Directed Graphical Models

Instructor: Alan Ritter

Many Slides from Tom Mitchell

Graphical Models

- Key Idea:
  - Conditional independence assumptions useful
  - but Naïve Bayes is extreme!
  - Graphical models express sets of conditional independence assumptions via graph structure
  - Graph structure plus associated parameters define *joint probability distribution over set of variables*

- Two types of graphical models:
  - Directed graphs (aka Bayesian Networks)
  - Undirected graphs (aka Markov Random Fields)

Graphical Models – Why Care?

- Among most important ML developments
- Graphical models allow combining:
  - Prior knowledge in form of dependencies/independencies
  - Prior knowledge in form of priors over parameters
  - Observed training data
- Principled and ~general methods for
  - Probabilistic inference
  - Learning
- Useful in practice
  - Diagnosis, help systems, text analysis, time series models, ...

Conditional Independence

Definition: X is conditionally independent of Y given Z, if the probability distribution governing X is independent of the value of Y, given the value of Z

$$(\forall i, j, k) P(X = x_i | Y = y_j, Z = z_k) = P(X = x_i | Z = z_k)$$

Which we often write $P(X|Y,Z) = P(X|Z)$

E.g., $P(Thunder|Rain, Lightning) = P(Thunder|Lightning)$

Marginal Independence

Definition: X is marginally independent of Y if

$$(\forall i, j) P(X = x_i, Y = y_j) = P(X = x_i) P(Y = y_j)$$

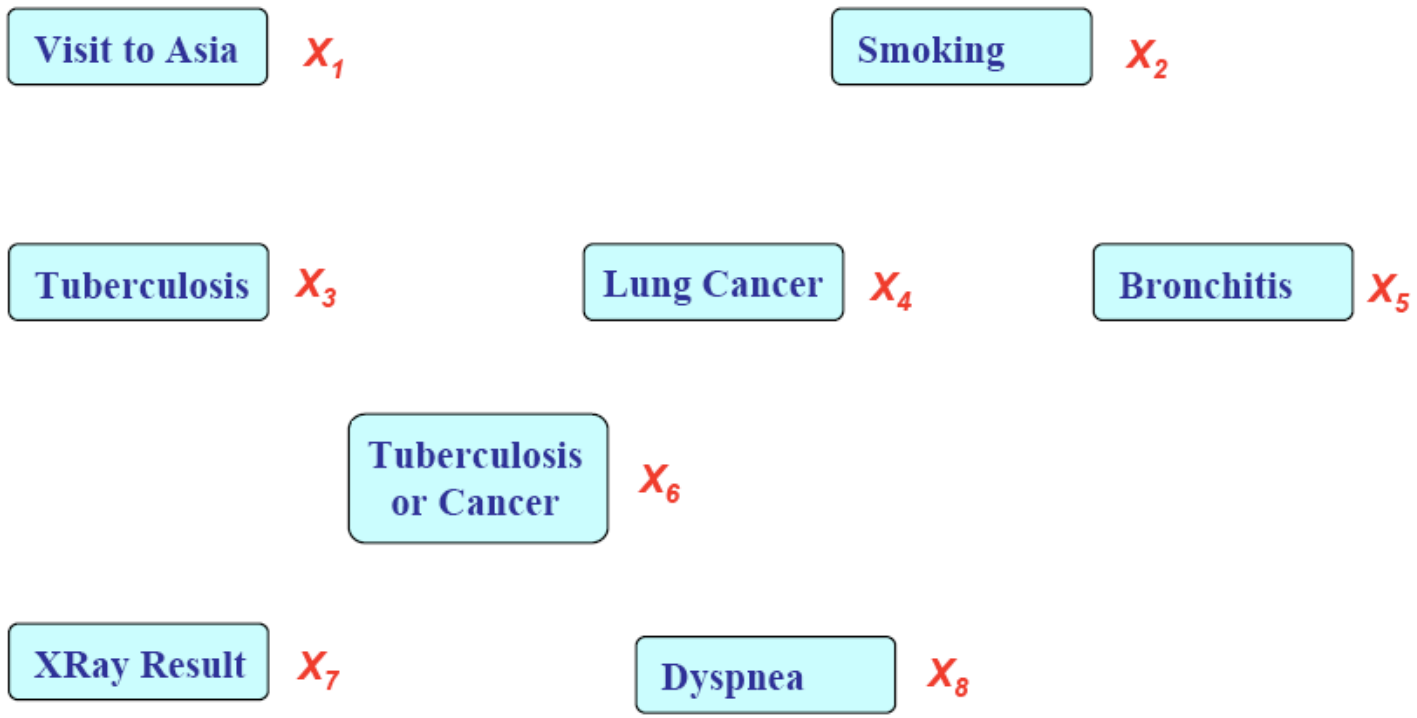
Equivalently, if

$$(\forall i, j) P(X = x_i | Y = y_j) = P(X = x_i)$$

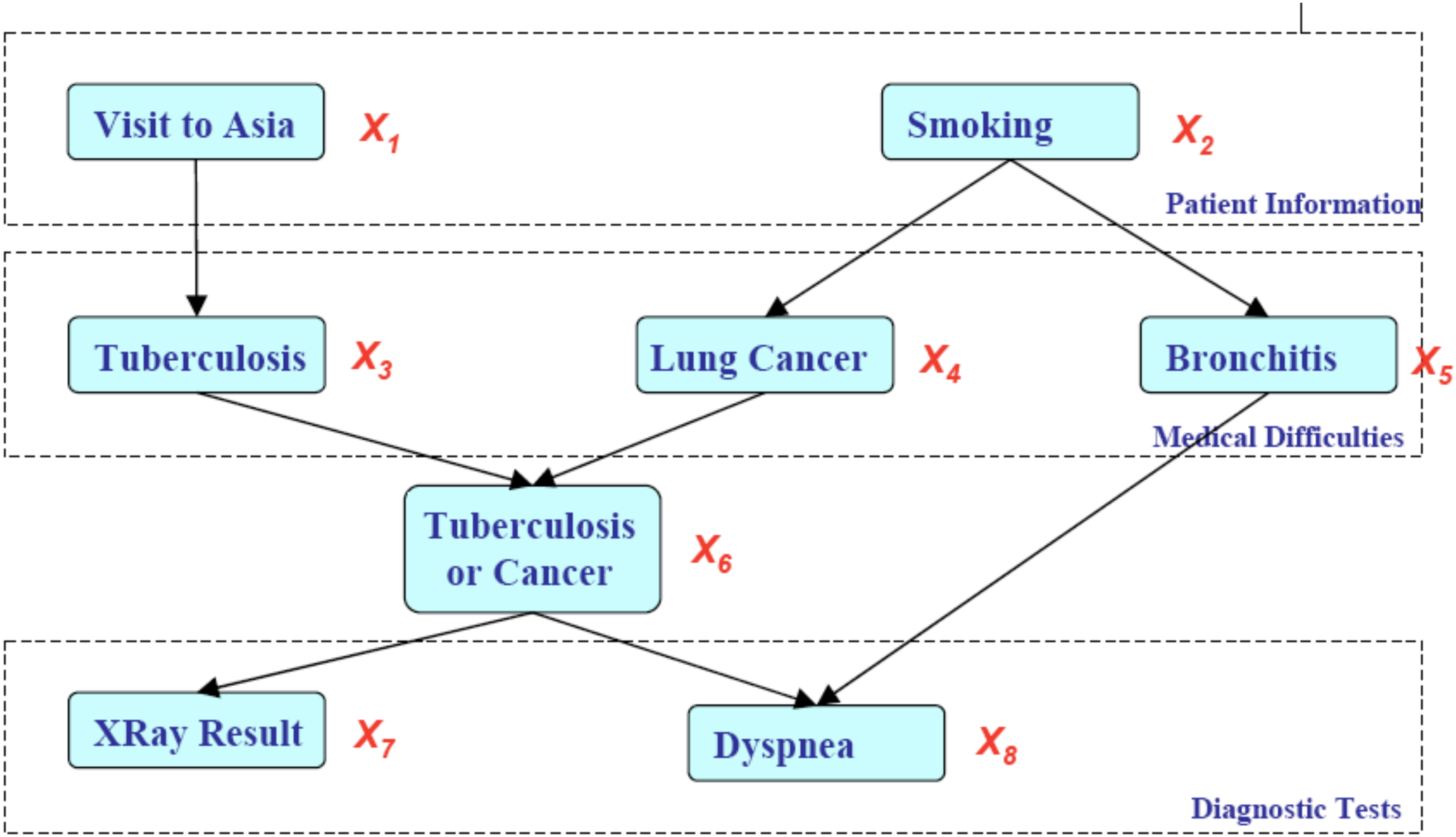
Equivalently, if

$$(\forall i, j) P(Y = y_i | X = x_j) = P(Y = y_i)$$

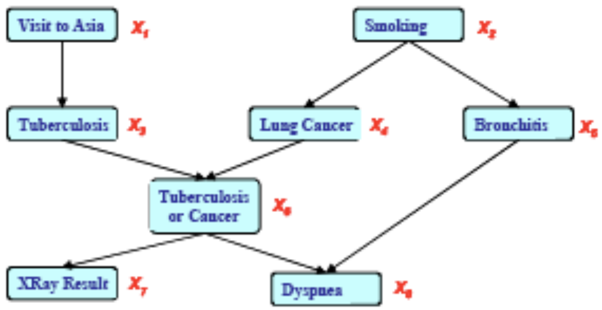
Represent Joint Probability Distribution over Variables



Describe network of dependencies



Bayes Nets define Joint Probability Distribution in terms of this graph, plus parameters



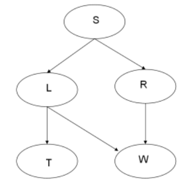
$$P(X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8)$$

$$= P(X_1)\, P(X_2)\, P(X_3 | X_1)\, P(X_4 | X_2)\, P(X_5 | X_2)\, P(X_6 | X_3, X_4)\, P(X_7 | X_6)\, P(X_8 | X_5, X_6)$$

Benefits of Bayes Nets:

- Represent the full joint distribution in fewer parameters, using prior knowledge about dependencies
- Algorithms for inference and learning

Bayesian Networks Definition



Parents	P(W|Pa)	P(¬W|Pa)
L, R	0	1.0
L, ¬R	0	1.0
¬L, R	0.2	0.8
¬L, ¬R	0.9	0.1

W

A Bayes network represents the joint probability distribution over a collection of random variables

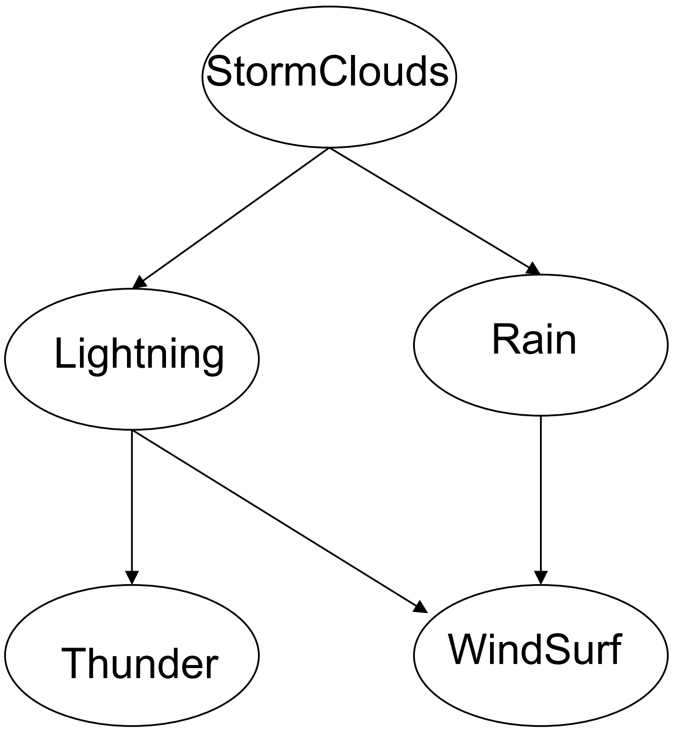
A Bayes network is a directed acyclic graph and a set of conditional probability distributions (CPD's)

