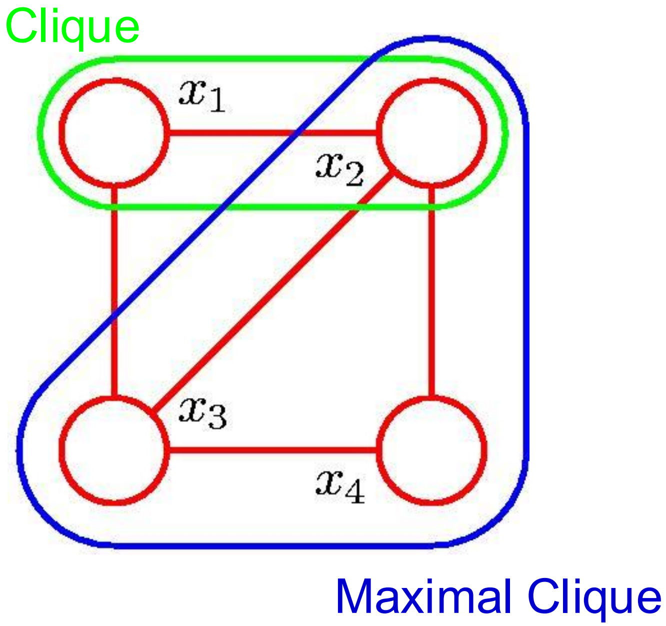
- Markov network = undirected graph + potential functions
  - For each clique (or max clique), a potential function is defined
    - A potential function is not locally normalized, i.e., it doesn't encode probabilities
  - A joint probability is proportional to the product of potentials

$$p(\mathbf{x}) = \frac{1}{Z} \prod_C \psi_C(\mathbf{x}_C)$$

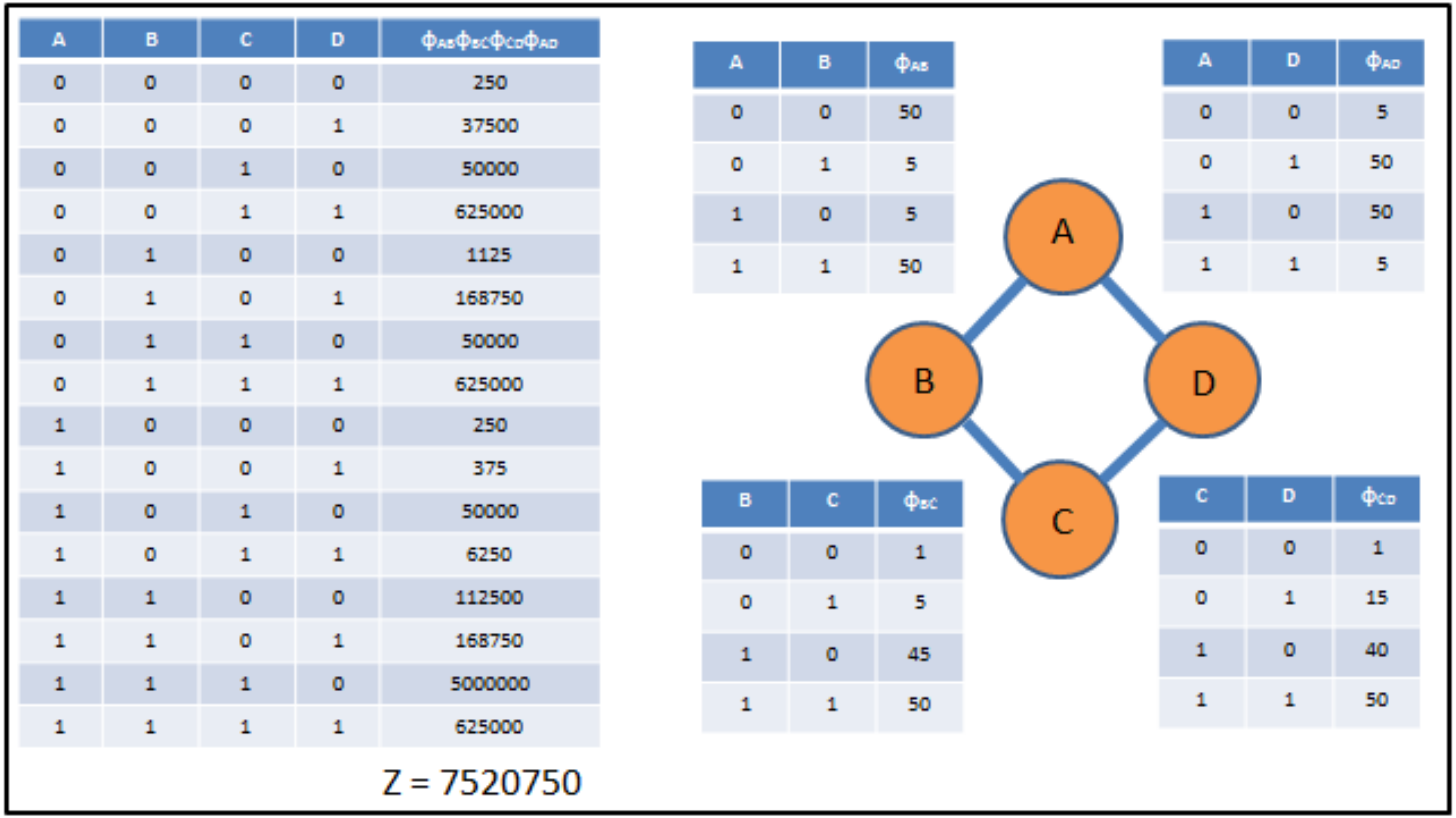
where $\psi_C(\mathbf{x}_C)$ is the potential over clique C and

