

Graphical Models

MLSS 2010, Canberra

Mostly
Bayes
Nets

Very
Introductory

Scott Sanner

NICTA / ANU

First.Last@nicta.com.au

Graphical Models

- One of the **most important tools** in your machine learning and AI inference toolbox

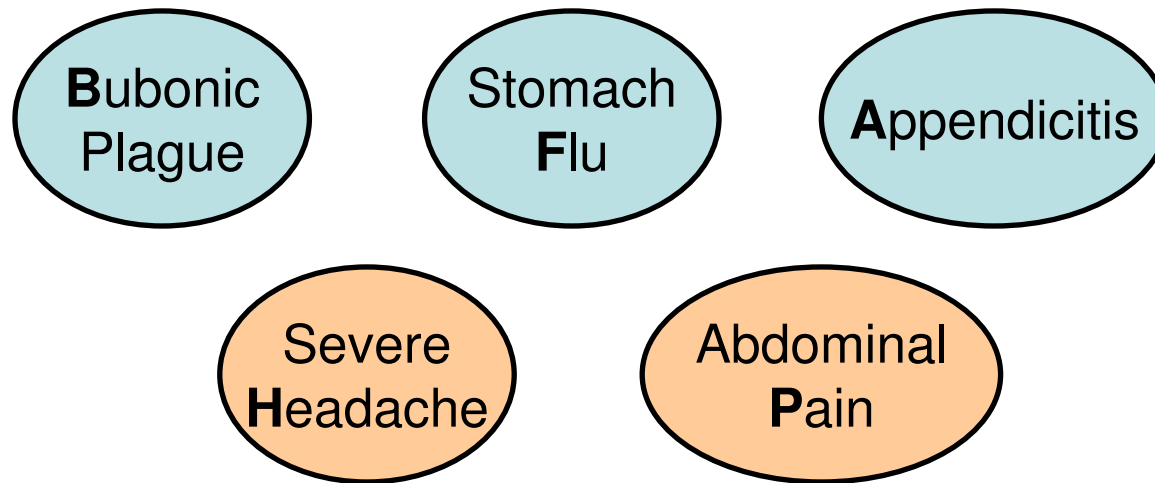


- Scott's applied view on GMs:
 - Formalizing intuitions
 - Build from ground up
 - Implementation
 - If you can implement it, then you understand it!
 - Models and data structures

Graphical Models

Definition:

- **compact specification of joint probability**
- e.g., have the binary variables B, F, A, H, P:

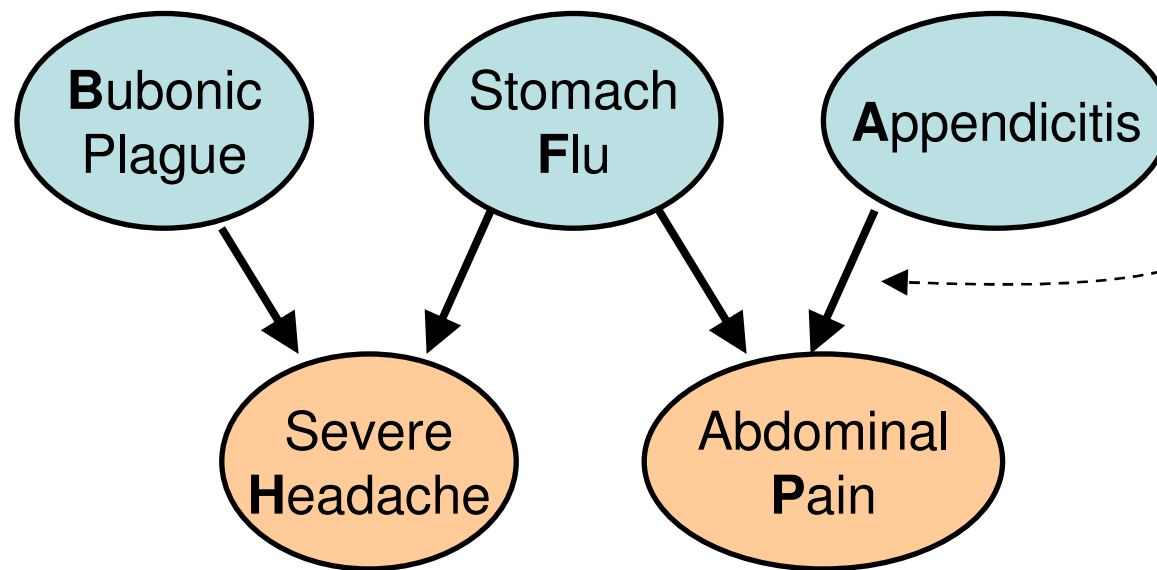


GMs can represent $P(B, F, A, H, P)$ compactly

Graphical Models

What makes it compact?

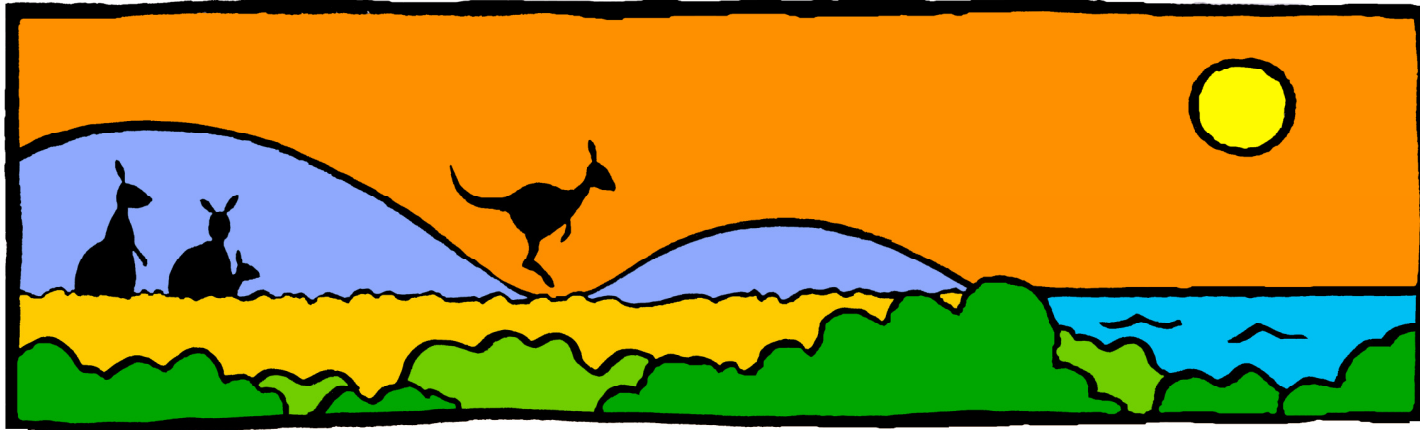
- specify conditional independence (CI) with edges



(note: *graphical* b/c graph properties \Rightarrow CI)

Graphical Models

- Why should you care?
 - Exponential **space savings** in representation
 - Exponential **time savings** in inference
 - Exponential **data complexity reduction**
 - # samples needed to learn “good” model

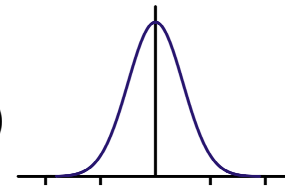


Graphical Models

Foundations

Random Variables

- For the purpose of this tutorial
 - **Random variable (RV)** denoted by uppercase letter
 - e.g., X
 - RVs take **value assignments** $X = x$ where X is a
 - Discrete RV if
 - x is in a countable set (binary: $x \in \{0,1\}$; or dice)
 - Continuous RV if
 - x is in an uncountable set (real: $x \in \text{Reals}$)
 - Write $x \in X$ for possible value assignments of X



Notation abuse

Random Variables

- Probability distributions
 - For all x , $P(X = x) \in [0,1]$
 - $P(X = x)$ is a proper distribution

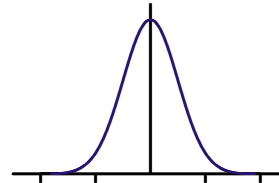
- Discrete RV:

$$\sum_{x \in X} P(X = x) = 1$$



- Continuous RV:

$$\int_{x \in X} P(X = x) dx = 1$$



- Write $P(x)$ for $P(X=x)$, write $P(X)$ for full distribution

Random Variables

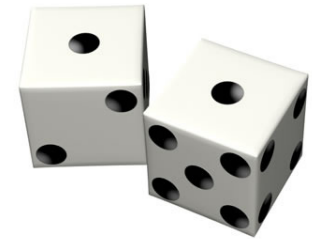
- **Representing** probability distributions

- Discrete RV: **tabular**

finite

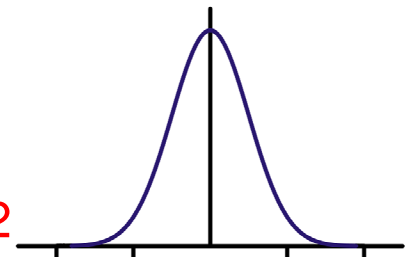
$P(X) =$

X	Pr
1	.1
2	.1
3	.1
4	.2
5	.2
6	.3



- Continuous / ∞ RV: **function**


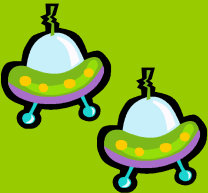

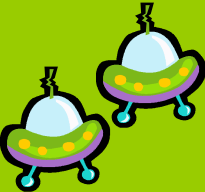
e.g., $P(X = x) \propto \exp (x - \mu)^2 / \sigma^2$



Joint Distributions on RVs

- Aliens in your backyard

$$P(R,C) =$$


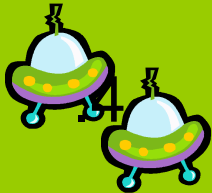

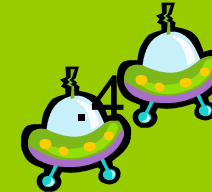
	C=1	C=2
R=1		
R=2		

Joint Distributions on RVs

- Aliens in your backyard

$P(R,C) =$

R	C	Pr
1	1	0
1	2	.4
2	1	.2
2	2	.4

	$C=1$	$C=2$
$R=1$		
$R=2$		

Joint Distributions on RVs

- Aliens in your backyard

$P(R,C) =$

R	C	Pr
1	1	0
1	2	.4
2	1	.2
2	2	.4

	$C=1$	$C=2$
$R=1$	0	.4
$R=2$.2	.4

- $P(R=2, C=2) = .4$

Marginalize over C

- $P(R=2) = \sum_{c \in \{1,2\}} P(R=2, C=c) = .6$

- $P(R=1|C=2) = P(R=1, C=2)/P(C=2) = .5$

Condition on $C=2$

Example from Andrew Moore @ CMU/Google

Rules of Probability

- Joint and conditional distributions:

$$P(A, B) = P(A|B) \cdot P(B) = P(B|A) \cdot P(A)$$

- Marginalization:

$$P(A) = \sum_{b \in B} P(A, B = b)$$

Don't memorize!
Derive from first
principles!

- Conditional probability & Bayes rule:

$$P(A|B) = \frac{P(A, B)}{P(B)} = \frac{P(B|A) \cdot P(A)}{\sum_{a \in A} P(B|a) \cdot P(a)}$$

Manipulating Distributions

- Sometimes we don't **just** want $P(R=1, C=2) = .4$
- We want to work with full distributions $P(R, C)$

<i>R</i>	<i>C</i>	<i>Pr</i>
1	1	0
1	2	.4
2	1	.2
2	2	.4

- How to apply previous rules to full distributions?
 - easy, just do once for each case and store in table...

Manipulating Discrete Distributions

- Marginalization

$$\sum_b P(A, b) = P(A)$$

$$\sum_b \begin{array}{|c|c|c|} \hline A & B & Pr \\ \hline 0 & 0 & .1 \\ \hline 0 & 1 & .3 \\ \hline 1 & 0 & .2 \\ \hline 1 & 1 & .4 \\ \hline \end{array} = \begin{array}{|c|c|} \hline A & Pr \\ \hline 0 & .4 \\ \hline 1 & .6 \\ \hline \end{array}$$

Manipulating Discrete Distributions

- Binary Multiplication

$$P(A) \cdot P(B|A) = P(A, B)$$

<i>A</i>	<i>Pr</i>		<i>A</i>	<i>B</i>	<i>Pr</i>		<i>A</i>	<i>B</i>	<i>Pr</i>
<i>0</i>	<i>.7</i>		<i>0</i>	<i>0</i>	<i>.1</i>		<i>0</i>	<i>0</i>	<i>.07</i>
<i>1</i>	<i>.3</i>	<i>·</i>	<i>0</i>	<i>1</i>	<i>.9</i>	<i>=</i>	<i>0</i>	<i>1</i>	<i>.63</i>
			<i>1</i>	<i>0</i>	<i>.2</i>		<i>1</i>	<i>0</i>	<i>.06</i>
			<i>1</i>	<i>1</i>	<i>.8</i>		<i>1</i>	<i>1</i>	<i>.24</i>

- Same principle holds for all binary ops
 - +, -, /, max, etc...