- Each node denotes a random variable
- Edges denote dependencies
- For each node X_i its CPD defines $P(X_i \mid Pa(X_i))$
- The joint distribution over all variables is defined to be

$$P(X_1 \ldots X_n) = \prod_i P(X_i | Pa(X_i))$$

Pa(X) = immediate parents of X in the graph

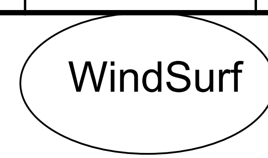
Bayesian Network



Nodes = random variables

A conditional probability distribution (CPD) is associated with each node N, defining P(N | Parents(N))

Parents	P(W|Pa)	P(¬W|Pa)
L, R	0	1.0
L, ¬R	0	1.0
¬L, R	0.2	0.8
¬L, ¬R	0.9	0.1



The joint distribution over all variables:

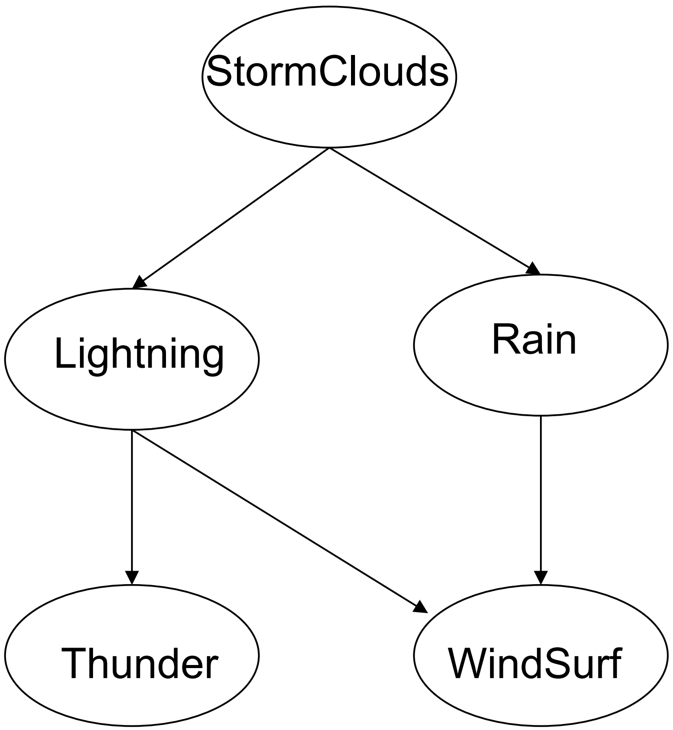
$$P(X_1 \ldots X_n) = \prod_i P(X_i | Pa(X_i))$$

Bayesian Network

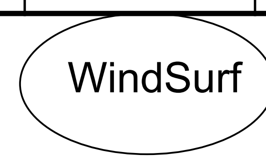
What can we say about conditional independencies in a Bayes Net?

One thing is this:

Each node is conditionally independent of its non-descendents, given only its immediate parents.



Parents	P(W|Pa)	P(¬W|Pa)
L, R	0	1.0
L, ¬R	0	1.0
¬L, R	0.2	0.8
¬L, ¬R	0.9	0.1



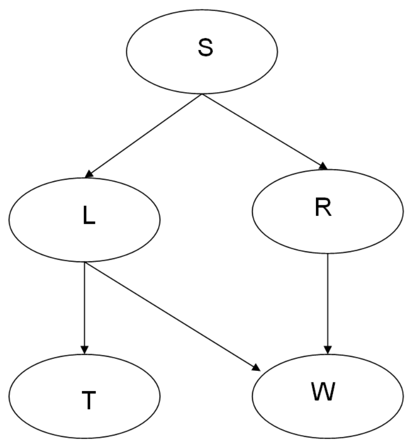
Some helpful terminology

Parents = Pa(X) = immediate parents

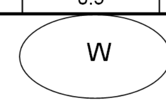
Antecedents = parents, parents of parents, ...

Children = immediate children

Descendents = children, children of children, ...

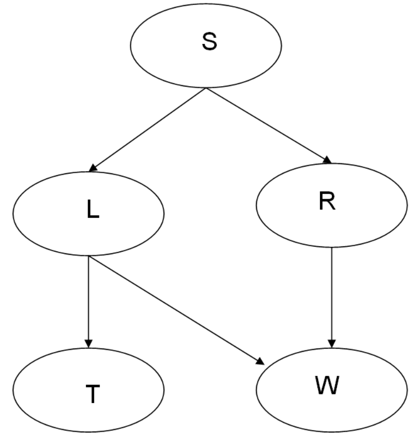


Parents	P(W|Pa)	P(¬W|Pa)
L, R	0	1.0
L, ¬R	0	1.0
¬L, R	0.2	0.8
¬L, ¬R	0.9	0.1

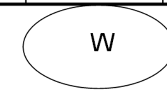


Bayesian Networks

- CPD for each node X_i describes $P(X_i \mid Pa(X_i))$



Parents	P(W|Pa)	P(¬W|Pa)
L, R	0	1.0
L, ¬R	0	1.0
¬L, R	0.2	0.8
¬L, ¬R	0.9	0.1

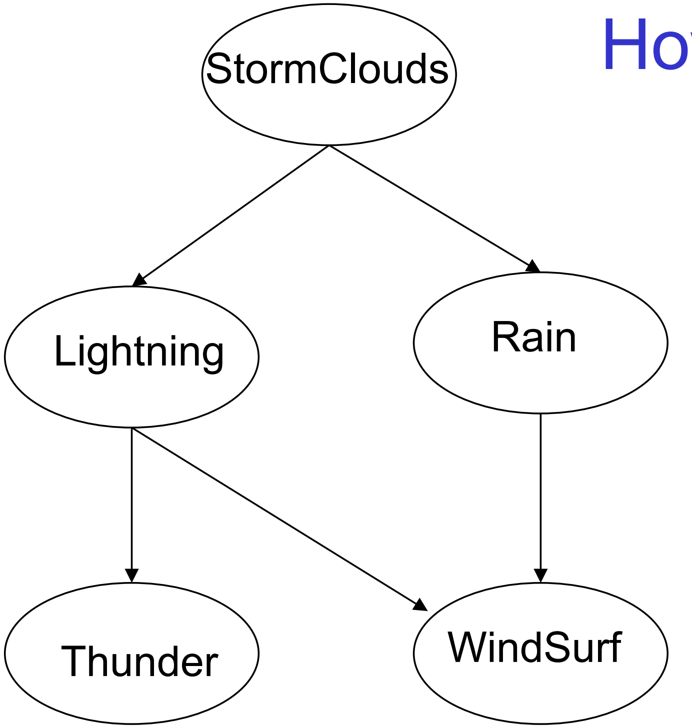


Chain rule of probability says that in general:

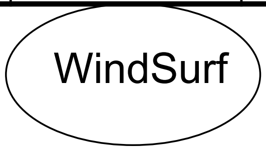
$$P(S,L,R,T,W) = P(S)P(L|S)P(R|S,L)P(T|S,L,R)P(W|S,L,R,T)$$

But in a Bayes net: $$P(X_1 \ldots X_n) = \prod_i P(X_i|Pa(X_i))$$

How Many Parameters?



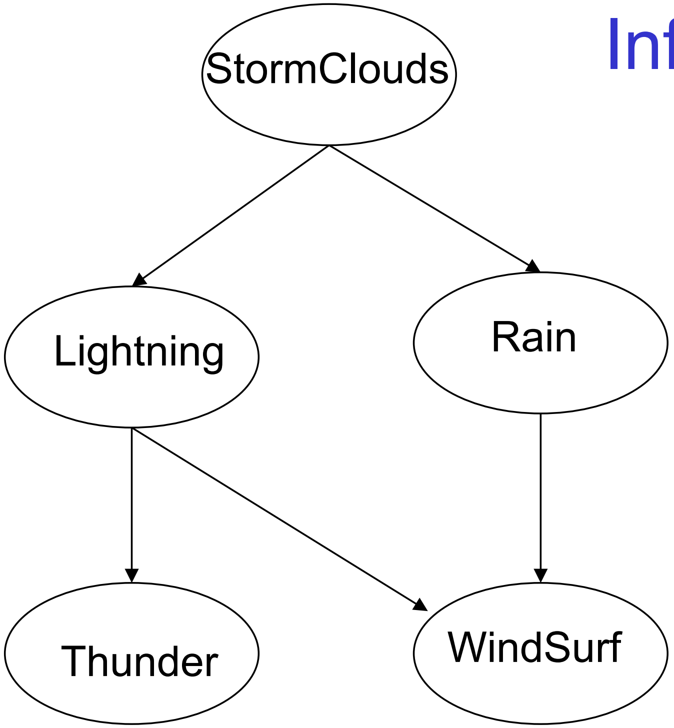
Parents	P(W|Pa)	P(¬W|Pa)
L, R	0	1.0
L, ¬R	0	1.0
¬L, R	0.2	0.8
¬L, ¬R	0.9	0.1



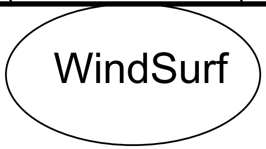
To define joint distribution in general?

To define joint distribution for this Bayes Net?

Inference in Bayes Nets

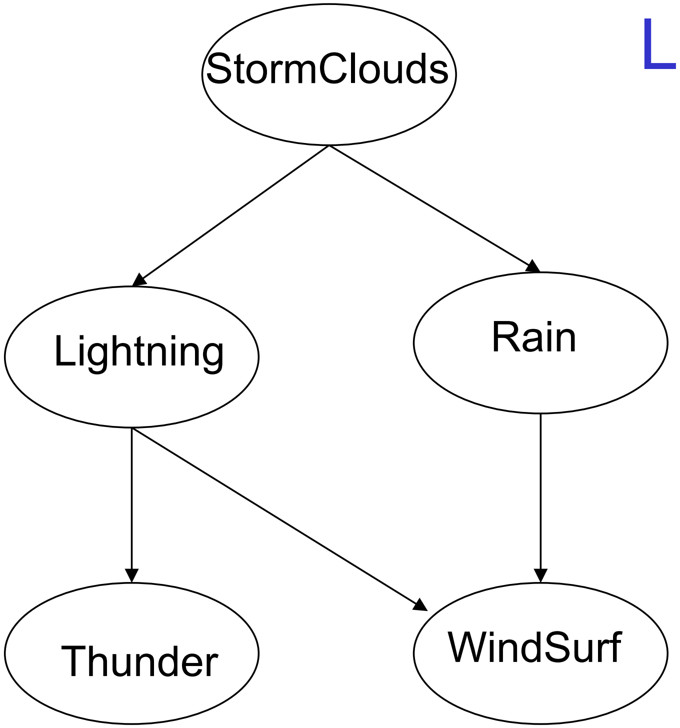


Parents	P(W|Pa)	P(¬W|Pa)
L, R	0	1.0
L, ¬R	0	1.0
¬L, R	0.2	0.8
¬L, ¬R	0.9	0.1

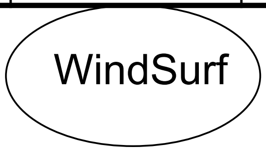


$P(S=1, L=0, R=1, T=0, W=1) =$

Learning a Bayes Net



Parents	P(W|Pa)	P(¬W|Pa)
L, R	0	1.0
L, ¬R	0	1.0
¬L, R	0.2	0.8
¬L, ¬R	0.9	0.1



Consider learning when graph structure is given, and data = { <s,l,r,t,w> }

What is the MLE solution? MAP?

