Games (cont'd)

Probability and Bayes Nets

Dr. Angelica Lim
Assistant Professor
School of Computing Science
Simon Fraser University, Canada

Nov. 19, 2024

Slide Credits:Dan Klein and Pieter Abbeel for CS188 Intro to AI at UC Berkeley (ai.berkeley.edu).
Pearson 2021

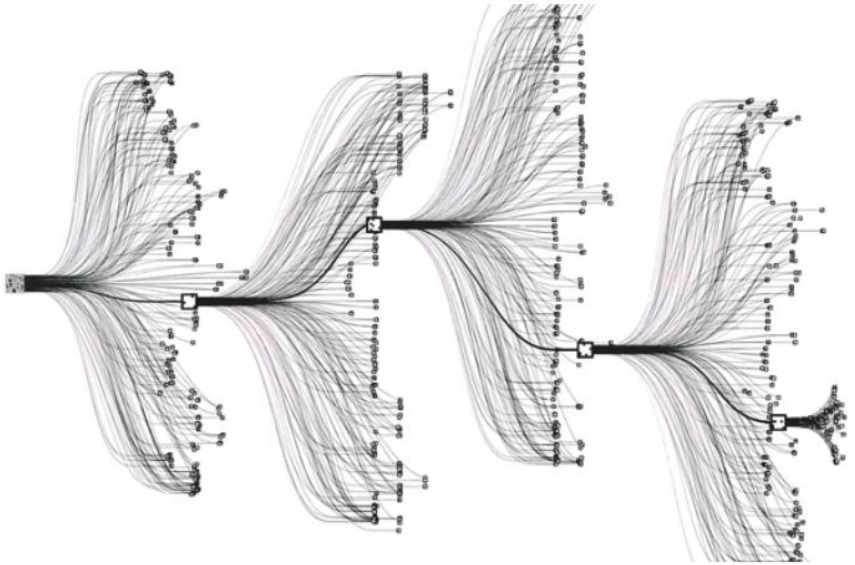
Course Overview

Week 1 : Getting to know you
Week 2 : Introduction to Artificial Intelligence
Week 3: Machine Learning I: Basic Supervised Models (Classification)
Week 4: Machine Learning II: Supervised Regression, Classification and Gradient Descent, K-Means
Week 5: Machine Learning III: Neural Networks and Backpropagation
Week 6 : Search
Week 7 : Markov Decision Processes
Week 8 : Midterm
Week 9 : Reinforcement Learning
Week 10 : Games
Week 11 : Probability and Bayesian Networks I
Week 12 : Bayesian Networks II and Markov Networks
Week 13 : Ethics and Explainability

State-based models

Key idea: Model the state of the world and transitions between states, triggered by actions

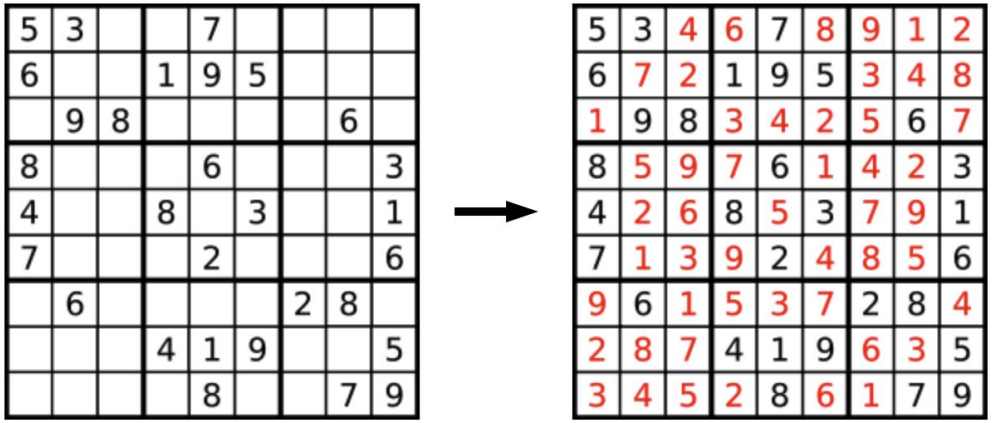
Solutions are procedural, step by step. These model tasks that require forethought, *e.g. playing chess or planning a big trip*



Applications:

- Games: Chess, Go, Pac-Man, Starcraft, etc.
- Robotics: motion planning

How do we fill in the elements?



5	3			7				
6			1	9	5			
	9	8					6	
8				6				3
4			8		3			1
7				2				6
	6					2	8	
			4	1	9			5
				8			7	9

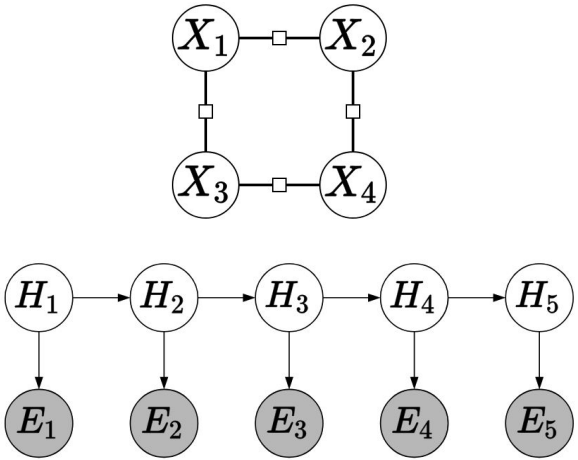
5	3	4	6	7	8	9	1	2
6	7	2	1	9	5	3	4	8
1	9	8	3	4	2	5	6	7
8	5	9	7	6	1	4	2	3
4	2	6	8	5	3	7	9	1
7	1	3	9	2	4	8	5	6
9	6	1	5	3	7	2	8	4
2	8	7	4	1	9	6	3	5
3	4	5	2	8	6	1	7	9

Goal: put digits in blank squares so each row, column, and 3x3 sub-block has digits 1–9

Key: unlike state-based models, the order of filling squares doesn't matter in the evaluation criteria!

Variable-based models

How is it different from a state-based model? In variable-based models, the order in which things are done is not important. Simply declare what you want, rather than micromanage how the solution is found.



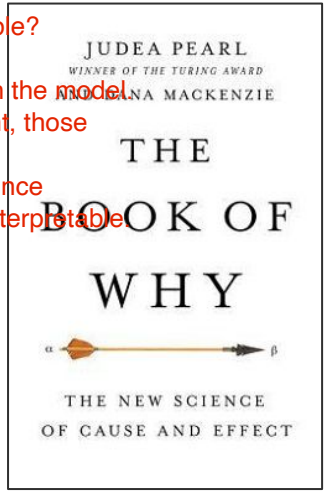
Constraint satisfaction problems: hard constraints (e.g., Sudoku, scheduling)

Bayesian networks: soft dependencies (e.g., tracking cars from sensors). Variables are random variables dependent on each other, e.g. location of airplane H_3 depends on radar reading E_3 and H_2

Bayesian Networks

- Allow for reasoning under uncertainty using variables and their dependency on one another, *e.g. P(airport on time | no traffic) = 0.90*
- Developed by Judea Pearl (1980s)
- Precursor to generative modeling and prediction

Used in medical field since we have values that are interpretable?
What does this mean?
An interpretable value is one which can be understood and used within the model.
If you extract features where you are unsure of what they represent, those features are not interpretable.
If you have a deep neural network, they are not interpretable since there are various hidden layers which extract features which are not interpretable.



Advantages

- Interpretable
- Need fewer training samples (due to incorporating prior knowledge into model)
- Can handle missing features (vs. deep network fixed size)
- Precursor to causal models, counterfactuals, etc.

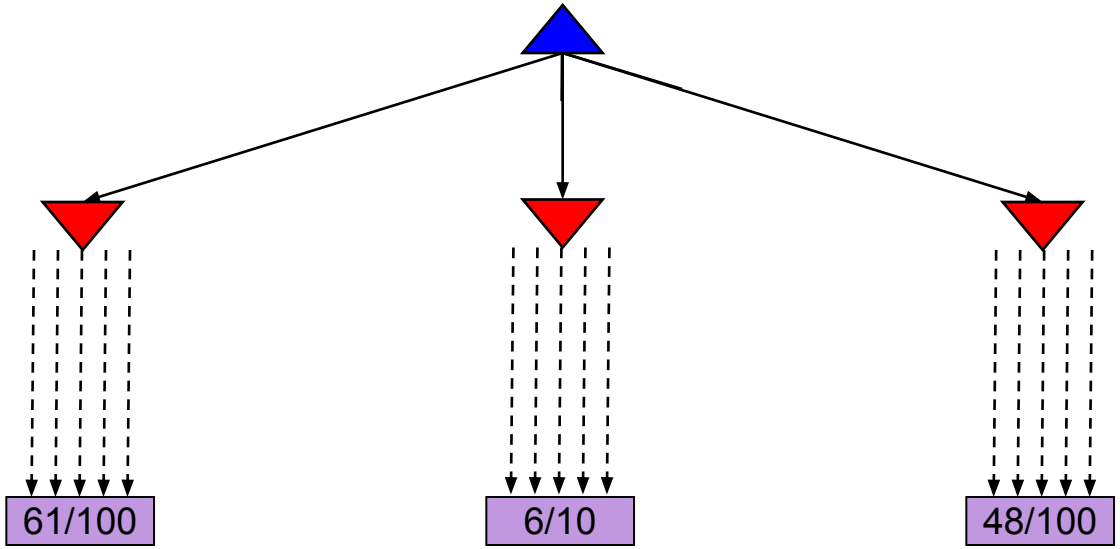
Applications

Decision-making and prediction in healthcare, risk assessment, spam filtering, robotics

Monte Carlo Tree Search

MCTS Version 1.0

- Allocate rollouts to more promising nodes
- Allocate rollouts to more uncertain nodes



UCB heuristics

- UCB1 formula combines "promising" and "uncertain":

$$UCB1(n) = \frac{U(n)}{N(n)} + C \times \sqrt{\frac{\log N(\text{PARENT}(n))}{N(n)}}$$

- $N(n)$ = number of rollouts from node n
- $U(n)$ = total utility of rollouts (e.g., # wins) for Player(Parent(n))
- A provably not terrible heuristic for ***bandit problems***
  - (which are not the same as the problem we face here!)