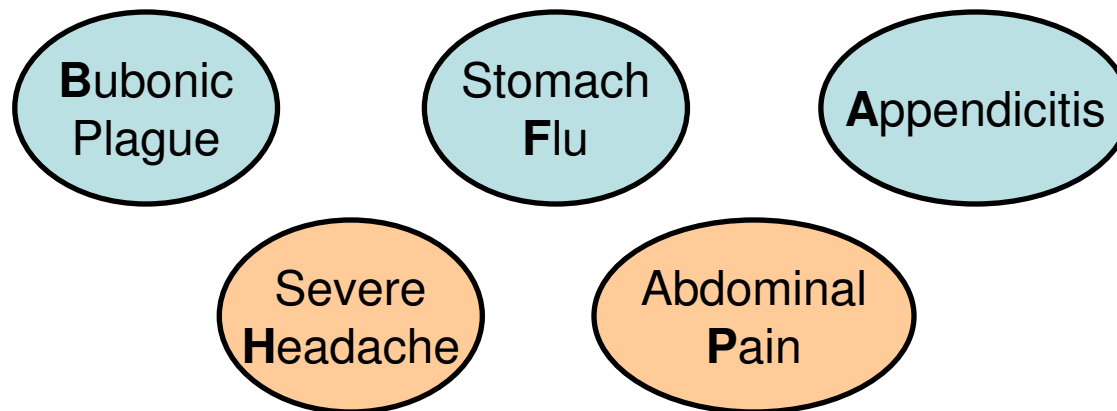
Nearing End of Prob 101

- We can
 - represent joint distributions
 - marginalize
 - condition
 - perform Bayes rule
- Q: But why is this useful?
- A: All you need to answer probabilistic queries

Fundamental Operation of Probabilistic Inference

- **Problem:**

- Given a **joint distribution** $P(B, F, A, H, P)$



- Given evidence: $H=\text{true}$, $P=\text{false}$
- Want to know probability of B given evidence

- **Answer:** evaluate $P(B \mid H=\text{true}, P=\text{false})$

Computing Probabilistic Queries

- Evaluate: $P(B \mid H=\text{true}, P=\text{false})$

given: $P(B, F, A, H, P)$ as table \longrightarrow

B	...	P	Pr
<i>true</i>	...	<i>true</i>	<i>.03</i>

- Step 0: Select lines for evidence in table
 - Reduce from 32 to 8 rows
- Step 1: Marginalize out non-query / non-evidence RVs
 - $P(B, H=\text{true}, P=\text{false}) = \sum_f \sum_a P(B, f, a, H=\text{true}, P=\text{false})$
- Step 2: Marginalize out query
 - $P(H=\text{true}, P=\text{false}) = \sum_b P(B=b, H=\text{true}, P=\text{false})$
- Step 3: Evaluate conditional probability
 - $$P(B \mid H=\text{true}, P=\text{false}) = \frac{P(B, H=\text{true}, P=\text{false})}{P(H=\text{true}, P=\text{false})}$$

Key Points

- Goal is to do probabilistic inference
- Need a joint distribution
 - RVs specified by human
 - Parameters can be learned (later)
- All probabilistic queries $P(Q|E)$ computed by
 - Instantiation of RVs (evidence)
 - Marginalization,
 - Multiplication & division *on distributions*

Learning Check

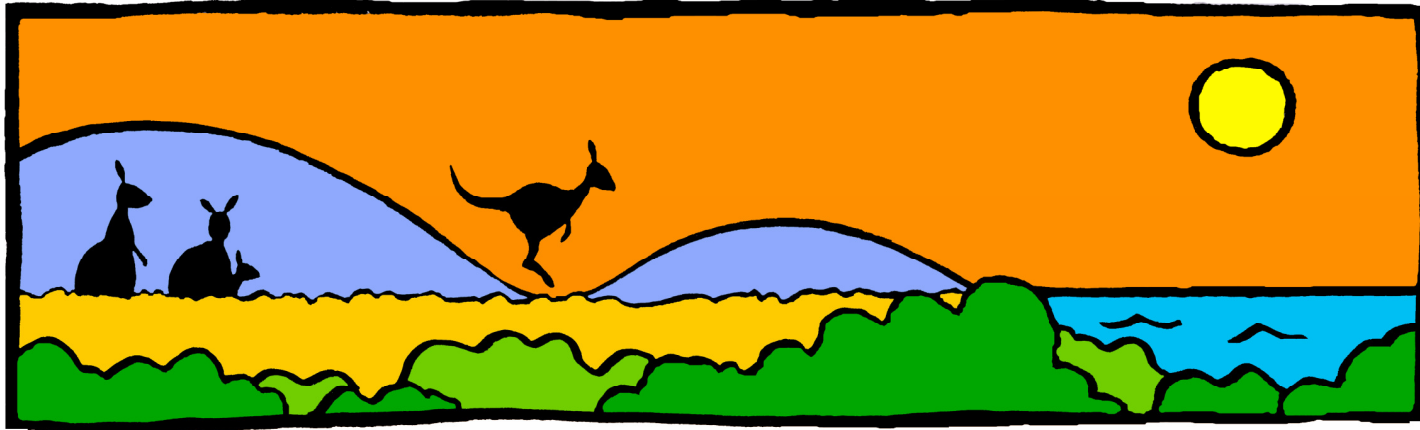
- Given **joint probability** over discrete RVs

$$P(X_1, \dots, X_{20}) =$$

X_1	...	X_{20}	Pr
0	...	0	.03
...

represented in **tabular format**

... can you **write code** to compute query, e.g.,
 $P(X_1, X_{11} \mid x_4, x_7, x_{17})$?



Graphical Models

Directed Graphical Models

Probabilistic Queries in the Real World

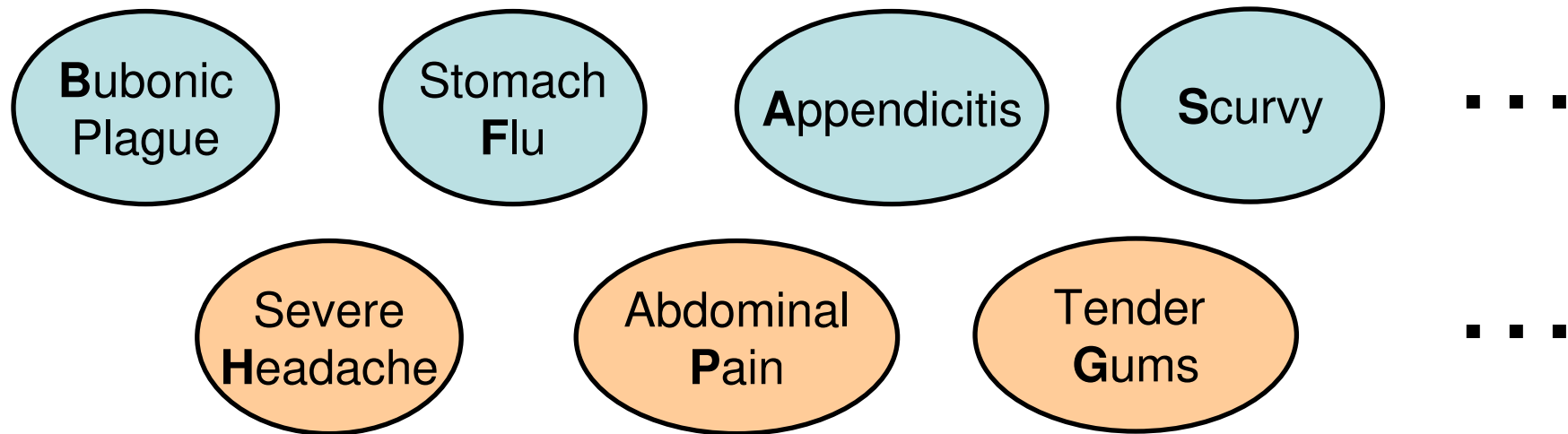
- Just need a
 - joint distribution
 - operations (marginalization, binary ops)and we can compute **any** prob. query, right?
- **If available time, space, and data are infinite**

Let's Look at a Medical Example

(one of the first fielded
applications of graphical models)

Scurvy, it's not just for Pirates

- You're a doctor
 - You regularly diagnose about 100 ailments
 - As evidence you use 200 tests / questions



Joint Distributions, Problem 1:

- How do you represent a joint distribution over 300 binary variables?

X_1	...	X_1	Pr
0	0	0	.03
0	0	1	.01
...
...
...
...

- Tabular? $2^{300} - 1$ rows!

Joint Distributions, Problem 2:

- You're Google and 2^{300} is a small number
 - So tabular doesn't scare you

X_1	...	X_{300}	Pr
0	...	0	.01
...

- How long does it take to compute $P(Q|E)$?
 - Note: have to **visit each row at least once**
- Takes time $\Omega(2^{300})$

Joint Distributions, Problem 3:

- You've sipped from the fountain of youth
 - So you have an eternity
- How much data does it take to learn probabilities
 - Remember alien spaceships? Just frequency ratios
 - Note: need a fair number of samples (30?) per row

X_1	...	X_{300}	Pr
0	...	0	.01
...

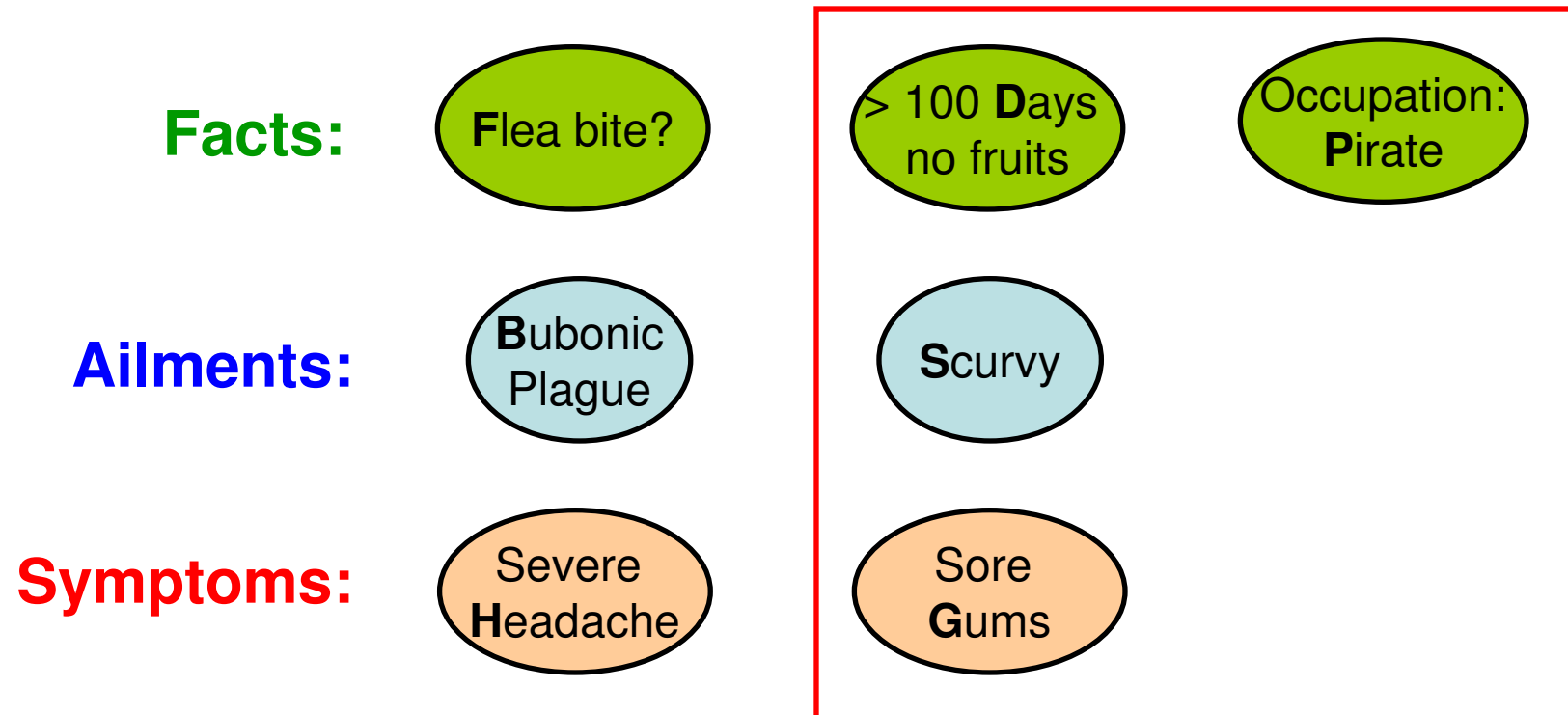
- Need at least $\Omega(30 \cdot 2^{300})$ labeled examples
 - i.e., doctor visits

Sorry folks, tutorial is over

Using probabilities in
practice is hopeless

Scurvy to the rescue, sort of

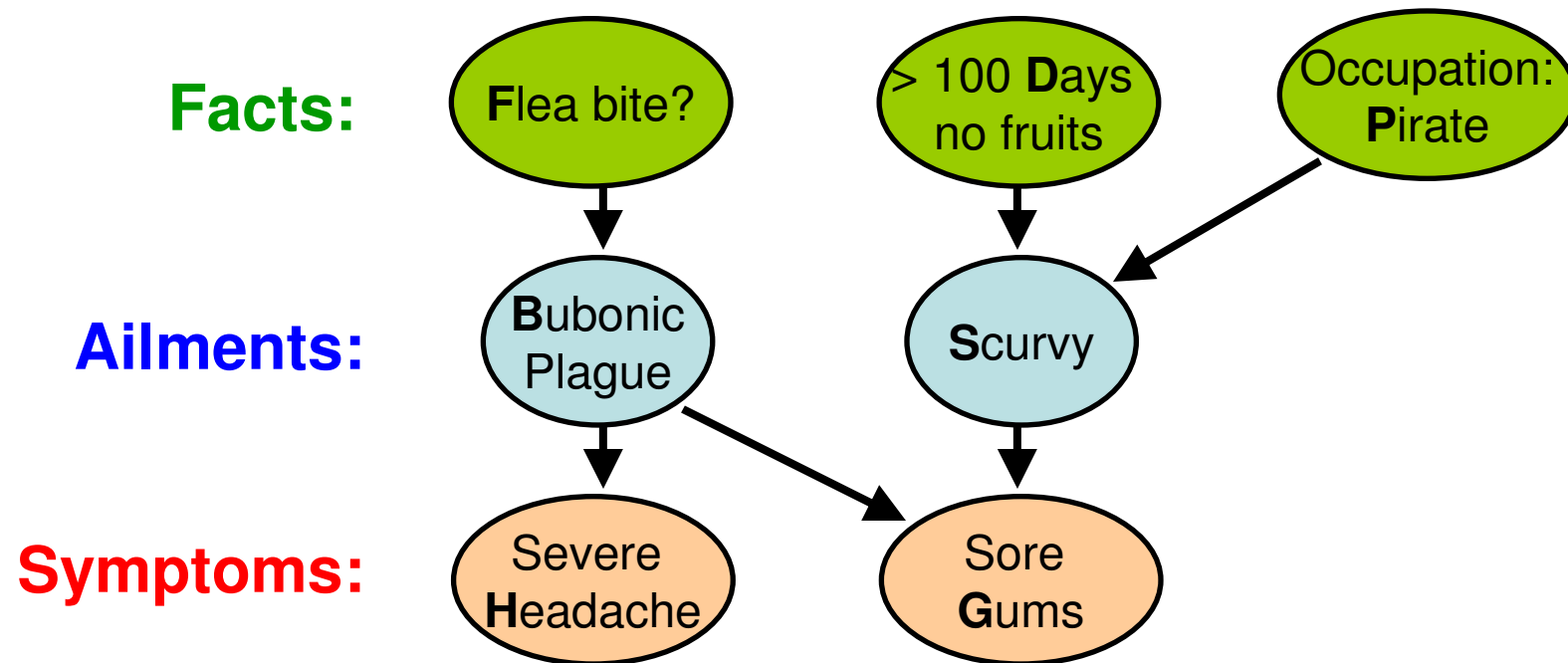
- Not all variables important in predicting others