Algorithm for Constructing Bayes Network

- Choose an ordering over variables, e.g., $X_1, X_2, \ldots X_n$
- For i=1 to n
  - Add X_i to the network
  - Select parents $Pa(X_i)$ as minimal subset of $X_1 \ldots X_{i-1}$ such that

$$P(X_i|Pa(X_i)) = P(X_i|X_1, \ldots, X_{i-1})$$

Notice this choice of parents assures

$$P(X_1 \ldots X_n) = \prod_i P(X_i|X_1 \ldots X_{i-1}) \quad \text{(by chain rule)}$$

$$= \prod_i P(X_i|Pa(X_i)) \quad \text{(by construction)}$$

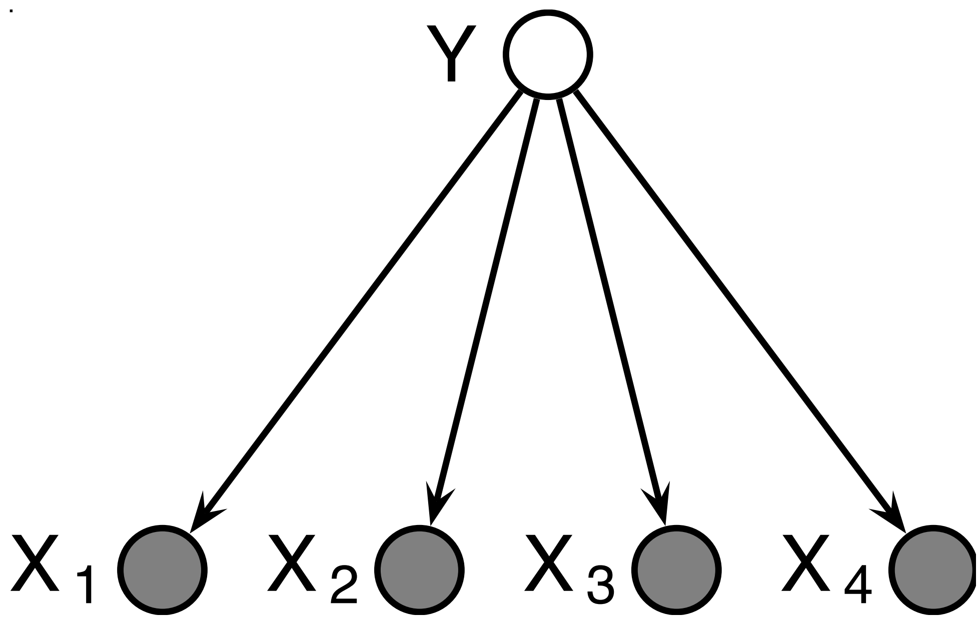
Example

- Bird flu and Allegies both cause Nasal problems
- Nasal problems cause Sneezes and Headaches

What is the Bayes Network for X1,…X4 with NO assumed conditional independencies?

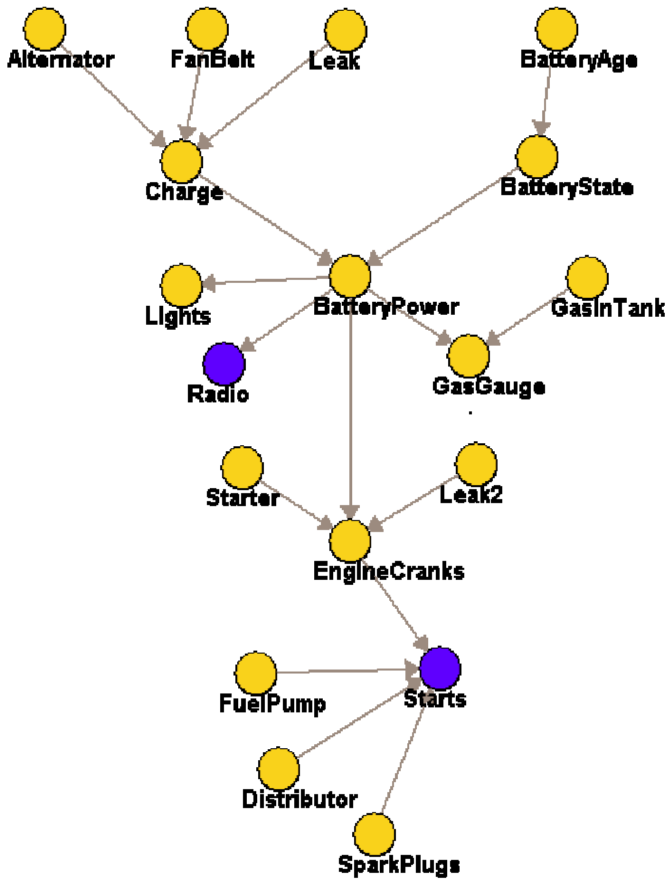
What is the Bayes Network for Naïve Bayes?

Naïve Bayes
(Same as Gaussian Mixture Model w/ Diagonal Covariance)



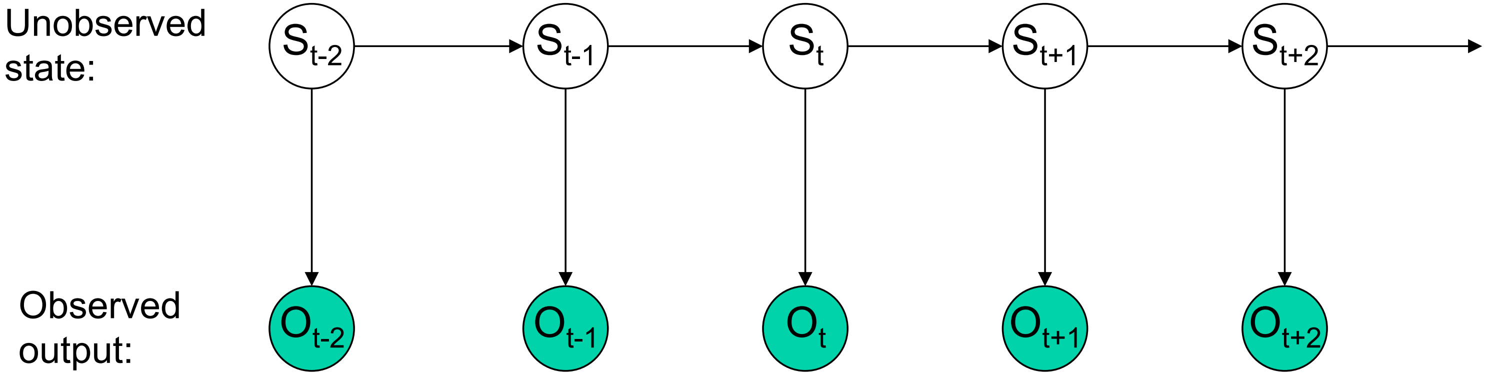
$$P(y, x_{1:D}) = P(y) \prod_{j=1}^{D} P(x_j | y)$$

What do we do if variables are mix of discrete and real valued?



Bayes Network for a Hidden Markov Model

Implies the future is conditionally independent of the past, given the present



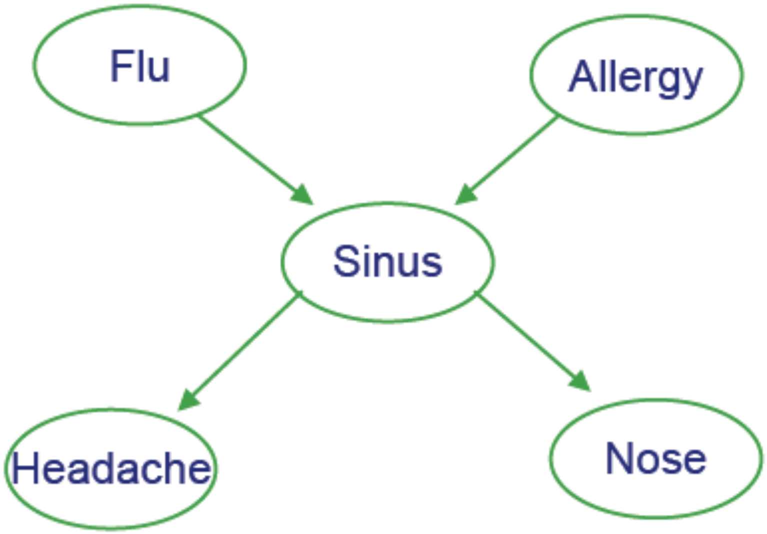
$$P(S_{t-2}, O_{t-2}, S_{t-1}, \ldots, O_{t+2}) =$$

Inference in Bayes Nets

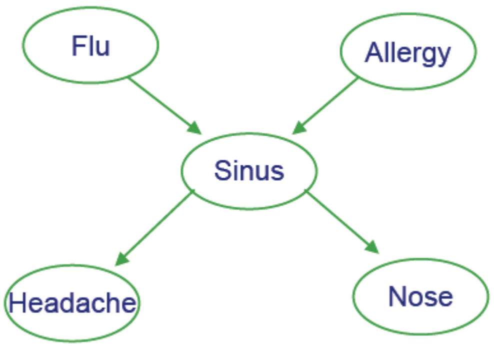
- In general, intractable (NP-complete)
- For certain cases, tractable
  - Assigning probability to fully observed set of variables
  - Or if just one variable unobserved
  - Or for singly connected graphs (ie., no undirected loops)
    - Variable elimination
    - Belief propagation
- For multiply connected graphs
    - Junction tree
- Sometimes use Monte Carlo methods
  - Generate many samples according to the Bayes Net distribution, then count up the results
- Variational methods for tractable approximate solutions

Example

- Bird flu and Allegies both cause Sinus problems
- Sinus problems cause Headaches and runny Nose



Prob. of joint assignment: easy

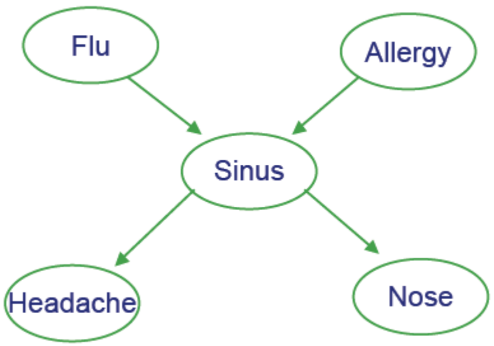


- Suppose we are interested in joint assignment <F=f,A=a,S=s,H=h,N=n>

What is P(f,a,s,h,n)?

let's use p(a,b) as shorthand for p(A=a, B=b)

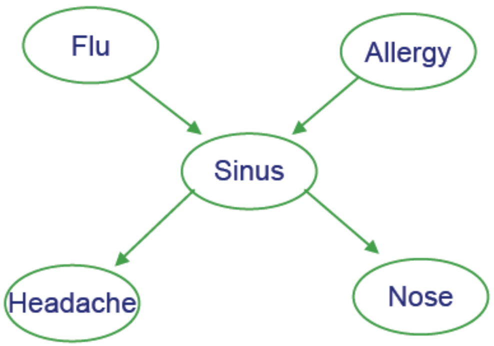
Prob. of marginals: not so easy



- How do we calculate P(N=n) ?

let's use p(a,b) as shorthand for p(A=a, B=b)

Generating a sample from joint distribution: easy



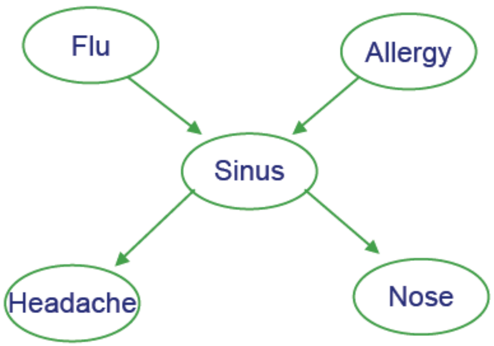
How can we generate random samples drawn according to P(F,A,S,H,N)?