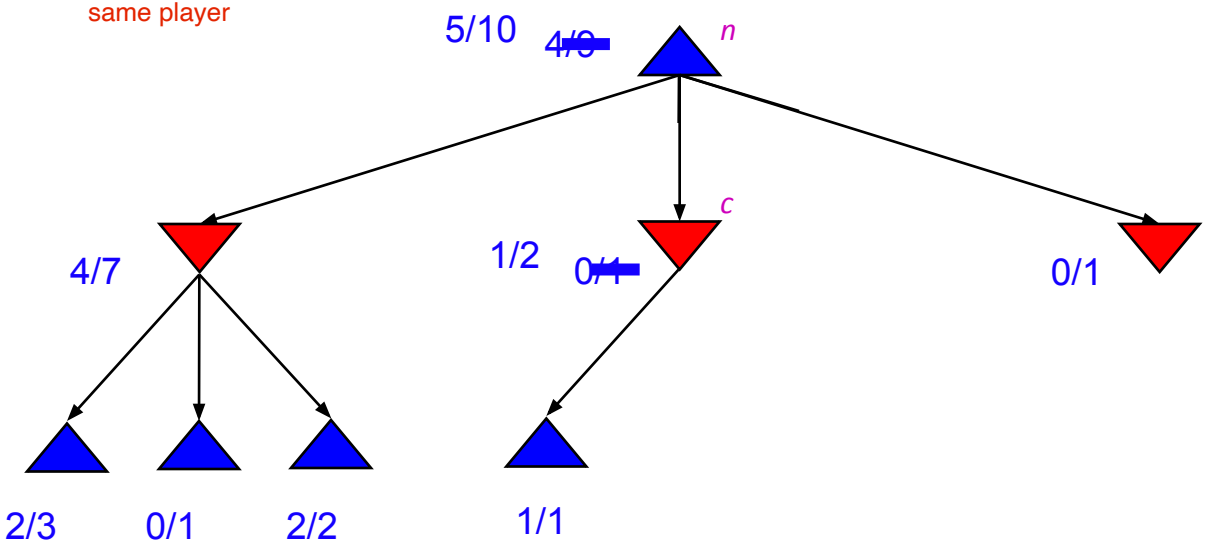
MCTS Version 2.0: UCT

UCT: Upper confidence bound applied to trees

- Repeat until out of time:
  - Given the current search tree, **recursively** apply UCB to choose a path down to a leaf (not fully expanded) node *n*
  - Add a new child *c* to *n* and run a rollout from *c*
  - Update the win counts from *c* back up to the root
- Choose the action leading to the child with highest *N*

UCT Example

The red and blue triangles represent the same player



- Given the current search tree, **recursively** apply UCB to choose a path down to a leaf (not fully expanded) node *n*
- Add a new child *c* to *n* and run a rollout from *c*
- Update the win counts from *c* back up to the root

Minimax vs. MCTS with UCT

- Alpha-beta pruning won't work for large state spaces (e.g. Go)
- With MCTS, we approximate using simulations

- "Value" of a node, $U(n)/N(n)$, is a weighted ***sum*** of child values!
- **Idea**: as $N \rightarrow \infty$, the vast majority of rollouts are concentrated in the best child(ren), so weighted average $\rightarrow$ minimax
- **Theorem**: as $N \rightarrow \infty$ UCT selects the minimax move
  - (but N never approaches infinity!)

Example Application of MCTS

Can a DRL be bayesian network?

Long-short term memory (LSTM)

Monte Carlo Tree Search (MCTS)

An MCTS-DRL Based Obstacle and Occlusion Avoidance Methodology in Robotic Follow-Ahead Applications

Sahar Leisiazar[1], Edward J. Park[1], Angelica Lim[2] and Mo Chen[2]

***Abstract*— We propose a novel methodology for robotic follow-ahead applications that address the critical challenge of obstacle and occlusion avoidance. Our approach effectively navigates the robot while ensuring avoidance of collisions and occlusions caused by surrounding objects. To achieve this, we developed a high-level decision-making algorithm that generates short-term navigational goals for the mobile robot. Monte Carlo Tree Search is integrated with a Deep Reinforcement Learning method to enhance the performance of the decision-making process and generate more reliable navigational goals. Through extensive experimentation and analysis, we demonstrate the effectiveness and superiority of our proposed approach in comparison to the existing follow-ahead human-following robotic methods. Our code is available at https://github.com/saharLeisiazar/follow-ahead-ros.**

I. INTRODUCTION

Human-robot interaction involves robots performing stable, real-time, and safe interactions with a target person in

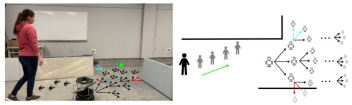
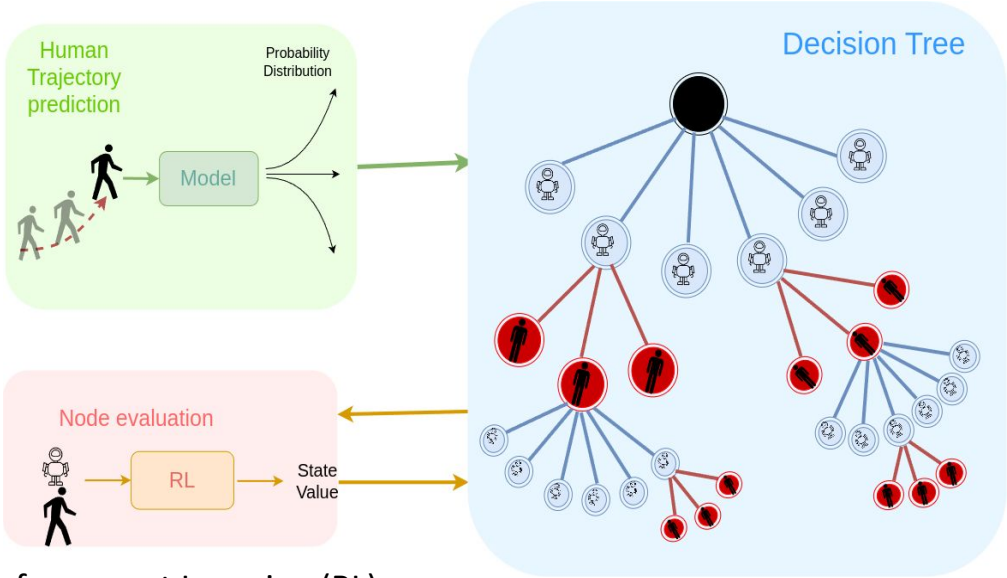


Fig. 1: Illustration of the MCTS-DRL framework utilizing human future estimation for goal generation. The search tree is expanded to identify the best goal point to follow ahead of the person, while avoiding occlusion and collision. The resulting goal point is indicated by a green star, while red and blue arrows represent paths leading to collision and occlusion, respectively. The MCTS algorithm expands a tree to find the best navigational goal for the robot in order to follow-ahead of the target person and avoid collision and occlusion caused by surrounding objects.



Reinforcement Learning (RL)

DRL: deep reinforcement learning

Probability and Bayes Nets

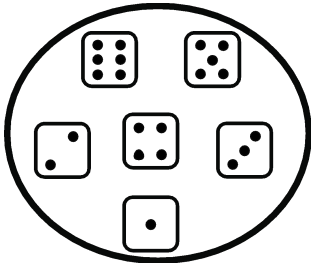
Slides mostly from Stuart Russell and Peyrin Kao

University of California, Berkeley

Uncertainty

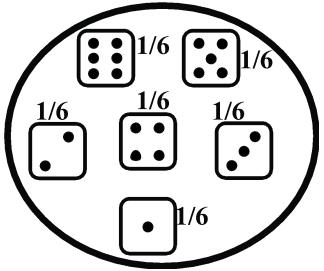
- The real world is rife with uncertainty!
  - E.g., if I leave for SFO 60 minutes before my flight, will I be there in time?
- Problems:
  - partial observability (road state, other drivers' plans, etc.)
  - noisy sensors (radio traffic reports, Google maps)
  - immense complexity of modelling and predicting traffic, security line, etc.
  - lack of knowledge of world dynamics (will tire burst? will I get in crash?)
- Probabilistic assertions summarize effects of ***ignorance*** and ***laziness***
- Combine probability theory + utility theory -> decision theory
  - ***Maximize expected utility*** : $a^* = \text{argmax}_a \sum_s P(s \mid a)\, U(s)$

Basic laws of probability (discrete)



- Begin with a set Ω of possible worlds
  - E.g., 6 possible rolls of a die, {1, 2, 3, 4, 5, 6}

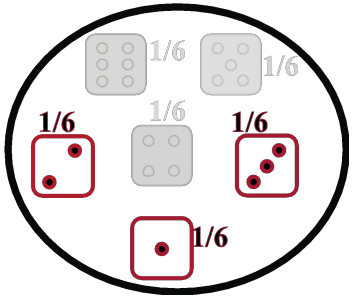
- A ***probability model*** assigns a number $P(\omega)$ to each world ω
  - E.g., $P(1) = P(2) = P(3) = P(5) = P(5) = P(6) = 1/6$.

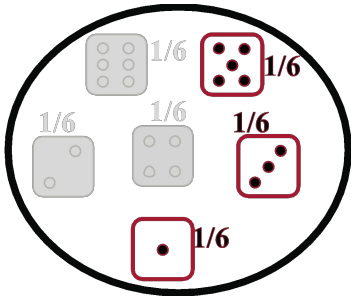


- These numbers must satisfy
  - $0 \leq P(\omega)$
  - $\sum_{\omega \in \Omega} P(\omega) = 1$

Basic laws contd.

- An ***event*** is any subset of Ω
  - E.g., "roll < 4" is the set {1,2,3}
  - E.g., "roll is odd" is the set {1,3,5}

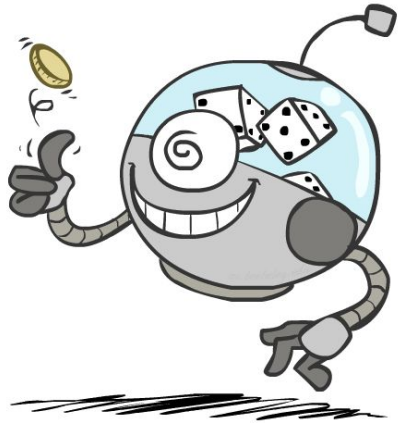




- The probability of an event is the ***sum*** of probabilities over its worlds
  - $P(A) = \sum_{\omega \in A} P(\omega)$
  - E.g., $P(roll < 4) = P(1) + P(2) + P(3) = 1/2$

Random Variables

- A **random variable** (usually denoted by a capital letter) is some aspect of the world about which we may be uncertain
- The ***range*** of a random variable is the set of possible values
  - *Odd* = Is the dice roll an odd number? → {true, false}
    - e.g. *Odd*(1)=true, *Odd*(6) = false
    - often write the event *Odd*=true as *odd*, *Odd*=false as ¬*odd*
  - *T* = Is it hot or cold? → {hot, cold}
  - *D* = How long will it take to get to the airport? → [0, ∞)
  - L_{Ghost} = Where is the ghost? → {(0,0), (0,1), …}
- The ***probability distribution*** of a random variable *X* gives the probability for each value *x* in its range (probability of the event *X*=*x*)
  - $P(X=x) = \sum_{\{\omega: X(\omega)=x\}} P(\omega)$
  - $P(x)$ for short (when unambiguous)
  - $P(X)$ refers to the entire distribution (think of it as a vector or table)



Probability Distributions

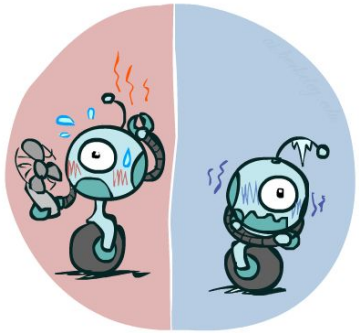
- Associate a probability with each value; sums to 1
  - Temperature:
  - Weather:

Marginal distributions

Joint distribution

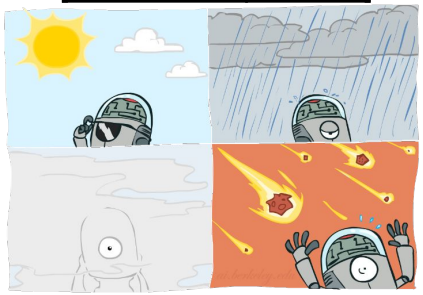
$P(T)$

T	P
hot	0.5
cold	0.5



$P(W)$

W	P
sun	0.6
rain	0.1
fog	0.3
meteor	0.0



$P(T,W)$

		Temperature	
		hot	cold
Weather	sun	0.45	0.15
	rain	0.02	0.08
	fog	0.03	0.27
	meteor	0.00	0.00

- *Can't* deduce joint from marginals
- *Can* deduce marginals from joint

Making possible worlds

- In many cases we
  - begin with random variables and their domains
  - construct possible worlds as assignments of values to all variables
- E.g., two dice rolls $Roll_1$ and $Roll_2$
  - How many possible worlds?
  - What are their probabilities?
- Size of distribution for n variables with range size d?

e.g. 6^2

d^n

- For all but the smallest distributions, cannot write out by hand!