$$Z = \sum_{\mathbf{x}} \prod_C \psi_C(\mathbf{x}_C)$$

is the normalization coefficient (aka. partition function).



Markov Networks



A	B	C	D	$\phi_{AB}\phi_{BC}\phi_{CD}\phi_{AD}$
0	0	0	0	250
0	0	0	1	37500
0	0	1	0	50000
0	0	1	1	625000
0	1	0	0	1125
0	1	0	1	168750
0	1	1	0	50000
0	1	1	1	625000
1	0	0	0	250
1	0	0	1	375
1	0	1	0	50000
1	0	1	1	6250
1	1	0	0	112500
1	1	0	1	168750
1	1	1	0	5000000
1	1	1	1	625000

Z = 7520750

A	B	ϕ_{AB}
0	0	50
0	1	5
1	0	5
1	1	50

A	D	ϕ_{AD}
0	0	5
0	1	50
1	0	50
1	1	5

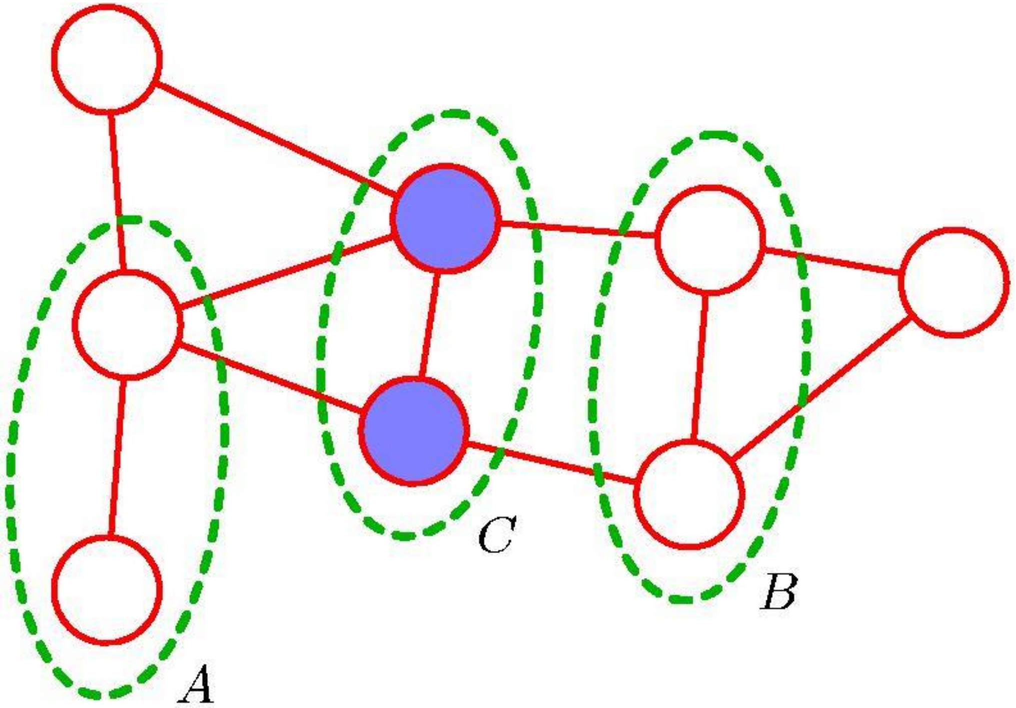
B	C	ϕ_{BC}
0	0	1
0	1	5
1	0	45
1	1	50

C	D	ϕ_{CD}
0	0	1
0	1	15
1	0	40
1	1	50

Non-negative scores

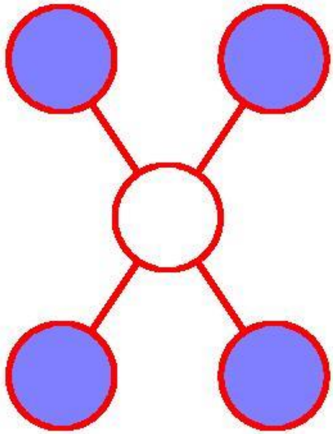
Markov Networks