Hint: random sample of F according to $P(F=1) = \theta_{F=1}$:

- draw a value of r uniformly from [0,1]
- if $r<\theta$ then output F=1, else F=0

let's use p(a,b) as shorthand for p(A=a, B=b)

Generating a sample from joint distribution: easy



How can we generate random samples drawn according to P(F,A,S,H,N)?

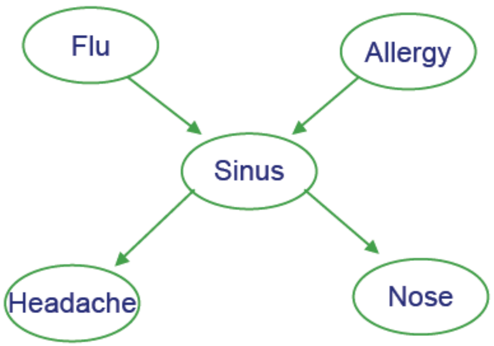
Hint: random sample of F according to $P(F=1) = \theta_{F=1}$:

- draw a value of r uniformly from [0,1]
- if $r<\theta$ then output F=1, else F=0

Solution:

- draw a random value f for F, using its CPD
- then draw values for A, for S|A,F, for H|S, for N|S

Generating a sample from joint distribution: easy



Note we can estimate marginals

like P(N=n) by generating many samples

from joint distribution, then count the fraction of samples for which N=n

Similarly, for anything else we care about

P(F=1|H=1, N=0)

→ weak but general method for estimating <u>any</u> probability term…

Learning of Bayes Nets

- Four categories of learning problems
  - Graph structure may be known/unknown
  - Variable values may be fully observed / partly unobserved

- Easy case: learn parameters for graph structure is *known*, and data is *fully observed*

- Interesting case: graph *known*, data *partly known*

- Gruesome case: graph structure *unknown*, data *partly unobserved*

Learning CPTs from Fully Observed Data

- Example: Consider learning the parameter

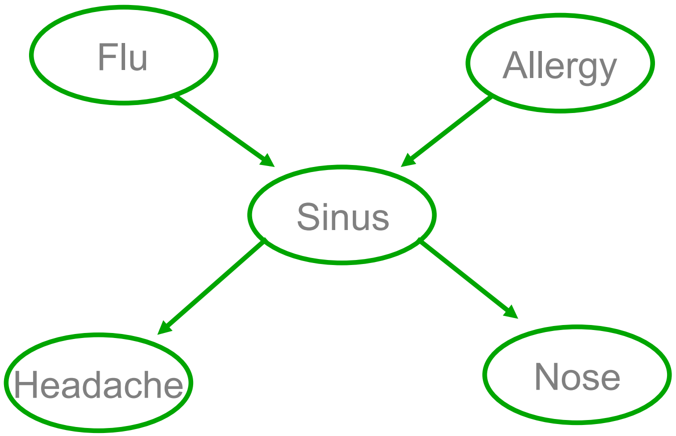
$$\theta_{s|ij} \equiv P(S=1|F=i, A=j)$$

- Max Likelihood Estimate is

$$\theta_{s|ij} = \frac{\sum_{k=1}^{K} \delta(f_k = i, a_k = j, s_k = 1)}{\sum_{k=1}^{K} \delta(f_k = i, a_k = j)}$$

k^{th} training example

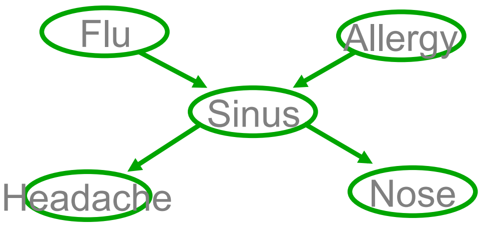
δ(x) = 1 if x=true,
= 0 if x=false



- Remember why?

let's use p(a,b) as shorthand for p(A=a, B=b)

MLE estimate of $\theta_{s|ij}$ from fully observed data



- Maximum likelihood estimate

$$\theta \leftarrow \arg\max_{\theta} \log P(data|\theta)$$

- Our case:

$$P(data|\theta) = \prod_{k=1}^{K} P(f_k, a_k, s_k, h_k, n_k)$$

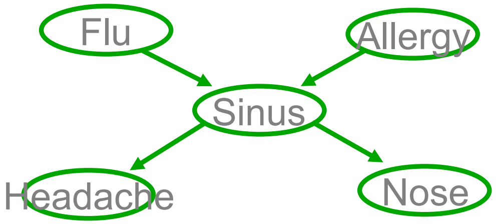
$$P(data|\theta) = \prod_{k=1}^{K} P(f_k)P(a_k)P(s_k|f_k a_k)P(h_k|s_k)P(n_k|s_k)$$

$$\log P(data|\theta) = \sum_{k=1}^{K} \log P(f_k) + \log P(a_k) + \log P(s_k|f_k a_k) + \log P(h_k|s_k) + \log P(n_k|s_k)$$

$$\frac{\partial \log P(data|\theta)}{\partial \theta_{s|ij}} = \sum_{k=1}^{K} \frac{\partial \log P(s_k|f_k a_k)}{\partial \theta_{s|ij}}$$

$$\theta_{s|ij} = \frac{\sum_{k=1}^{K} \delta(f_k = i, a_k = j, s_k = 1)}{\sum_{k=1}^{K} \delta(f_k = i, a_k = j)}$$

Estimate θ from partly observed data



- What if FAHN observed, but not S?
- Can't calculate MLE

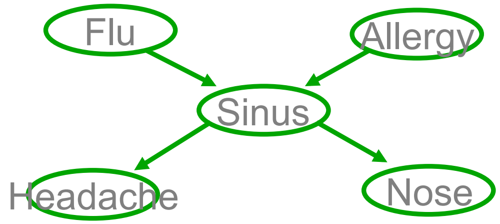
$$\theta \leftarrow \arg\max_\theta \log \prod_k P(f_k, a_k, s_k, h_k, n_k | \theta)$$

- Let X be all *observed* variable values (over all examples)
- Let Z be all *unobserved* variable values
- Can't calculate MLE:

$$\theta \leftarrow \arg\max_\theta \log P(X, Z | \theta)$$

- WHAT TO DO?

Estimate θ from partly observed data



- What if FAHN observed, but not S?
- Can't calculate MLE

$$\theta \leftarrow \arg\max_\theta \log \prod_k P(f_k, a_k, s_k, h_k, n_k | \theta)$$

- Let X be all *observed* variable values (over all examples)
- Let Z be all *unobserved* variable values
- Can't calculate MLE:

$$\theta \leftarrow \arg\max_\theta \log P(X, Z|\theta)$$

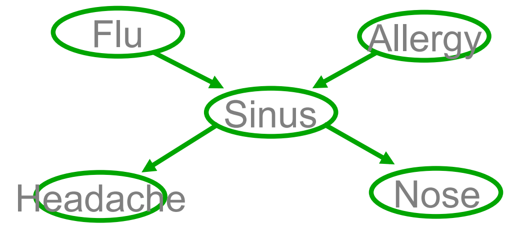
- EM seeks* to estimate:

$$\theta \leftarrow \arg\max_\theta E_{Z|X,\theta}[\log P(X, Z|\theta)]$$

* EM guaranteed to find local maximum

- EM seeks estimate:

$$\theta \leftarrow \arg\max_{\theta} E_{Z|X,\theta}[\log P(X, Z|\theta)]$$



- here, observed X={F,A,H,N}, unobserved Z={S}

$$\log P(X, Z|\theta) = \sum_{k=1}^{K} \log P(f_k) + \log P(a_k) + \log P(s_k|f_k a_k) + \log P(h_k|s_k) + \log P(n_k|s_k)$$

$$E_{P(Z|X,\theta)} \log P(X, Z|\theta) = \sum_{k=1}^{K} \sum_{i=0}^{1} P(s_k = i|f_k, a_k, h_k, n_k)$$

$$[logP(f_k) + \log P(a_k) + \log P(s_k|f_k a_k) + \log P(h_k|s_k) + \log P(n_k|s_k)]$$

EM Algorithm

EM is a general procedure for learning from partly observed data

Given observed variables X, unobserved Z (X={F,A,H,N}, Z={S}) ✓

Define $Q(\theta'|\theta) = E_{P(Z|X,\theta)}[\log P(X, Z|\theta')]$

current

M step new

Iterate until convergence:

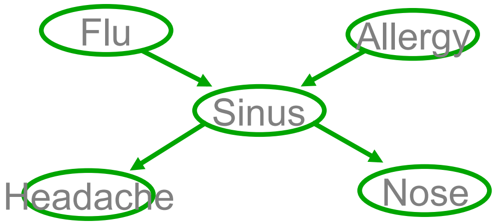
- E Step: Use X and current θ to calculate $P(Z|X,\theta)$
- M Step: Replace current θ by

$$\theta \leftarrow \arg\max_{\theta'} Q(\theta'|\theta)$$

Guaranteed to find local maximum.
Each iteration increases $E_{P(Z|X,\theta)}[\log P(X, Z|\theta')]$

E Step: Use X, θ, to Calculate P(Z|X,θ)

observed X={F,A,H,N},
unobserved Z={S}



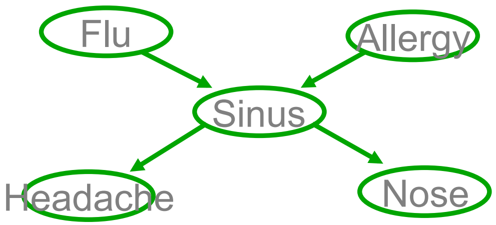
- How? Bayes net inference problem.

$$P(S_k = 1 | f_k a_k h_k n_k, \theta) =$$

let's use p(a,b) as shorthand for p(A=a, B=b)

E Step: Use X, θ, to Calculate P(Z|X,θ)

observed X={F,A,H,N},
unobserved Z={S}



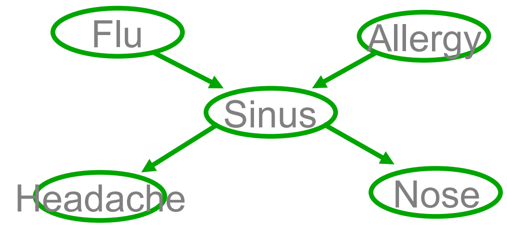
- How? Bayes net inference problem.

$$P(S_k = 1 | f_k a_k h_k n_k, \theta) =$$

$$P(S_k = 1 | f_k a_k h_k n_k, \theta) = \frac{P(S_k = 1, f_k a_k h_k n_k | \theta)}{P(S_k = 1, f_k a_k h_k n_k | \theta) + P(S_k = 0, f_k a_k h_k n_k | \theta)}$$

let's use p(a,b) as shorthand for p(A=a, B=b)

