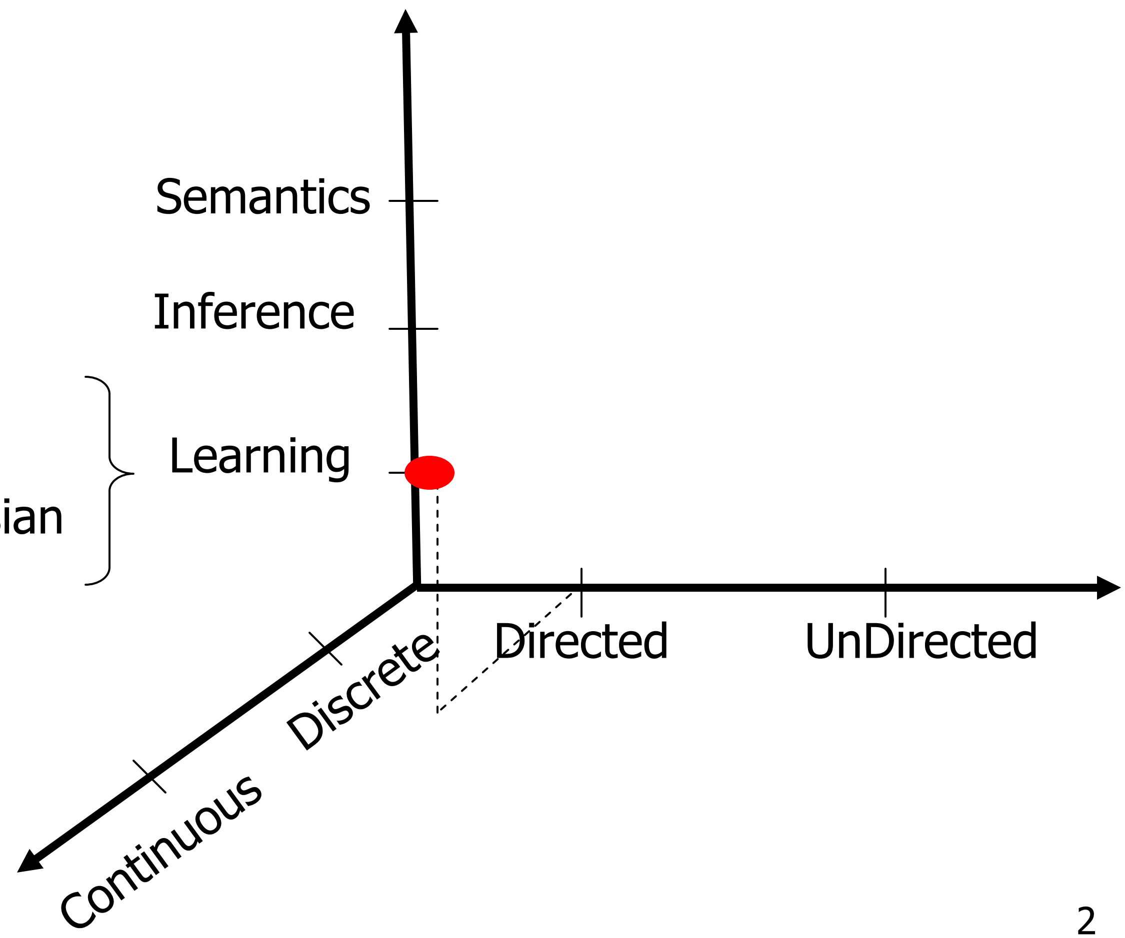
Learning Bayes Net Structures

KF, Chapter 17

Some material taken from C Guesterin (CMU), K Murphy (UBC)

Space of Topics

Learning…
- Parameter, Structure
- Framework: Frequentist, Bayesian
- Data: Complete, Missing

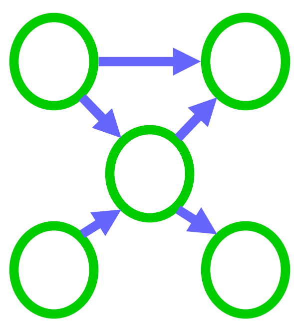


Learning Bayes Nets

Data		Structure: Known	Structure: Unknown
	Complete	Easy	NP-hard
	Missing	Hard ... EM	Very hard!!

Data
$\mathbf{x}^{(1)}$
...
$\mathbf{x}^{(m)}$

$\Rightarrow$



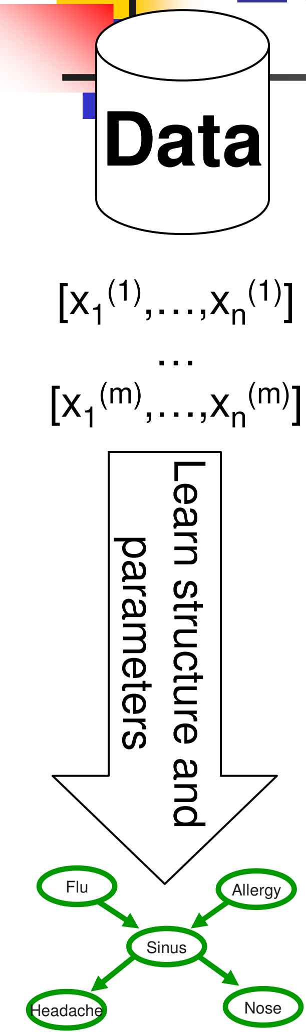
structure

$+$

CPTs :
$P(X_i | \mathbf{Pa}_{X_i})$

parameters

Learning the structure of a BN



- **Constraint-based approach**
  - BN encodes conditional independencies
  - Test conditional independencies in data
  - Find an I-map (?P-map?)
- **Score-based approach**
  - Finding structure + parameters is *density estimation*
  - Evaluate *model* as we evaluated *parameters*
    - Maximum likelihood
    - Bayesian
    - etc.

Outline

- Constraint-based
  - Learn_PDAG
- Score Based (Frequentist)
- Score Based (Bayesian)

Remember: Obtaining a P-map?

- Given $\mathcal{I}(P) = \{ (\mathbf{X},\mathbf{Y}; \mathbf{Z}) : P(\mathbf{X},\mathbf{Y}|\mathbf{Z}) = P(\mathbf{X}|\mathbf{Z})\ P(\mathbf{Y}|\mathbf{Z}) \}$
  = independence assertions that are true for P
  1. Obtain skeleton
  2. Obtain immoralities
  3. Using skeleton and immoralities,
     obtain every (and only) BN structures from the equivalence class

- **Constraint-based approach**:
  - Use Learn_PDAG algorithm
  - Key question: **Independence test**

Independence tests

- Statistically difficult task!
- Intuitive approach: **Mutual information**

$$I(X,Y) = \sum_{x,y} P(x,y) \log \frac{P(x,y)}{P(x)P(y)}$$

- Mutual information and independence:
  - X and Y independent if and only if I(X,Y)=0
  - $X \perp Y \Rightarrow P(x,y) = P(x)\,P(y) \Rightarrow \log[\,P(x,y)/P(x)P(y)\,] = 0$

- **Conditional mutual information:**

$$I(X,Y|Z) = E_Z[\,I[X,Y|Z=z] = \sum_{z} \sum_{x,y} P(x,y|z) \log \frac{P(x,y|z)}{P(x|z)P(y|z)}$$

$X \perp Y \mid Z$ iff $P(X,Y|Z) = P(X|Z)\ P(Y|Z)$ iff $I(X,Y|Z) = 0$

Independence tests and the Constraint-based approach

- Using the data *D*
  - Empirical distribution: $\hat{P}(x_i, x_j) = \frac{\text{Count}(x_i, x_j)}{m}$
  - Mutual information: $\hat{I}(X_i, X_j) = \sum_{x_i, x_j} \hat{P}(x_i, x_j) \log \frac{\hat{P}(x_i, x_j)}{\hat{P}(x_i)\hat{P}(x_j)}$
  - Similarly for conditional MI
- Use Learn_PDAG algorithm:
  When algorithm asks: $(X \perp Y | \mathbf{U})$?
  - Use $I(X, Y \mid \mathbf{U}) = 0$?
    - No… doesn't happen
  - Use $I(X, Y \mid \mathbf{U}) < t$ for some t>0 ?
    - … based on some statistical text "t s.t. $p<0.05$"
- Many other types of independence tests …

Independence Tests – II

- For discrete data: χ^2 statistic
  - measures how far the counts are,
    from expectation given independence:

$$d_{\chi^2}(D) = \sum_{x,y} \frac{(O_{x,y} - E_{x,y})^2}{E_{x,y}} = \sum_{x,y} \frac{(N(x,y) - NP(x)P(y))^2}{NP(x)P(y)}$$

- *p-value* requires averaging over all datasets of size N:

$$p(t) = P(\{D : d(D) > t\} \mid H_0, N)$$

- Expensive… $\Rightarrow$ approximation
  - consider the expected distribution of d(D)
    (under the null hypothesis)
    as $N \to \infty$
  - … to define thresholds for a given significance

Ex of classical hypothesis testing

- Spin Belgian one-euro coin
  - N = 250… heads Y = 140; tails 110.
- Distinguish two models,
  - H_0 = coin is unbiased: so p = 0.5)
  - H_1 = coin is biased: $p \neq 0.5$
- p-value is "less than 7%"
  - $p = P(Y \geq 140) + P(Y \leq 110) = 0.066$:

  n=250; p = 0.5; y = 140;

  p = (1-binocdf(y-1,n,p)) + binocdf(n-y,n,p)
- If Y = 141: p = 0.0497
  $\Rightarrow$ reject the null hypothesis at significance level 0.05.
- But is the coin really biased?