- In predicting **Scurvy**, do **Flea bites** matter?
 - Should be able to predict scurvy probability from

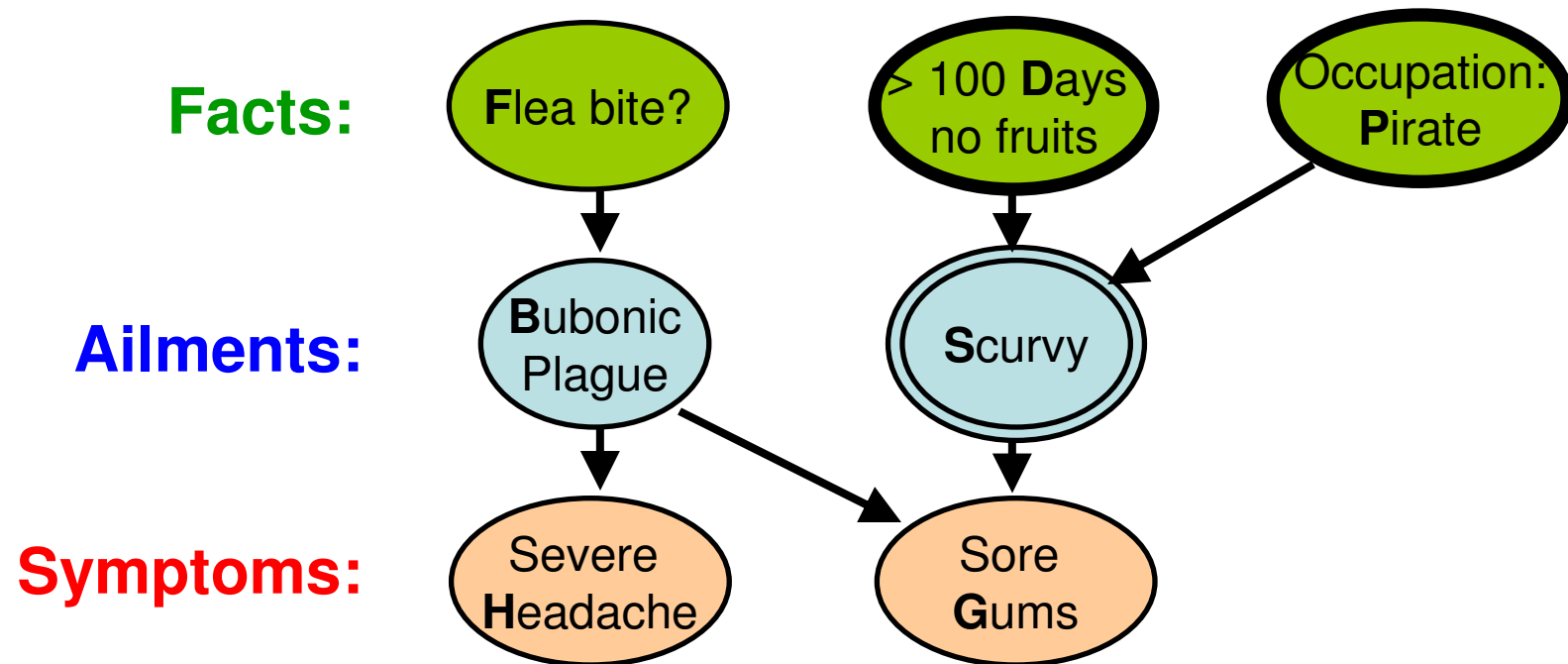
Exploiting Structure

- Key idea is to exploit known dependences
 - Draw arc when one variable known to influence other
 - For directed GMs, any directed acyclic graph (DAG) is allowed



Conditional Independence I

- Every DAG implies set of conditional independences
 - Use thick ovals for evidence, double ovals for query

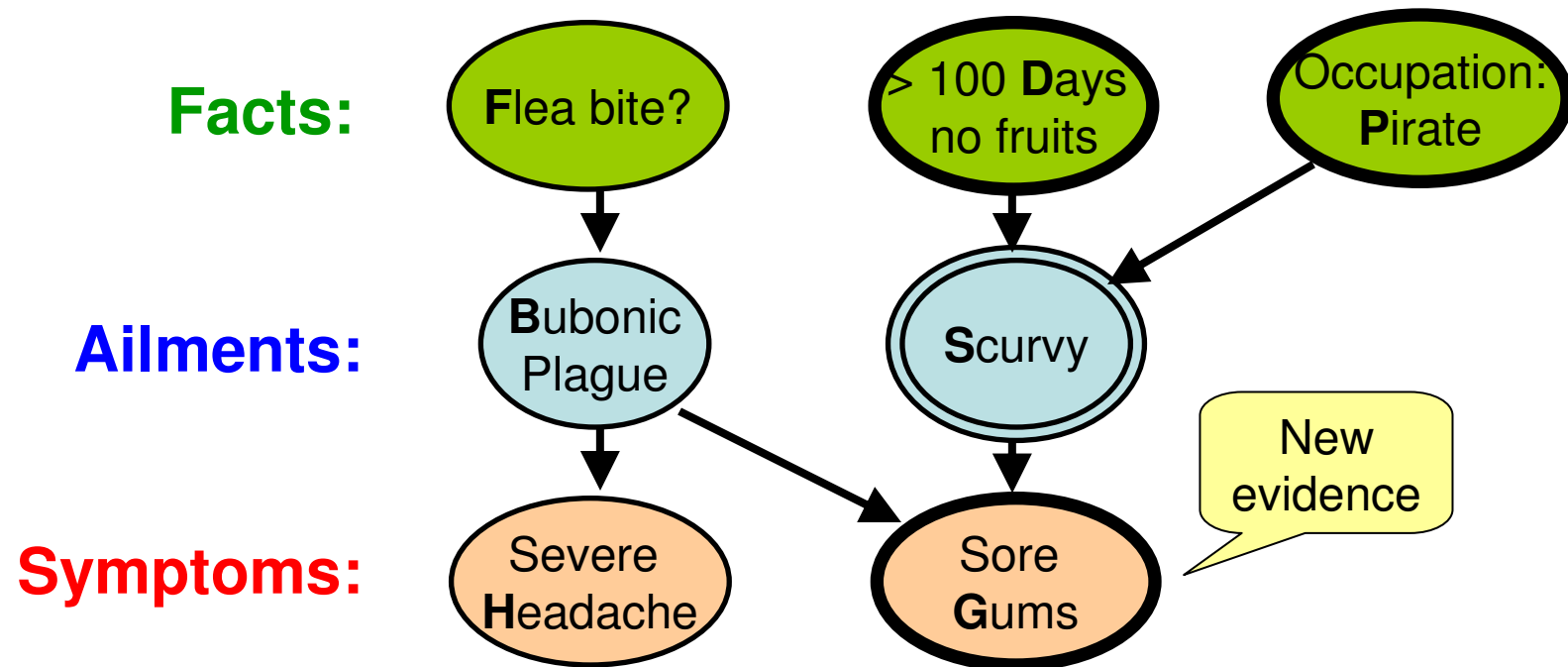


- Then following are two of the CI ($\perp\!\!\!\perp$) implications:

$$S \perp\!\!\!\perp B | D, P \qquad S \perp\!\!\!\perp F | D, P$$

Conditional Independence II

- Every DAG implies a set of conditional independences
 - Use thick circles for evidence, double circle for query

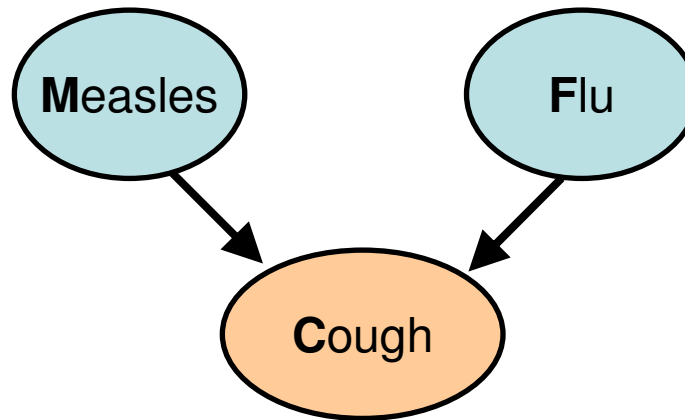


- Then following two are **not** CI ($\perp\!\!\!\perp$) implications:

$$\cancel{S \perp\!\!\!\perp B | D, P, G} \quad \cancel{S \perp\!\!\!\perp F | D, P, G}$$

Conditional Independence III

- That's odd... **adding evidence** made previously independent variables now dependent
- Diagnosis example
 - **M** measles, **F** flu, **C** cough



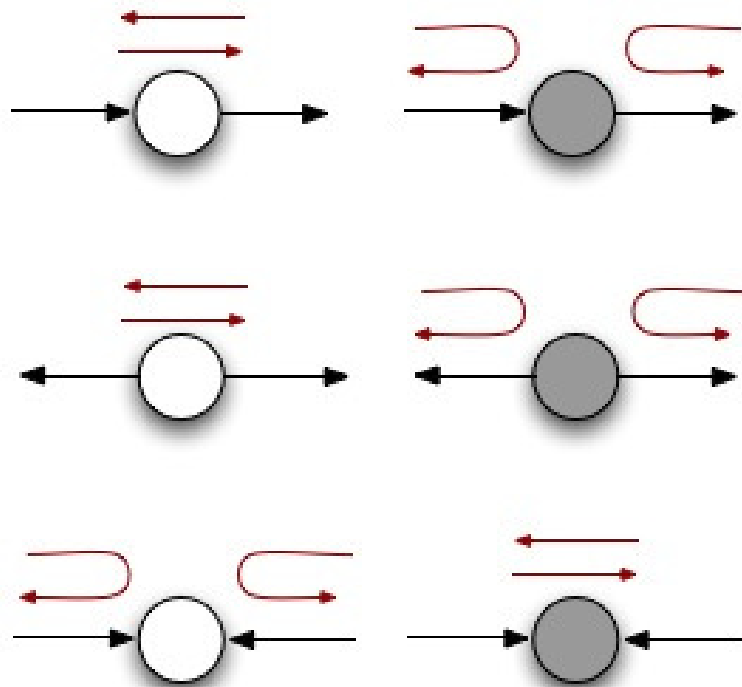
- Note what happens to CI when C (un)observed:

$$M \perp\!\!\!\perp F | \emptyset$$

~~$$M \perp\!\!\!\perp F | C$$~~

Explaining away

Conditional Independence IV

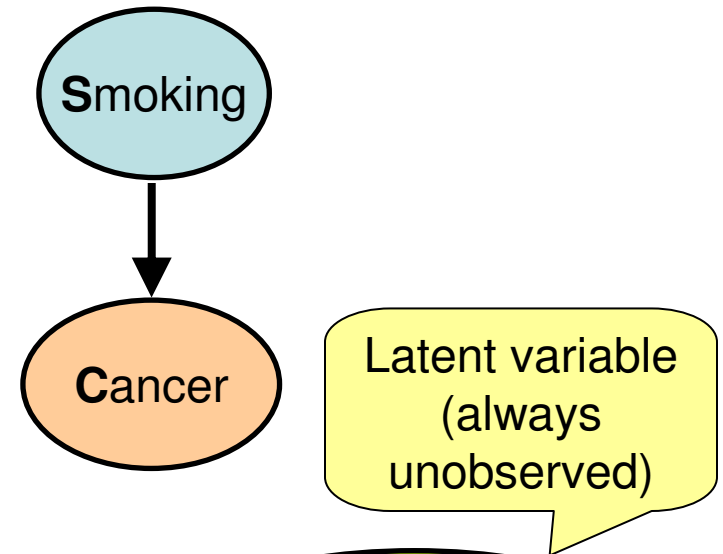


Bayes-Ball Rules In A Nutshell

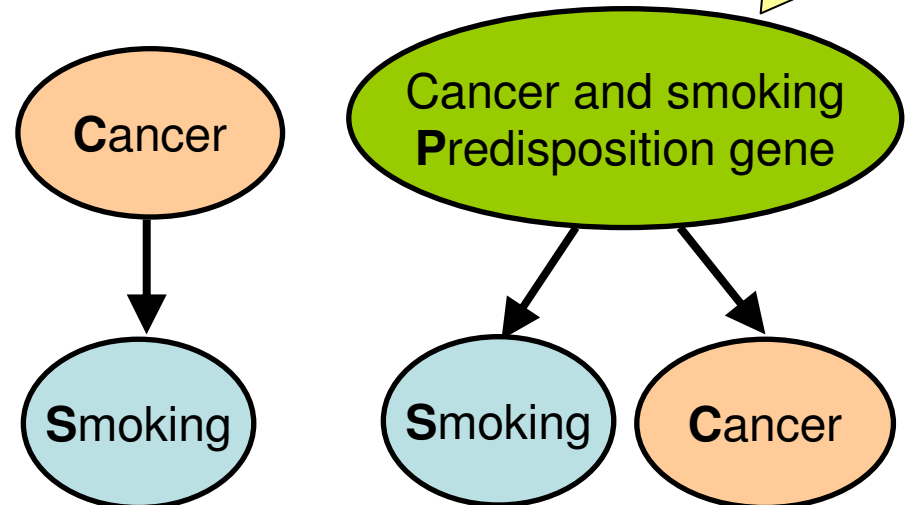
- How to tell what CIs implied by directed GM?
- Graph-theoretic property called **dependency separation (D-separation)**
- Compute using “Bayes ball”
 - Observed nodes shaded
 - X & Y are CI iff bounce a ball from X to Y (or reverse) using bouncing rules at left

Conditional Independence V

- Is a directed link causal?