Probabilities of events

- Recall that the probability of an event is the sum of probabilities of its worlds:
  - $P(A) = \sum_{\omega \in A} P(\omega)$
- So, given a joint distribution over all variables, can compute any event probability
  - Probability that it's hot AND sunny?

0.45

  - Probability that it's hot?

0.5

  - Probability that it's hot OR not foggy?

p(hot U not foggy) = p(hot) + p(not foggy) - p(hot intersect not foggy) = 0.5 + 0.47 + 0.23 - 0.47 = 0.73

- ***Joint distribution***

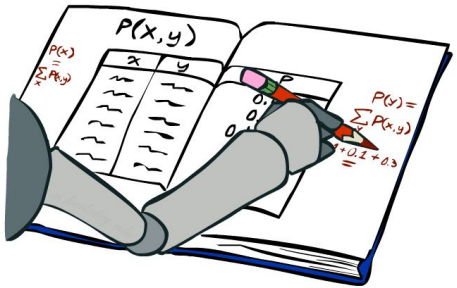
$P(T,W)$

		Temperature	
		hot	cold
Weather	sun	0.45	0.15
	rain	0.02	0.08
	fog	0.03	0.27
	meteor	0.00	0.00

Marginal Distributions

- Marginal distributions are sub-tables which eliminate variables
- ***Marginalization*** (***summing out***): Collapse a dimension by adding

$$P(X=x) = \sum_y P(X=x, Y=y)$$



		Temperature		
		hot	cold	
Weather	sun	0.45	0.15	0.60
	rain	0.02	0.08	0.10
	fog	0.03	0.27	0.30
	meteor	0.00	0.00	0.00
		0.50	0.50	

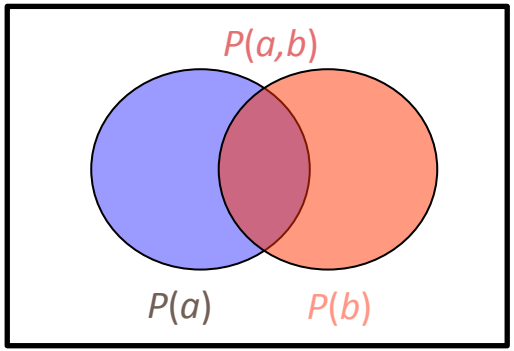
$P(W)$

$P(T)$

Conditional Probabilities

- A simple relation between joint and conditional probabilities
  - In fact, this is taken as the *definition* of a conditional probability

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



$P(T,W)$

		Temperature	
		hot	cold
Weather	sun	0.45	0.15
	rain	0.02	0.08
	fog	0.03	0.27
	meteor	0.00	0.00

$$P(W=s \mid T=c) = \frac{P(W=s, T=c)}{P(T=c)} = 0.15/0.50 = 0.3$$

$= P(W=s,T=c) + P(W=r,T=c) + P(W=f,T=c) + P(W=m,T=c)$
$= 0.15 + 0.08 + 0.27 + 0.00 = 0.50$

Conditional Distributions

- Distributions for one set of variables given another set

		Temperature	
		hot	cold
Weather	sun	0.45	0.15
	rain	0.02	0.08
	fog	0.03	0.27
	meteor	0.00	0.00

$P(W \mid T=h)$

hot
0.90
0.04
0.06
0.00

$P(W \mid T=c)$

cold
0.30
0.16
0.54
0.00

$P(W \mid T)$

hot	cold
0.90	0.30
0.04	0.16
0.06	0.54
0.00	0.00

divided by P(T=h)
=0.5 from previous slide

Normalizing a distribution

- (Dictionary) To bring or restore to a normal condition

All entries sum to ONE

- Procedure:
  - Multiply each entry by $\alpha = 1/(\text{sum over all entries})$

$P(W,T)$

		Temperature	
		hot	cold
Weather	sun	0.45	0.15
	rain	0.02	0.08
	fog	0.03	0.27
	meteor	0.00	0.00

$P(W,T=c)$

0.15
0.08
0.27
0.00

Normalize

$\alpha = 1/0.50 = 2$

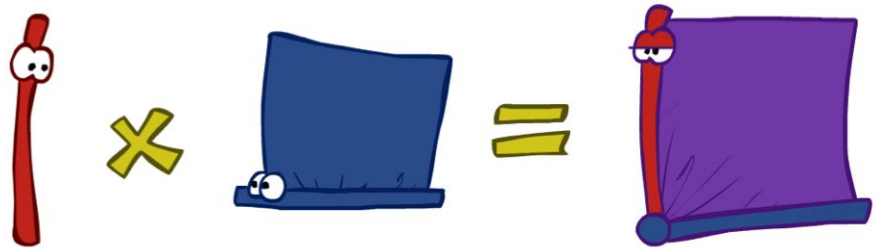
$P(W \mid T=c) = P(W,T=c)/P(T=c) = \alpha\, P(W,T=c)$

0.30
0.16
0.54
0.00

The Product Rule

- Sometimes we have conditional distributions but want the joint

$$P(a \mid b)\, P(b) = P(a, b) \quad \Leftrightarrow \quad P(a \mid b) = \frac{P(a, b)}{P(b)}$$



The Product Rule: Example

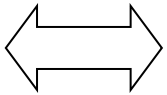
$$P(W \mid T)\, P(T) = P(W, T)$$

$P(W \mid T)$

hot	cold
0.90	0.30
0.04	0.16
0.06	0.54
0.00	0.00

$P(T)$

T	P
hot	0.5
cold	0.5



$P(W, T)$

		Temperature	
		hot	cold
Weather	sun	0.45	0.15
	rain	0.02	0.08
	fog	0.03	0.27
	meteor	0.00	0.00

The Chain Rule

A joint distribution can be written as a **product** of **conditional distributions** by repeated application of the product rule:

$$P(x_1, x_2, x_3) = P(x_3 \mid x_1, x_2)\, P(x_1, x_2) = P(x_3 \mid x_1, x_2)\, P(x_2 \mid x_1)\, P(x_1)$$

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

Recap: Probability

- **Basic laws**: $0 \leq P(\omega) \leq 1 \quad \sum_{\omega \in \Omega} P(\omega) = 1$
- **Events**: subsets of Ω: $P(A) = \sum_{\omega \in A} P(\omega)$
- **Random variable** $X(\omega)$ has a value in each ω
  - **Distribution** $P(X)$ gives probability for each possible value x
  - **Joint distribution** $P(X,Y)$ gives total probability for each combination x,y
- Summing out/**marginalization**: $P(X=x) = \sum_y P(X=x,Y=y)$
- **Conditional probability**: $P(X|Y) = P(X,Y)/P(Y)$
- **Product rule**: $P(X|Y)P(Y) = P(X,Y) = P(Y|X)P(X)$
  - Generalize to **chain rule**: $P(X_1,..,X_n) = \prod_i P(X_i \mid X_1,..,X_{i-1})$

Probabilistic Inference

- **Probabilistic inference**: compute a desired probability from a probability model
  - Typically for a ***query variable*** given ***evidence***
  - E.g., P(airport on time | no accidents) = 0.90
  - These represent the agent's *beliefs* given the evidence

- Probabilities change with new evidence:
  - P(airport on time | no accidents, 5 a.m.) = 0.95
  - P(airport on time | no accidents, 5 a.m., raining) = 0.80
  - Observing new evidence causes ***beliefs to be updated***

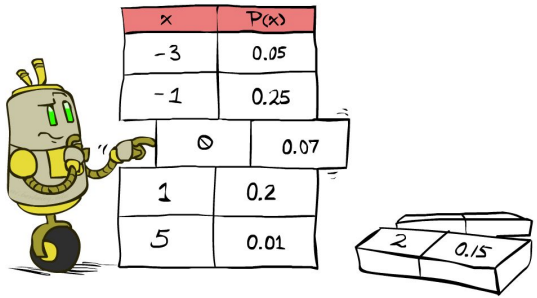
Inference by Enumeration

- Probability model $P(X_1, \ldots, X_n)$ is given
- Partition the variables $X_1, \ldots, X_n$ into sets as follows:
  - Evidence variables: $\boldsymbol{E} = \boldsymbol{e}$
  - Query variables: $\boldsymbol{Q}$
  - Hidden variables: $\boldsymbol{H}$

- We want:

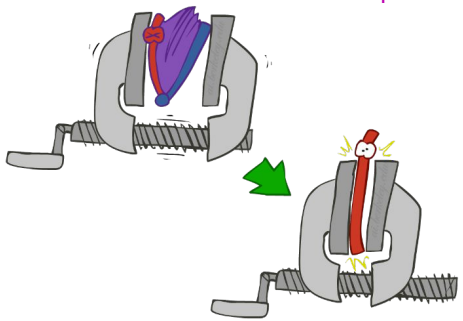
$$P(\boldsymbol{Q} \mid \boldsymbol{e})$$

- Step 1: Select the entries consistent with the evidence



- Step 2: Sum out $\boldsymbol{H}$ from model to get joint of query and evidence

$$P(\boldsymbol{Q}, \boldsymbol{e}) = \sum_{\boldsymbol{h}} \underbrace{P(\boldsymbol{Q}, \boldsymbol{h}, \boldsymbol{e})}_{X_1, \ldots, X_n}$$



- Step 3: Normalize

$$P(\boldsymbol{Q} \mid \boldsymbol{e}) = \alpha\, P(\boldsymbol{Q}, \boldsymbol{e})$$

Inference by Enumeration

Query variable

Evidence variable

P(S | sun)?

- 1. Enumerate options with sun
- 2. Sum out irrelevant variable(s)
- 3. Normalize

S = season

The assumption made here is that all outcomes are known ahead of time.