EM and estimating $\theta_{s|ij}$



observed X = {F,A,H,N}, unobserved Z={S}

E step: Calculate $P(Z_k|X_k; \theta)$ for each training example, k

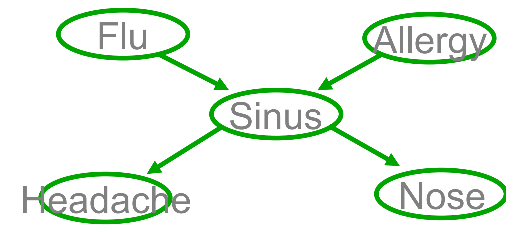
$$P(S_k = 1|f_k a_k h_k n_k, \theta) = \underset{P(Z|X;\theta)}{E[s_k]} = \frac{P(S_k = 1, f_k a_k h_k n_k|\theta)}{P(S_k = 1, f_k a_k h_k n_k|\theta) + P(S_k = 0, f_k a_k h_k n_k|\theta)}$$

M step: update all relevant parameters. For example:

$$\theta_{s|ij} \leftarrow \frac{\sum_{k=1}^{K} \delta(f_k = i, a_k = j) \; E[s_k]}{\sum_{k=1}^{K} \delta(f_k = i, a_k = j)}$$

Recall MLE was: $\theta_{s|ij} = \dfrac{\sum_{k=1}^{K} \delta(f_k = i, a_k = j, s_k = 1)}{\sum_{k=1}^{K} \delta(f_k = i, a_k = j)}$

EM and estimating θ



More generally,
Given observed set X, unobserved set Z of boolean values

E step: Calculate for each training example, k

the expected value of each unobserved variable

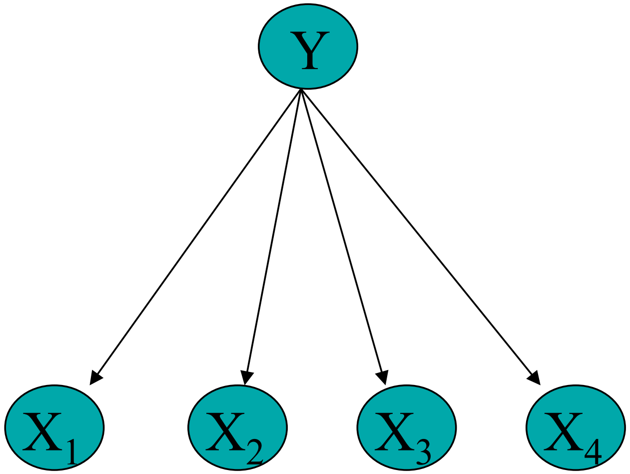
M step:

Calculate estimates similar to MLE, but replacing each count by its <u>expected count</u>

$$\delta(Y=1) \rightarrow E_{Z|X,\theta}[Y] \qquad \delta(Y=0) \rightarrow (1 - E_{Z|X,\theta}[Y])$$

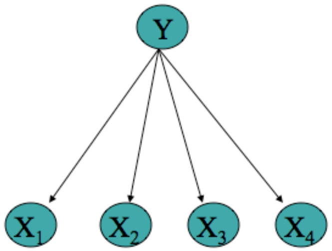
Using Unlabeled Data to Help Train Naïve Bayes Classifier

Learn P(Y|X)



Y	X1	X2	X3	X4
1	0	0	1	1
0	0	1	0	0
0	0	0	1	0
?	0	1	1	0
?	0	1	0	1

EM and estimating θ



Given observed set X, unobserved set Y of boolean values

E step: Calculate for each training example, k

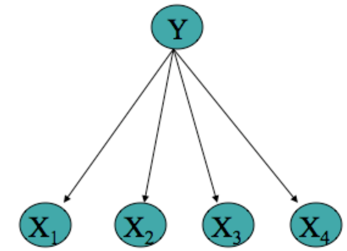
the expected value of each unobserved variable Y

$$E_{P(Y|X_1 \ldots X_N)}[y(k)] = P(y(k)=1|x_1(k), \ldots x_N(k); \theta) = \frac{P(y(k)=1) \prod_i P(x_i(k)|y(k)=1)}{\sum_{j=0}^{1} P(y(k)=j) \prod_i P(x_i(k)|y(k)=j)}$$

M step: Calculate estimates similar to MLE, but replacing each count by its <u>expected count</u>

let's use y(k) to indicate value of Y on kth example

EM and estimating θ



Given observed set X, unobserved set Y of boolean values

E step: Calculate for each training example, k

the expected value of each unobserved variable Y

$$E_{P(Y|X_1 \ldots X_N)}[y(k)] = P(y(k) = 1 | x_1(k), \ldots x_N(k); \theta) = \frac{P(y(k) = 1) \prod_i P(x_i(k)|y(k) = 1)}{\sum_{j=0}^{1} P(y(k) = j) \prod_i P(x_i(k)|y(k) = j)}$$

M step: Calculate estimates similar to MLE, but replacing each count by its expected count

$$\theta_{ij|m} = \hat{P}(X_i = j | Y = m) = \frac{\sum_k P(y(k) = m | x_1(k) \ldots x_N(k)) \ \delta(x_i(k) = j)}{\sum_k P(y(k) = m | x_1(k) \ldots x_N(k))}$$

MLE would be: $\hat{P}(X_i = j | Y = m) = \frac{\sum_k \delta((y(k) = m) \wedge (x_i(k) = j))}{\sum_k \delta(y(k) = m)}$

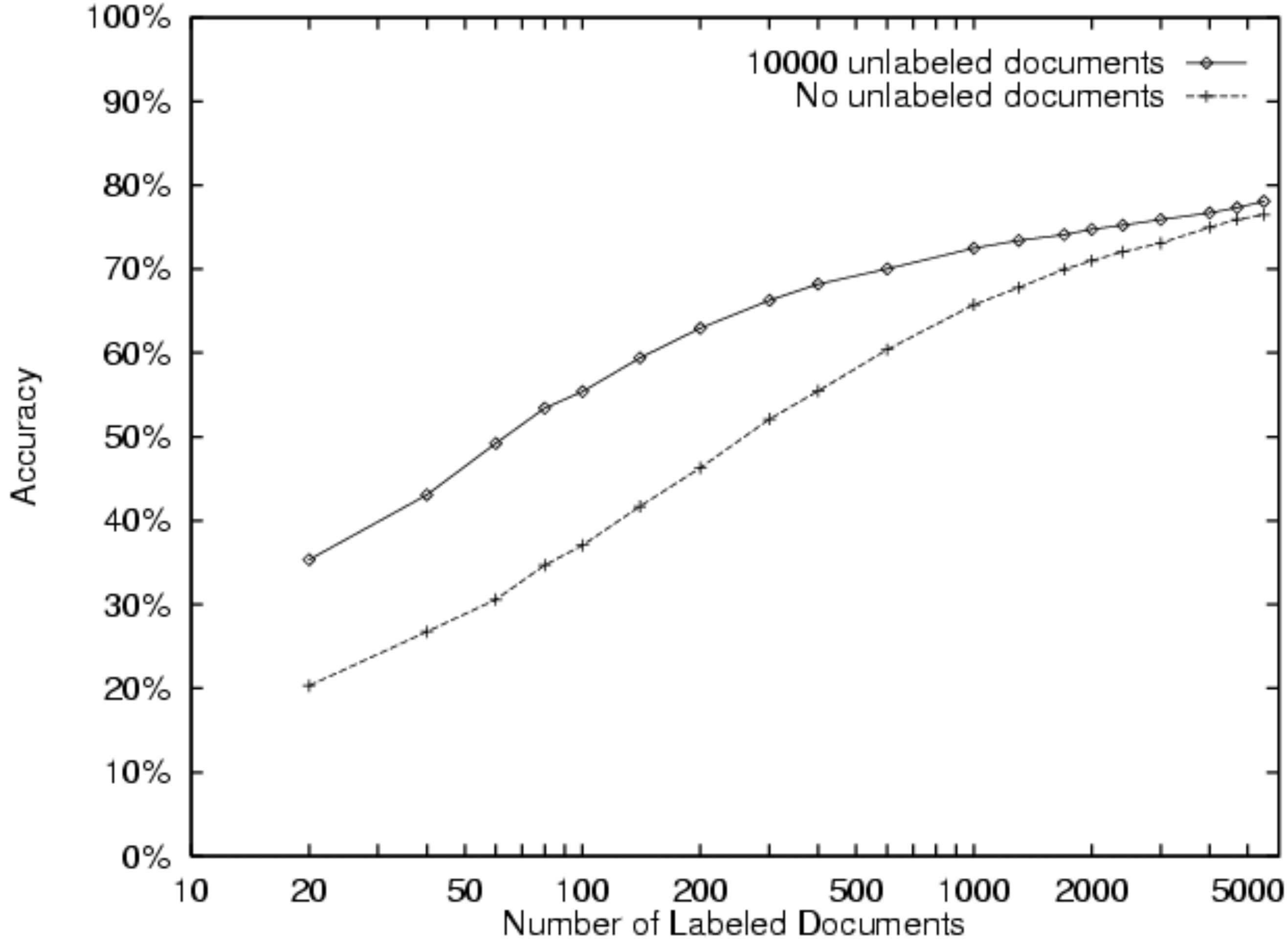
- **Inputs:** Collections $\mathcal{D}^l$ of labeled documents and $\mathcal{D}^u$ of unlabeled documents.
- Build an initial naive Bayes classifier, $\hat{\theta}$, from the labeled documents, $\mathcal{D}^l$, only. Use maximum a posteriori parameter estimation to find $\hat{\theta} = \arg\max_\theta \mathrm{P}(\mathcal{D}|\theta)\mathrm{P}(\theta)$ (see Equations 5 and 6).
- Loop while classifier parameters improve, as measured by the change in $l_c(\theta|\mathcal{D};\mathbf{z})$ (the complete log probability of the labeled and unlabeled data
  - **(E-step)** Use the current classifier, $\hat{\theta}$, to estimate component membership of each unlabeled document, *i.e.*, the probability that each mixture component (and class) generated each document, $\mathrm{P}(c_j|d_i;\hat{\theta})$ (see Equation 7).
  - **(M-step)** Re-estimate the classifier, $\hat{\theta}$, given the estimated component membership of each document. Use maximum a posteriori parameter estimation to find $\hat{\theta} = \arg\max_\theta \mathrm{P}(\mathcal{D}|\theta)\mathrm{P}(\theta)$ (see Equations 5 and 6).
- **Output:** A classifier, $\hat{\theta}$, that takes an unlabeled document and predicts a class label.