- Conditional independence and Markov blanket in MN



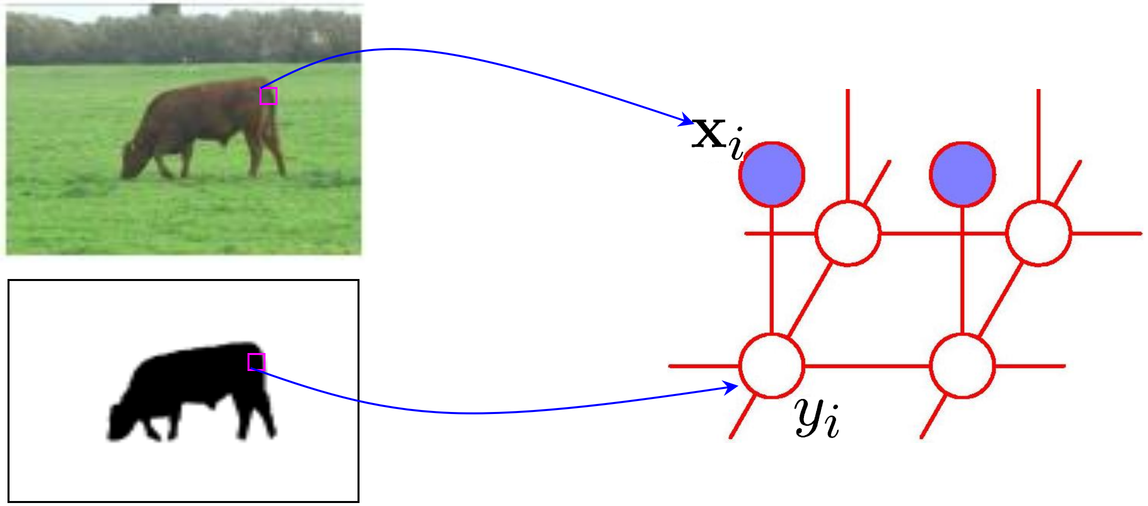
$A \perp\!\!\!\perp B | C$

Markov Blanket



Example – Image Segmentation

- Binary segmentation



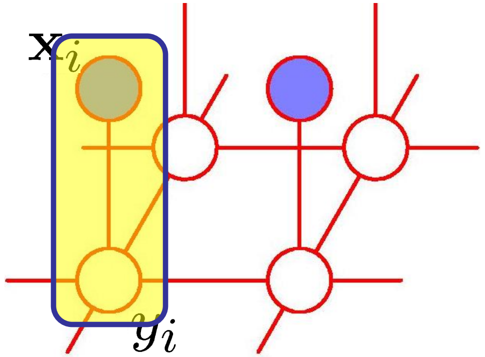
$\mathbf{x}_i$ image pixel

$y_i \in \{0, 1\}$

0: background
1: foreground

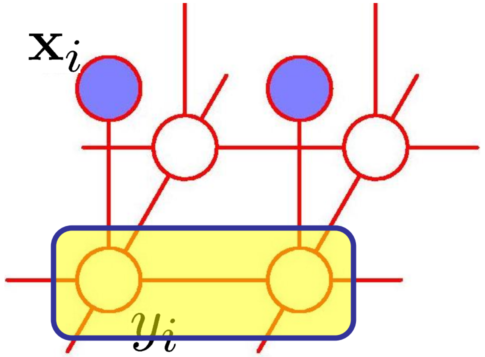
Example – Image Segmentation

- Unary potential
  - Indicating how likely a pixel is a background vs. foreground
    - Ex: $\psi(\mathbf{x}_i, y_i) = \exp\left(w^T \phi(\mathbf{x}_i, y_i)\right)$, where $\phi(\mathbf{x}_i, y_i)$ is a feature vector
    - Ex: we may assign a large weight to the feature: $\{\mathbf{x}_i$ is dark and $y_i = 0\}$