P(S | sun) =

{summer: 0.45/(0.45+0.25), winter: 0.25/(0.45+0.25)}

0.45

0.25

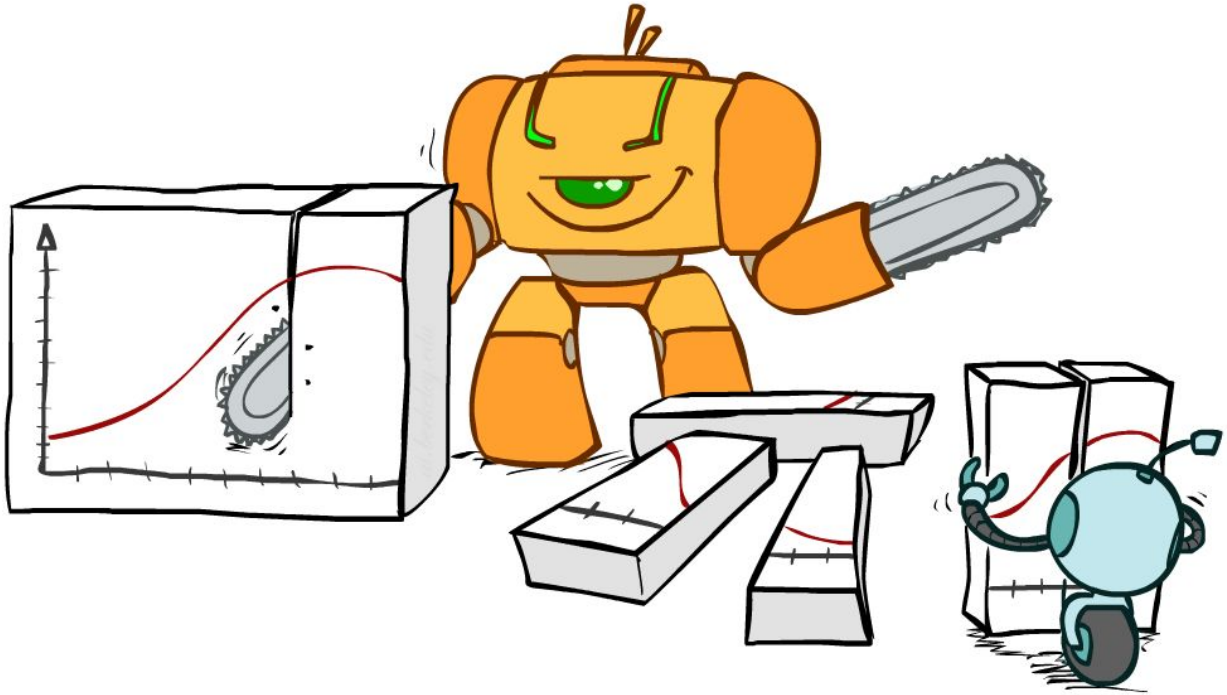
Season	Temp	Weather	P
summer	hot	sun	0.35
summer	hot	rain	0.01
summer	hot	fog	0.01
summer	hot	meteor	0.00
summer	cold	sun	0.10
summer	cold	rain	0.05
summer	cold	fog	0.09
summer	cold	meteor	0.00
winter	hot	sun	0.10
winter	hot	rain	0.01
winter	hot	fog	0.02
winter	hot	meteor	0.00
winter	cold	sun	0.15
winter	cold	rain	0.20
winter	cold	fog	0.18
winter	cold	meteor	0.00

Here, temperature was a hidden variable

Inference by Enumeration

- Obvious problems:
  - Worst-case time complexity $O(d^n)$ (exponential in #hidden variables)
  - Space complexity $O(d^n)$ to store the joint distribution
  - $O(d^n)$ data points to estimate the entries in the joint distribution

Bayes' Rule



Bayes' Rule

- Write the product rule both ways:

$$P(a \mid b)\, P(b) = P(a, b) = P(b \mid a)\, P(a)$$

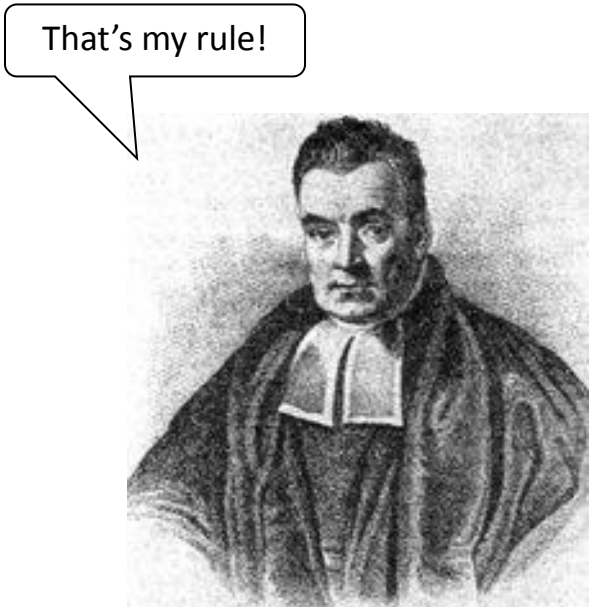
- Dividing left and right expressions, we get:

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

- Why is this at all helpful?
  - Often one conditional is tricky but the other one is simple
  - Lets us build one conditional from its reverse
  - Describes an "update" step from prior $P(a)$ to posterior $P(a \mid b)$

- E.g.: d is data, h is hypothesis:

$$P(h \mid d) = \frac{P(d \mid h)\, P(h)}{P(d)} \qquad \frac{P(d \mid h)\, P(h)}{\Sigma_{h'} P(d \mid h')\, P(h')}$$



Inference with Bayes' Rule

- Example: Diagnostic probability from causal probability:

$$P(cause \mid effect) = \frac{P(effect \mid cause)\, P(cause)}{P(effect)}$$

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

$$P(s \mid m) = 0.8$$
$$P(m) = 0.0001$$
$$P(s) = 0.01$$

Example givens

$$P(m \mid s) = \frac{P(s \mid m)\, P(m)}{P(s)} = \frac{0.8 \times 0.0001}{0.01}$$

  - Note: posterior probability of meningitis still very small: 0.008 (80x bigger – why?)
  - Note: you should still get stiff necks checked out! Why?

Independence

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

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

  - I.e., the joint distribution ***factors*** into a product of two simpler distributions

- Equivalently, via the product rule $P(x,y) = P(x|y)P(y)$,

$$P(x \mid y) = P(x) \quad \text{or} \quad P(y \mid x) = P(y)$$

Models have issue with determining causal relationships

- Example: two dice rolls $Roll_1$ and $Roll_2$
  - $P(Roll_1=5, Roll_2=3) = P(Roll_1=5)\, P(Roll_2=3) = 1/6 \times 1/6 = 1/36$
  - $P(Roll_2=3 \mid Roll_1=5) = P(Roll_2=3)$

Example: Strict Independence

P(Rain, Traffic, Umbrella)

Rain	Traffic	Umbrella	P	P_indep.
F	F	F	0.504	0.314
F	F	T	0.056	0.141
F	T	F	0.126	0.169
F	T	T	0.014	0.076
T	F	F	0.018	0.135
T	F	T	0.072	0.060
T	T	F	0.042	0.072
T	T	T	0.168	0.033

P(Rain) * *P*(Traffic) * *P*(Umbrella) =

F	T
0.7	0.3

*

F	T
0.65	0.35

*

F	T
0.69	0.31

=

Example: Chain Rule

0.14 / 0.7 = 14/100 * 10/ 7 2/10 = 0.2

0.21 / 0.3 = 0.7

P(Rain, Traffic, Umbrella)

Rain	Traffic	Umbrella	P
F	F	F	0.504
F	F	T	0.056
F	T	F	0.126
F	T	T	0.014
T	F	F	0.018
T	F	T	0.072
T	T	F	0.042
T	T	T	0.168

✓

0.09 + 0.21 0.3

P(Rain)

F	T
0.7	0.3

*

P(Traf.|Rain)

	Traffic	
Rain	F	T
F	0.8	0.2
T	0.3	0.7

*

P(Umbr.|Rain, Traf.)

		Umbrella	
Rain	Traffic	F	T
F	F	0.9	0.1
F	T	0.9	0.1
T	F	0.2	0.8
T	T	0.2	0.8

conditional independence

=

traffic column does not inform us of whether an umbrella is needed.

Example: Chain Rule

P(Rain, Traffic, Umbrella)

Rain	Traffic	Umbrella	P
F	F	F	0.504
F	F	T	0.056
F	T	F	0.126
F	T	T	0.014
T	F	F	0.018
T	F	T	0.072
T	T	F	0.042
T	T	T	0.168

✓

P(Rain)

F	T
0.7	0.3

*

P(Traf.|Rain)

	Traffic	
Rain	F	T
F	0.8	0.2
T	0.3	0.7

*

P(Umbr.|Rain)

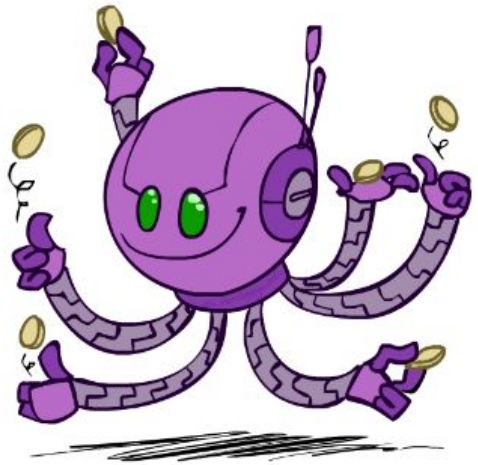
	Umbrella	
Rain	F	T
F	0.9	0.1
T	0.2	0.8

conditional independence

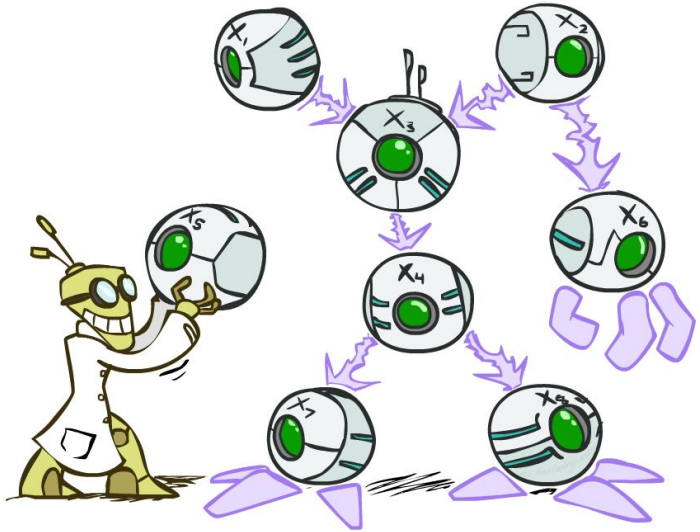
=

Independence, contd.

- Independence is incredibly powerful
  - Exponential reduction in representation size
- Independence is extremely rare!
- ***Conditional*** independence is ubiquitous!!



Bayes Nets



[These slides were created by Dan Klein and Pieter Abbeel for CS188 Intro to AI at UC Berkeley. All CS188 materials are available at http://ai.berkeley.edu.]