- Not necessarily
 - can represent exact same GM using...
 - Implies same CIs



OK, So DAG implies Cls

But still don't know how to
represent, infer, learn...

Bayes nets (BNs): Directed GMs

- Simple BN rule for joint

- Write down product of all variables conditional on parents (if any)

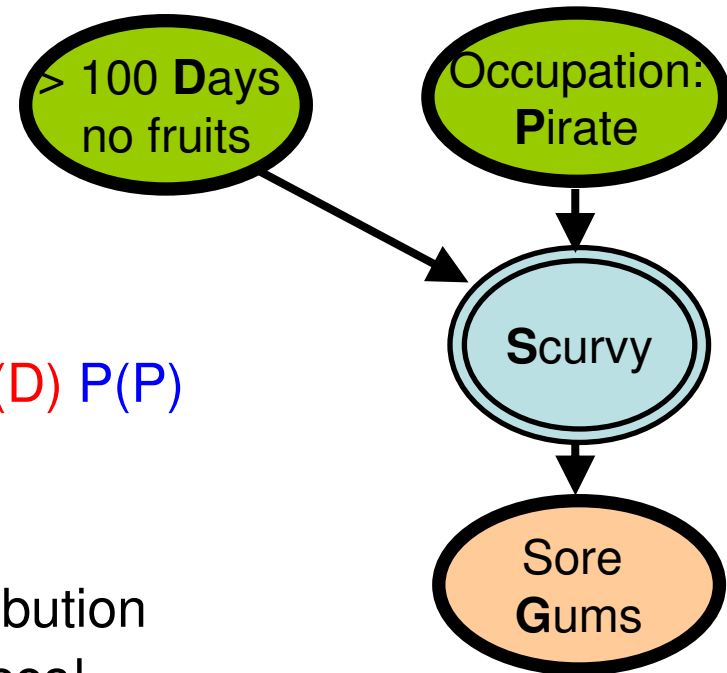
- $P(D,P,S,G) = P(G|S) P(S|D,P) P(D) P(P)$

- If network is a DAG

- **Always** gives a proper joint distribution
- CIs are probabilistic independences!

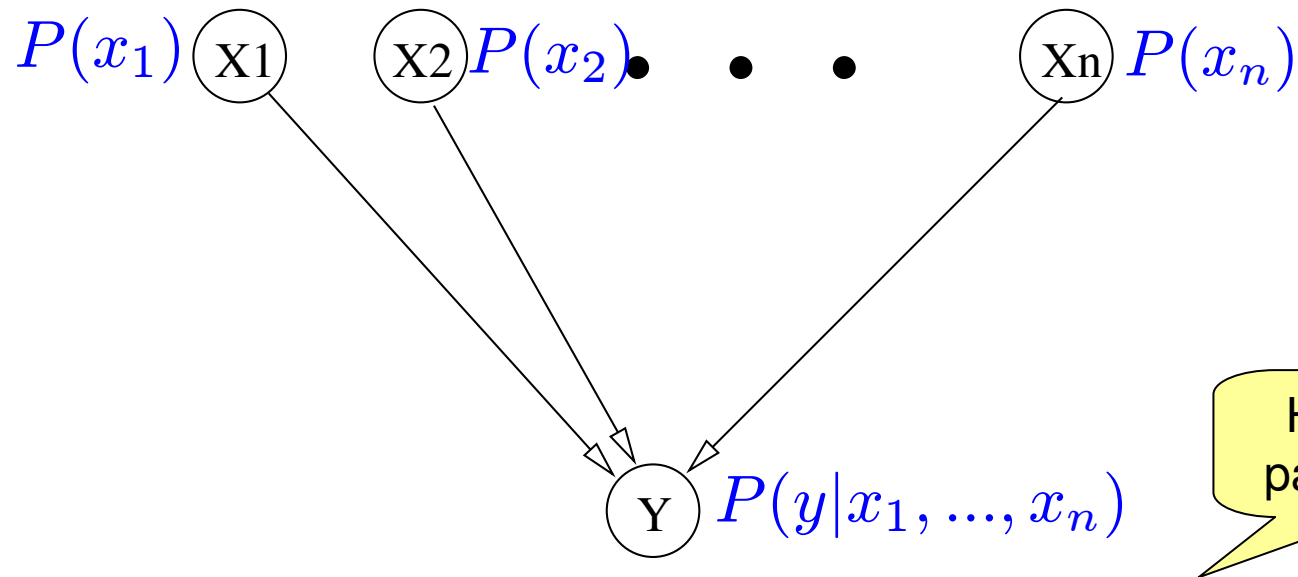
- Note compactness by exploiting (in)dependences

- How many parameters in tabular joint? 31
- How many parameters in this BN? 8



Graphical Model Example II

- Each node has associated conditional probability
- Root nodes correspond to prior probabilities

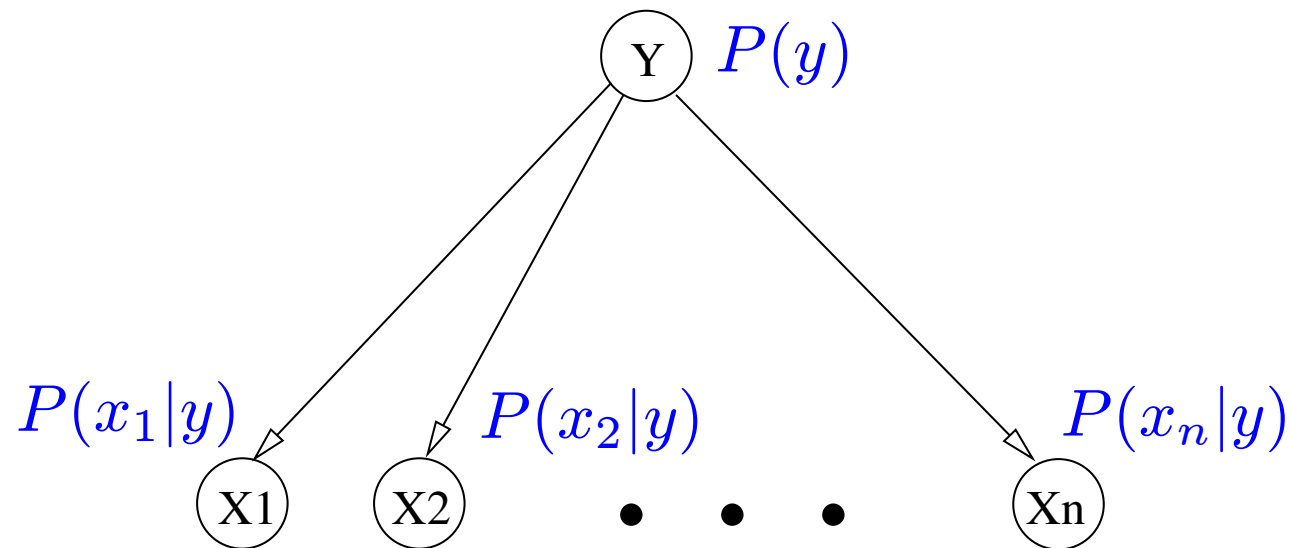


How many parameters?

$$P(y, x_1, \dots, x_n) = P(y|x_1, \dots, x_n)P(x_1) \cdots P(x_n)$$

Graphical Model Example III

- Model for naïve Bayes classifier



$$P(y, x_1, \dots, x_n) = P(x_1|y) \cdots P(x_n|y)P(y)$$

How many
parameters?

Aside: Probabilistic Independence

- Conditional independence for Bayes nets implies probabilistic independence

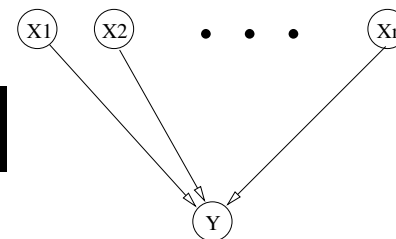
$$A \perp\!\!\!\perp B \Rightarrow P(A, B) = P(A) \cdot P(B)$$

$$A \perp\!\!\!\perp B|C \Rightarrow P(A, B|C) = P(A|C) \cdot P(B|C)$$

It's Query Time

- Clear **space savings** for joint distribution using some **graphical models**
- Can **also** exploit graphical model structure during probabilistic query evaluation...
 - e.g., **variable elimination (VE)**

Variable Elimination (VE) I



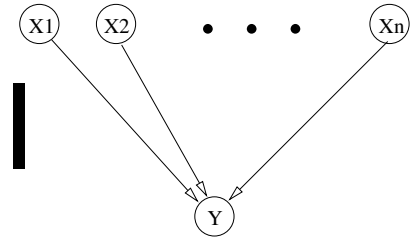
- When marginalizing over y , try to factor out all probabilities independent of y :

$$\begin{aligned} P(X_1) &= \sum_{y, x_2, \dots, x_n} P(y|x_1, \dots, x_n) P(X_1) \cdots P(x_n) \\ &= \underbrace{P(X_1)}_{O(1)} \underbrace{\sum_{x_2, \dots, x_n} P(x_2) \cdots P(x_n)}_{=O(n)} \sum_y \underbrace{P(y|X_1, \dots, x_n)}_{=O(1)} \end{aligned}$$

- Curly braces show number of FLOPS
- So this query can be done efficiently in GM

complexity in
tabular case?

Variable Elimination (VE) II



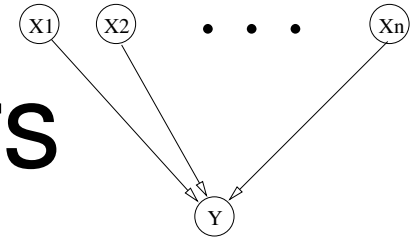
- When marginalizing over x , try to factor out all probabilities independent of x :

Different Query

$$\begin{aligned} P(Y) &= \sum_{x_1, \dots, x_n} P(Y|x_1, \dots, x_n) P(x_1) \cdots P(x_n) \\ &= \underbrace{\sum_{x_1, \dots, x_n} P(Y|x_1, \dots, x_n) P(x_1) \cdots P(x_n)}_{O(2^{n+1})} \end{aligned}$$

- Curly braces show number of FLOPS
- So this query **cannot** be done efficiently in GM

Variable Order Matters



- Original query, different variable elim. order:

$$P(X_1) = \sum_{y, x_2, \dots, x_n} P(y|x_1, \dots, x_n) P(X_1) \cdots P(x_n)$$

Original
Query

$$= \sum_{y, x_3, \dots, x_n} P(X_1) P(x_3) \cdots P(x_n) \underbrace{\sum_{x_2} P(x_2) P(y|X_1, \dots, x_n)}_{=O(2^{n+1})}$$

- With different variable order: $O(n) \rightarrow O(2^{n+1})$
 - Good variable order:
 - minimize #vars in largest intermediate factor
 - a.k.a., \sim tree width (TW) = $n+1$
 - Graphical model inference is $\sim O(2^{TW})$

Actually TW+1
but the point is
exponential

Query Types

- Marginals
 - $P(X)$, $P(Y|\text{evidence})$
 - As previously shown using VE
- Clique marginals
 - $P(X,Y)$, $P(X,Y|\text{evidence})$
 - Trivial for VE, not so for some other inference algorithms
- Most probable explanation (MPE)
 - Also known as MAP (but not in *MAP parameter* sense)
 - Instead of $\sum_{x_1} \Pi$ for marginals, use $\text{argmax}_{x_1} \Pi$
 - Still uses VE
 - Just *generalized distributive law* (SM Aji, 2000)
(works in any *commutative semiring* like $\sum \Pi$ or $\text{max} \Pi$)

Parameter Estimation

- Maximum Likelihood (ML)

$$\vec{\theta}^* = \arg \max_{\vec{\theta}} P(D|\vec{\theta})$$

Closed-form
for Gaussian,
Binomial, etc.