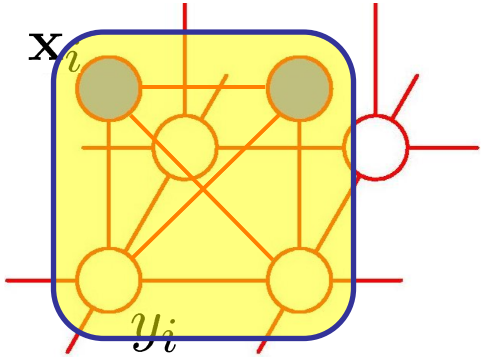


Example – Image Segmentation

- Pairwise potential
  - Encouraging adjacent pixels to have same labels (smoothing)
    - Ex. $\psi(y_i, y_j) = \exp\left(\alpha\, I(y_i = y_j)\right)$

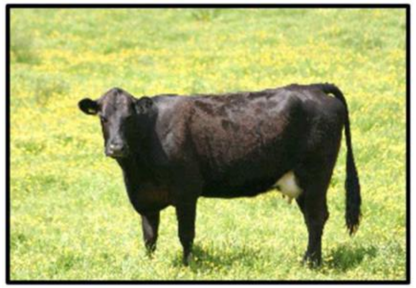


  - A better design is to incorporate pixel info, e.g., similar pixels are more likely to have same labels.
    - Need to change the graph structure

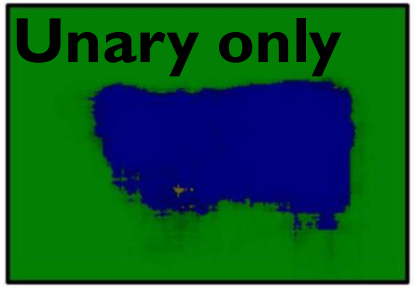


Example – Image Segmentation

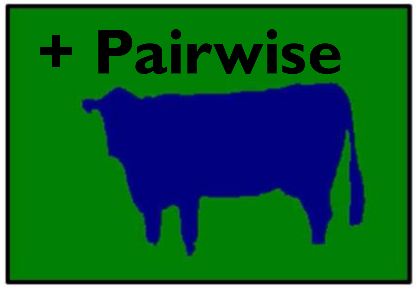
- Inferring labels from image pixels



X



Y

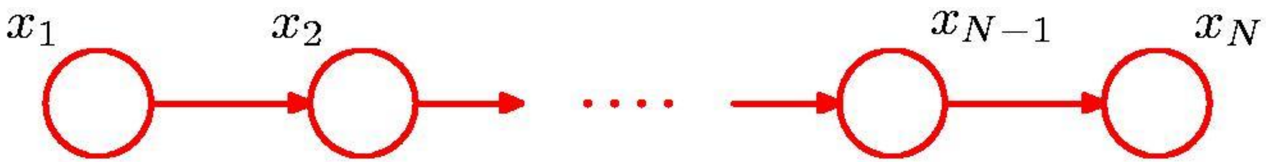


Y

Graphical Models

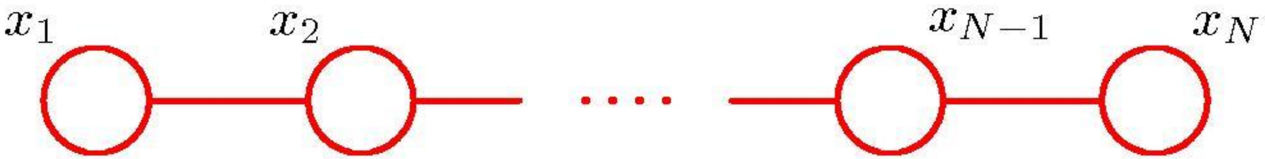
- A graphical model is a probabilistic model for which a graph expresses conditional dependence between random variables
  - Bayesian networks: directed acyclic graph
  - Markov networks: undirected graph
  - Factor graphs, conditional random fields, etc.

Converting Directed to Undirected Graphs (1)