Build-PDAG Algorithm

Also called IC or PC algorithm

Build-PDAG can recover the true structure

- up to I-equivalence

in $O(N^3 2^d)$ time

if

- maximum number of parents over nodes is d
- independence test oracle can handle $\leq$ 2d + 2 variables
- $\exists$ G = a $\mathcal{I}$-map of P
  - underlying distribution P is *faithful* to G
  - $\neg\exists$ spurious independencies not sanctioned by G

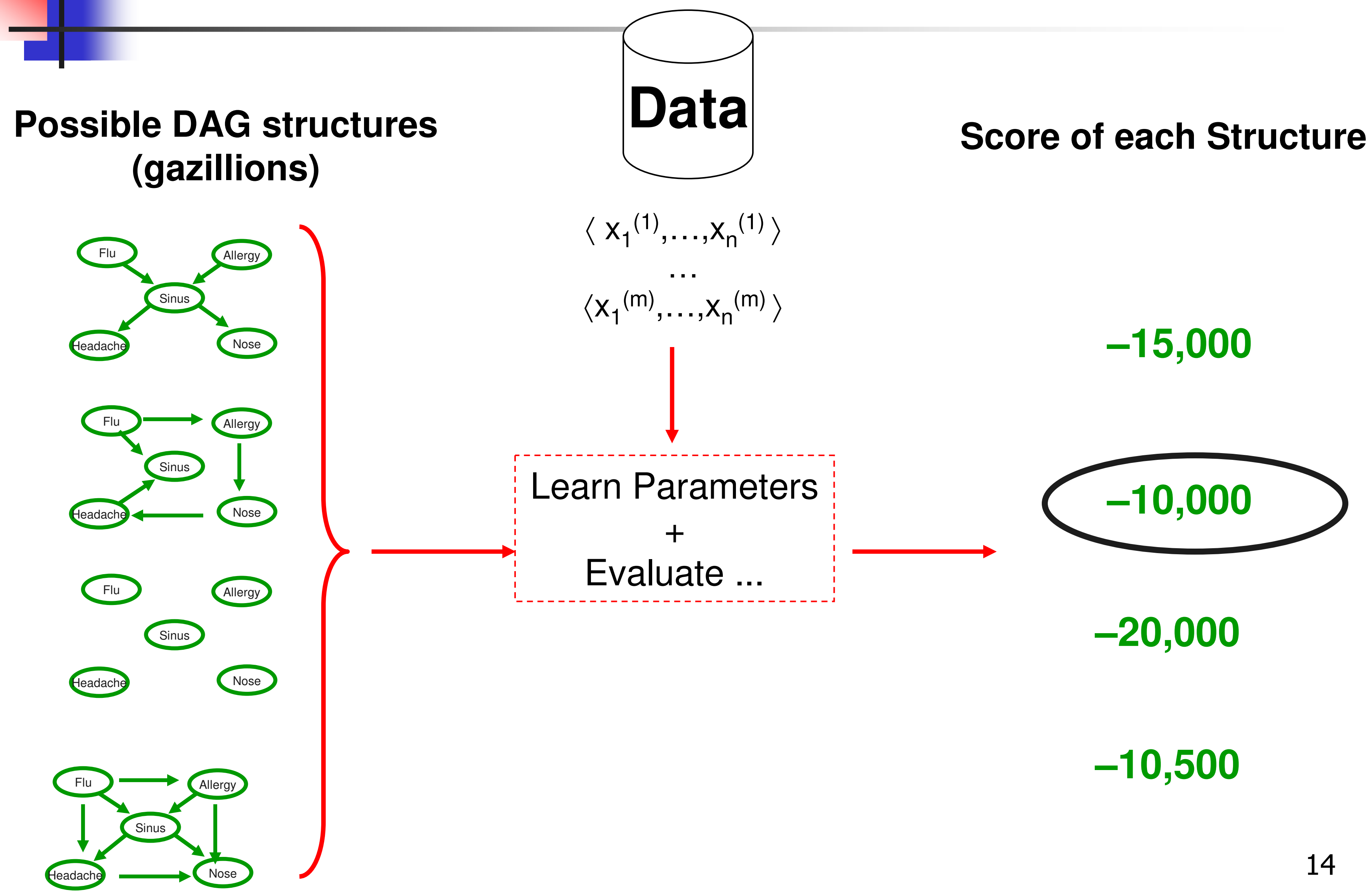
Eval of IC / PC alg

- Good
  - PC algorithm is less dumb than local search
- Bad
  - Faithfulness assumption rules out certain CPDs
    - (noisy) XOR
  - Independence test typically unreliable
    - … especially given small data sets
    - make many errors
  - One misleading independence test result can result in multiple errors in the resulting PDAG
    $\Rightarrow$ overall the approach is not robust to noise

Outline

- Constraint-based
- Score Based (Frequentist)
  - Use MLE parameters
  - Best parents are very informative
  - Best Tree Structure
  - Overfitting
- Score Based (Bayesian)

Score-based Approach

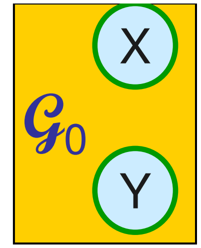


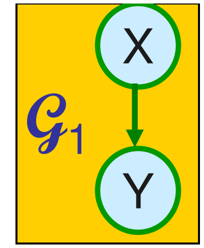
Just use MLE parameters

- $\max_{\mathcal{G},\, \theta_{\mathcal{G}}} L(\, \langle \mathcal{G}, \theta_{\mathcal{G}} \rangle : \mathcal{D}\,) =$
  $\max_{\mathcal{G}} \boxed{\max_{\theta_{\mathcal{G}}} L(\, \langle \mathcal{G}, \theta_{\mathcal{G}} \rangle : \mathcal{D}\,)} =$
  $\max_{\mathcal{G}} L(\, \langle \mathcal{G}, \boxed{\theta^*_{\mathcal{G}}} \rangle : \mathcal{D}\,)$
- So…
  seek the structure $\mathcal{G}$ that achieves highest likelihood,
  given its MLE parameters $\theta^*_{\mathcal{G}}$

- $\text{Score}(\mathcal{G}, \mathcal{D}) = \log L(\, \langle \mathcal{G}, \theta^*_{\mathcal{G}} \rangle : \mathcal{D}\,)$

Comparing Models





- $\mathcal{D} = \{\langle x[1],y[1] \rangle, \ldots, \langle x[M],y[M] \rangle\}$

- $\text{Score}(\mathcal{G}_0, \mathcal{D}) = \sum_m \log \theta^*_{x[m]} + \log \theta^*_{y[m]}$
- $\text{Score}(\mathcal{G}_1, \mathcal{D}) = \sum_m \log \theta^*_{x[m]} + \log \theta^*_{y[m] \mid x[m]}$

- $\text{Score}(\mathcal{G}_1, \mathcal{D}) - \text{Score}(\mathcal{G}_0, \mathcal{D})$

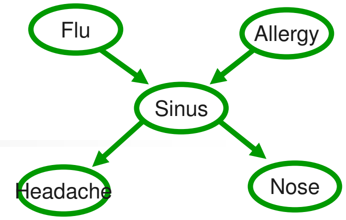
  $= \sum_{x,y} M[x,y] \log \theta^*_{y[m]} - \sum_y M[y] \log \theta^*_{y[m]}$

  $= M \sum_{x,y} p^*(x,y) \log[p^*(y|x) / p(y)$

  $= M \; I_{p^*}(X,Y)$

- $I_{p^*}(X,Y)$ = mutual information between X and Y in P^*
- … higher mutual info $\Rightarrow$ stronger $X \rightarrow Y$ dependency

Information-theoretic interpretation of maximum likelihood



- Given structure $\mathcal{G}$, parameters $\theta_{\mathcal{G}}$, log likelihood of data $\mathcal{D}$:

$$
\begin{aligned}
\log P(\mathcal{D} \mid \theta_{\mathcal{G}}, \mathcal{G}) &= \sum_{j=1}^{m} \sum_{i=1}^{n} \log P\left(X_i = x_i^{(j)} \mid \mathbf{Pa}_{X_i} = \mathbf{x}^{(j)}\left[\mathbf{Pa}_{X_i}\right]\right) \\
&= \sum_{i=1}^{n} \sum_{j=1}^{m} \log P\left(X_i = x_i^{(j)} \mid \mathbf{Pa}_{X_i} = \mathbf{x}^{(j)}\left[\mathbf{Pa}_{X_i}\right]\right)
\end{aligned}
$$

N_{ijk} θ_{ijk}

$$
= \sum_{i=1}^{n} \sum_{x_i, \mathbf{u}} \#(X_i = x_i, \mathbf{Pa}_{X_i} = u) \log P\left(X_i = x_i \mid \mathbf{Pa}_{X_i} = \mathbf{u}\right)
$$

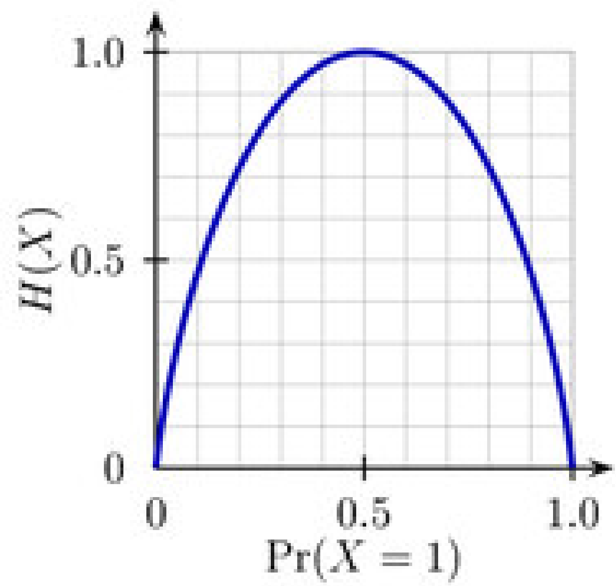
$$
= m \sum_{i=1}^{n} \sum_{x_i, \mathbf{u}} \frac{\#(X_i = x_i, \ \mathbf{Pa}_{X_i} = u)}{m} \log P\left(X_i = x_i \mid \mathbf{Pa}_{X_i} = \mathbf{u}\right)
$$

$$
\boxed{\widehat{P}(X_i = x_i, \mathbf{Pa}_{X_i} = u)}
$$

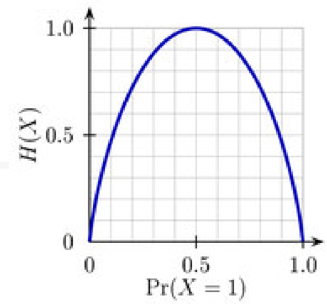
$$
= m \sum_{i=1}^{n} \sum_{x_i, \mathbf{u}} \hat{P}(X_i = x_i, \ \mathbf{Pa}_{X_i} = \mathbf{u}) \log P\left(X_i = x_i \mid \mathbf{Pa}_{X_i} = \mathbf{u}\right)
$$

Entropy

- Entropy of $V = [p(V = 1), p(V = 0)]$:
  $H(V) = -\sum_{v_i} P(V = v_i) \log_2 P(V = v_i)$
  ≡ # of bits needed to obtain full info
  ...average surprise of result of one "trial" of V
- Entropy ≈ measure of uncertainty