Conditional Independence

Conditional Independence

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

- X is conditionally independent of Y given Z if and only if:

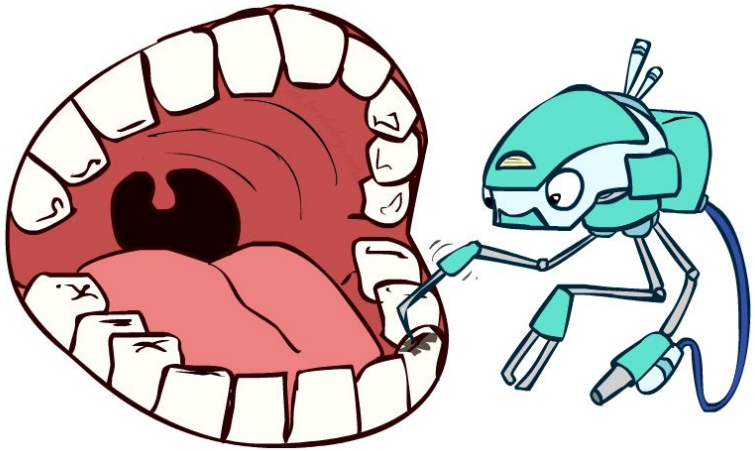
$$\forall x,y,z \qquad P(x \mid y, z) = P(x \mid z)$$

or, equivalently, if and only if

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

Conditional Independence

- P(Toothache, Cavity, Catch)
- If I have a cavity, the probability that the probe catches it doesn't depend on whether I have a toothache:
  - P(+catch | +toothache, +cavity) = P(+catch | +cavity)
- The same independence holds if I don't have a cavity:
  - P(+catch | +toothache, -cavity) = P(+catch| -cavity)
- Catch is *conditionally independent* of Toothache given Cavity:
  - P(Catch | Toothache, Cavity) = P(Catch | Cavity)

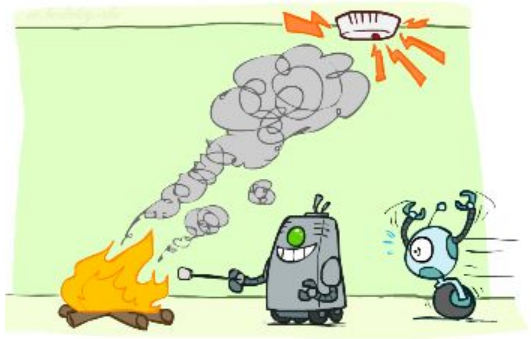


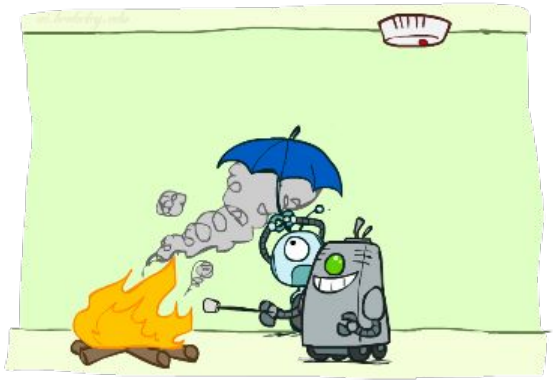
- Equivalent statements:
  - P(Toothache | Catch , Cavity) = P(Toothache | Cavity)
  - P(Toothache, Catch | Cavity) = P(Toothache | Cavity) P(Catch | Cavity)
  - One can be derived from the other easily

Conditional Independence

- What about this domain:
  - Fire
  - Smoke
  - Alarm

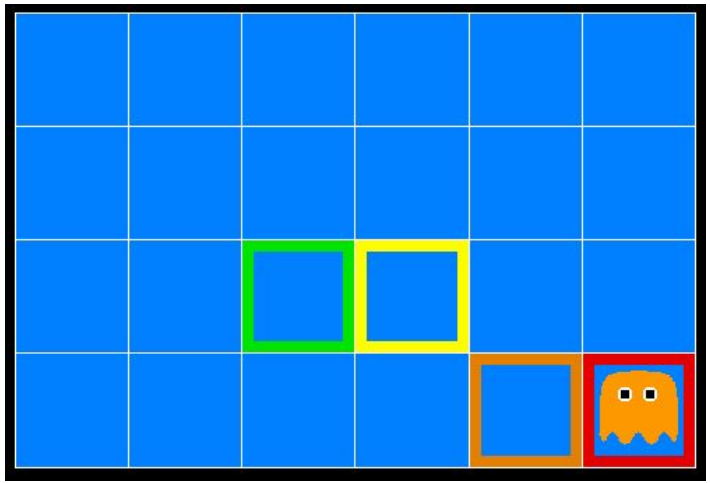
$P(A|S) = P(A|S,F)$





Ghostbusters

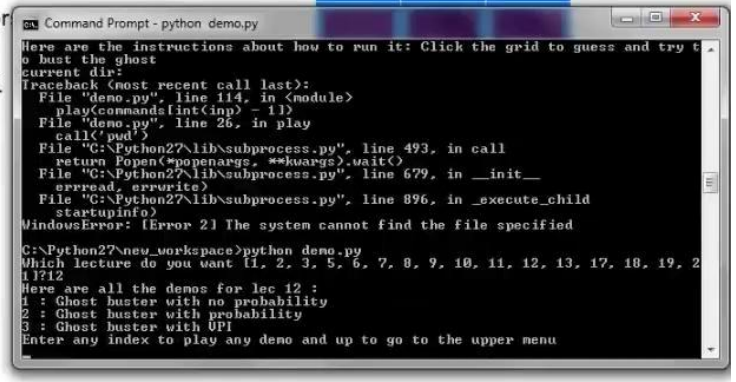
- A ghost is in the grid somewhere
- **Sensor** readings tell how close a square is to the ghost
  - On the ghost: usually red
  - 1 or 2 away: usually orange
  - 3 or 4 away: usually yellow
  - 5+ away: usually green
- Click on squares until confident of location, then "***bust***"



Video of Demo Ghostbusters with Probability

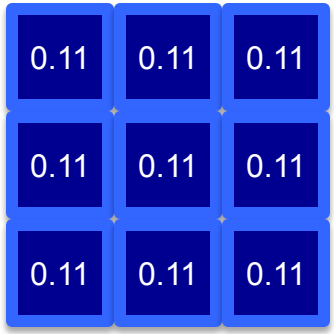
Ghostbusters, Revisited

- Let's say we have two distributions:
  - Prior distribution over ghost location: P(G)
    - Let's say this is uniform
  - Sensor reading model: P(C | G)
    - Given: we know what our sensor
    - C = color measured at (1,1)
    - E.g. P(C = yellow | G=(1,1)) = 0.1



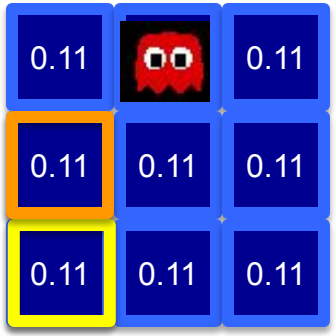
Ghostbusters model

- Variables and ranges:
  - G (ghost location) in {(1,1),…,(3,3)}
  - $C_{x,y}$ (color measured at square x,y) in {red,orange,yellow,green}



- Ghostbuster physics:
  - ***Uniform prior distribution*** over ghost location: $P(G)$
  - ***Sensor model***: $P(C_{x,y} \mid G)$ (depends only on distance to G)
    - E.g. $P(C_{1,1} = \text{yellow} \mid G = (1,1)) = 0.1$

Ghostbusters model, contd.



- $P(G, C_{1,1}, \ldots C_{3,3})$ has $9 \times 4^9 = 2{,}359{,}296$ entries!!!

- Ghostbuster independence:
  - Are $C_{1,1}$ and $C_{1,2}$ independent?
    - E.g., does $P(C_{1,1} = \text{yellow}) = P(C_{1,1} = \text{yellow} \mid C_{1,2} = \text{orange})$?

Given that G is known, the colors of different locations are conditionally independent of each other

- Ghostbuster physics again:
  - $P(C_{x,y} \mid G)$ ***depends only on distance to G***
    - So $P(C_{1,1} = \text{yellow} \mid G = (2,3)) = P(C_{1,1} = \text{yellow} \mid G = (2,3), C_{1,2} = \text{orange})$
    - I.e., $C_{1,1}$ is ***conditionally independent*** of $C_{1,2}$ ***given*** G

Ghostbusters model, contd.

If we are not able to simplify using conditional independence, would that mean that for the last expression, the state space would be 9 * 4^8 many spaces large after removing c_{3, 3}?

Apply the chain rule to decompose the joint probability model:

- $P(G, C_{1,1}, \ldots C_{3,3}) =$
  $P(G)\, P(C_{1,1} \mid G)\, P(C_{1,2} \mid G, C_{1,1})\, P(C_{1,3} \mid G, C_{1,1}, C_{1,2}) \ldots P(C_{3,3} \mid G, C_{1,1}, \ldots, C_{3,2})$

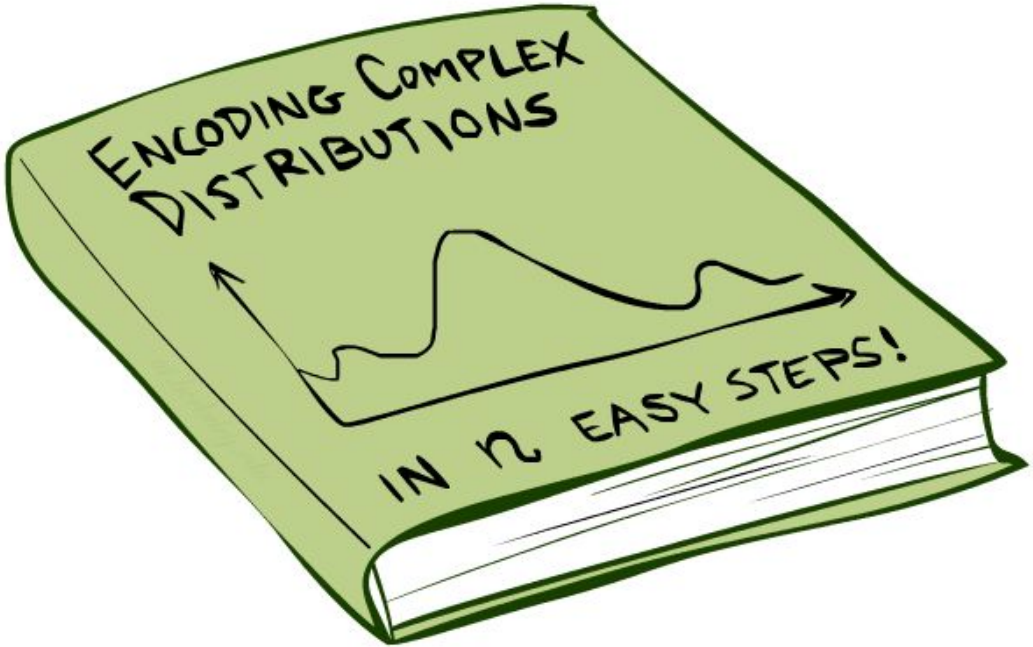
Now simplify using conditional independence:

- $P(G, C_{1,1}, \ldots C_{3,3}) = P(G)\, P(C_{1,1} \mid G)\, P(C_{1,2} \mid G)\, P(C_{1,3} \mid G) \ldots P(C_{3,3} \mid G)$

I.e., conditional independence properties of ghostbuster physics simplify the probability model from ***exponential*** to ***linear*** in the number of squares

Why is the original model exponential?