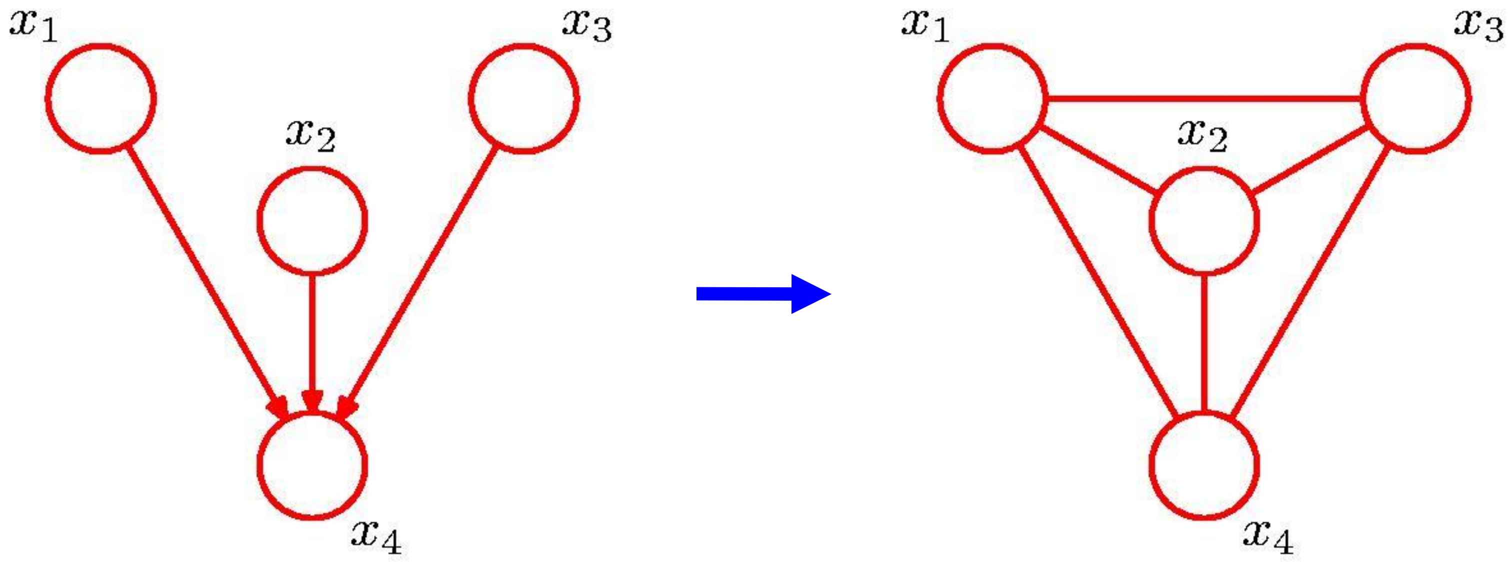
$$p(\mathbf{x}) = \underbrace{p(x_1)p(x_2|x_1)}\, p(x_3|x_2)\cdots p(x_N|x_{N-1})$$

$$p(\mathbf{x}) = \frac{1}{Z}\,\psi_{1,2}(x_1, x_2)\,\psi_{2,3}(x_2, x_3)\cdots\psi_{N-1,N}(x_{N-1}, x_N)$$



Converting Directed to Undirected Graphs (2)

- Additional links (moralization)

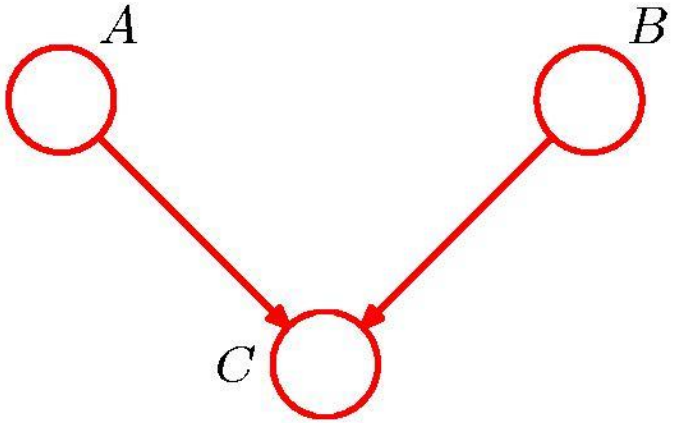


$$
\begin{aligned}
p(\mathbf{x}) &= p(x_1)p(x_2)p(x_3)p(x_4|x_1,x_2,x_3) \\
&= \frac{1}{Z}\psi(x_1,x_2,x_3,x_4)
\end{aligned}
$$

Bayesian Network → Markov Network

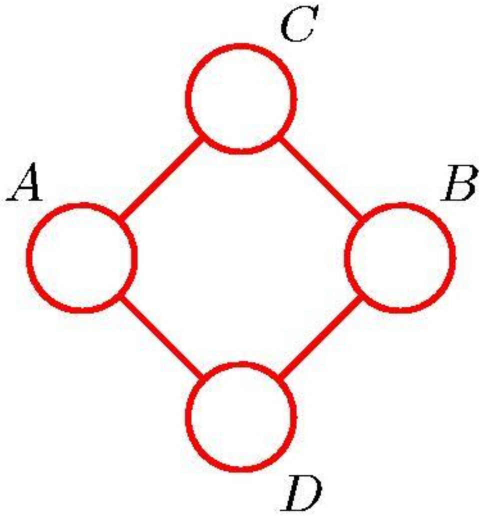
- Steps
  1. Moralization
  2. Construct potential functions from CPTs
- The BN and MN encode the same distribution
- Do they encode the same set of conditional independence?