Examples of Entropy



- Fair coin:
  - $H(½, ½) = -½ \log_2(½) - ½ \log_2(½) = 1$ bit
  - ie, need 1 bit to convey the outcome of coin flip)

- Biased coin:

  $H(1/100, 99/100) =$
  $-1/100 \log_2(1/100) - 99/100 \log_2(99/100) = 0.08$ bit

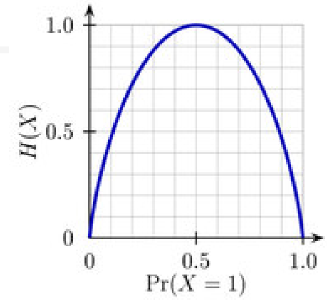
- As P(heads) $\mapsto$ 1, info of actual outcome $\mapsto$ 0

  $H(0, 1) = H(1, 0) = 0$ bits

  ie, no uncertainty left in source

  $(0 \times \log_2(0) = 0)$

Entropy & Conditional Entropy



- Entropy of Distribution
  - $H(X) = -\sum_i P(x_i) \log P(x_i)$
  - "How `surprising' variable is"
  - Entropy = 0 when know everything… eg P(+x)=1.0
- Conditional Entropy $H(X \mid \mathbf{U})$ …
  - $H(X|\mathbf{U}) = -\sum_{\mathbf{u}} P(\mathbf{u}) \sum_i P(x_i|\mathbf{u}) \log P(x_i|\mathbf{u})$
  - How much uncertainty is left in X, after observing **U**

$$H(X_i \mid \mathbf{Pa}_{X_i}) = -\sum_{x_i,\mathbf{u}} \hat{P}(X_i = x_i, \mathrm{Pa}_{X_i} = \mathrm{u}) \log P\left(X_i = x_i^{(j)} \mid \mathbf{Pa}_{X_i} = \mathrm{u}\right)$$

Information-theoretic interpretation of maximum likelihood … 2

- Given structure $\mathcal{G}$, parameters $\theta_{\mathcal{G}}$, log likelihood of data $\mathcal{D}$ is…

$$\log \hat{P}(\mathcal{D} \mid \theta, \mathcal{G}) = m \sum_i \sum_{x_i, \mathbf{u}} \hat{P}(x_i, \mathbf{Pa}_{x_i,\mathcal{G}} = \mathbf{u}) \log \hat{P}(x_i \mid \mathbf{Pa}_{x_i,\mathcal{G}} = \mathbf{u})$$

$$= m \sum_i -\hat{H}(X_i | \mathbf{Pa}_{x_i,\mathcal{G}})$$

$$= -m \sum_i \hat{H}(X_i | \mathbf{Pa}_{x_i,\mathcal{G}})$$

So $\log P(\mathcal{D} | \theta, \mathcal{G})$ is LARGEST

when each $H(X_i \mid Pa_{X_i, \mathcal{G}})$ is SMALL…

…ie, when parents of X_i are very INFORMATIVE about X_i !

Score for Bayesian Network

- $I(X, \mathbf{U}) = H(X) - H(X \mid \mathbf{U})$
  $\Rightarrow \; H(X \mid Pa_{X,\mathcal{G}}) = H(X) - \mathcal{I}(X, Pa_{X,\mathcal{G}})$

Doesn't involve the structure, $\mathcal{G}$!

- Log data likelihood

$$\log \hat{P}(\mathcal{D} \mid \theta, \mathcal{G}) = m \sum_i \hat{I}(X_i, \mathbf{Pa}_{X_i,\mathcal{G}}) - m \sum_i \hat{H}(X_i)$$

  - $\neg(X \perp Pa_X)$ … not very independent ☺

- So use score: $\sum_i I(X_i, Pa_{Xi,\mathcal{G}})$

Decomposable Score

- Log data likelihood

$$\log \widehat{P}(\mathcal{D} \mid \theta, \mathcal{G}) = m \sum_i \widehat{I}(X_i, \mathbf{Pa}_{X_i,\mathcal{G}}) - m \sum_i \widehat{H}(X_i)$$

- … or perhaps just score: $\sum_i I(X_i, Pa_{Xi,\, \mathcal{G}})$

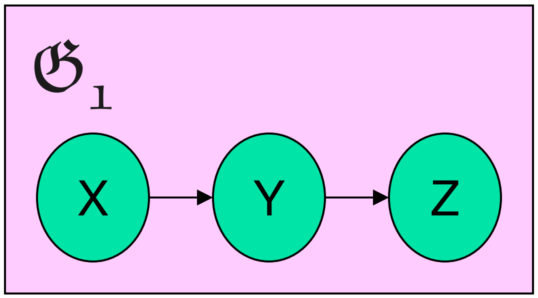
- Decomposable score:
  - Decomposes over families in BN (node and its parents)
  - Will lead to significant computational efficiency!!!
  - Score($G : D$) = $\sum_i$ FamScore(X_i | $\mathbf{Pa}_{Xi}$: D)
  - For MLE: FamScore(X_i | $\mathbf{Pa}_{Xi}$: D) = m[I(X_i, Pa_{xi}) – H(X_i)]

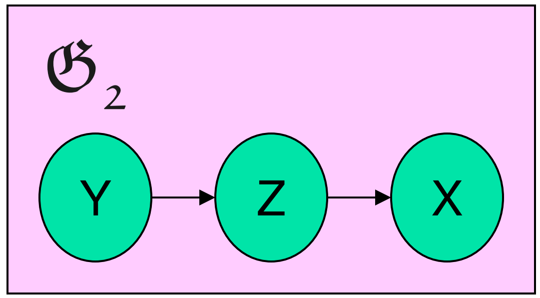
Using DeComposability

$$\log \widehat{P}(\mathcal{D} \mid \theta, \mathcal{G}) = m \sum_i \hat{I}(x_i, \mathbf{Pa}_{x_i,\mathcal{G}}) - m \sum_i \hat{H}(X_i)$$

$$\mapsto \sum_i I(X_i, Pa_{X_i, \mathfrak{G}}) + c$$

- Compare

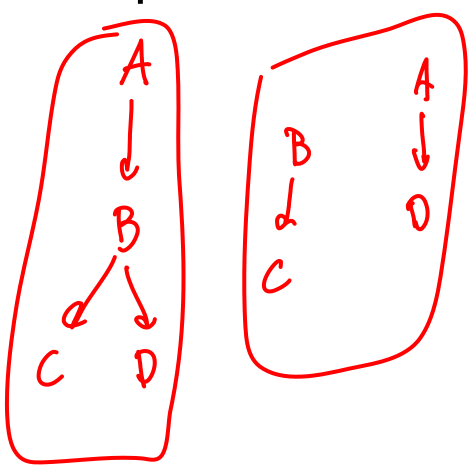


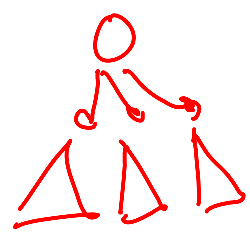


- $\mathfrak{G}_1$: $\sum_i I(X_i, Pa_{X_i, \mathfrak{G}_1}) = \underbrace{I(X, \{\})}_{0} + I(Y, X) + I(Z, Y)$
  $= I(Y, X) + I(Z, Y)$
- $\mathfrak{G}_2$: $\sum_i I(X_i, Pa_{X_i, \mathfrak{G}_2}) = \underbrace{I(Y, \{\})}_{0} + I(Z,Y) + I(X, Z)$
  $= I(Z,Y) + I(X, Z)$
- … so diff is $I(Y, X) - I(X, Z)$

How many trees are there?

- Tree:
  - ∃ one path between any two nodes (in skeleton)
  - Most nodes have 1 parent (+ root with 0 parents)
- How many:
  - One: pick root
  - pick children … for each child … another tree





$\sim 2^{\Theta(n \lg n)}$

Nonetheless… ∃ efficient optimal alg to find OPTIMAL tree

Best Tree Structure

$$\log \widehat{P}(\mathcal{D} \mid \theta, \mathcal{G}) = m \sum_i \widehat{I}(x_i, \mathbf{Pa}_{x_i,\mathcal{G}}) - m \sum_i \widehat{H}(X_i)$$

- Identify tree with set $\mathfrak{F}$ = { Pa(X) }
  - each Pa(X) is {}, or another variable
- Optimal tree, given data, is

  $\mathrm{argmax}_{\mathfrak{F}}\ m \sum_i I(X_i, Pa(X_i)) - m \sum_i H(X_i)$

  $= \mathrm{argmax}_{\mathfrak{F}} \sum_i I(X_i, Pa(X_i))$
  - … as $\sum_i H(X_i)$ does not depend on structure
- So … want parents $\mathfrak{F}$ s.t.
  - tree structure
  - maximizes $\sum_i I(X_i, Pa(X_i))$

Chow-Liu Tree Learning Alg

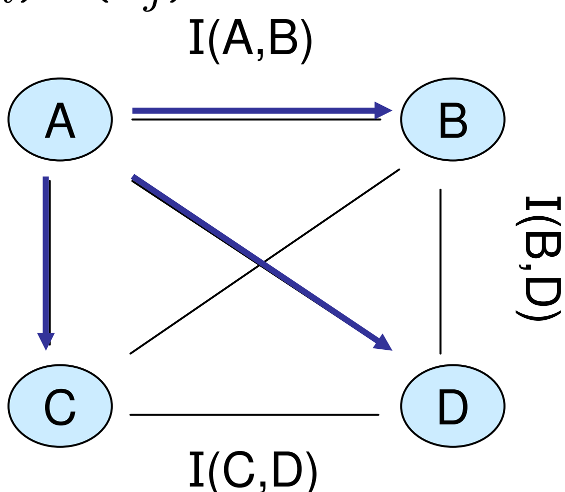
- For each pair of variables X_i, X_j
  - Compute empirical distribution:

$$\hat{P}(x_i, x_j) = \frac{\text{Count}(x_i, x_j)}{m}$$

  - Compute mutual information:

$$\hat{I}(X_i, X_j) = \sum_{x_i, x_j} \hat{P}(x_i, x_j) \log \frac{\hat{P}(x_i, x_j)}{\hat{P}(x_i)\hat{P}(x_j)}$$

- Define a graph
  - Nodes $X_1, \ldots, X_n$
  - Edge (i,j) gets weight $\hat{I}(X_i, X_j)$
- Find Maximal Spanning Tree
- Pick a node for root, dangle…