From [Nigam et al., 2000]

Experimental Evaluation

- Newsgroup postings
  - 20 newsgroups, 1000/group
- Web page classification
  - student, faculty, course, project
  - 4199 web pages
- Reuters newswire articles
  - 12,902 articles
  - 90 topics categories

20 Newsgroups



Conditional Independence Properties

- A is independent of B given C

$$X_A \perp_G X_B | X_C$$

- I(G) is the set of all such conditional independence assumptions encoded by G
- G is an I-map for P iff I(G) $\subseteq$ I(P)
  - Where I(P) is the set of all CI statements that hold for P
  - In other words: G doesn't make any assertions that are not true about P

Conditional Independence Properties (cont)

- Note: fully connected graph is an I-map for all distributions
- G is a **minimal I-map** of P if:
  - G is an I-map of P
  - There is no G' $\subseteq$ G which is an I-map of P
- Question:
  - How to determine if $X_A \perp_G X_B | X_C$?
  - Easy for undirected graphs
  - Kind of complicated for DAGs (Bayesian Nets)

D-separation

- Definitions:
  - An undirected path P is d-separated by a set of nodes E (containing evidence) iff at least one of the following conditions hold:
    - P contains a chain *s* -> *m* -> *t* or *s* <- *m* <- *t* where *m* is evidence
    - P contains a **fork** *s* <- *m* -> *t* where *m* is in the evidence
    - P contains a **v-structure** *s* -> *m* <- *t* where *m* is **not** in the evidence, nor any descendent of *m*

D-seperation (cont)

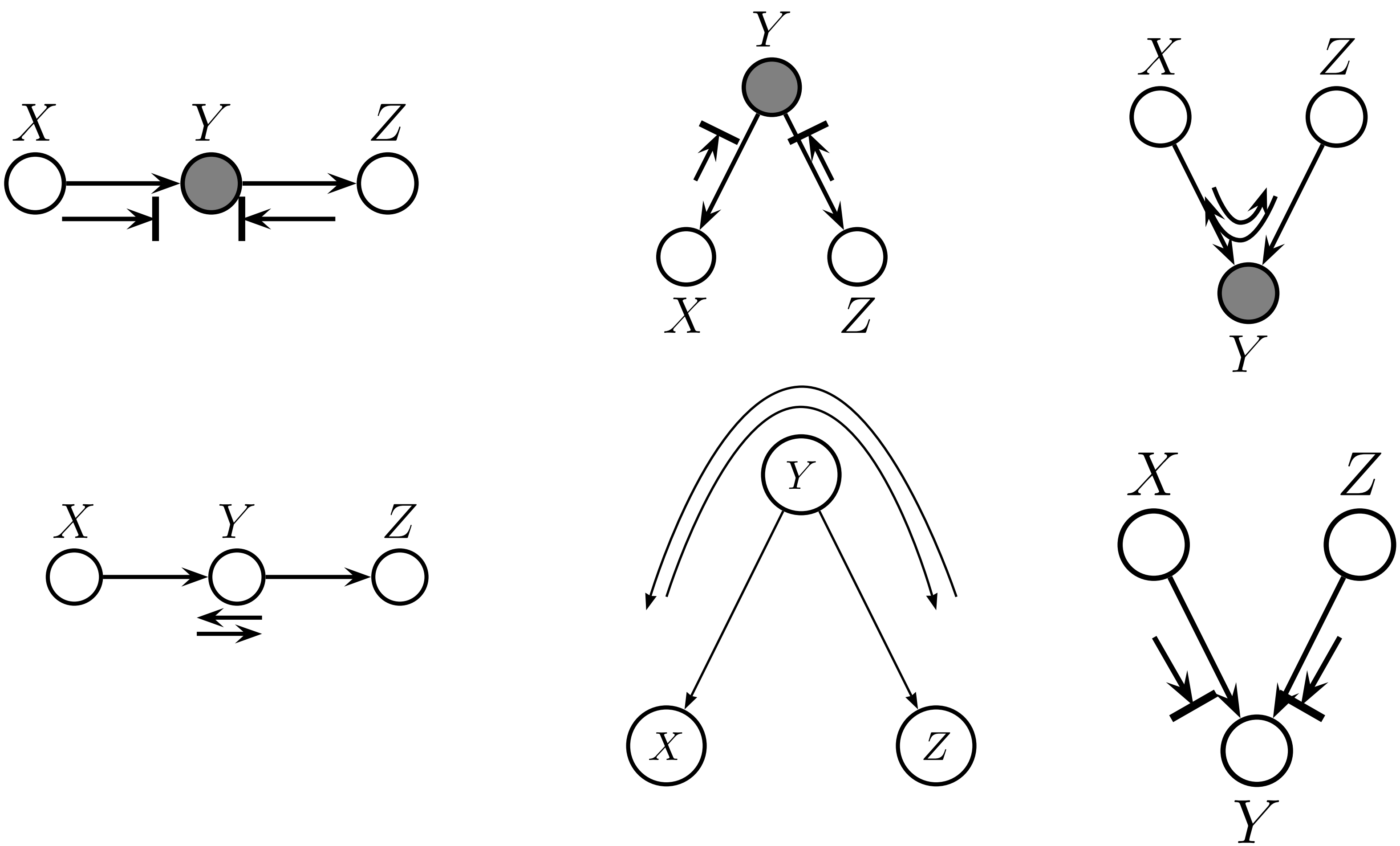
- A set of nodes A is **D-separated** from a set of nodes B, if given a third set of nodes E iff each undirected path from every node in A to every node in B is d-seperated by E
- Finally, define the CI properties of a DAG as follows:

$$X_A \perp_G X_B | X_E \iff \text{A is d-seperated from B given E}$$

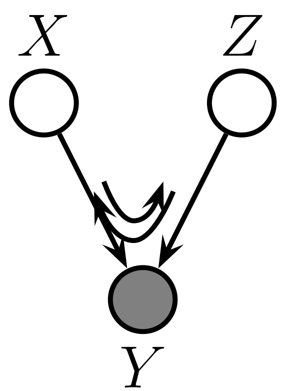
Bayes Ball Algorithm

- Simple way to check if A is d-separated from B given E
  1. Shade in all nodes in E
  2. Place “balls” in each node in A and let them “bounce around” according to some rules
     - Note: balls can travel in either direction
  3. Check if any balls from A reach nodes in B

Bayes Ball Rules



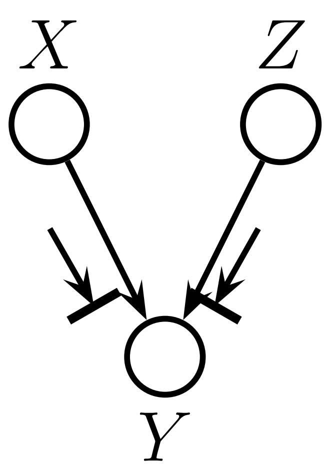
Explaining Away (inter-causal reasoning)



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

$$\implies x \not\perp z|y$$

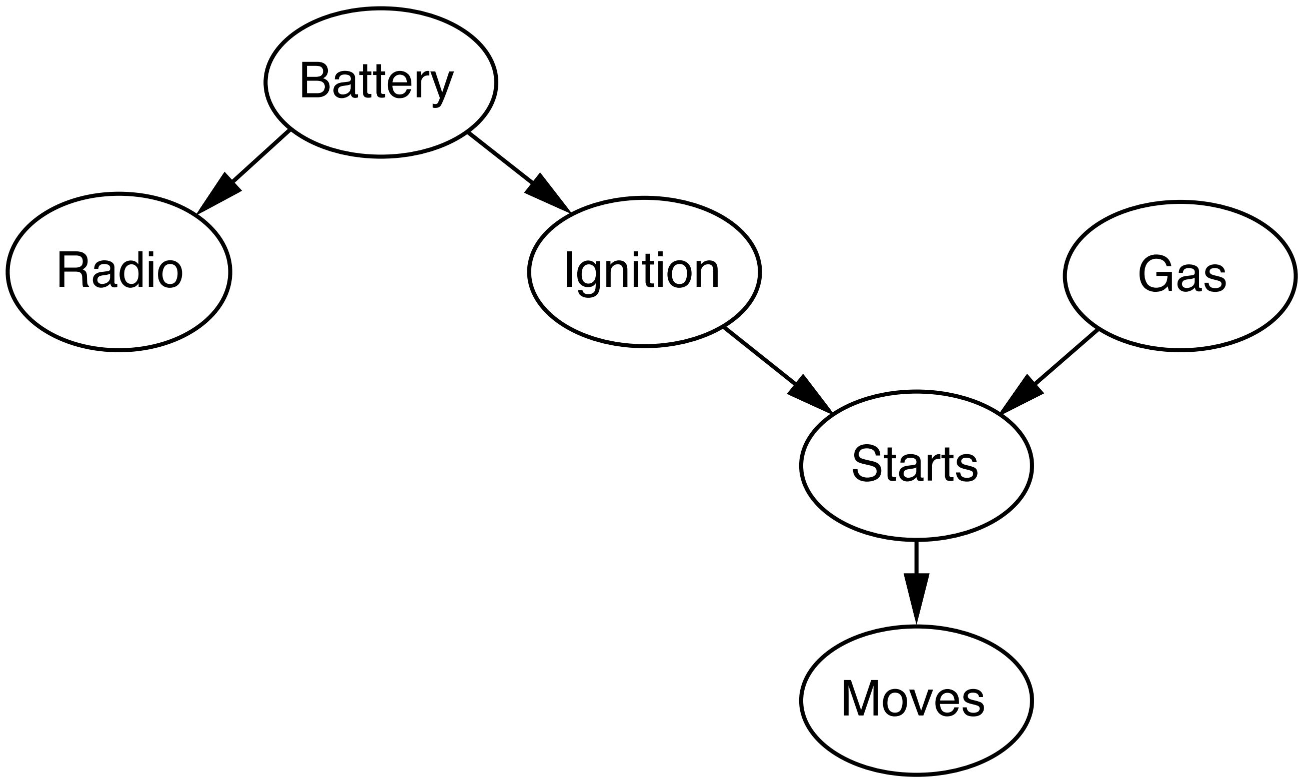
Example: Toss two coins and observe their sum