- Maximum a Posteriori (MAP)

$$\vec{\theta}^* = \arg \max_{\vec{\theta}} P(\vec{\theta}|D)$$

$$\propto \arg \max_{\vec{\theta}} P(D|\vec{\theta})P(\vec{\theta})$$

Just ML
with a prior

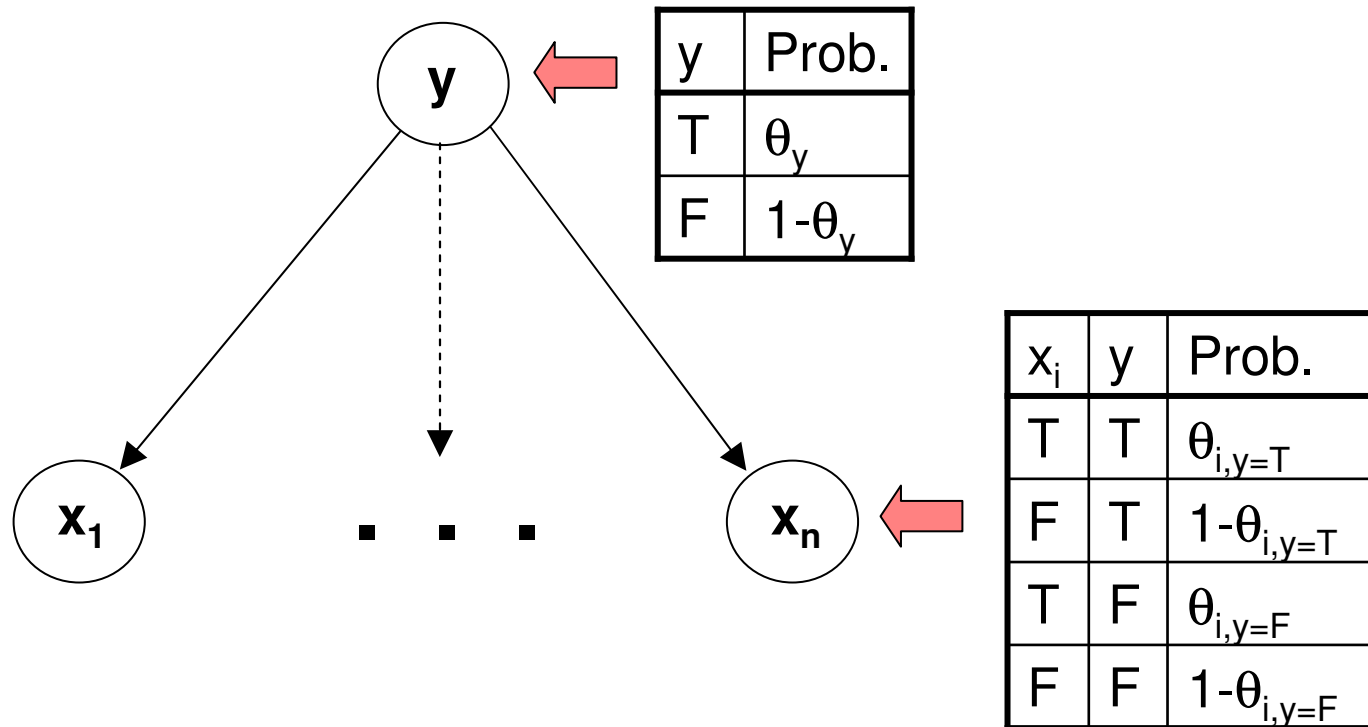
- Bayesian: maintain distribution over models

$$P(\vec{\theta}|D) \propto P(D|\vec{\theta})P(\vec{\theta})$$

Inference
requires
integral

ML for Naïve Bayes Graphical Model

- Estimate ML parameters for following GM
 - Assume all variables are binary



Max Likelihood for Naïve Bayes

$$\begin{aligned}
 \vec{\theta}^* &= \arg \max_{\vec{\theta}} \overbrace{P(D|\vec{\theta})}^{L(\theta)} \\
 &= \arg \max_{\vec{\theta}} \prod_{d \in D} P(y^d, x_1^d, \dots, x_n^d | \vec{\theta}) \\
 &= \arg \max_{\vec{\theta}} \prod_{d \in D} P(x_1^d, \dots, x_n^d | y^d, \vec{\theta}) P(y^d | \vec{\theta}) \\
 &= \arg \max_{\vec{\theta}} \prod_{d \in D} P(y^d | \vec{\theta}) \prod_{i=1}^n P(x_i^d | y^d, \vec{\theta}) \\
 &= \arg \max_{\vec{\theta}} \prod_{d \in D} \theta_y^{\mathbb{I}_{y=T}[d]} (1 - \theta_y)^{\mathbb{I}_{y=F}[d]} \prod_{i=1}^n \prod_{v \in \{F, T\}} \theta_{y=v, i}^{\mathbb{I}_{y=v, x_i=T}[d]} (1 - \theta_{y=v, i})^{\mathbb{I}_{y=v, x_i=F}[d]} \\
 &= \arg \max_{\vec{\theta}} \theta_y^{\#D_{y=T}} (1 - \theta_y)^{\#D_{y=F}} \prod_{i=1}^n \prod_{v \in \{F, T\}} \theta_{y=v, i}^{\#D_{y=v, x_i=T}} (1 - \theta_{y=v, i})^{\#D_{y=v, x_i=F}} \\
 &= \arg \max_{\vec{\theta}} \overbrace{\#D_{y=T} \log \theta_y + \#D_{y=F} \log(1 - \theta_y) +} \\
 &\quad \underbrace{\sum_{i=1}^n \sum_{v \in \{F, T\}} \left[\#D_{y=v, x_i=T} \log \theta_{y=v, i} + \#D_{y=v, x_i=F} \log(1 - \theta_{y=v, i}) \right]}_{l(\vec{\theta})}
 \end{aligned}$$

Max Likelihood for NB (Cont.)

- Unique maxima for log-linear models at slope = 0

$$\begin{aligned}\frac{\partial l(\vec{\theta})}{\partial \theta_y} &= \frac{\#D_{y=T}}{\theta_y} + \frac{\#D_{y=F}}{(1 - \theta_y)} = 0 \\ \frac{\theta_y}{(1 - \theta_y)} &= \frac{\#D_{y=T}}{\#D_{y=F}} \\ \theta_y &= \frac{\#D_{y=T}}{\#D_{y=T} + \#D_{y=F}}\end{aligned}$$

$$\begin{aligned}\frac{\partial l(\vec{\theta})}{\partial \theta_{y=v,i}} &= \frac{\#D_{y=v,x_i=T}}{\theta_{y=v,i}} + \frac{\#D_{y=v,x_i=F}}{(1 - \theta_{y=v,i})} = 0 \\ \frac{\theta_{y=v,i}}{(1 - \theta_{y=v,i})} &= \frac{\#D_{y=v,x_i=T}}{\#D_{y=v,x_i=F}} \\ \theta_{y=v,i} &= \frac{\#D_{y=v,x_i=T}}{\#D_{y=v,x_i=T} + \#D_{y=v,x_i=F}}\end{aligned}$$

- ML parameters are just the empirical probabilities!

ML Parameters for General Bayes Nets

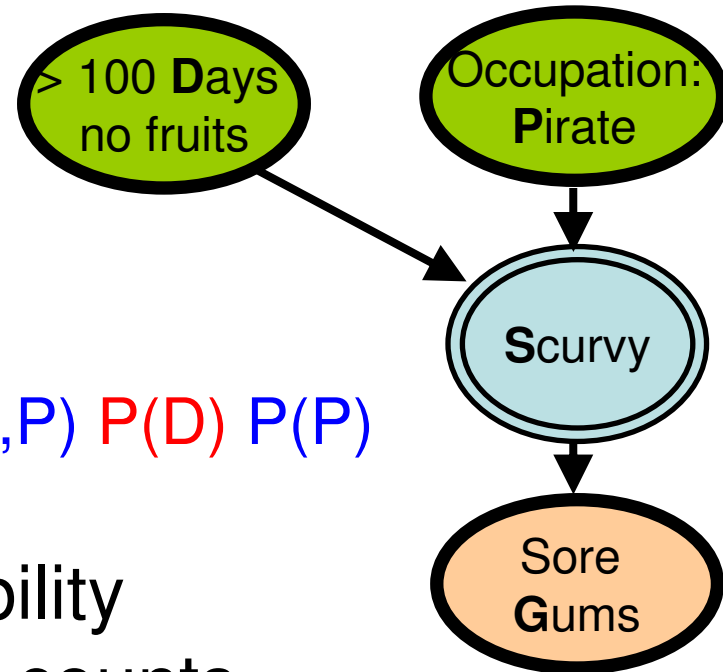
- Recall the scurvy network

- Joint is product of four conditional probabilities

- $P(D, P, S, G) = P(G|S) P(S|D, P) P(D) P(P)$

- Get ML conditional probability tables from empirical data counts

- E.g., $P(G=\text{true}|S=\text{true}) = \frac{\text{Freq}(G=\text{true}, S=\text{true})}{\text{Freq}(S=\text{true})}$



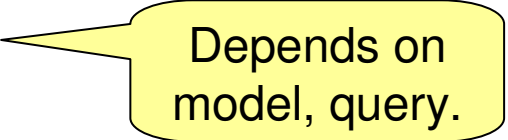
Bayes Net Recap

- What we covered

- Laws of probabilities
- Discrete representation and operations
- Use graphical models to structure joint dist.
 - Edges represent conditional dependences
- Variable elimination inference to exploit structure

- Benefits

- Compact representation
- Robust learning of params (more data per parameter!)
- Can often do efficient inference



Depends on
model, query.

Other Inference Algorithms

- Exact

- Junction tree

- think of as VE on modified graph with cached pre-computations of intermediate factors
 - only use if repeated queries with same evidence

- ...

- Approximate

- Biased

- Loopy belief propagation
 - just like it sounds, exact on trees, efficient for all marginals
 - ...

- Unbiased

- Sampling (Rejection, Importance, MCMC – Gibbs)
 - ...

Structure Learning

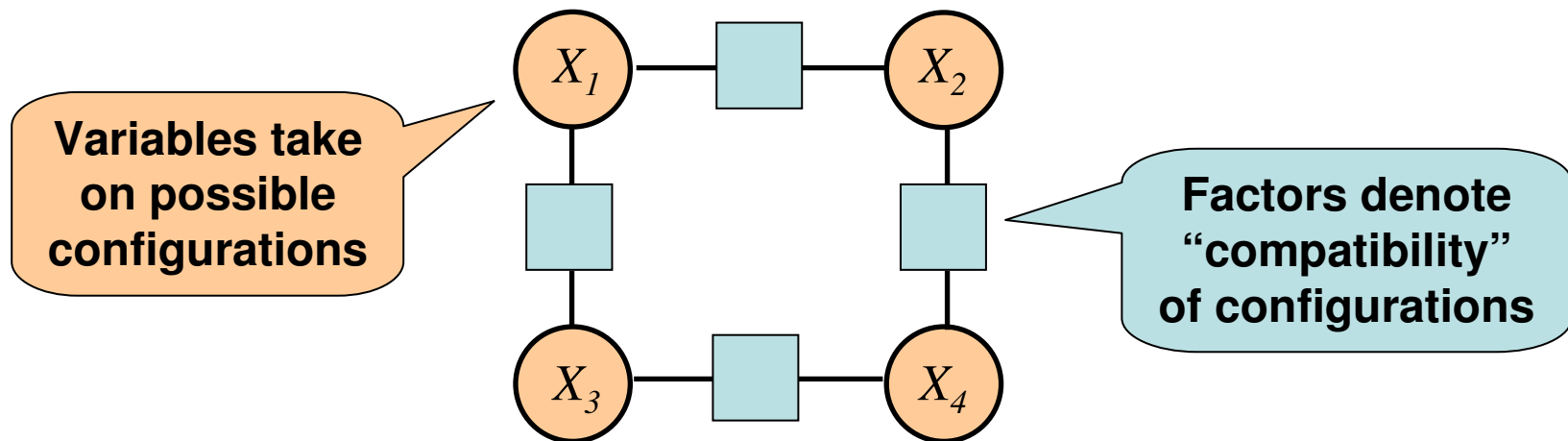
- Have explored parameter learning for a fixed DAG
- But what about learning the DAG itself?
- For Bayes Nets, many approaches
 - All search in space of DAGs
 - Usually greedy search with incremental modifications
 - Find DAG that trades off log likelihood with model complexity
 - Local changes in structure incur local changes in log likelihood
 - So search is relatively efficient
- Not so well explored for undirected models
 - Local changes require recalculating all parameters
 - See Koller, Welling for some ideas in recent years

Other RVs

- For continuous / ∞ integer case
 - Gaussians
 - Closed-form
 - Nice properties for loopy BP (exact means)
 - ∞ Integer or non-Gaussians
 - Requires symbolic function representation, integration
 - Some useful approximation techniques
 - MCMC Sampling (Gibbs)
 - Variational
 - Expectation propagation
- We only focused on discrete probabilities
 - But all ideas generalize (CI & distributive law)

Other Graphical Models

- Markov Random Fields (MRFs)
- Factor Graphs (generalization of BNs and MRFs)



- Still product of factors, but need $1/Z$ normalizer
 - Product no longer guaranteed to sum out to 1
- Inference same ($1/Z$ is a constant)
- Parameter estimation different (if not Bayes net)
 - No closed-form solution like Bayes net empirical counts
 - Have to use gradient descent or other method

Other Training Criteria

- **Max likelihood**

- generative models, shown previously

Just empirical counts, used most in practice

- **Max conditional likelihood**

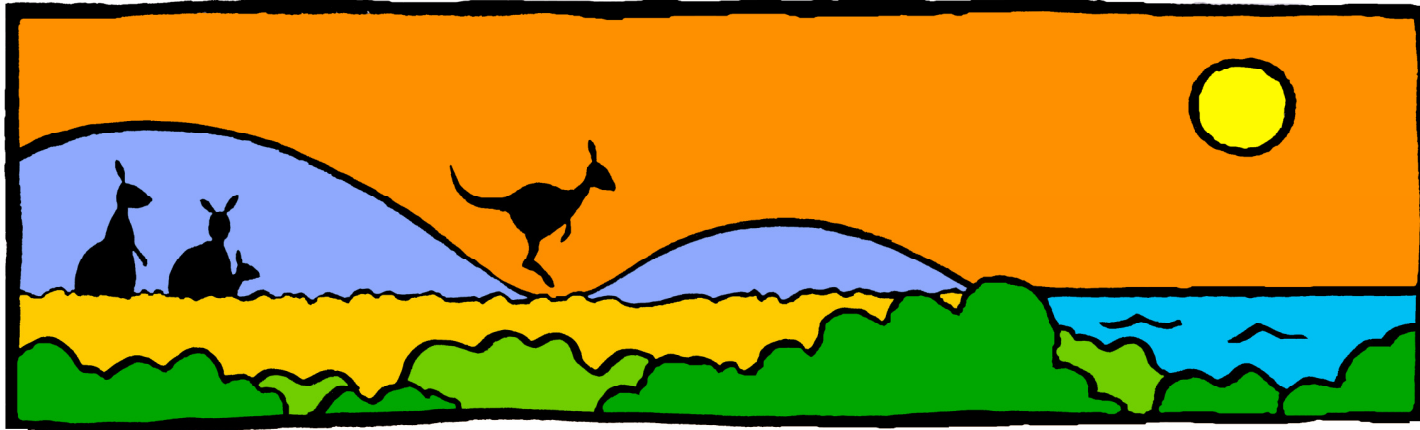
- discriminative models (conditional random fields)
 - don't model features!
- No closed-form gradient descent
 - inference required to compute gradient