Bayes Nets: Big Picture



Bayes Nets: Big Picture

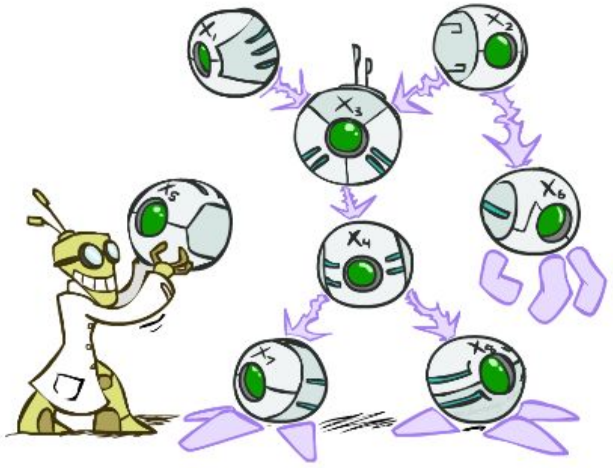
- Bayes nets: a technique for describing complex joint distributions (models) using simple, conditional distributions
  - A subset of the general class of graphical models

Use local causality/conditional independence:

- the world is composed of many variables,
- each interacting locally with a few others

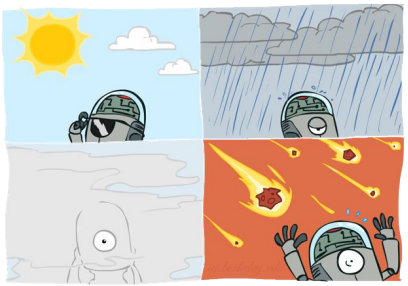
Outline

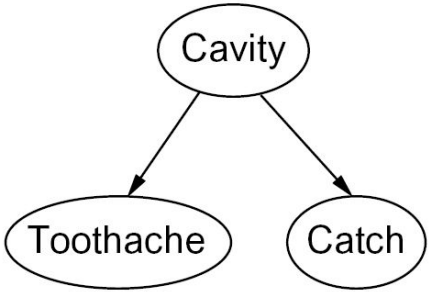
- Representation
- Exact inference
- Approximate inference

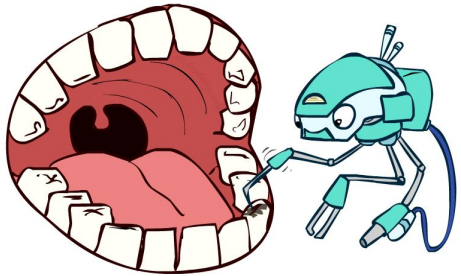


Graphical Model Notation

- **Nodes**: variables (with domains)
  - Can be assigned (observed) or unassigned (unobserved)
- **Arcs**: interactions
  - Indicate “direct influence” between variables
  - Formally: absence of arc encodes conditional independence (more later)
- For now: imagine that arrows mean direct **causation** (in general, they don’t!)

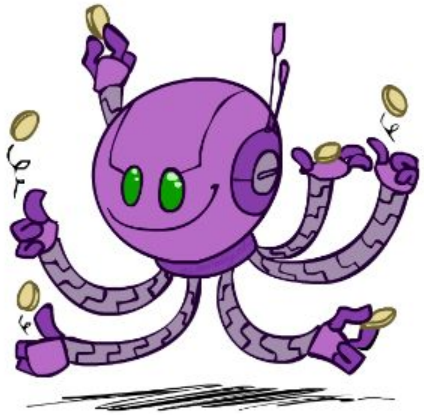






Example: Coin Flips

- *n* independent coin flips

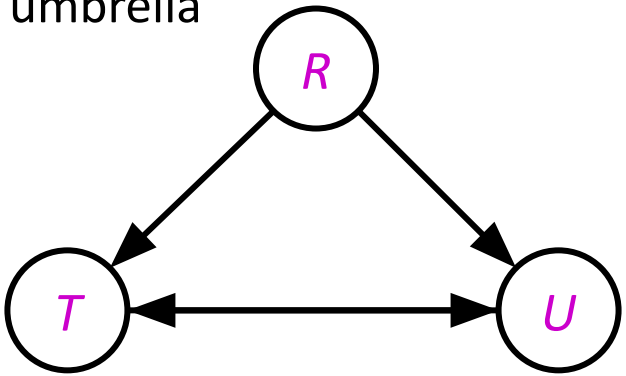


X_1 X_2 . . . X_n

- No interactions between variables: strict independence

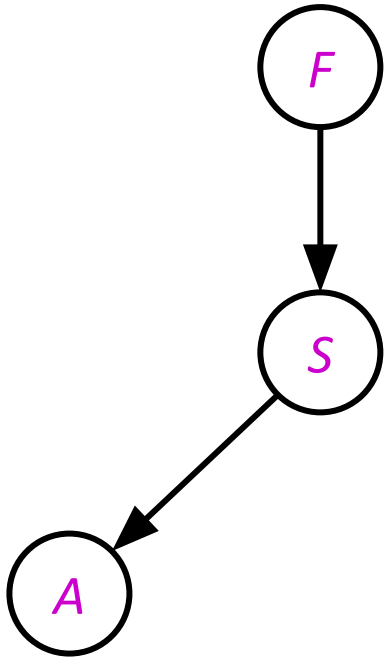
Example: Traffic

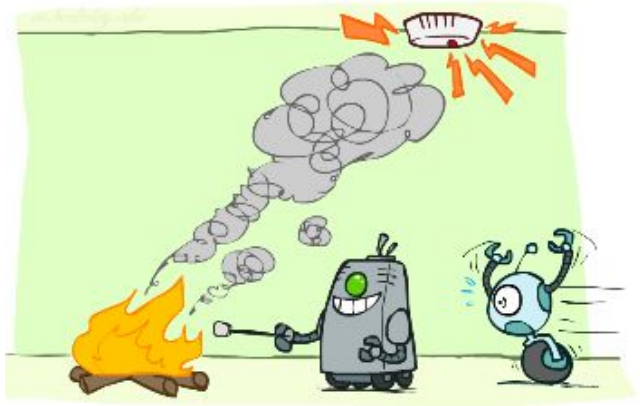
- Variables:
  - ***T***: There is traffic
  - ***U***: I'm holding my umbrella
  - ***R***: It rains

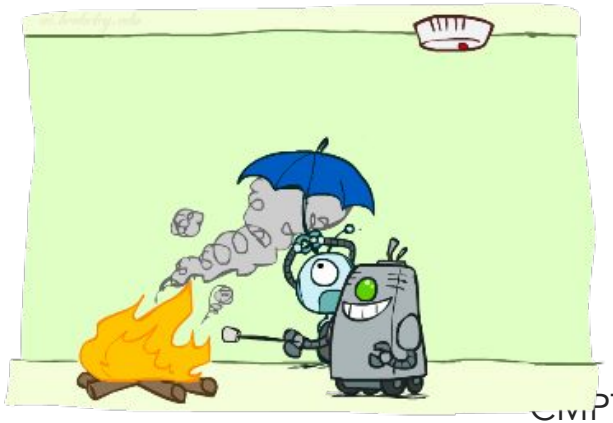


Example: Smoke alarm

- Variables:
  - ***F***: There is fire
  - ***S***: There is smoke
  - ***A***: Alarm sounds

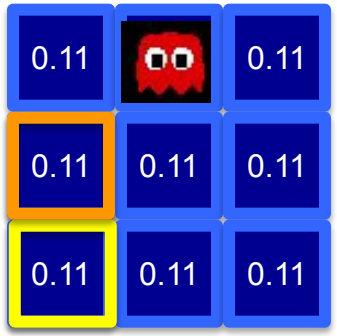




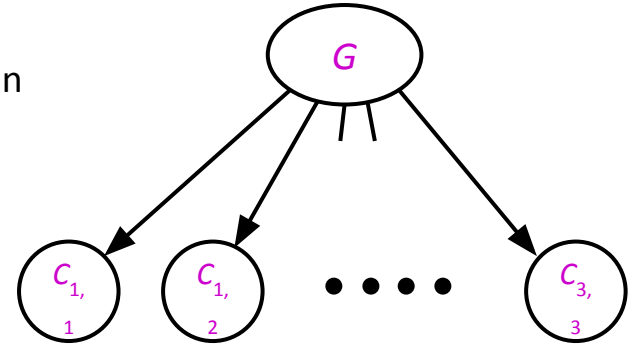


Example: Ghostbusters

- Variables:
  - G: The ghost's location
  - $C_{1,1}, \ldots C_{3,3}$:
    The observation at each location



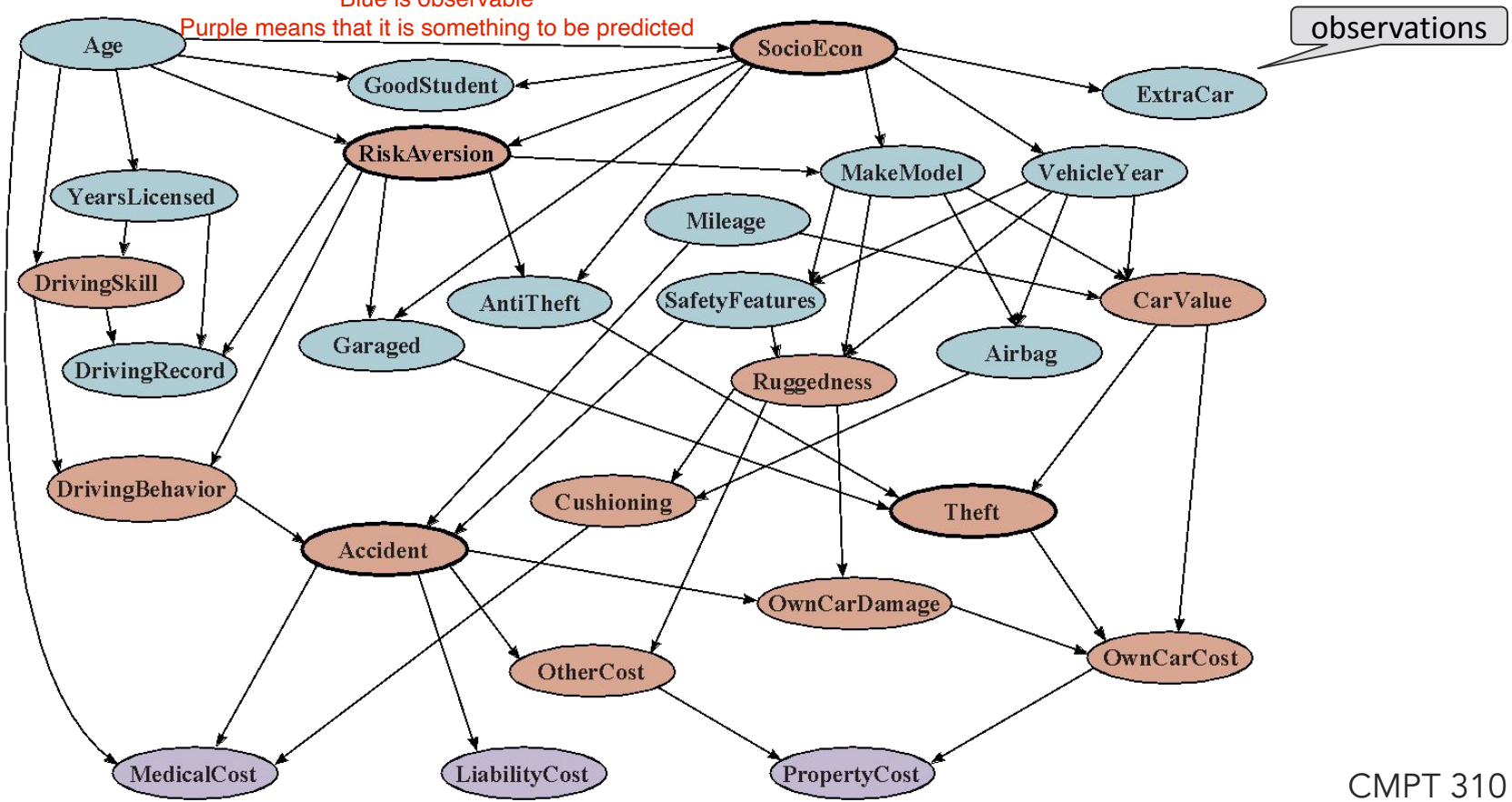
- Want to estimate:
  $P(G \mid C_{1,1}, \ldots C_{3,3})$