Encoding Conditional Independence



$$A \perp\!\!\!\perp B \mid \emptyset$$

$$A \not\perp\!\!\!\perp B \mid C$$

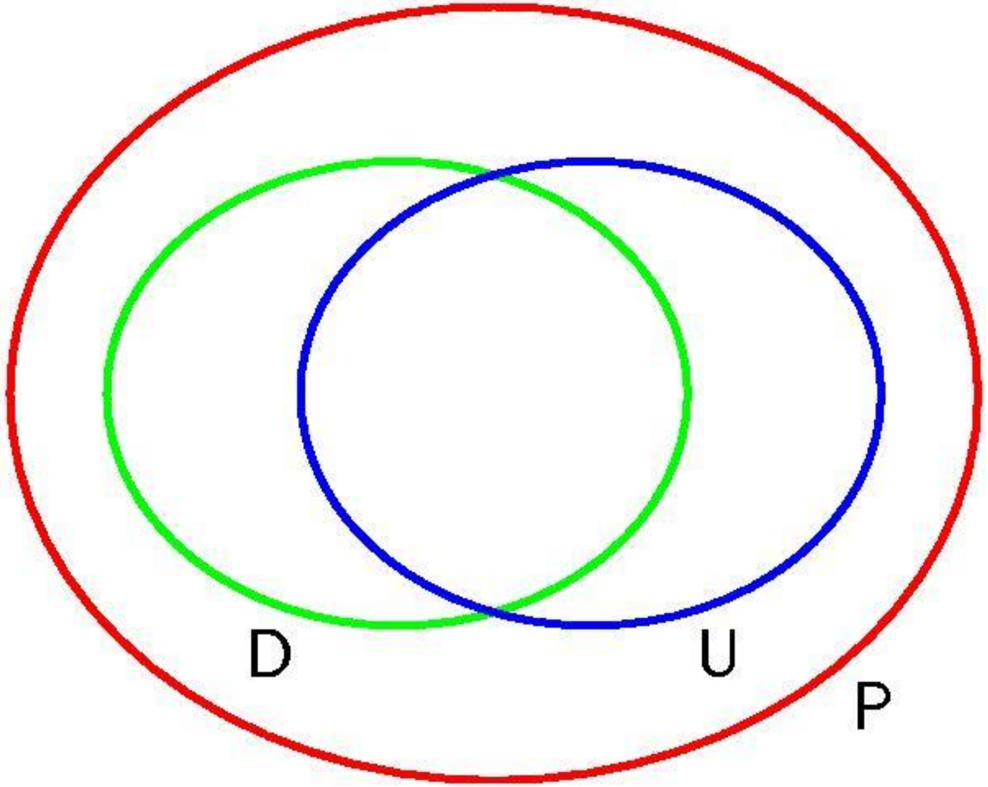


$$A \not\perp\!\!\!\perp B \mid \emptyset$$

$$A \perp\!\!\!\perp B \mid C \cup D$$

$$C \perp\!\!\!\perp D \mid A \cup B$$

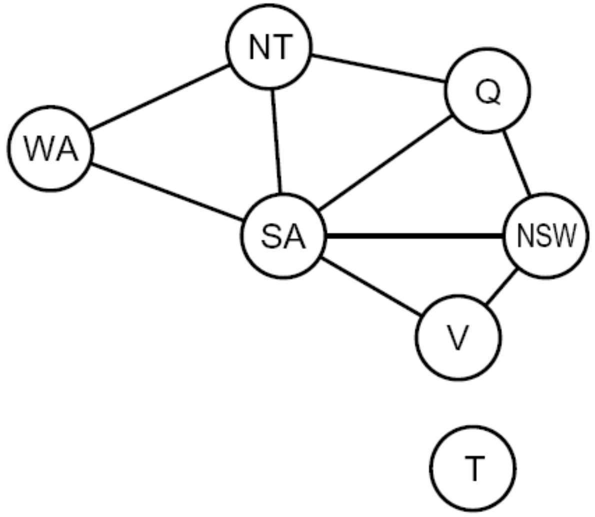
Encoding Conditional Independence

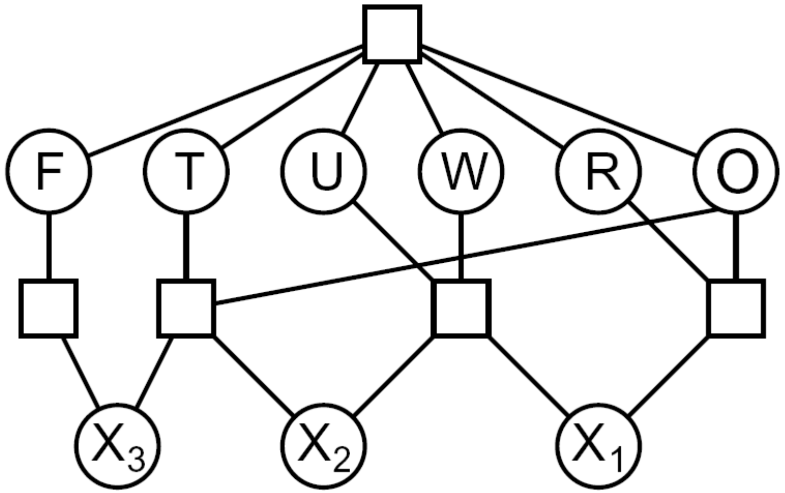