- **Pseudolikelihood (Besag, 1975)**

- “local” training of discriminative models
- asymptotically consistent

- **Max / large margin methods**

- choose weights to separate correct configuration from rest

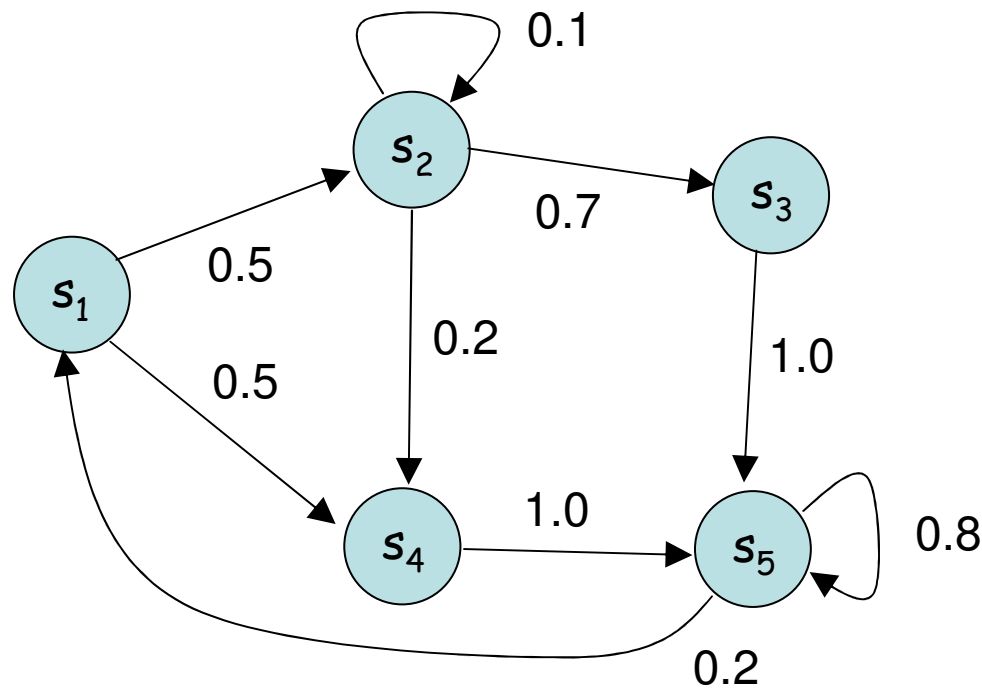


Graphical Models

(Hidden) Markov Models

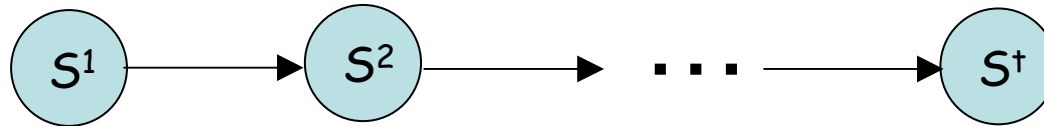
Markov Models (or Markov Chains)

- At each time step, probabilistically transition from current state to next state ($S = \{s_1, s_2, \dots, s_n\}$)
- Finite State Machine (FSM) view for $n=5$:



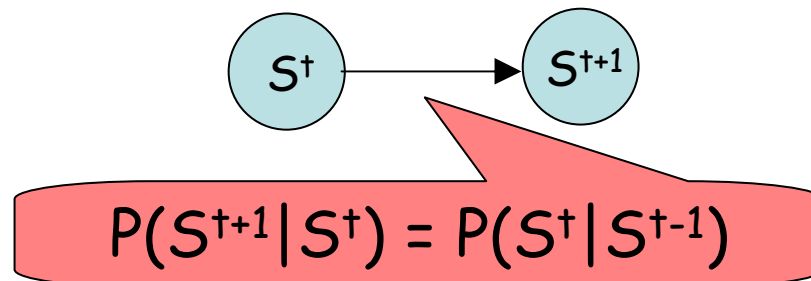
Markov Models

- The graphical model view for t steps:



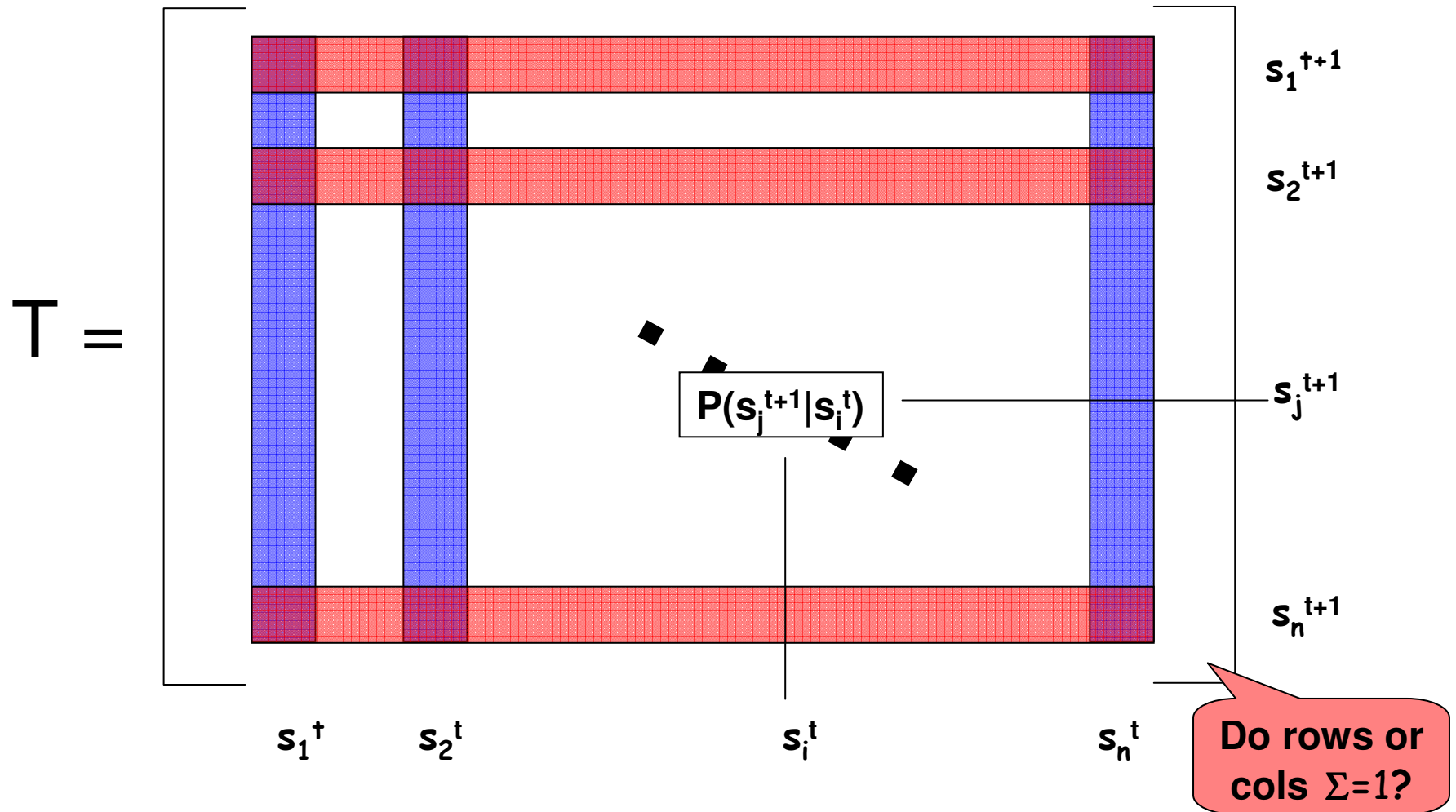
– Note: for $t = \infty$, an infinite graphical model!

- Or assuming transition stationarity, just:



Transition Matrix

- Represent $P(s^{t+1}|s^t)$ as transition matrix:



Transition Probabilities

- Formally
 - Define state set $S^t = \{s_1, s_2, \dots, s_n\} ; \forall t$
 - Define transition matrix $T_{ij}^t = P(S_i^{t+1} | S_j^t) ; \forall t$
- Properties of T_{ij}
 - *Stationary*: $T_{ij}^t = T_{ij}^{t-1}$ OR $P(S^{t+1} | S^t) = P(S^t | S^{t-1}) ; \forall t$
 - *Ergodic*: any state can be reached from any other state in a finite number of steps

Examples?

Distribution at Time t

- Given $P(s^0)$, what is $P(s^t)$?
- Use var. elim. to marginalize over intermediate time steps
 - $P(s^t) = \sum_{s^1, \dots, s^{t-1}} P(s^0) \prod_{i=0 \dots t-1} P(s^{i+1} | s^i)$
- Or let $P s^0$ & $P s^t$ be column vectors...
 - Then simply: $P s^t = (T^t) P s^0$
 - Note: Intimate connection between matrix ops and var. elim.
 - When $P(s^{i+1} | s^i)$ factors as a DBN...
capture many efficiencies of var. elim. via sparse matrix ops

If no evidence after time t, all factors for t+1 and after marginalize out

Stationary Distribution

- Stationary Distribution π at $t=\infty$
 - $\pi = (T^{\infty}) P s^0$
 - If T ergodic, $P s^0$ irrelevant
 - Reaches *unique* steady-state distribution: $\pi = T\pi$
 - So π = any column of T^{∞}
 - Can solve via eigenvector analysis (note: $\lambda=1$)
 - Related to (Krylov) iterated eigenvector computation
 - Or use fixed point to solve linear system
 - $T\pi - \pi = 0 \Rightarrow \pi' T' - \pi' = 0 \Rightarrow \pi' (T' - I) = 0$
s.t. constraints on π
 - Can solve linear system via matrix inversion
 - » $(T' - I)$ guaranteed full rank \therefore invertible

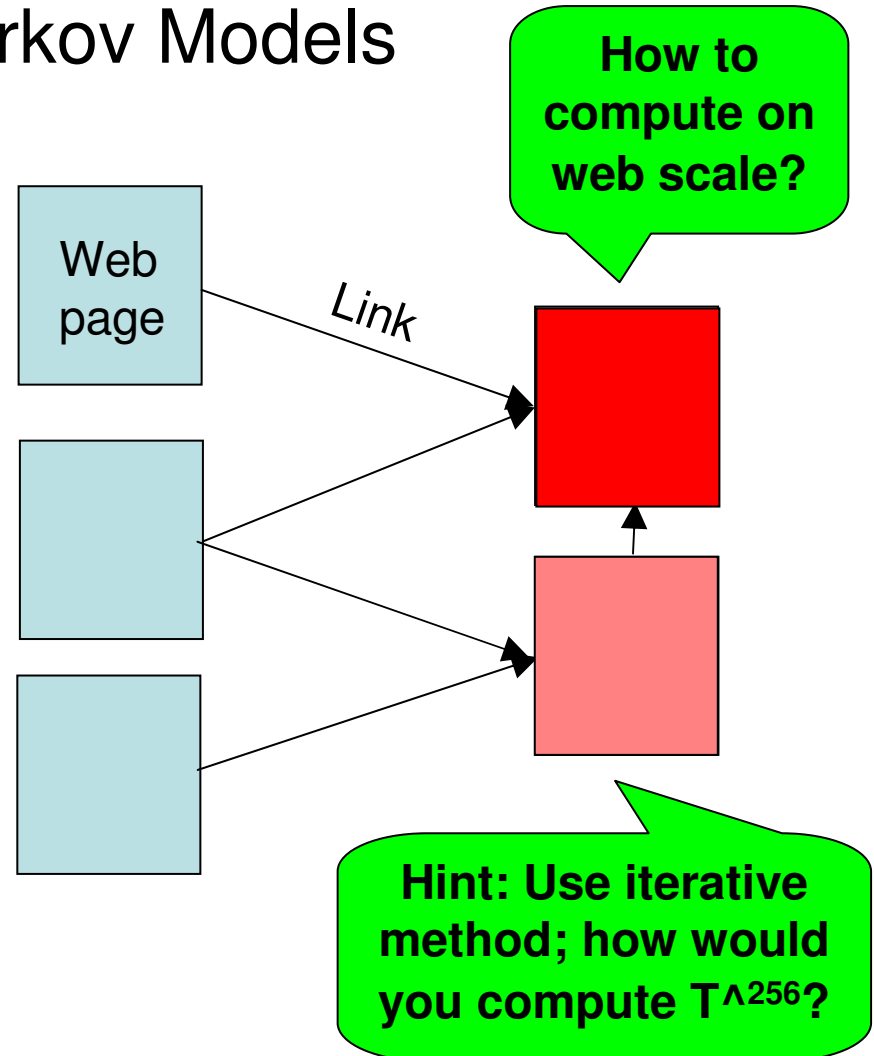
**Why? What
are they?**

Markov Model Applications

- Simple theory, ingenious applications:
 - n^{th} -order Markov models
 - Relax Markovian assumption to previous n states
 - Used in text and speech processing
 - N-grams for predicting next word occurrence
 - Colocation identification
 - [Dasher](#) for text input, try it in your [web browser](#)
 - More generally
 - Physics (states of systems)
 - Queuing theory (random entries and exits)
 - Economics, Biology, Chemistry, etc...
 - Google!