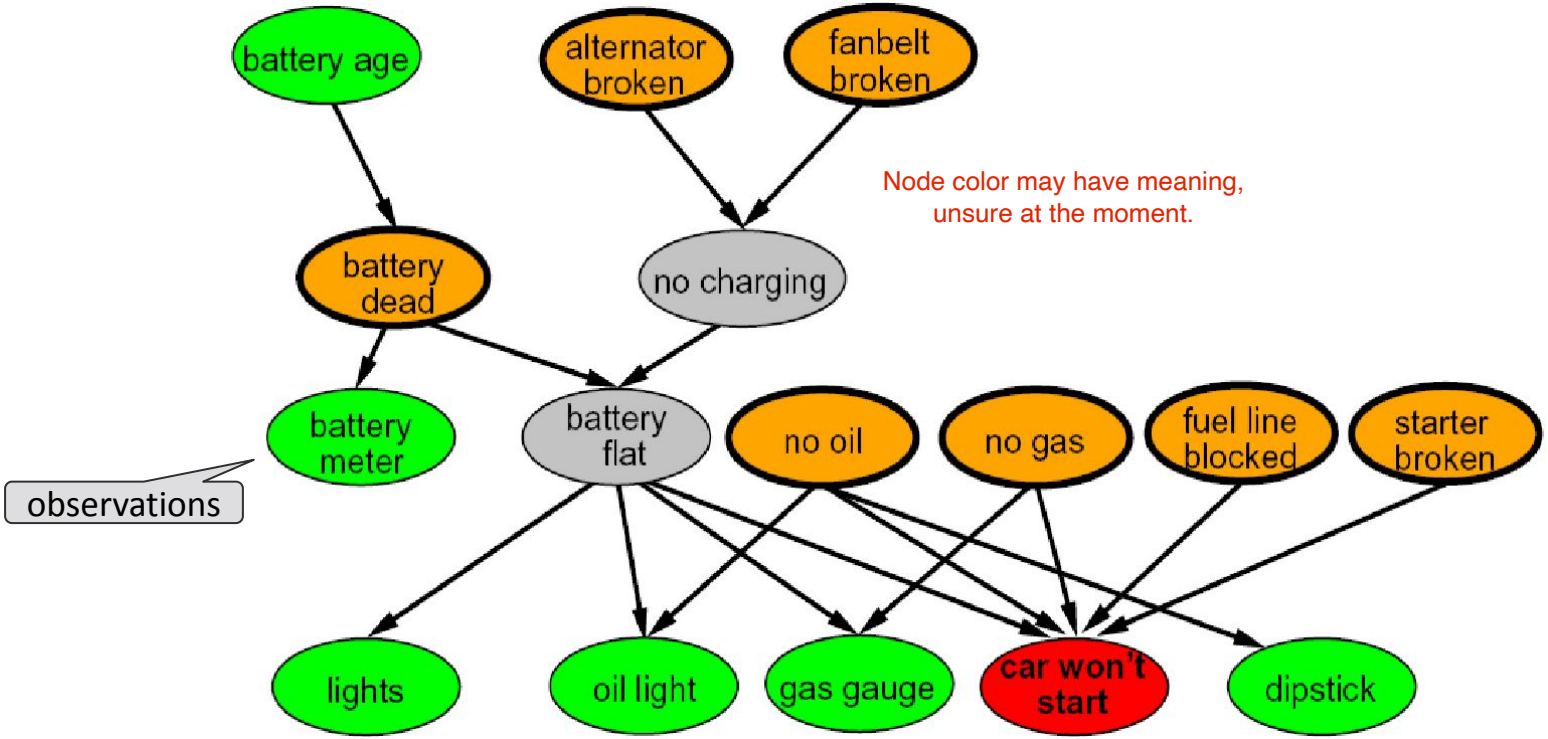
- This is called a ***Naïve Bayes*** model:
  - One discrete query variable (often called the ***class*** or ***category*** variable)
  - All other variables are (potentially) evidence variables
  - Evidence variables are all conditionally independent given the query variable

Example: Car Insurance

Red variables are unobserved, meaning derived?
Blue is observable
Purple means that it is something to be predicted



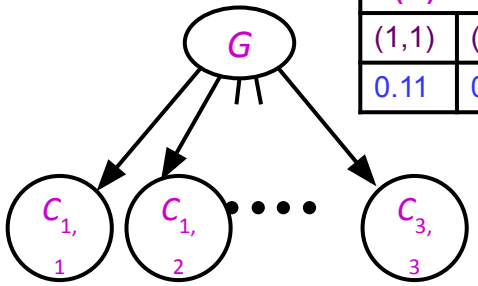
Example: Car Won't Start



Bayes Net Syntax and Semantics

Bayes Net Syntax

- A set of nodes, one per variable X_i
- A directed, acyclic graph
- A conditional distribution for each node given its ***parent variables*** in the graph
  - ***CPT*** (conditional probability table); each row is a distribution for child given values of its parents



P(G)			
(1,1)	(1,2)	(1,3)	…
0.11	0.11	0.11	…

G	P($C_{1,1}$ | G)			
	g	y	o	r
(1,1)	0.01	0.1	0.3	0.59
(1,2)	0.1	0.3	0.5	0.1
(1,3)	0.3	0.5	0.19	0.01
…				

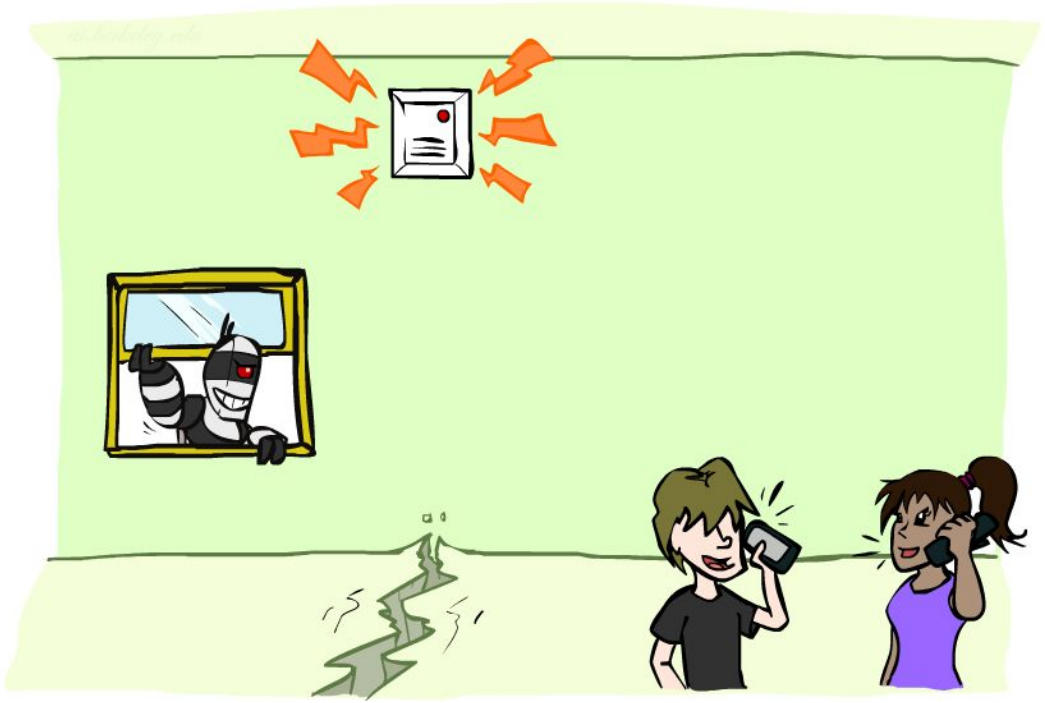
Bayes net = Topology (graph) + Local Conditional Probabilities

Example: Alarm Network

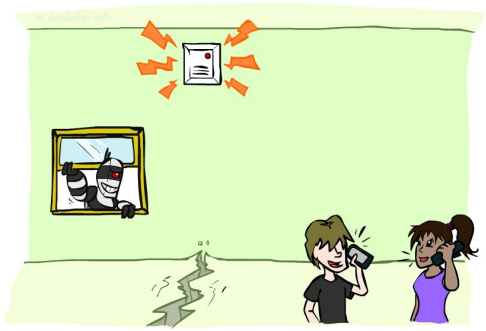
- Variables
  - B: Burglary
  - E: Earthquake
  - A: Alarm goes off
  - J: John calls
  - M: Mary calls

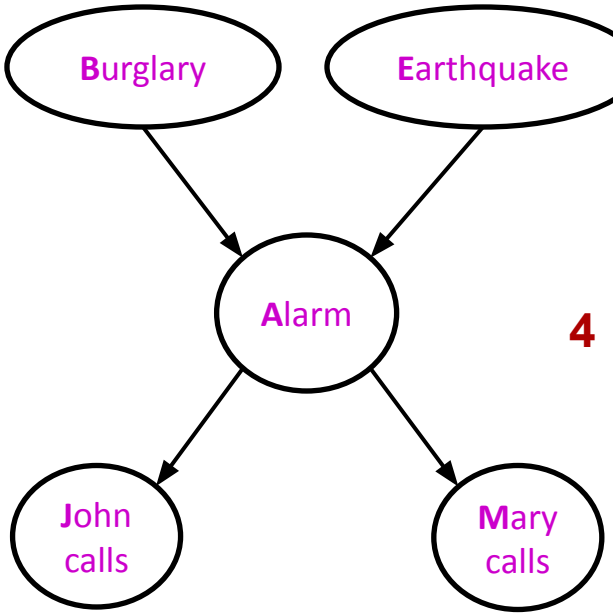
John and Mary are calling about the alarm

Earthquakes can cause the alarm



Example: Alarm Network





P(B)	
true	false
0.001	0.999

1

P(E)	
true	false
0.002	0.998

1

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

4

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

2

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

2

Number of ***free parameters*** in each CPT:

Parent range sizes $d_1,\ldots,d_k$

Child range size d
Each table row must sum to 1

$(d-1) \prod_i d_i$

Bayes net global semantics

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

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

- Exploits sparse structure: number of parents is usually small

Size of a Bayes Net

- How big is a joint distribution over N variables, each with d values?

  d^N

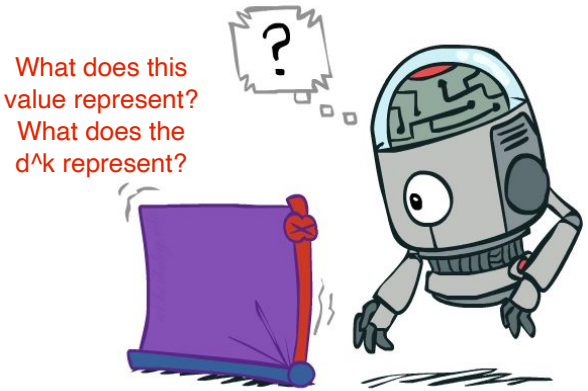
- How big is an N-node net if nodes have at most k parents?