The set of distributions whose conditional independence can be exactly (i.e., no more, no less) represented by a **directed**/**undirected** graph

Markov networks vs. Constraint graphs

- Constraint graphs can be seen as Markov networks with 0/1 potentials



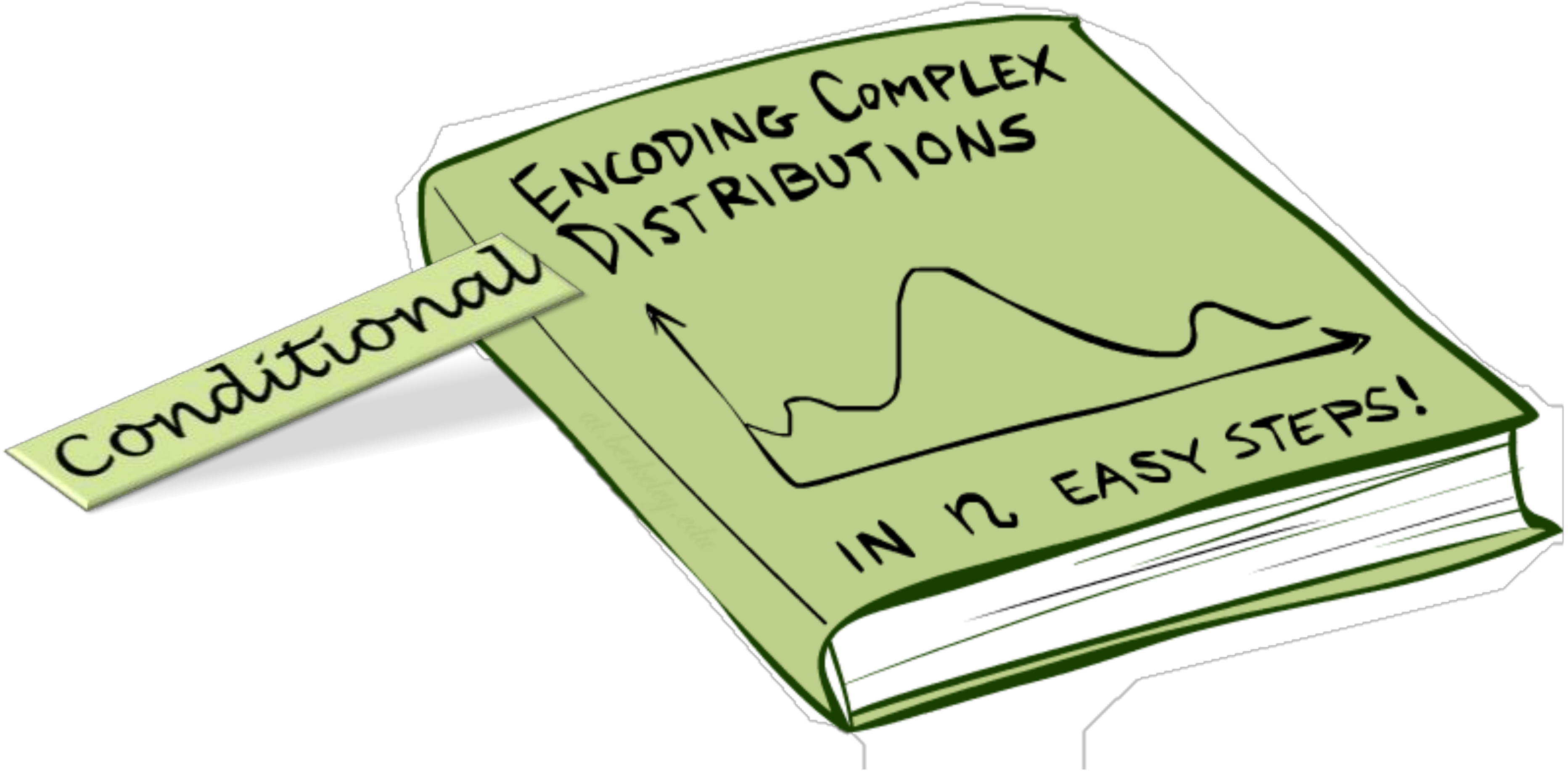


BN/MN vs. Logic

- Which logic is BN/MN more similar to: PL? FOL?
  - Boolean nodes represent propositions
  - No explicit representation of objects, relations, quantifiers