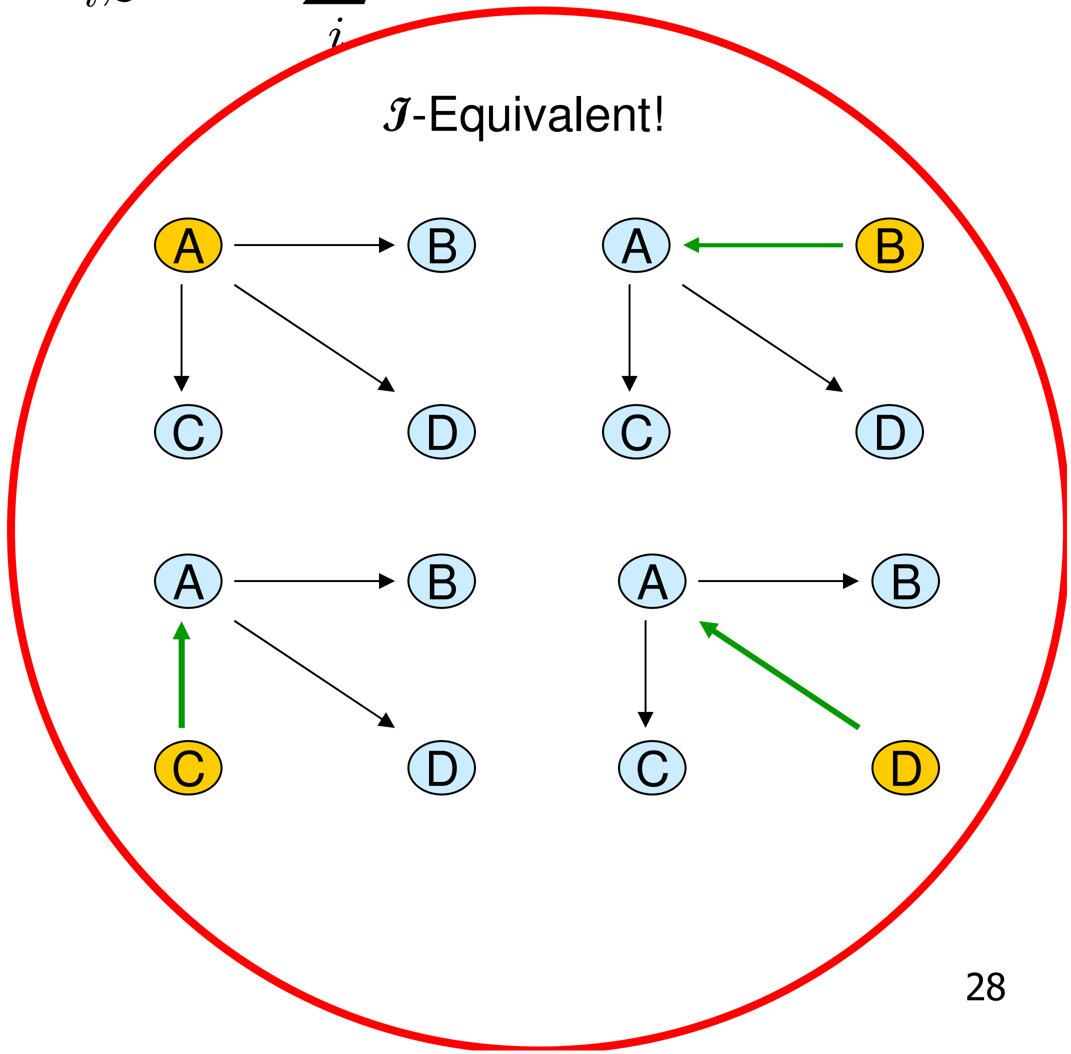


Chow-Liu Tree Learning Alg … 2

$$\log \hat{P}(\mathcal{D} \mid \theta, \mathcal{G}) = m \sum_i \hat{I}(x_i, \mathbf{Pa}_{x_i,\mathcal{G}}) - m \sum_i \hat{H}(X_i)$$

- Optimal tree BN
  - …
  - Compute maximum weight spanning tree
  - Directions in BN:
    - pick any node as root, …doesn't matter which!
    - breadth-first-search defines directions
- Score Equivalence:

  If $\mathcal{G}$ and $\mathcal{G}'$ are $\mathcal{I}$-equiv, then scores are same

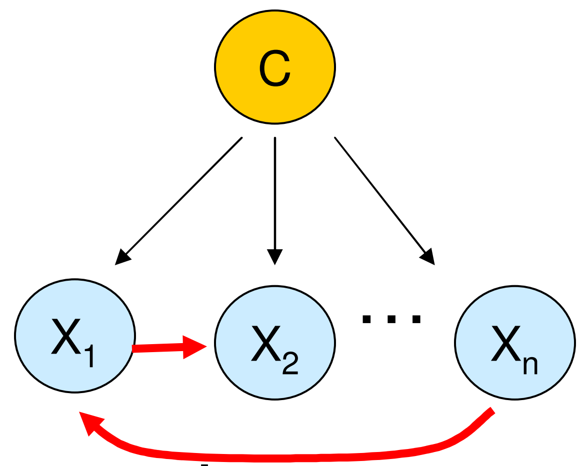


Chow-Liu (CL) Results

- If distribution P is tree-structured,
  CL finds CORRECT one
- If distribution P is NOT tree-structured,
  CL finds tree structured Q that
  has min'l KL-divergence – $\arg\min_Q KL(P; Q)$
- Even though $2^{\theta(n \log n)}$ trees,
  CL finds BEST one in poly time $O(n^2 [m + \log n])$

Extending Chow-Liu… #1

- Naïve Bayes model
  - $X_i \perp X_j \mid C$
  - Ignores correlation between features
  - What if $X_1 = X_2$? **Double count…**



- Avoid by conditioning features on one another

- Tree Augmented Naïve bayes (TAN)
  [Friedman et al. '97]

$$\hat{I}(X_i, X_j \mid C) = \sum_{c,x_i,x_j} \hat{P}(c, x_i, x_j) \log \frac{\hat{P}(x_i, x_j \mid c)}{\hat{P}(x_i \mid c)\hat{P}(x_j \mid c)}$$

All but ONE feature have 2 parents: C, X_i

Extending Chow-Liu… #2

- (Approximately learning) models with tree-width up to *k*
  - [Narasimhan & Bilmes '04]
  - But, $O(n^{k+1})$…
    - and more subtleties

Learning BN structures… so far

- Decomposable scores
  - Maximum likelihood
  - Information theoretic interpretation
- Best tree (Chow-Liu)
- Best TAN
- Nearly best k-treewidth (in $O(N^{k+1})$)
- … all frequentist…

Maximum likelihood score overfits!

$$\log \hat{P}(\mathcal{D} \mid \theta, \mathcal{G}) = m \sum_i \hat{I}(X_i, \mathbf{Pa}_{X_i,\mathcal{G}}) - m \sum_i \hat{H}(X_i)$$

- Adding a parent never decreases score!!!
  - ***Facts:*** $H(X \mid Pa_{X,\mathcal{G}}) = H(X) - I(X, Pa_{X,\mathcal{G}})$

    $H(X \mid A) \geq H(X \mid A \cup Y)$
  - $I(X_i, Pa_{Xi,\mathcal{G}} \cup Y) = H(X_i) - H(X_i \mid Pa_{Xi,\mathcal{G}} \cup Y)$

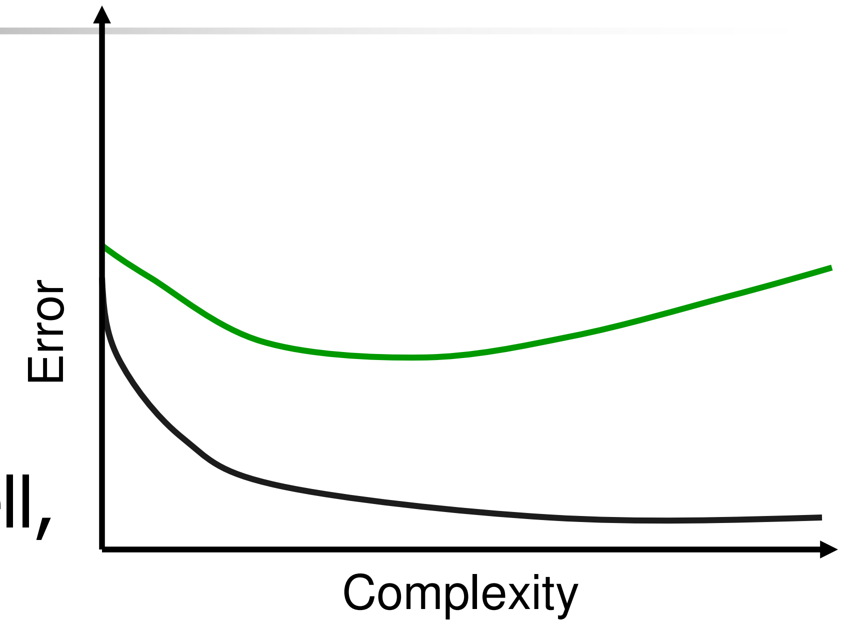
    $\geq H(X_i) - H(X_i \mid Pa_{Xi,\mathcal{G}})$

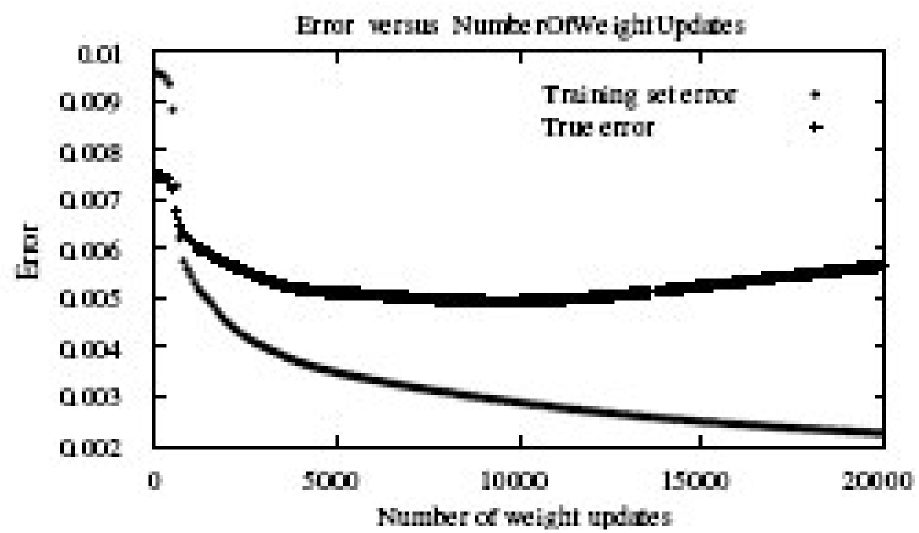
    $= I(X_i, Pa_{Xi,\mathcal{G}})$
- So score increases as we add edges!
  - Best is COMPLETE Graph
  - … overfit !