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

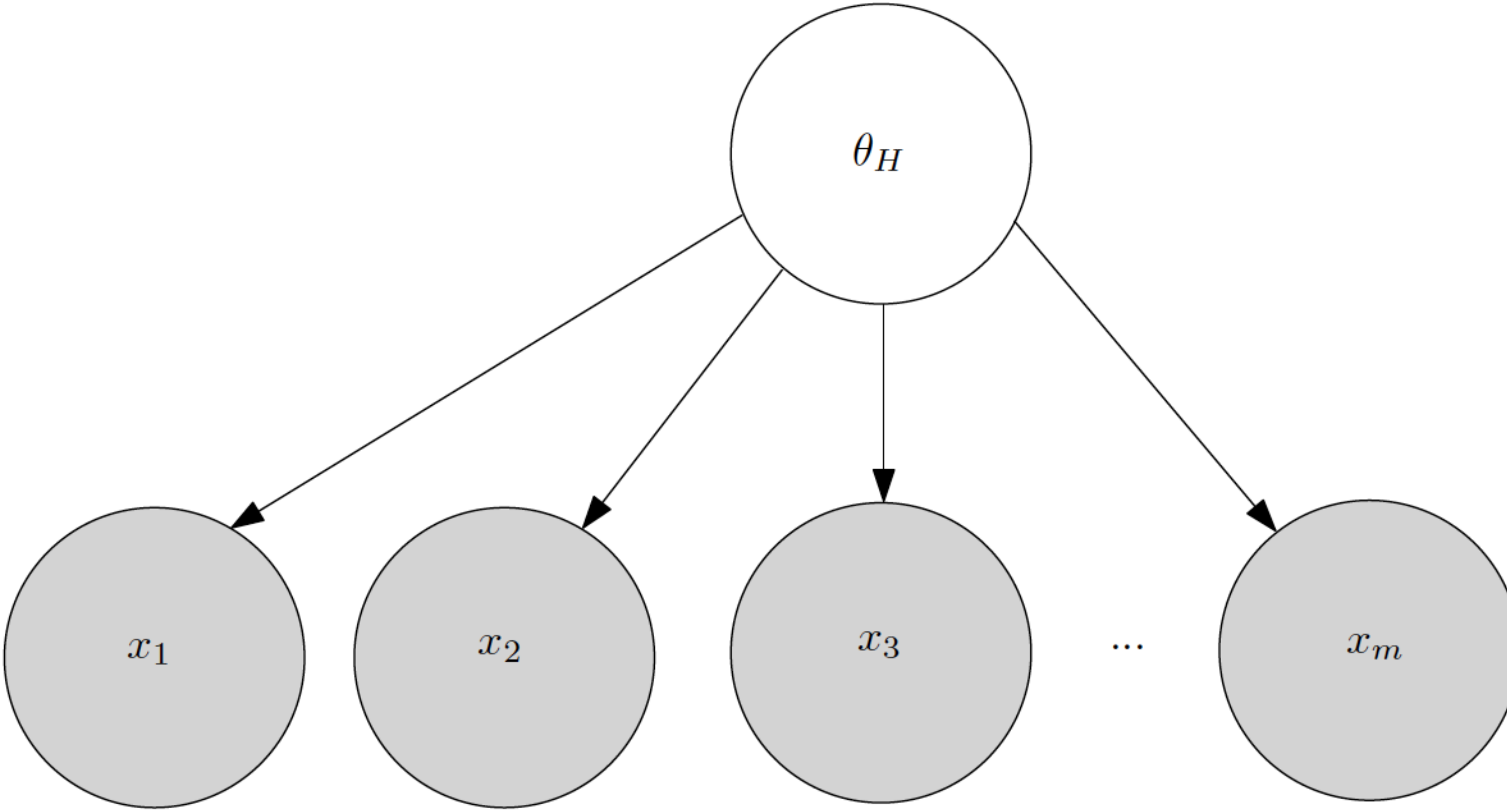
$$\implies x \perp z$$

Example



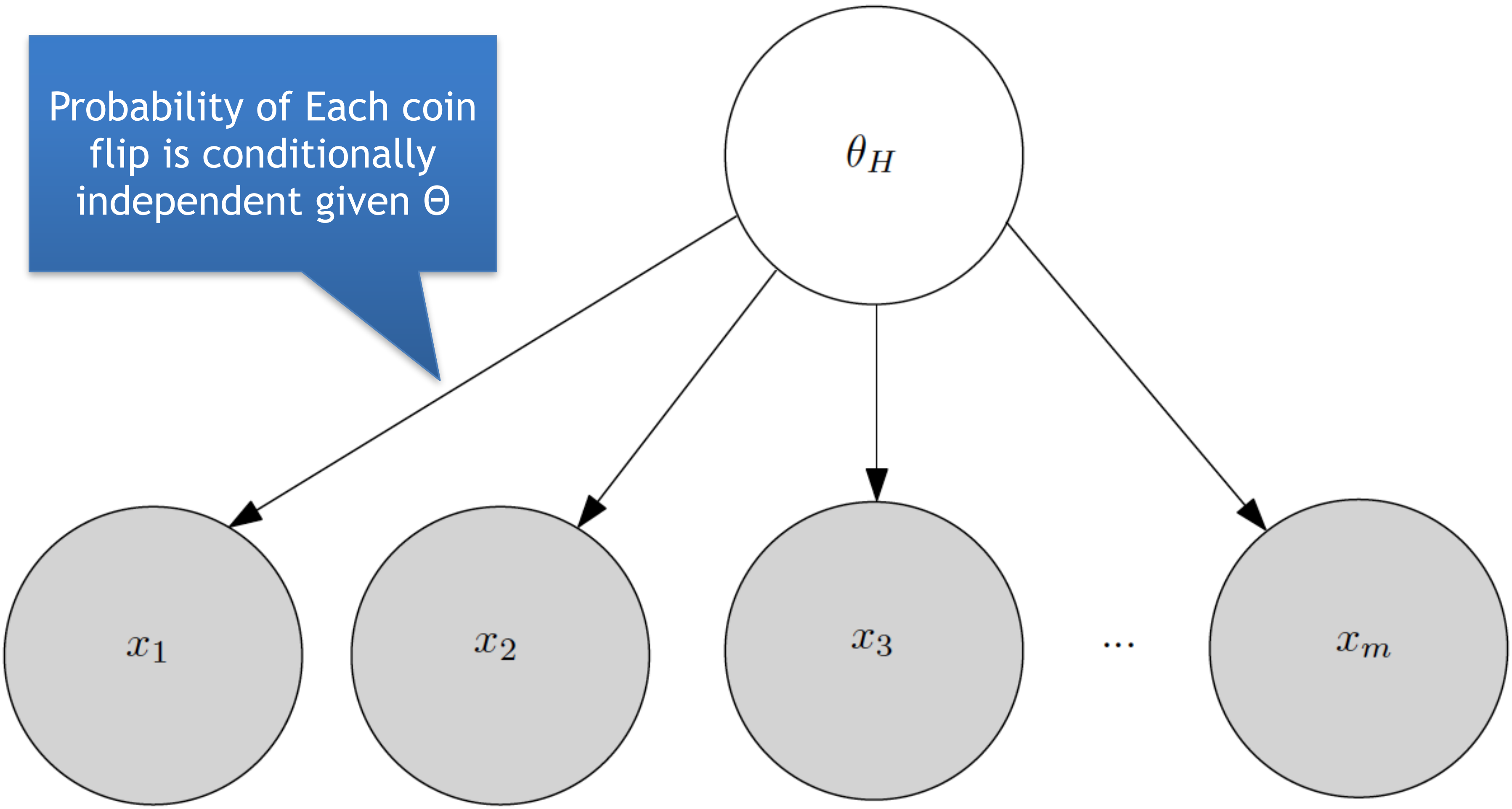
Are Gas and Radio independent? Given Battery? Ignition? Starts? Moves?

Bent Coin Bayesian Network



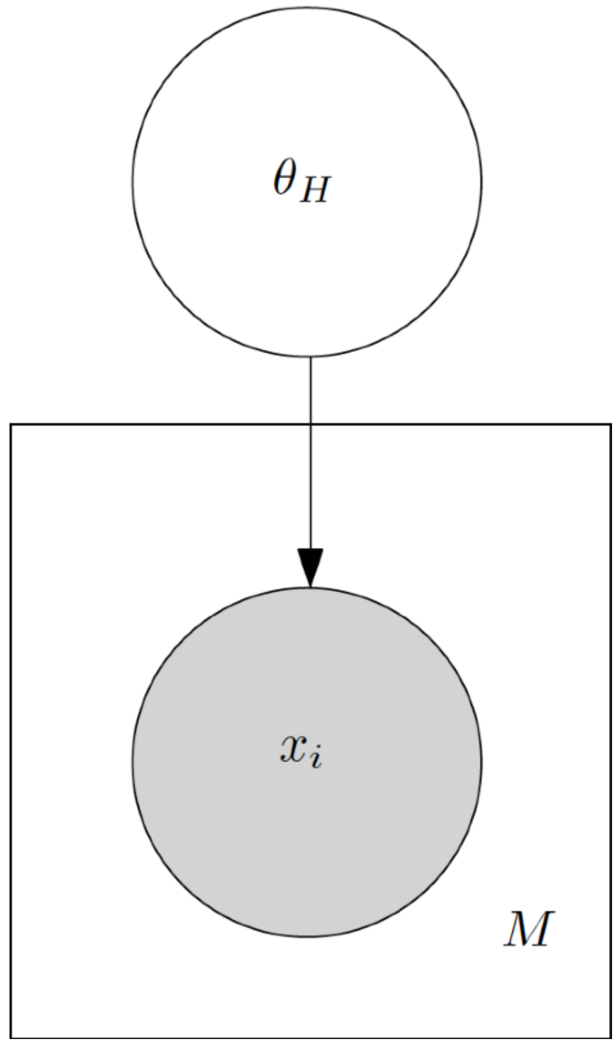
$$P(x_1, x_2, \ldots, x_n|\theta_H) = P(\theta_H)P(x_1|\theta_H)P(x_2|\theta_H)\ldots P(x_n|\theta_H)$$

Bent Coin Bayesian Network



$$P(x_1, x_2, \ldots, x_n|\theta_H) = P(\theta_H)P(x_1|\theta_H)P(x_2|\theta_H)\ldots P(x_n|\theta_H)$$

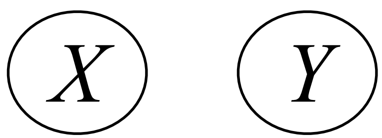
Bent Coin Bayesian Network (Plate Notation)



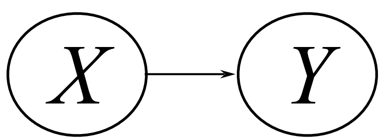
Learning Bayes-net structure

Given data, which model is correct?

model 1:

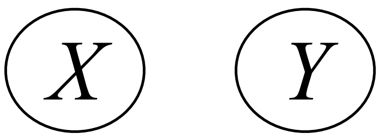


model 2:



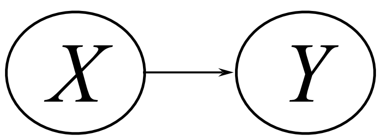
Bayesian approach

Given data, which model is ~~correct?~~ more likely?

model 1: 

$p(m_1) = 0.7$ $\qquad$ $p(m_1 \mid \mathbf{d}) = 0.1$

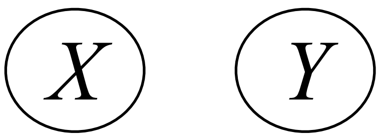
Data $\mathbf{d}$ $\longrightarrow$

model 2: 

$p(m_2) = 0.3$ $\qquad$ $p(m_2 \mid \mathbf{d}) = 0.9$

Bayesian approach: Model averaging

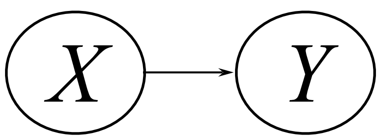
Given data, which model is ~~correct~~? more likely?

model 1: 

$p(m_1) = 0.7$

$p(m_1 \mid \mathbf{d}) = 0.1$

Data $\mathbf{d}$

model 2: 

$p(m_2) = 0.3$

$p(m_2 \mid \mathbf{d}) = 0.9$

average predictions

Bayesian approach: Model selection

Given data, which model is ~~correct~~? more likely?

model 1: $X \quad Y$ $\quad p(m_1) = 0.7 \qquad p(m_1 \mid \mathbf{d}) = 0.1$

Data $\mathbf{d}$ $\longrightarrow$

model 2: $X \rightarrow Y$ $\quad p(m_2) = 0.3 \qquad p(m_2 \mid \mathbf{d}) = 0.9$

Keep the best model:
- Explanation
- Understanding
- Tractability

To score a model, use Bayes' theorem

Given data $\mathbf{d}$:

model score

$$p(m \mid \mathbf{d}) \propto p(m)\, \underline{p(\mathbf{d} \mid m)}$$

"marginal likelihood"

likelihood

$$p(\mathbf{d} \mid m) = \int p(\mathbf{d} \mid \theta, m)\, p(\theta \mid m)\, d\theta$$

Thumbtack example

X heads/tails

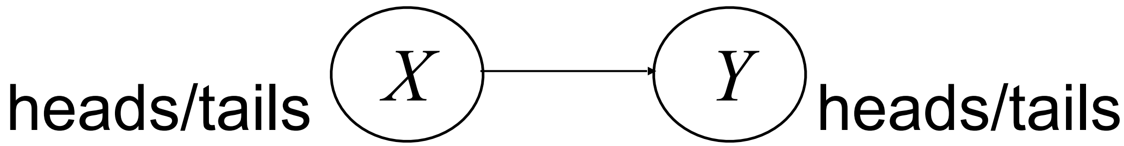
$$p(\mathbf{d} \mid m) = \int \theta^{\#h} (1-\theta)^{\#t} \; p(\theta \mid m)\, d\theta$$

$$= \int \theta^{\#h+\alpha_h-1} (1-\theta)^{\#t+\alpha_t-1} \; d\theta$$

conjugate prior

$$= \frac{\Gamma(\alpha_h + \alpha_t)}{\Gamma(\alpha_h + \alpha_t + \#h + \#t)} \frac{\Gamma(\alpha_h + \#h)}{\Gamma(\alpha_h)} \frac{\Gamma(\alpha_t + \#t)}{\Gamma(\alpha_t)}$$