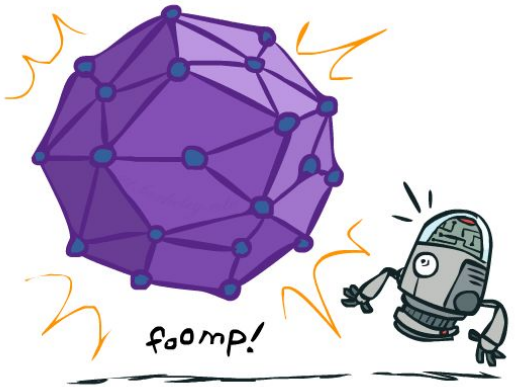
  $O(N * d^k)$

- Both give you the power to calculate $P(X_1, X_2, \ldots, X_N)$

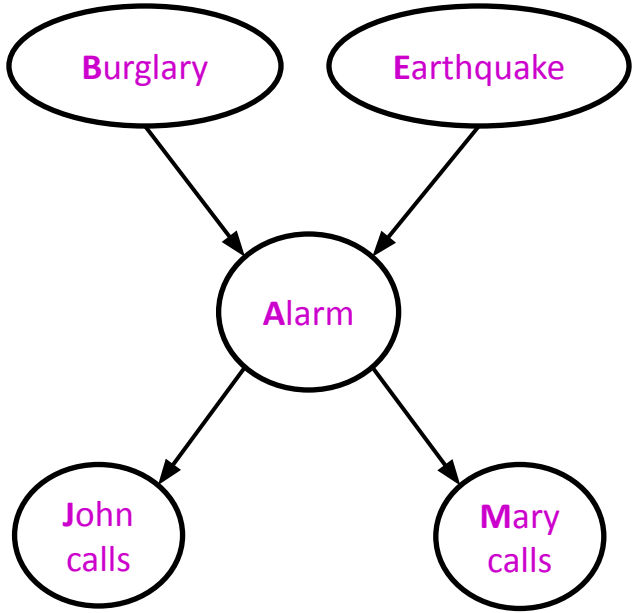
  Sparsity refers to having multiple, smaller distributions compared to one large distribution.

- Bayes Nets: huge space savings with sparsity!
- Also easier to elicit local CPTs
- Also faster to answer queries (coming)





Example



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

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

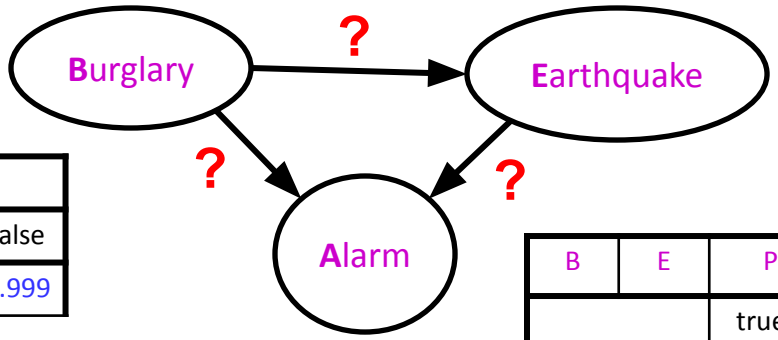
$P(b,\neg e, a, \neg j, \neg m) =$

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

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

Example: Burglary

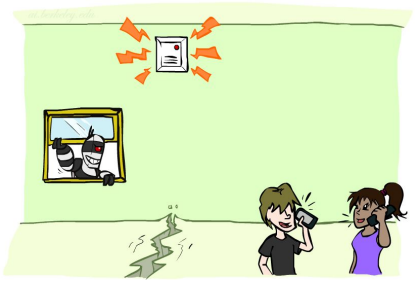
- Burglary
- Earthquake
- Alarm



P(E)	
true	false
0.002	0.998

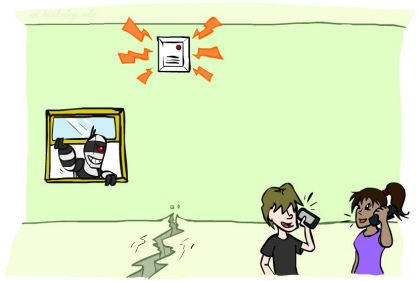
P(B)	
true	false
0.001	0.999

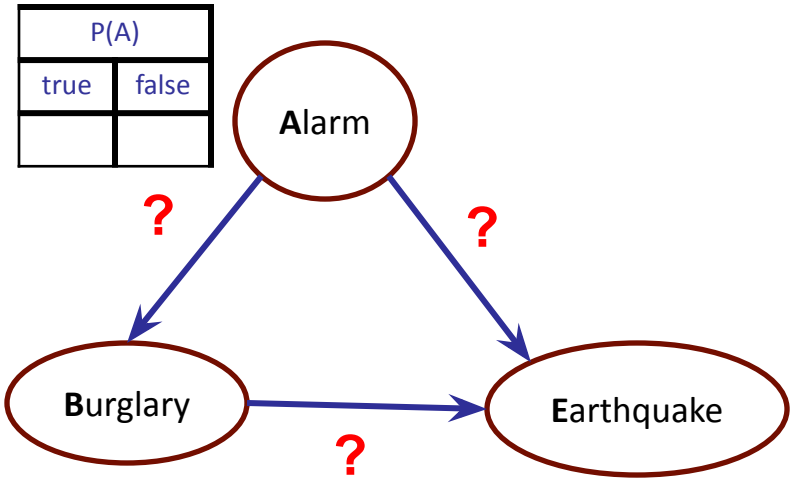
B	E	P(A|B,E)	
		true	false
true	true	0.95	0.05
true	false	0.94	0.06
false	true	0.29	0.71
false	false	0.001	0.999



Example: Burglary

- Alarm
- Burglary
- Earthquake



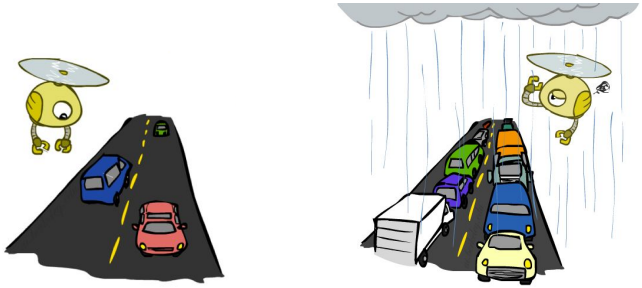


A	P(B|A)	
	true	false
true	?	
false		

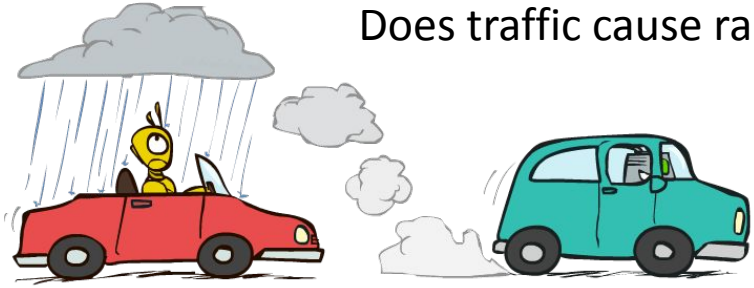
A	B	P(E|A,B)	
		true	false
true	true	?	
true	false		
false	true		
false	false		

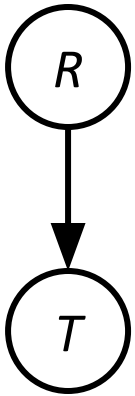
Example: Traffic

Does rain cause traffic?



Does traffic cause rain?





$P(R)$

+r	1/4
-r	3/4

$P(T|R)$

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

$P(T,R)$

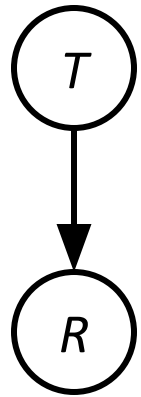
+r	+t	3/16
+r	-t	1/16
-r	+t	6/16
-r	-t	6/16

$P(T)$

+t	9/16
-t	7/16

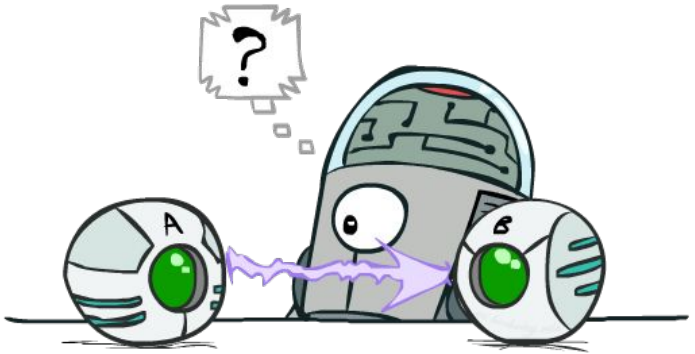
$P(R|T)$

+t	+r	1/3
	-r	2/3
-t	+r	1/7
	-r	6/7



Causality?

- When Bayes' nets reflect the true causal patterns:
  - Often simpler (nodes have fewer parents)
  - Often easier to think about
  - Often easier to elicit from experts



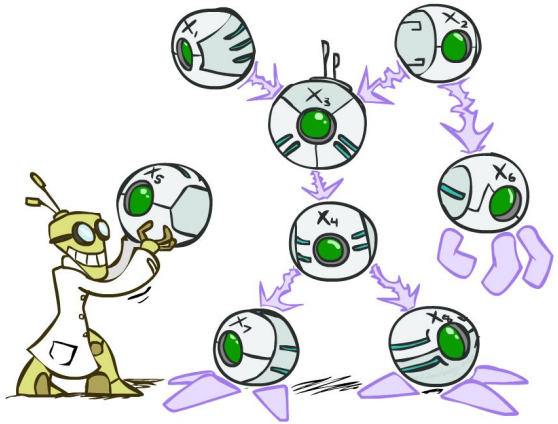
- BNs need not actually be causal
  - Sometimes no causal net exists over the domain (especially if variables are missing)
  - E.g. consider the variables *Traffic* and *Rain*
  - End up with arrows that reflect correlation, not causation

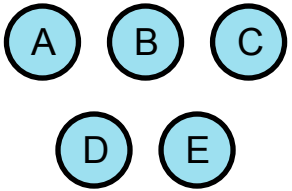
- What do the arrows really mean?
  - Topology may happen to encode causal structure
  - Topology really encodes conditional independence

$$P(x_i|x_1, \ldots x_{i-1}) = P(x_i|parents(X_i))$$

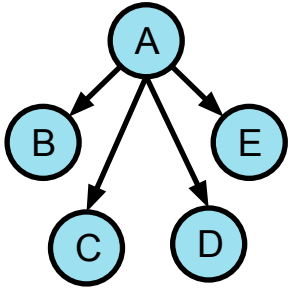
Summary

- Independence and conditional independence are important forms of probabilistic knowledge
- **Bayes net** encode **joint distributions** efficiently by taking advantage of **conditional independence**
  - Global joint probability = product of local conditionals
- Allows for flexible tradeoff between model accuracy and memory/compute efficiency

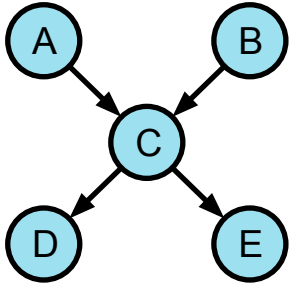




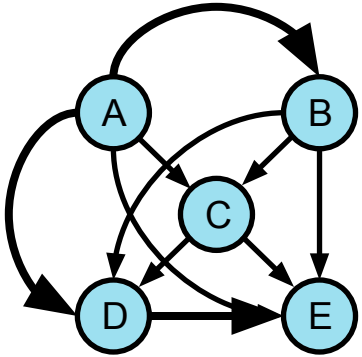
Strict Independence



Naïve Bayes



Sparse Bayes Net



Joint Distribution