Overfitting

- So far:
  Find parameters/structure
  that "fit" the training data
- If too many parameters,
  will match TRAINING data well,
  but NOT new instances
- Overfitting!

- Regularizing,
  Bayesian approach, …





Outline

- Constraint-based
- Score Based (Frequentist)
- Score Based (Bayesian)
  - Marginal posterior
  - BIC approx’n
  - Consistency
  - BDE Priors
  - Learning General DAGs
  - Model Averaging

Bayesian Score

- Prior distributions:
  - Over structures
  - Over parameters of a structure

  Goal: Prefer simpler structures... regularization ...
- Posterior over structures given data:
  - $\underbrace{\mathrm{P}(\mathcal{G}|\mathcal{D})}_{\text{Posterior}} \propto \underbrace{\mathrm{P}(\mathcal{D}|\mathcal{G})}_{\text{Likelihood}} \times \underbrace{\mathrm{P}(\mathcal{G})}_{\text{Prior over Graphs}}$

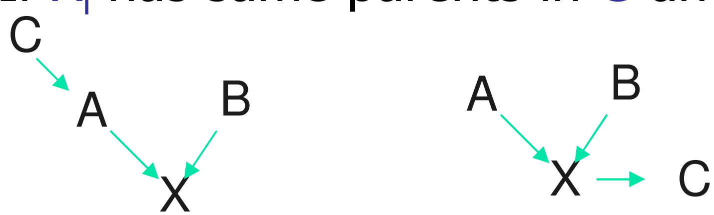
    Prior over Parameters
  - $\mathrm{P}(\mathcal{D}|\mathcal{G}) = \int_{\Theta} \mathrm{P}(\mathcal{D} \mid \mathcal{G}, \Theta)\, \mathrm{P}(\Theta|\mathcal{G})\, d\Theta$

$$\log P(\mathcal{G} \mid D) \approx \log P(\mathcal{G}) + \log \int_{\theta_{\mathcal{G}}} P(D \mid \mathcal{G}, \theta_{\mathcal{G}}) P(\theta_{\mathcal{G}}|\mathcal{G}) d\theta_{\mathcal{G}}$$

Towards a decomposable Bayesian score

$$\log P(\mathcal{G} \mid D) \approx \log P(\mathcal{G}) + \log \int_{\theta_{\mathcal{G}}} P(D \mid \mathcal{G}, \theta_{\mathcal{G}}) P(\theta_{\mathcal{G}} | \mathcal{G}) d\theta_{\mathcal{G}}$$

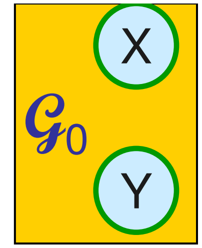
- **Local and global parameter independence** $\theta_{Y|+x} \perp \theta_X$
- Prior satisfies **parameter modularity**:
  - If X_i has same parents in G and G′, then parameters have same prior



$\Theta(X; A,B)$ same in both structures

- Structure prior $P(\mathcal{G})$ satisfies **structure modularity**
  - Product of terms over families
  - Eg, $P(\mathcal{G}) \propto c^{|\mathcal{G}|}$ $|\mathcal{G}|$=#edges; c<1
- … then … Bayesian score decomposes along families!
  - $\log P(\mathcal{G}|\mathcal{D}) = \sum_X \text{ScoreFam}(X \mid Pa_X : \mathcal{D})$

Factoring Marginal



$$P(\mathcal{D}|\mathcal{G}_0) = \int P(\mathcal{D}, \theta_X, \theta_Y|\mathcal{G}_0)\, P(\theta_X, \theta_Y \mid \mathcal{G}_0)\, d\theta_X\, d\theta_Y$$

$$= \int P(x[1], \ldots, x[M], y[1], \ldots, y[M], \theta_X, \theta_Y|\mathcal{G}_0)\, P(\theta_X, \theta_Y \mid \mathcal{G}_0)\, d\theta_X\, d\theta_Y$$

$$= \int P(x[1], \ldots, x[M] \mid \cancel{y[1], \ldots, y[M]}, \theta_X, \cancel{\theta_Y, \mathcal{G}_0}) \times P(y[1], \ldots, y[M] \mid \cancel{\theta_X}, \theta_Y, \cancel{\mathcal{G}_0})\, P(\theta_X \mid \cancel{\theta_Y}, \mathcal{G}_0)\, P(\theta_Y \mid \mathcal{G}_0)\, d\theta_X\, d\theta_Y$$

- As $x[i] \perp y[j]$, $x[i] \perp \theta_Y$, $x[i] \perp \mathcal{G}_0 \mid \theta_X$, $y[j] \perp \mathcal{G}_0 \mid \theta_Y$, $\theta_X \perp \theta_Y \mid \mathcal{G}_0$

$$P(\mathcal{D}|\mathcal{G}_0) =$$

$$\int \prod_m P(x[m] \mid \theta_X, x[1:m-1])\, \prod_m P(y[m] \mid \theta_y, y[1:m-1])\, P(\theta_X \mid \mathcal{G}_0)\, P(\theta_Y \mid \mathcal{G}_0)\, d\theta_X\, d\theta_Y$$

$$= \int P(\theta_X \mid \mathcal{G}_0) \prod_m P(x[m] \mid \theta_X, x[1:m-1])\, d\theta_X$$

$$\int P(\theta_y \mid \mathcal{G}_0) \prod_m P(y[m] \mid \theta_y, y[1:m-1])\, d\theta_y$$

Marginal Posterior

- Given $\theta \sim \text{Beta}(1,1)$,
  what is probability of $\langle H, T, T, H, H \rangle$?
- $P(f_1{=}H, f_2{=}T, f_3{=}T, f_4{=}H, f_5{=}H \mid \theta \sim \text{Beta}(1,1))$

$$= P(f_1{=}H \mid \theta \sim \text{Beta}(1,1)) \times$$
$$P(f_2{=}T, f_3{=}T, f_4{=}H, f_5{=}H \mid f_1{=}H, \theta \sim \text{Beta}(1,1))$$
$$= ½ \times P(f_2{=}T, f_3{=}T, f_4{=}H, f_5{=}H \mid \theta \sim \text{Beta}(2,1))$$
$$= ½ \times P(f_2{=}T \mid \theta \sim \text{Beta}(2,1)) \text{ x}$$
$$P(f_3{=}T, f_4{=}H, f_5{=}H \mid f_2{=}T, \theta \sim \text{Beta}(2,1))$$
$$= ½ \times 1/3 \times P(f_3{=}T, f_4{=}H, f_5{=}H \mid \theta \sim \text{Beta}(2,2))$$
$$= ½ \times 1/3 \times 2/4 \times 2/5 \times P(f_5{=}H \mid \theta \sim \text{Beta}(2,3))$$
$$= ½ \times 1/3 \times 2/4 \times 2/5 \times 3/6$$
$$= \underbrace{(1 \times 2 \times 3)}_{\text{3 heads}} \times \underbrace{(1 \times 2)}_{\text{2 tails}} / \underbrace{(2 \times 3 \times 4 \times 5)}_{\text{5 flips}}$$

Marginal Posterior… con't

- Given $\theta \sim \text{Beta}(a,b)$, what is $P[\langle H, T, T, H, H \rangle]$?
- $P(f_1=H, f_2=T, f_3=T, f_4=H, f_5=H \mid \theta \sim \text{Beta}(a,b))$

  $= P(f_1=H \mid \theta \sim \text{Beta}(a,b)) \times$

  $\quad P(f_2=T, f_3=T, f_4=H, f_5=H \mid f_1=H, \theta \sim \text{Beta}(a,b))$

  $= a/(a+b) \times$

  $\quad P(f_2=T, f_3=T, f_4=H, f_5=H \mid \theta\sim\text{Beta}(a+1,b))$

$$= \frac{a}{a+b}\,\frac{b}{a+b+1}\,\frac{b+1}{a+b+2}\,\frac{a+1}{a+b+3}\,\frac{a+2}{a+b+4}$$

$$= \frac{a\times(a+1)\times(a+2)\ \times\ b\times(b+1)}{(a+b)(a+b+1)(a+b+2)(a+b+3)(a+b+4)}$$

$$= \boxed{\frac{\Gamma(\alpha_H+m_H)}{\Gamma(\alpha_H)}\,\frac{\Gamma(\alpha_T+m_T)}{\Gamma(\alpha_T)}\,\frac{\Gamma(\alpha_H+\alpha_T)}{\Gamma(\alpha_H+\alpha_T+m_H+m_T)}}$$

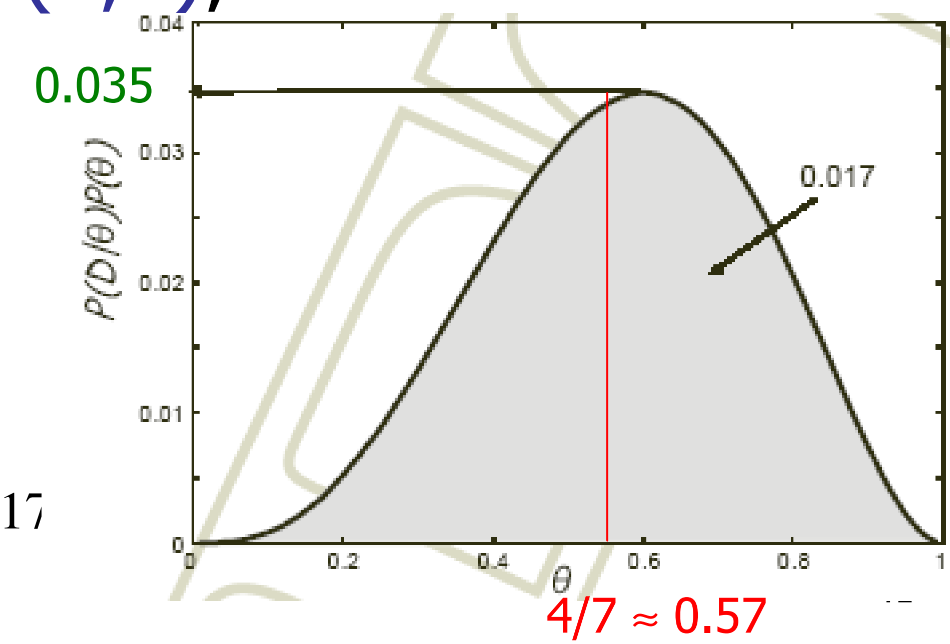
Marginal, vs Maximal, Likelihood

- Data $\mathcal{D} = \langle$ H, T, T, H, H $\rangle$
- MLE: $\theta^* = \text{argmax}_\theta\ P(\mathcal{D} \mid \theta) = 3/5$
  - … Here: $P(\mathcal{D} \mid \theta^*) = (3/5)^3\,(2/5)^2 \approx 0.035$
- Bayesian, …from Beta(1,1),