More complicated graphs



3 separate thumbtack-like learning problems

$$
\begin{aligned}
p(\mathbf{d} \mid m) = {} & \frac{\Gamma(\alpha_h + \alpha_t)}{\Gamma(\alpha_h + \alpha_t + \#h + \#t)} \frac{\Gamma(\alpha_h + \#h)}{\Gamma(\alpha_h)} \frac{\Gamma(\alpha_t + \#t)}{\Gamma(\alpha_t)} && \text{X} \\
& \cdot \frac{\Gamma(\alpha_h + \alpha_t)}{\Gamma(\alpha_h + \alpha_t + \#h + \#t)} \frac{\Gamma(\alpha_h + \#h)}{\Gamma(\alpha_h)} \frac{\Gamma(\alpha_t + \#t)}{\Gamma(\alpha_t)} && \text{Y|X=heads} \\
& \cdot \frac{\Gamma(\alpha_h + \alpha_t)}{\Gamma(\alpha_h + \alpha_t + \#h + \#t)} \frac{\Gamma(\alpha_h + \#h)}{\Gamma(\alpha_h)} \frac{\Gamma(\alpha_t + \#t)}{\Gamma(\alpha_t)} && \text{Y|X=tails}
\end{aligned}
$$

Model score for a discrete Bayes net

$$p(\mathbf{d} \mid m) = \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})}$$

N_{ijk}: # cases where $X_i = x_{\mathrm{i}}^{\mathrm{k}}$ and $\mathbf{Pa}_i = \mathbf{pa}_i^j$

r_i: number of states of X_i

q_i: number of instances of parents of X_i

$$\alpha_{ij} = \sum_{k=1}^{r_i} \alpha_{ijk} \qquad N_{ij} = \sum_{k=1}^{r_i} N_{ijk}$$

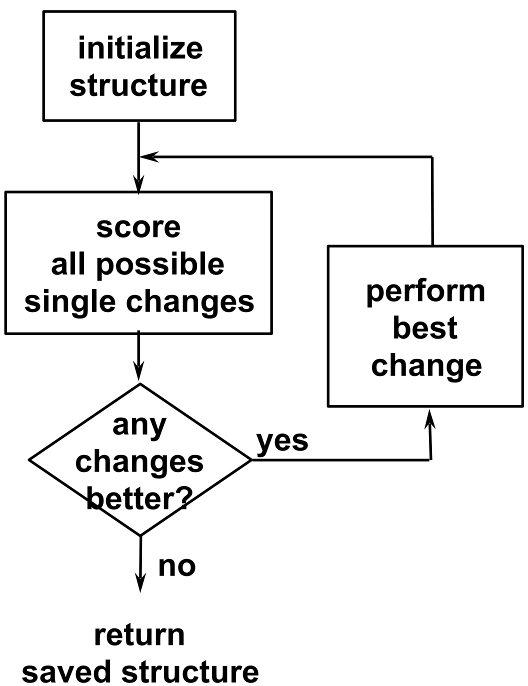
Computation of marginal likelihood

Efficient closed form if

- Local distributions from the exponential family (binomial, poisson, gamma, ...)
- Parameter independence
- Conjugate priors
- No missing data (including no hidden variables)

Structure search

- Finding the BN structure with the highest score among those structures with at most *k* parents is NP hard for *k*>1 (Chickering, 1995)
- Heuristic methods
  - Greedy
  - Greedy with restarts
  - MCMC methods



Structure priors

1. All possible structures equally likely
2. Partial ordering, required / prohibited arcs
3. Prior(m) α Similarity(m, prior BN)

Parameter priors

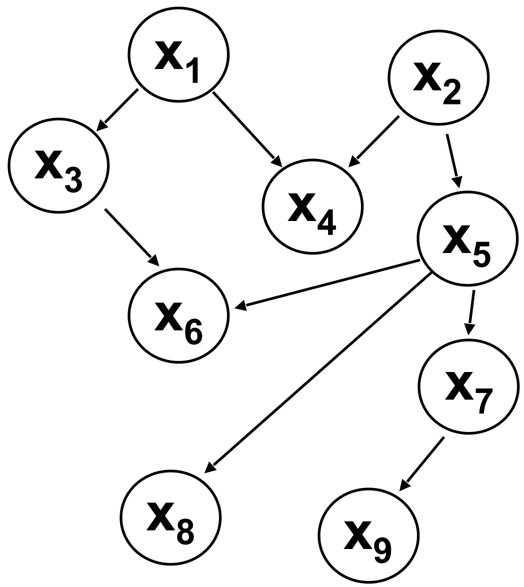
- All uniform: Beta(1,1)
- Use a prior Bayes net

Parameter priors

Recall the intuition behind the Beta prior for the thumbtack:

- The hyperparameters α_h and α_t can be thought of as imaginary counts from our prior experience, starting from "pure ignorance"
- Equivalent sample size = $\alpha_h + \alpha_t$
- The larger the equivalent sample size, the more confident we are about the long-run fraction

Parameter priors



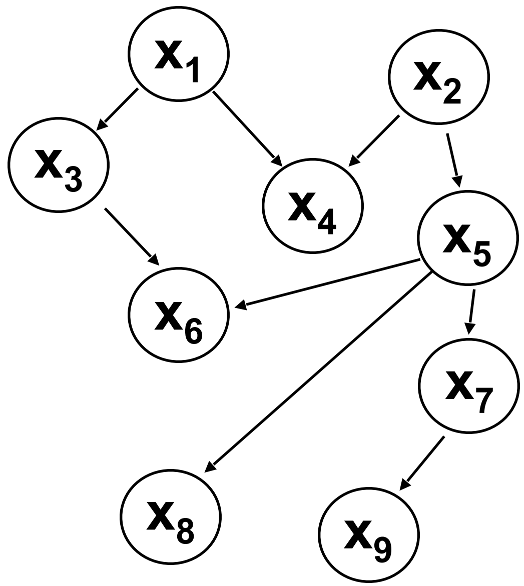
+ equivalent sample size → imaginary count for any variable configuration

parameter modularity

parameter priors for any Bayes net structure for $X_1 \ldots X_n$

Combining knowledge & data

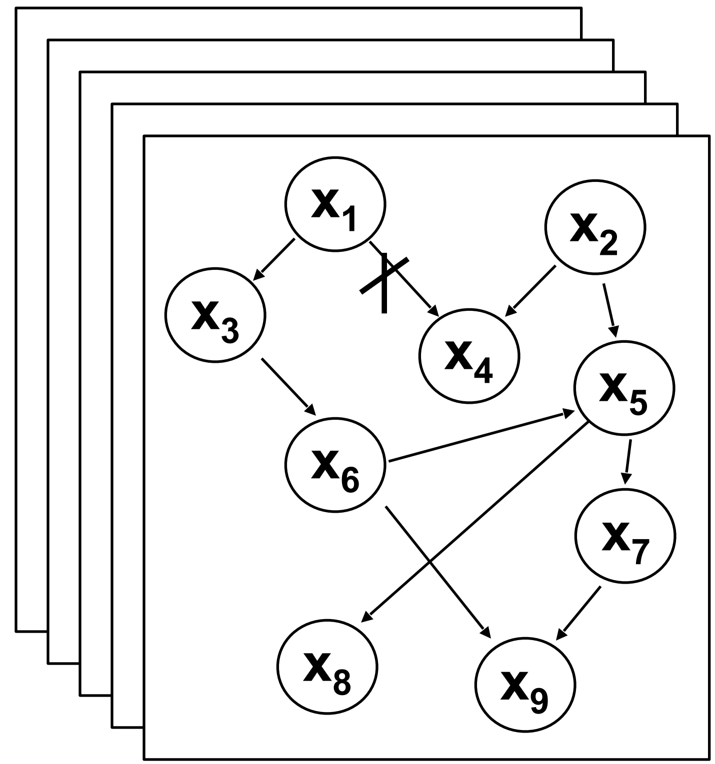
prior network+equivalent sample size



data

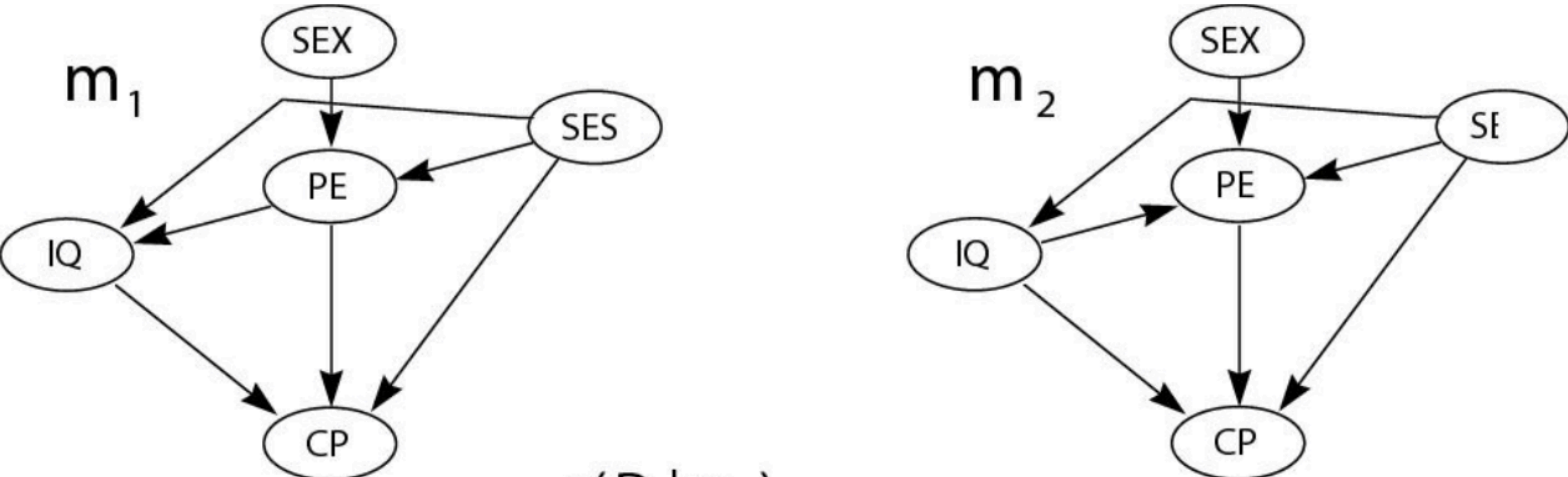
x_1	x_2	x_3	
true	false	true	
false	false	true	
false	false	false	...
true	true	false	
	⋮		⋱

improved network(s)



Example: College Plans Data (Heckerman et. Al 1997)

- Data on 5 variables that might influence high school students’ decision to attend college:
  - **Sex:** Male or Female
  - **SES:** Socio economic status (low, lower-middle, middle, upper-middle, high)
  - **IQ:** discritized into low, lower middle, upper middle, high
  - **PE:** Parental Encouragement (low or high)
  - **CP:** College plans (yes or no)
- 128 possible joint configurations
- Heckerman et. al. computed the exact posterior over all 29,281 possible 5 node DAGs
  - Except those in which Sex or SAS have parents and/or CP have children (prior knowledge)



$$\frac{p(D \mid m_1)}{p(D \mid m_2)} \cong 8.3 \bullet 10^9$$

Bayes Nets – What You Should Know

- Representation
  - Bayes nets represent joint distribution as a DAG + Conditional Distributions
  - D-separation lets us decode conditional independence assumptions
- Inference
  - NP-hard in general
  - For some graphs, some queries, exact inference is tractable
  - Approximate methods too, e.g., Monte Carlo methods, …
- Learning
  - Easy for known graph, fully observed data (MLE's, MAP est.)
  - EM for partly observed data, known graph