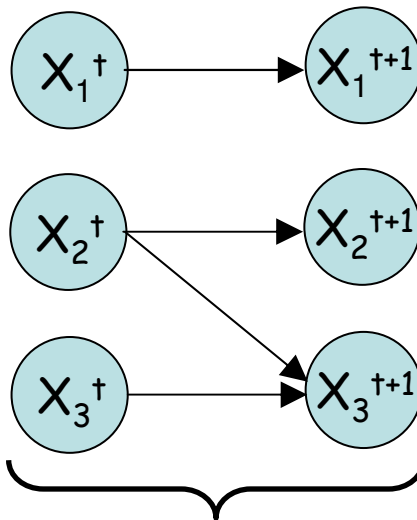
Google PageRank Example

- Very beautiful use of Markov Models
- Model of web browsing:
 - Probabilistically take link with $\sim 1/k$ chance if k links
 - Small chance of random transition
- Stationary distribution π gives PageRank!
 - Measure of “authority”



Factored Markov Models

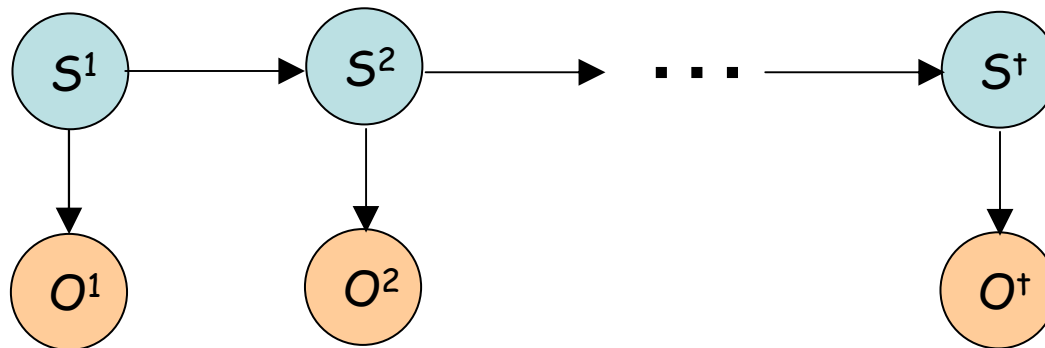
- The Dynamic Bayes Net (DBN) view:
 - State factors into variables: X_1, X_2, \dots, X_k
 - Capture transition independences



$$P(x_1^{t+1}, x_2^{t+1}, \dots, x_k^{t+1} \mid x_1^t, x_2^t, \dots, x_k^t) = \prod \dots$$

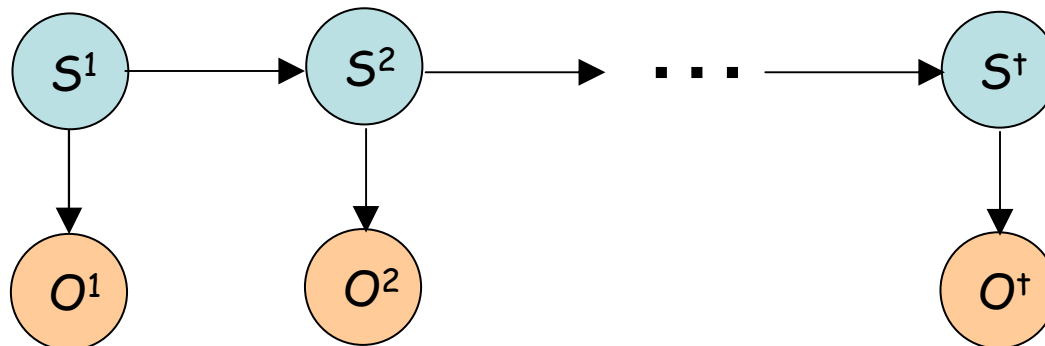
Hidden Markov Models

- Formally
 - Define state set $S^{\dagger} = \{s_1, s_2, \dots, s_n\} ; \forall t$
 - Define observation set $O^{\dagger} = \{o_1, o_2, \dots, o_m\} ; \forall t$
 - Define observation prob $P(O^{\dagger} | S^{\dagger}) ; \forall t$
 - Define transition prob $P(S^{\dagger+1} | S^{\dagger}) ; \forall t$
- Graphical Model view:



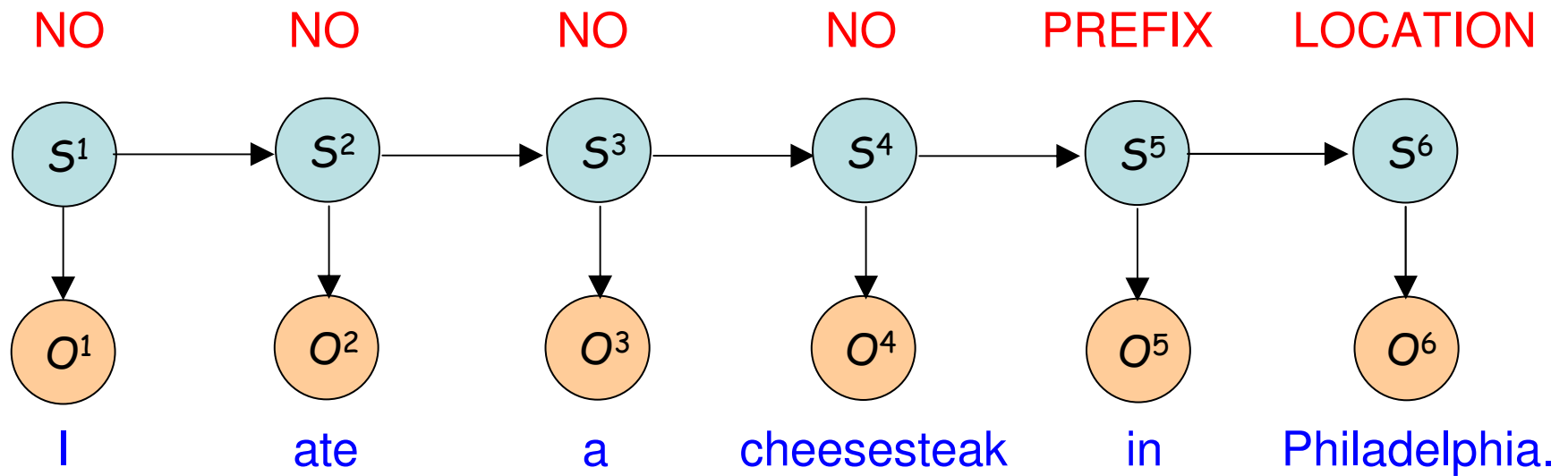
Primary Operations

- Operations
 - **Viterbi**: state estimation
 - **Forward-backward**: marginal prob of each state
 - **Baum-Welch**: EM estimation of parameters given only observations
- Graphical Model view:



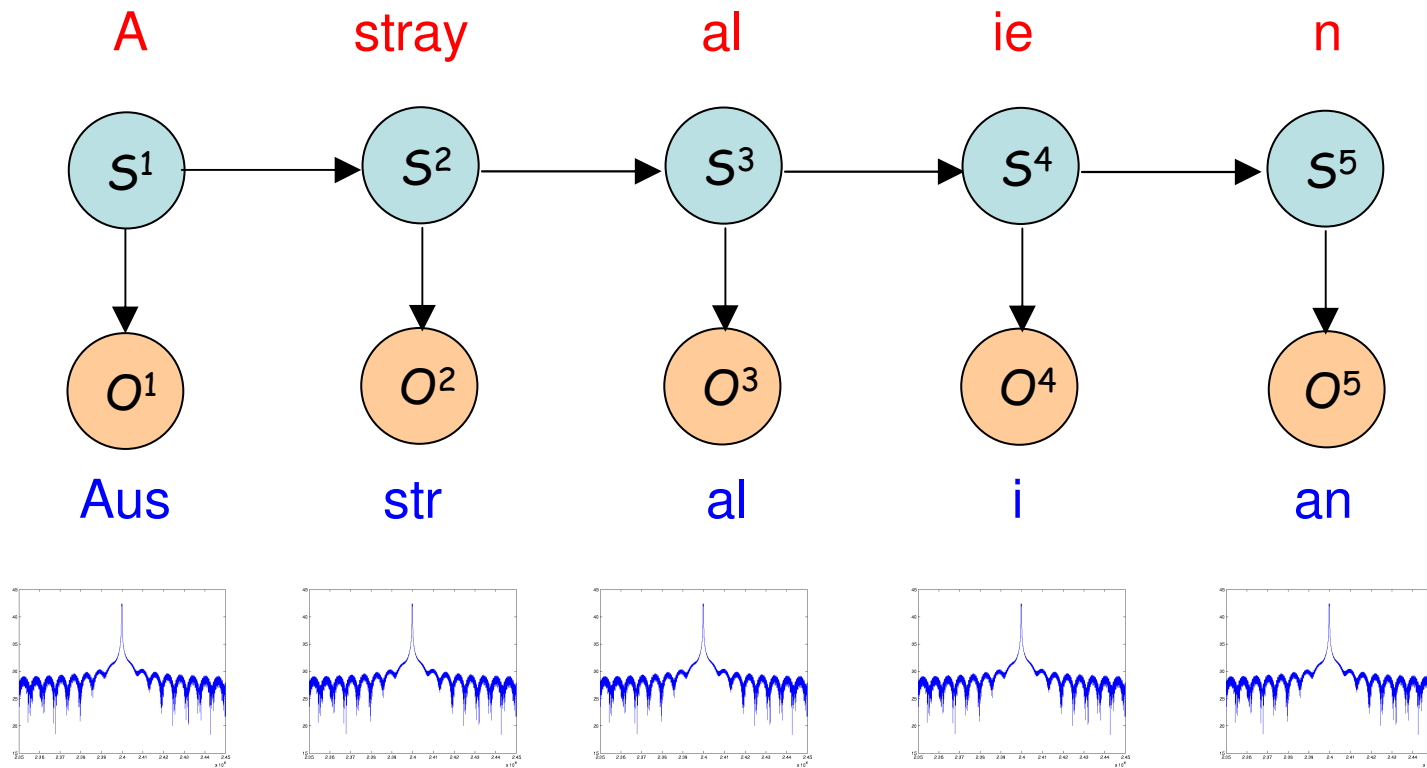
HMMs for Information Extraction

- Information Extraction
 - Want to extract entities from text (e.g., places)
 - Context helps (“at X”, “X located in Y”)
 - Observations are words, infer state via MPE query



HMMs for Speech Recognition

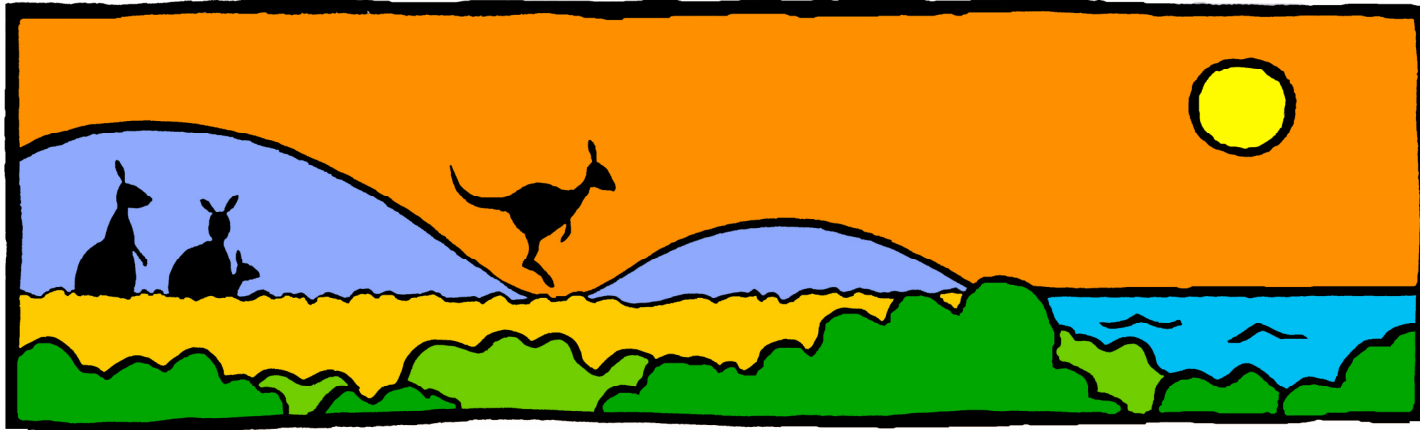
- Speech to text
 - Observations are power spectrum for time intervals
 - Each state is a syllable (inferred via MPE query)



Sequential Decision Theory

- We've looked at sequential *prediction*
 - *But* what if choose actions that affect model?
 - And differing utilities for different states?
- Fully observable case (MDP):
 - Markov Decision Process (MDP) = MM + Actions
- Partially observable case (POMDP):
 - Partially Observable MDP = HMM + Actions

More in Will
Uther's RL lecture



Graphical Models

Data Structures for Factors

Factor Representation (Tables)

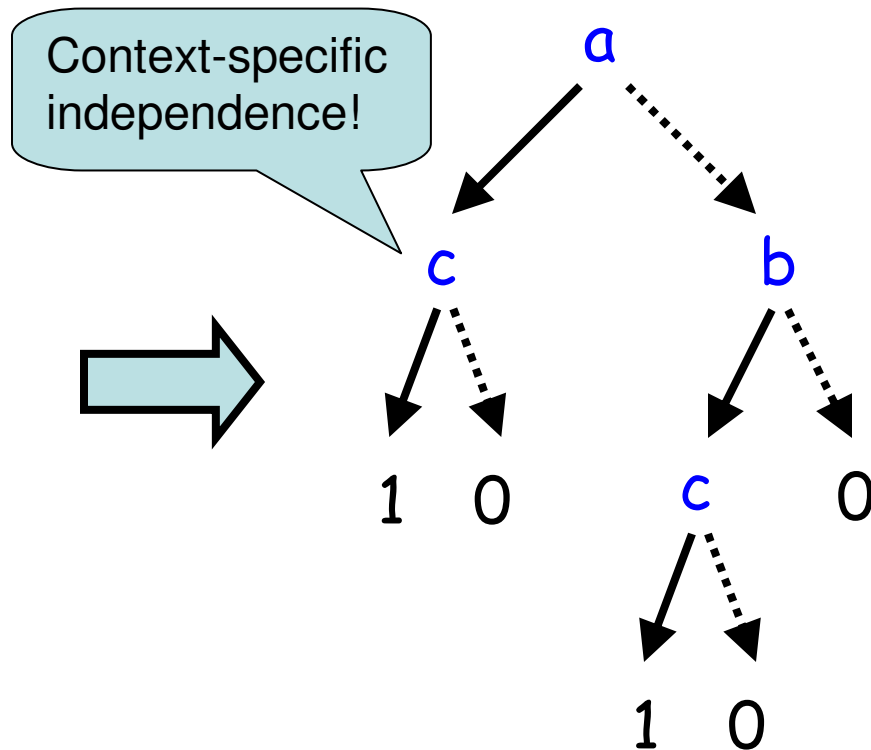
- How do we represent a function from $B^n \rightarrow \mathbb{R}$?
- How about a fully enumerated table...
- ...OK, but can we be more compact?

a	b	c	F(a,b,c)
0	0	0	0.00
0	0	1	0.00
0	1	0	0.00
0	1	1	1.00
1	0	0	0.00
1	0	1	1.00
1	1	0	0.00
1	1	1	1.00

Factor Representation (Trees)

- How about a tree? Sure, now can simplify.