  $\theta_{B(1,1)\mid\mathcal{D}} \sim \text{Beta}(4, 3)$
  - Expected posterior:

    $E[\,\theta_{B(1,1)\mid\mathcal{D}}\,] = 4/7$
- Marginal

$$P(D \mid \Theta) = \frac{\Gamma(1+3)}{\Gamma(1)}\frac{\Gamma(1+2)}{\Gamma(1)}\frac{\Gamma(1+1)}{\Gamma(1+1+3+2)} \approx 0.017$$



Marginal Probability of Graph

$$\log P(D \mid \mathcal{G}) = \log \int_{\theta_\mathcal{G}} P(D \mid \mathcal{G}, \theta_\mathcal{G}) P(\theta_\mathcal{G} | \mathcal{G}) d\theta_\mathcal{G}$$

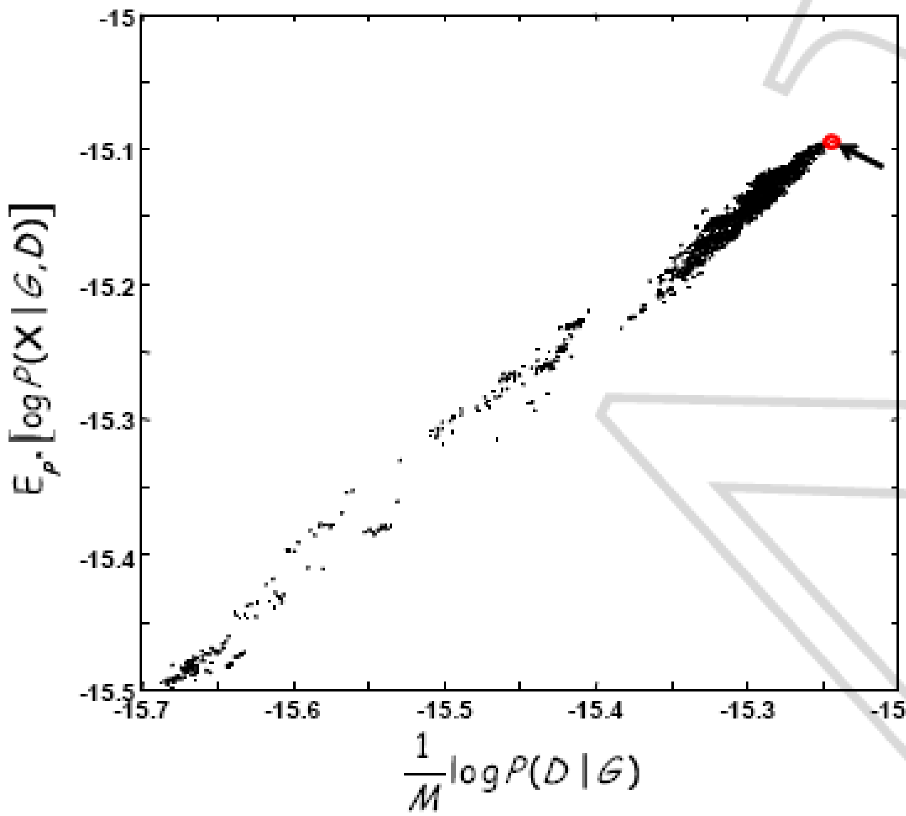
- Given complete data, independent parameters, …

$$\boxed{P(D \mid G) = \prod_i \prod_{u_i \in Val(Pa_{X_i})} \frac{\Gamma(\alpha^G_{X_i|u_i})}{\Gamma(\alpha^G_{X_i|u_i} + M[u_i])} \prod_{x_i^j \in Val(X_i)} \frac{\Gamma(\alpha^G_{x_i^j|u_i} + M[x_i^j, u_i])}{\Gamma(\alpha^G_{x_i^j|u_i})}}$$

Marginal Probability ≈ Validation Set!

- $P(\mathcal{D} \mid \mathcal{G}) = \prod_m P(\xi[m] \mid \xi[1], \ldots, \xi[m-1], \mathcal{G})$
- Each $P(\xi[m] \mid \xi[1], \ldots, \xi[m-1], \mathcal{G})$ is prob of *m^{th} instance* using parameters learned from *first m-1 instances*
- kinda like cross validation:
  Evaluate each instance,
  wrt previous instance
  - Suggests… $\frac{1}{M} \log P(D \mid G) \approx E_{P^*}[\log P(\xi \mid G, D)]$

Average Training Log Likelihood vs Expected Log Likelihood



10000 instances

Approx'n of Bayesian Score

- In general, Bayesian has difficult integrals
- For *Dirichlet prior over parameters*, can use simple Bayes information criterion (BIC) approximation
  - In the limit, we can forget prior!
- **Theorem**: Given Dirichlet priors for a BN with Dim($\mathcal{G}$) independent parameters, as m$\rightarrow\infty$:

$$\log P(D \mid \mathcal{G}) = \underbrace{\log P(D \mid \mathcal{G}, \overset{\text{max likelihood estimate for } \theta}{\widehat{\theta}_{\mathcal{G}}})}_{\text{likelihood score… prefers fully-connected graph}} - \underbrace{\frac{\log m}{2} \text{Dim}(\mathcal{G})}_{\text{regularizer… penalizes edges}} + O(1)$$

BIC approximation

- BIC: $\boxed{\text{Score}_{\text{BIC}}(\mathcal{G} : D) = \log P(D \mid \mathcal{G}, \theta_{\mathcal{G}}) - \frac{\log m}{2}\text{Dim}(\mathcal{G})}$
  - Dim[G] = #parameters
    $= \sum_i \sum_j \text{Dim}[\,\theta_{Xi|Pa\_ij}] = \sum_i (k-1)\, k^{|Pa\_i|}$
  - $|X_i| = k$
  - Scales exponentially with #parents – Bad!
- As m grows, -log m "compensates"
  - … so complex models become ok…
- $\text{Score}_{\text{BIC}}(\mathcal{G} : D) = m \sum_i \underbrace{\hat{I}(X_i, \mathbf{Pa}_{X_i,\mathcal{G}}) - m \sum_i \hat{H}(X_i) - \frac{\log m}{2} \sum_i \text{Dim}(P(X_i \mid \mathbf{Pa}_{X_i,\mathcal{G}}))}_{}$

$\text{ScoreFam}_{\text{BIC}}(\, X_i \mid Pa_{Xi}, \mathcal{D})$
$= m\, I(X_i, Pa_{Xi,G}) - m\, H(X_i) - ½ \log m\ \text{Dim}[\, P(I(X_i, Pa_{Xi,G})\,]$

Consistency of BIC, Bayesian scores

- A scoring function is **consistent** if, for true model $\mathcal{G}^*$, as m$\rightarrow\infty$, with probability 1,
  - $\mathcal{G}^*$ maximizes the score
  - All structures **not $\mathcal{I}$-equivalent** to $\mathcal{G}^*$ have *strictly* lower score

- **Theorem**: BIC score (with Dirichlet prior) is consistent
- **Corollary**: the Bayesian score is consistent
- What about likelihood score?

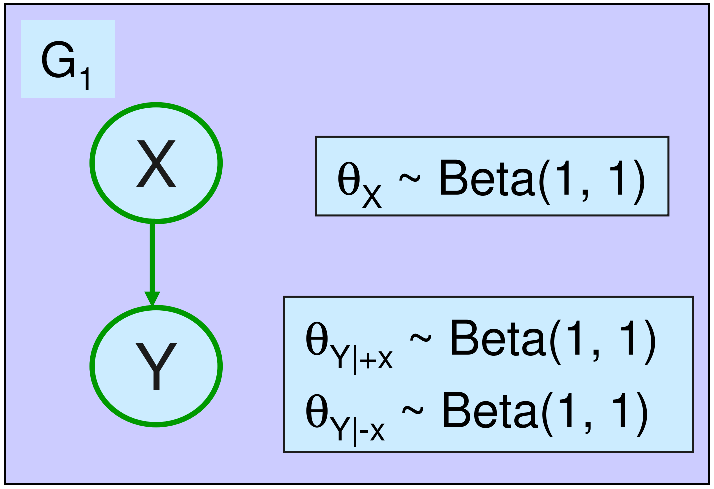
NO! True, Likelihood of optimal is MAX.
But fully-connected graph (which is NOT $\mathcal{I}$-equiv) also max's score!

Consistency is limiting behavior…
says nothing wrt finite sample size!!!