- BN/MN can be seen as a probabilistic extension of PL
- PL can be seen as BN/MN with deterministic CPTs/potentials

Conditional Random Fields



Generative vs. Discriminative Models

- Generative models
  - A generative model represents a joint distribution $P(X_1, X_2, \ldots, X_n)$
  - Both BN and MN are generative models

- Discriminative models
  - In some scenarios, we only care about predicting queries from evidence
    - E.g., image segmentation
  - A discriminative model represents a conditional distribution $P(Y_1, Y_2, \ldots, Y_n | X)$
  - It does not model $P(X)$

Conditional Random Fields (CRF)

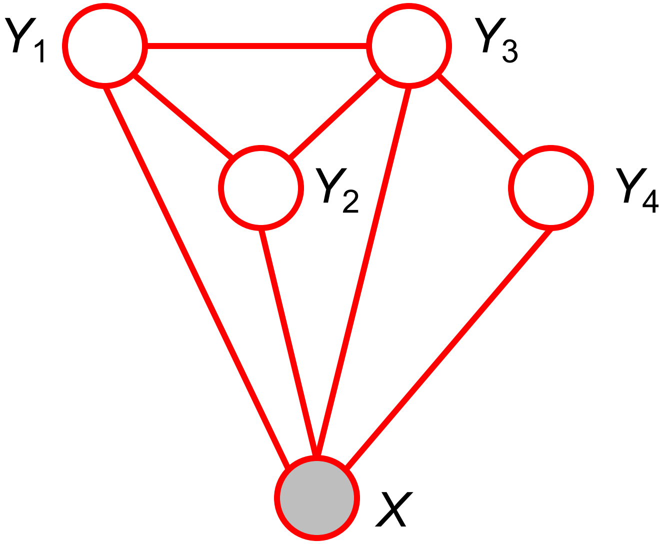
- An extension of MN (aka. Markov random field) where everything is conditioned on an input

$$P(\mathrm{y}|x) = \frac{1}{Z(x)} \prod_C \psi_C(\mathrm{y}_C, x)$$

where $\psi_C(y_C, x)$ is the potential over clique C and

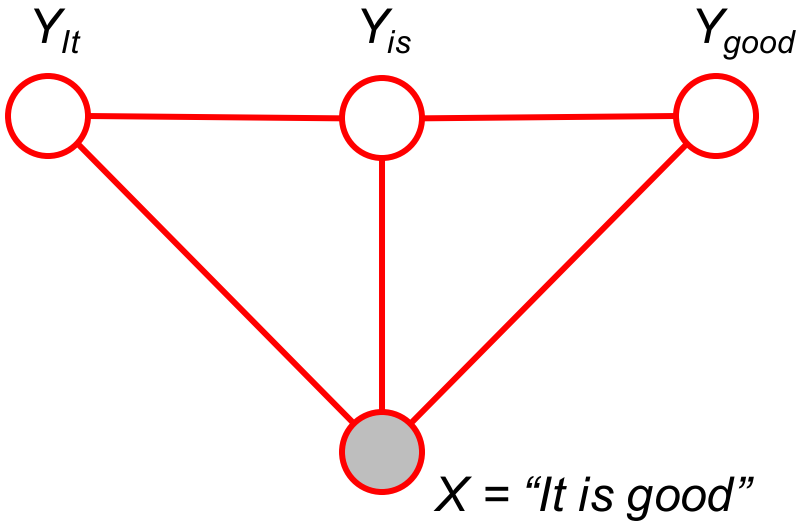
$$Z(x) = \sum_{\mathrm{y}} \prod_C \psi_C(\mathrm{y}_C, x)$$

is the normalization coefficient.



CRF Applications

- NLP
  - POS tagging
  - Named entity recognition
  - Syntactic parsing
- CV
  - Image segmentation
  - Posture recognition



Summary

- A Bayesian network encodes a joint distribution
  - Syntax: DAG+CPTs
  - Semantics
    - Global semantics
    - Conditional independence semantics
    - D-separation

- Markov networks
  - Syntax: undirected graph + potentials
  - Semantics
  - Extension: CRF