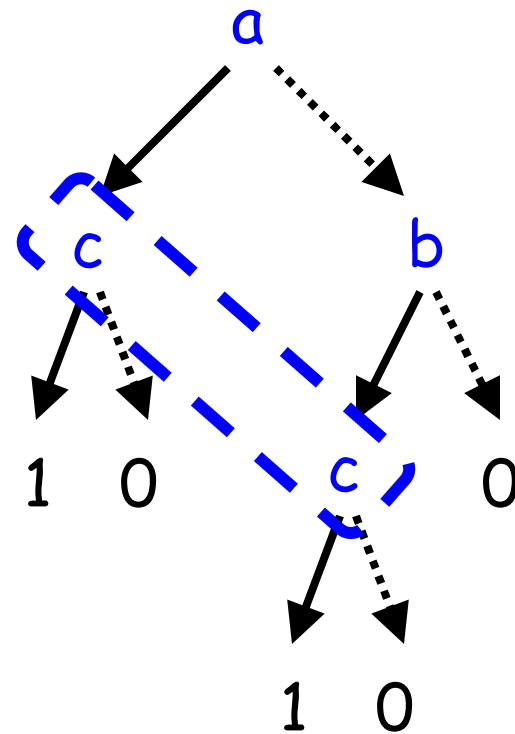
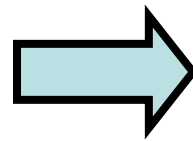
a	b	c	$F(a,b,c)$
0	0	0	0.00
0	0	1	0.00
0	1	0	0.00
0	1	1	1.00
1	0	0	0.00
1	0	1	1.00
1	1	0	0.00
1	1	1	1.00



Factor Representation (ADDs)

- Why not a directed acyclic graph (DAG)?

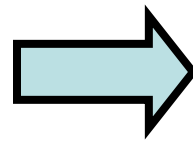
a	b	c	$F(a,b,c)$
0	0	0	0.00
0	0	1	0.00
0	1	0	0.00
0	1	1	1.00
1	0	0	0.00
1	0	1	1.00
1	1	0	0.00
1	1	1	1.00



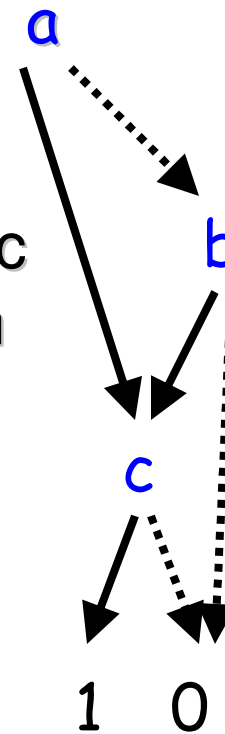
Factor Representation (ADDs)

- Why not a directed acyclic graph (DAG)?

a	b	c	$F(a,b,c)$
0	0	0	0.00
0	0	1	0.00
0	1	0	0.00
0	1	1	1.00
1	0	0	0.00
1	0	1	1.00
1	1	0	0.00
1	1	1	1.00

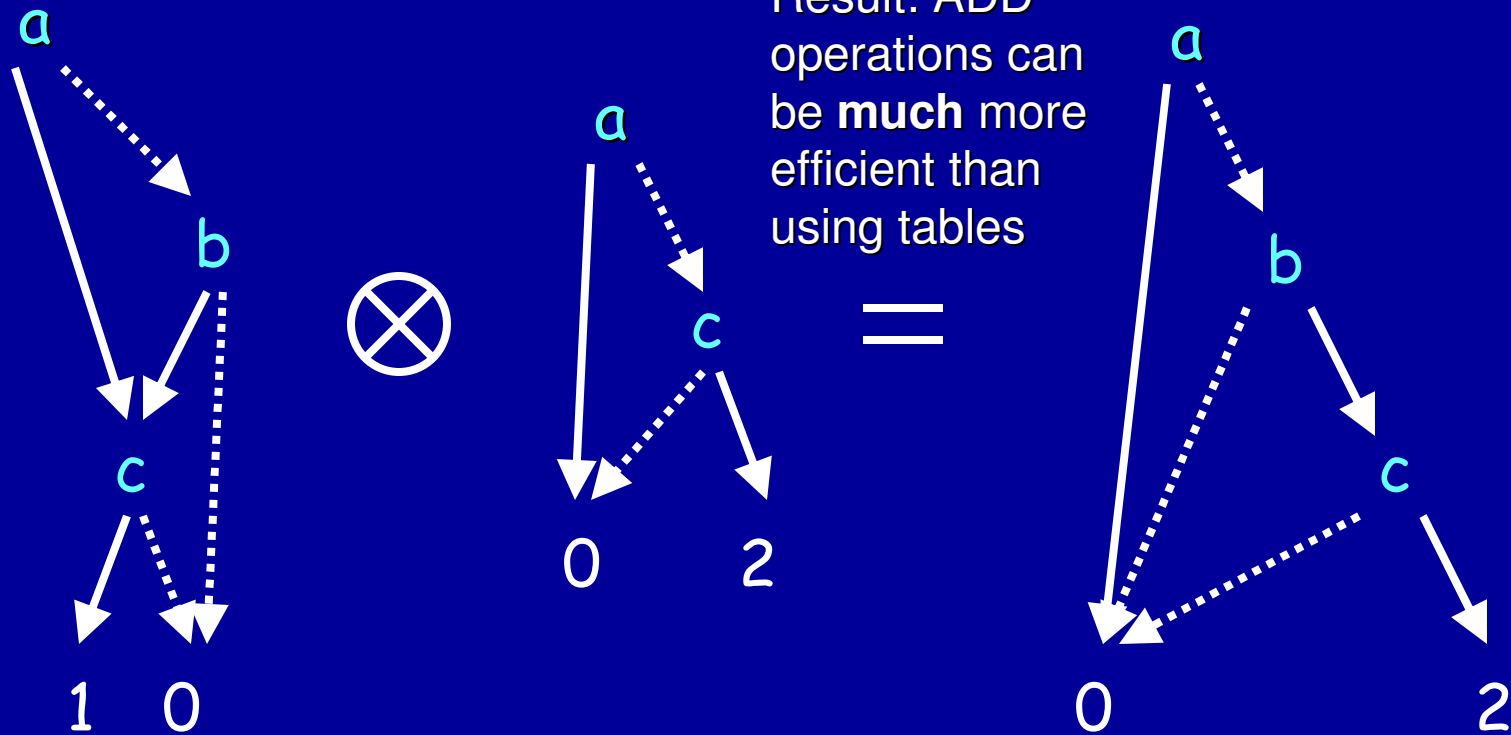


Algebraic
Decision
Diagram
(ADD)




Binary Operations (ADDs)

- Why do we order the variables?
- This enables us to do efficient binary operations...



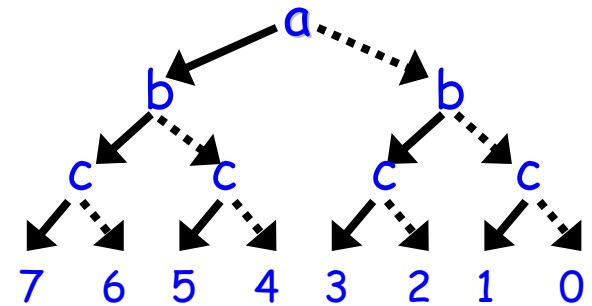
Replacing Tables with ADDs

- Instead of representing discrete probability functions as tables...
- Represent as ADDs
 - Can add, multiply, divide, max ADDs
 - Can even marginalize How?
 - No worse than table space / time
 - Often much better when repeated values in tables

ADD Inefficiency

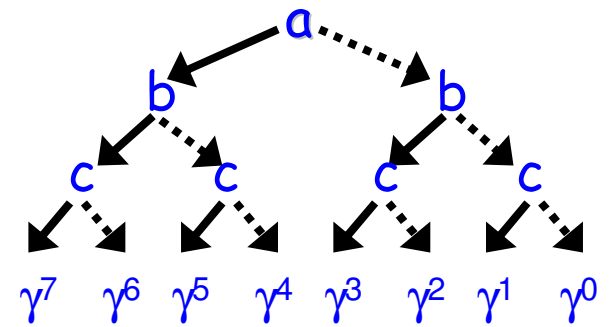
- Is context-specific independence enough?
- Or do we need more compactness?
- Example 1: Additive reward/utility functions

$$\begin{aligned} - R(a,b,c) &= R(a) + R(b) + R(c) \\ &= 4a + 2b + c \end{aligned}$$



- Example 2: Multiplicative value functions

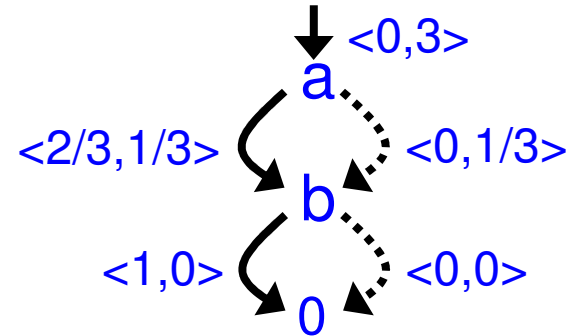
$$\begin{aligned} - V(a,b,c) &= V(a) \cdot V(b) \cdot V(c) \\ &= \gamma^{(4a + 2b + c)} \end{aligned}$$



Affine ADDs to Rescue

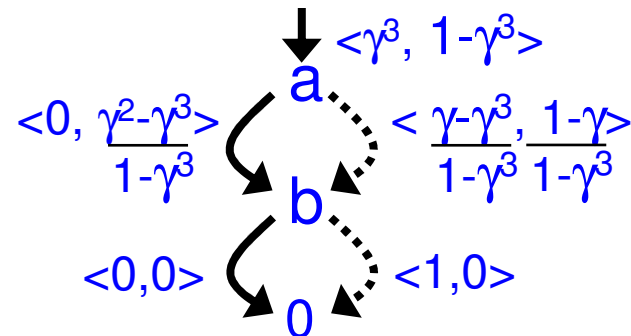
- AADDs can be exp. smaller than ADDs
- Ex. 1: Additive reward/utility functions

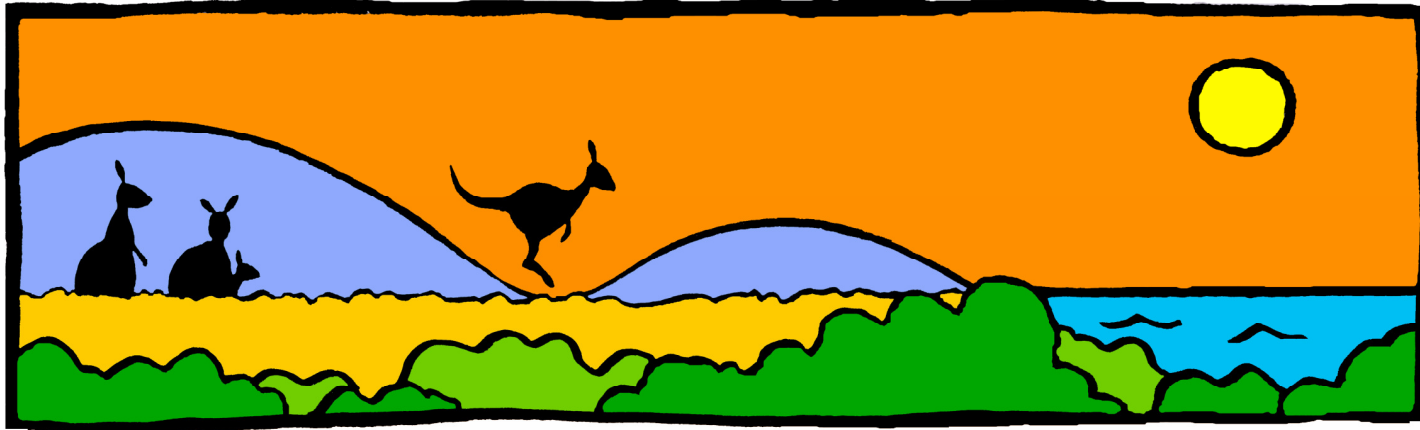
- $R(a,b) = R(a) + R(b)$
 $= 2a + b$



- Ex. 2: Multiplicative value functions

- $V(a,b) = V(a) \cdot V(b)$
 $= \gamma^{(2a + b)}; \gamma < 1$





Graphical Models

Summary

Probabilistic Inference

- Fundamental Operation
 - $P(\text{Query} \mid \text{Evidence})$
- To compute, just need
 - Joint probability distribution over RVs
 - Graphical model is a compact representation that exploits known (in)dependences (e.g., Markov)
 - Ability to do marginalization, multiplication, and division on distributions
 - For discrete distributions, can use tables / DDs

Graphical Models

- A tool you **need** in your toolbox:
 - Exponential **space savings** in representation
 - Exponential **time savings** in inference
 - Exponential **data complexity reduction**
 - # samples needed to learn “good” model

For More Information

- Kevin Murphy's "Bayes Net Tutorial"

<http://www.cs.ubc.ca/~murphyk/Bayes/bnintro.html>

- Thorough discussion
- Great reference list
- Links to software!