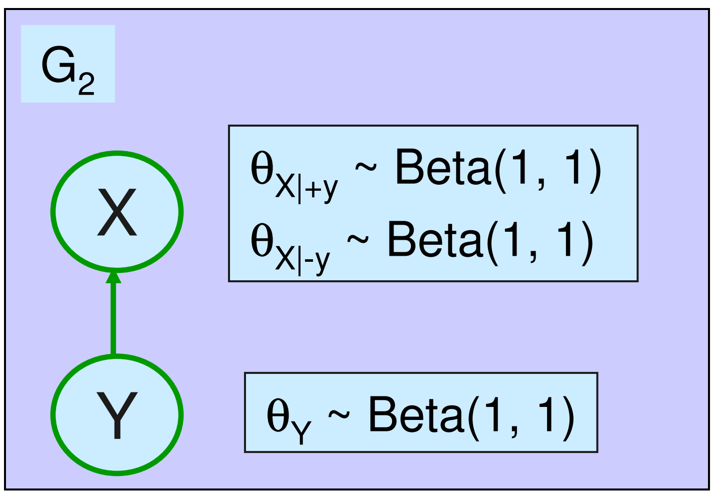
Priors for General Graphs

- For finite datasets, prior is important!
- Prior over structure satisfying prior modularity
  - Eg, $P(\mathcal{G}) \propto c^{|\mathcal{G}|}$ $|\mathcal{G}|$=#edges; $c<1$
- What is good prior over *all* parameters?
  - *K2 prior*: fix $\alpha \in \Re^+$, set $\theta_{X_i|\mathbf{Pa}_{X_i}} \sim \text{Dirichlet}(\alpha, \ldots, \alpha)$
  - Effective sample size, wrt X_i ?
    - If 0 parents: $k \times \alpha$
    - If 1 binary parent: $2\ k \times \alpha$
    - If d k-ary parents: $k^d\ k \times \alpha$
  - So X_i *"effective sample size"* depends on #parental assignments
    - More parents ⇒ strong prior… doesn't make sense!
  - K2 is "inconsistent"

Priors for Parameters

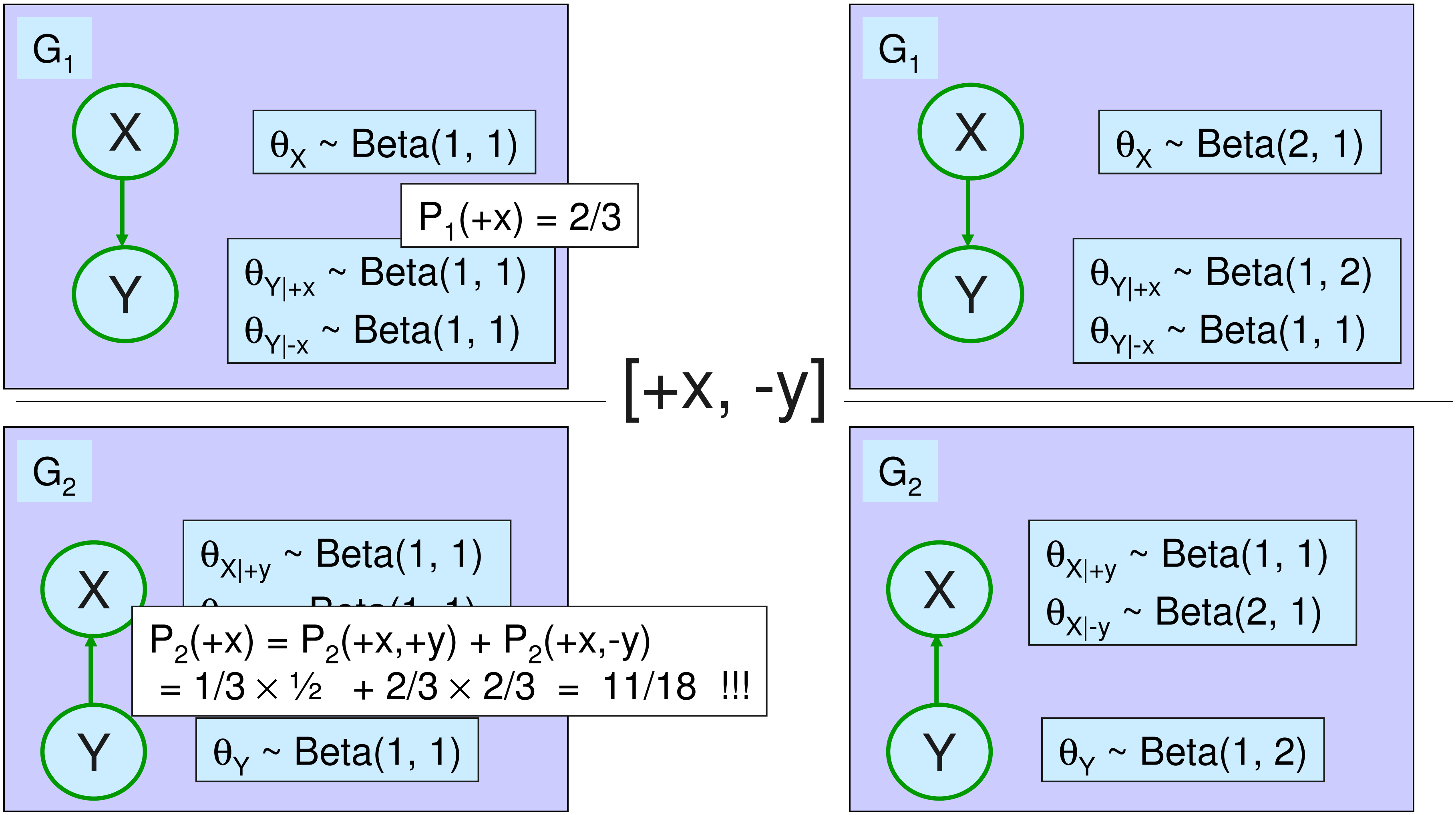


- Does this make sense?
  - EffectiveSampleSize($\theta_{Y|+x}$) = 2
  - But only 1 example ~ "+x" ??

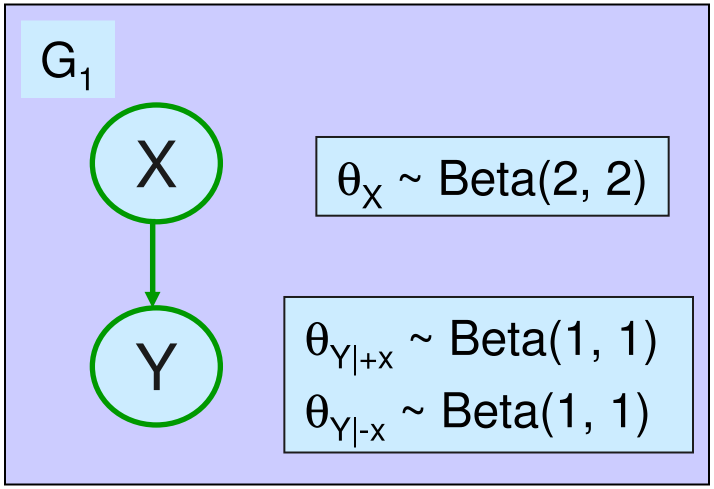


- $\mathcal{I}$-Equivalent structure
- What happens after [+x, -y] ?
  - Should be the same!!

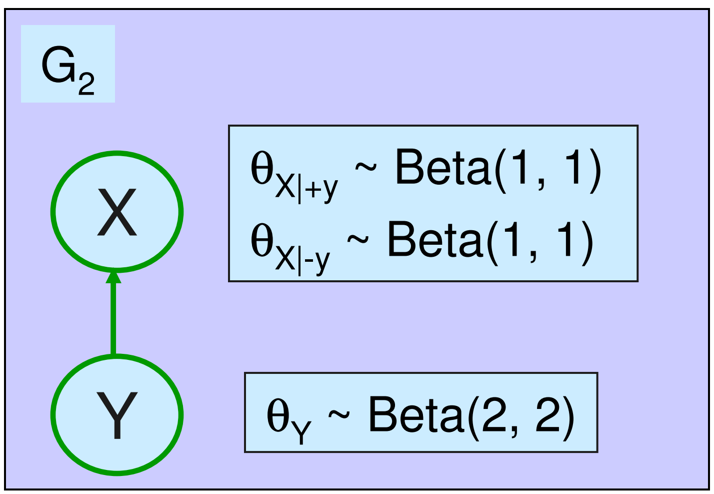
Priors for Parameters



BDe Priors



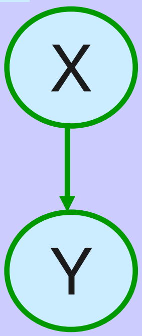
- This makes more sense:
  - EffectiveSampleSize($\theta_{Y|+x}$) = 2
  - Now ≈∃ 2 examples ~ "+x" ??



- I-Equivalent structure
- Now what happens after [+x, -y] ?

BDe Priors

G_1



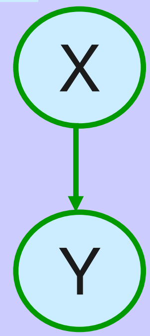
$\theta_X \sim \mathrm{Beta}(2, 2)$

$P_1(+x) = 3/5$

$\theta_{Y|+x} \sim \mathrm{Beta}(1, 1)$

$\theta_{Y|-x} \sim \mathrm{Beta}(1, 1)$

G_1



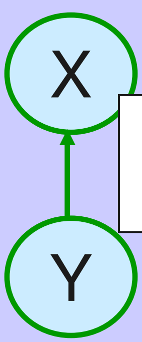
$\theta_X \sim \mathrm{Beta}(3, 2)$

$\theta_{Y|+x} \sim \mathrm{Beta}(1, 2)$

$\theta_{Y|-x} \sim \mathrm{Beta}(1, 1)$

[+x, -y]

G_2

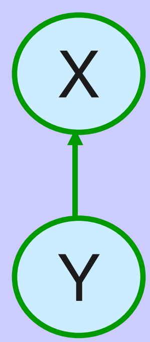


$\theta_{X|+y} \sim \mathrm{Beta}(1, 1)$

$P_2(+x) = P_2(+x,+y) + P_2(+x,-y)$
$= 2/5 \times ½ + 3/5 \times 2/3 = 3/5$!!!

$\theta_Y \sim \mathrm{Beta}(2, 2)$

G_2



$\theta_{X|+y} \sim \mathrm{Beta}(1, 1)$

$\theta_{X|-y} \sim \mathrm{Beta}(2, 1)$

$\theta_Y \sim \mathrm{Beta}(2, 3)$

BDe Prior

- View Dirichlet parameters as "fictitious samples" – equivalent sample size
- Pick a fictitious sample size m'
- For each possible family, define a prior distribution $P(X_i, \mathbf{Pa}_{X_i})$
  - Represent with a BN
  - Usually independent (product of marginals)
    - $P(X_i, Pa_{X_i}) = P'(x_i) \prod_{x_j \in Pa[X_i]} P'(x_j)$
    - $P(\theta[x_i \mid Pa_{X_i} = u]) = Dir(m' P'(x_i{=}1, Pa_{X_i} = u), \ldots, m' P'(x_i{=}k, Pa_{X_i} = u))$
    - Typically, $P'(X_i)$ = uniform

Score Equivalence

- If $\mathcal{G}$ and $\mathcal{G}$' are $\mathcal{I}$-equivalent, then they have **same** score

- **Theorem 1**: Maximum likelihood score and BIC score satisfy score equivalence.

- **Theorem 2**:

  If

  - P($\mathcal{G}$) assigns same prior to $\mathcal{I}$-equivalent structures (eg, edge counting), and
  - each parameter prior is Dirichlet

  then

  - **Bayesian score satisfies score equivalence**

    if and only if

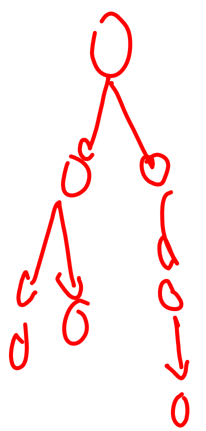
    prior over parameters represented as a **BDe prior**!

Learning General DAGs

- In a tree, every node only has ≤ 1 parent
- **Theorem**:
  - The problem of learning a BN structure with at most d parents that optimizes BDe is NP-hard for any (fixed) $d \geq 2$
- Most structure learning approaches use heuristics
  - Exploit score decomposition
  - (Quickly) Describe two heuristics that exploit decomposition in different ways

Learn BN structure using local search

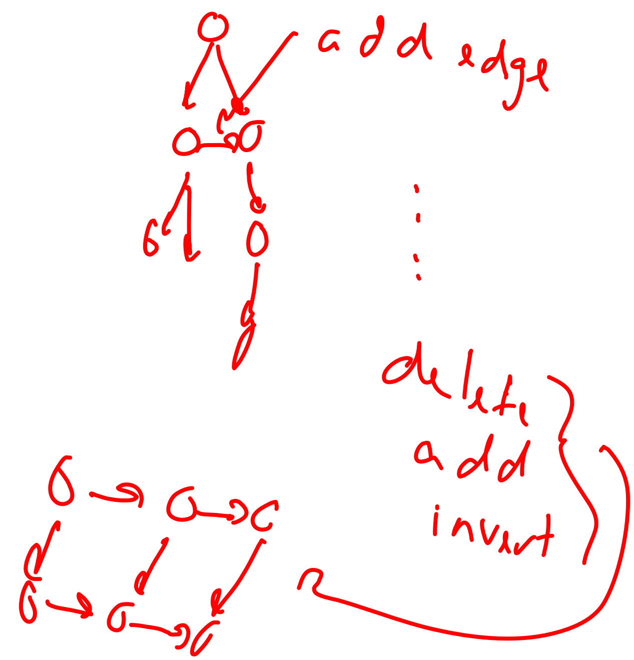
Starting from Chow-Liu tree



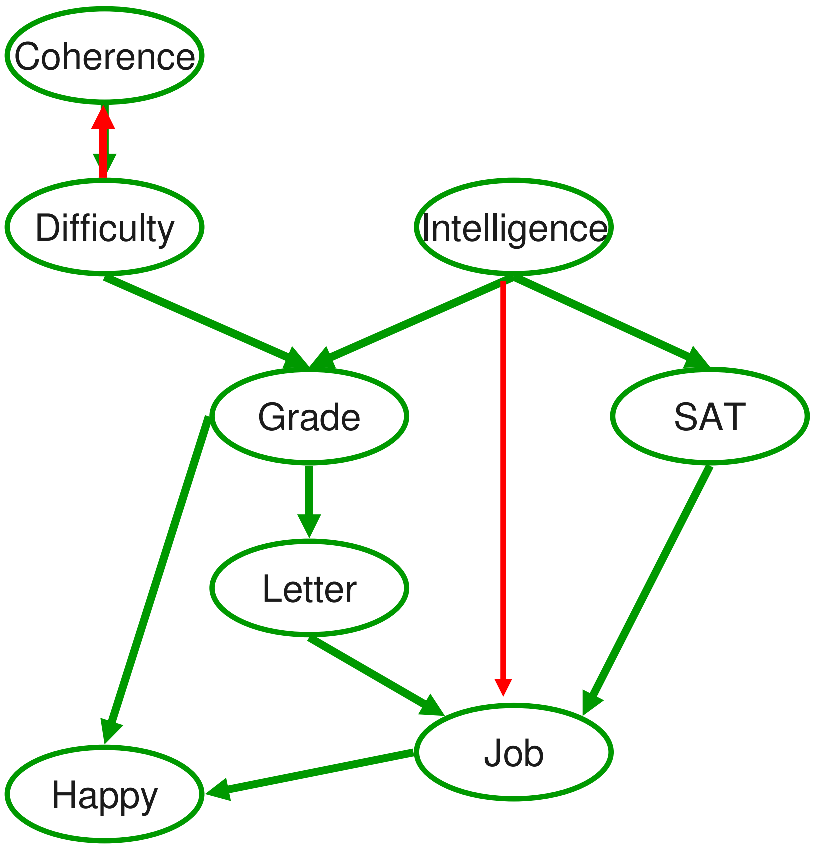
Local search,
possible moves:
Only if acyclic!!!
- Add edge
- Delete edge
- Invert edge

Computed locally (⇒ efficiently) thanks to Score Decomposition…
FamScore

Select using favorite score

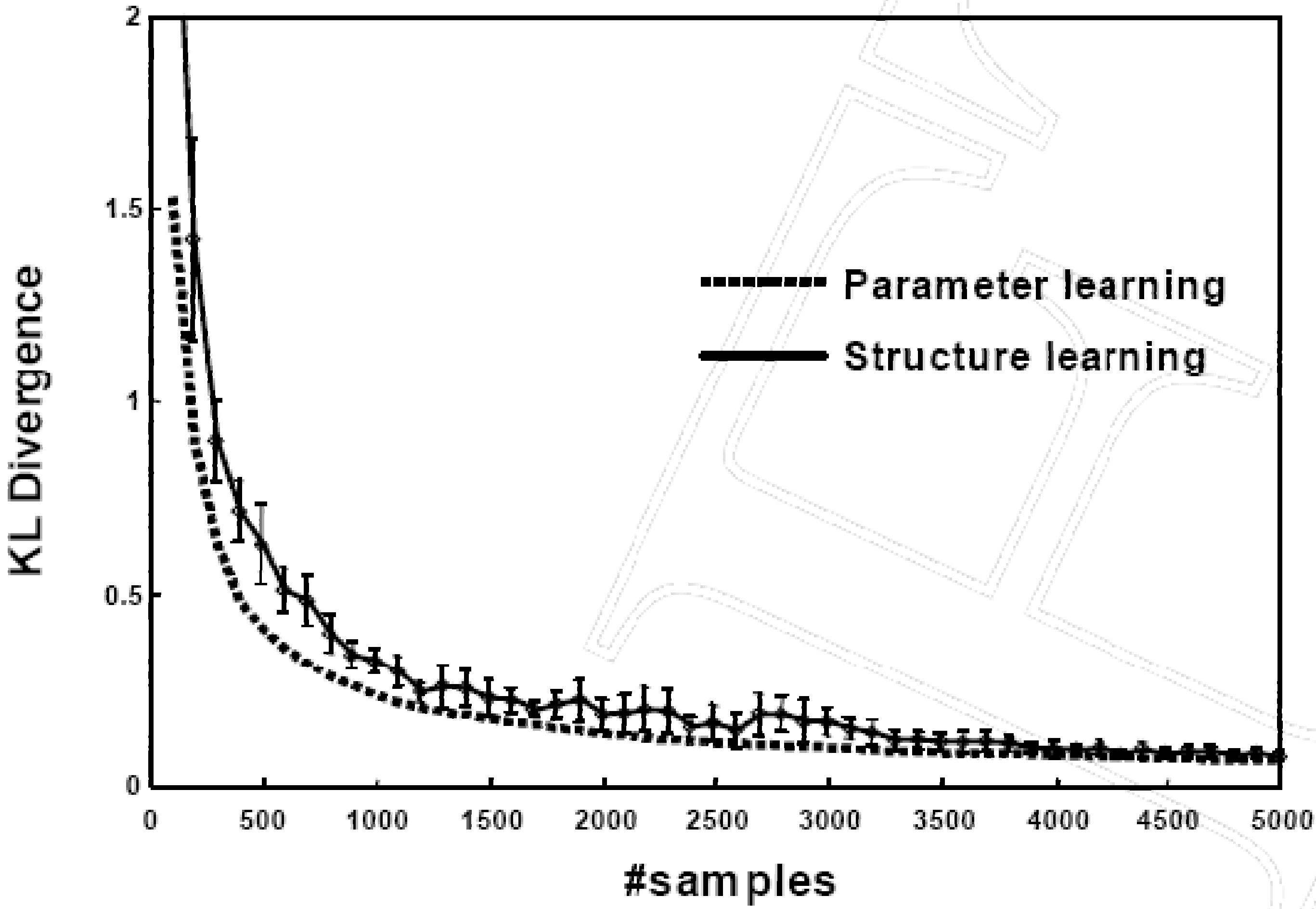


Exploit score decomposition in local search



- Add edge:
  - Re-score only one family!
- Delete edge:
  - Re-score only one family!
- Reverse edge
  - Re-score only two families

Some Experiments



Alarm network

Order search versus Graph search

- Order search advantages
  - For fixed order, optimal BN – more "global" optimization
  - Space of orders ($n!$) much smaller than space of graphs $\Omega(2^{n^2})$
- Graph search advantages
  - Not restricted to k parents
    - Especially if exploiting CPD structure, such as CSI
  - Cheaper per iteration
  - Finer moves within a graph

Bayesian Model Averaging

- So far, we have selected a single structure
- But, if you are really Bayesian…
  must average over structures
  - Similar to averaging over parameters

    $$\log P(D \mid \mathcal{G}) = \log \int_{\theta_{\mathcal{G}}} P(D \mid \mathcal{G}, \theta_{\mathcal{G}}) P(\theta_{\mathcal{G}} | \mathcal{G}) d\theta_{\mathcal{G}}$$

  - P($\mathcal{G}$|$\mathcal{D}$) $\rightarrow$ probability for each graph
- Inference for structure averaging is very hard!!!
  - Clever tricks in KF text

Summary wrt Learning BN Structure

- Decomposable scores
  - Data likelihood
  - Information theoretic interpretation
  - Bayesian
  - BIC approximation
- Priors
  - Structure and parameter assumptions
  - BDe if and only if score equivalence
- Best tree (Chow-Liu)
- Best TAN
- Nearly best k-treewidth (in $O(N^{k+1})$)
- Bayesian model averaging