Readings:
K&F: 15.1, 15.2, 15.3, 15.4, 15.5

Structure Learning in BNs 2: (the good,) the bad, the ugly

Graphical Models – 10708
Carlos Guestrin
Carnegie Mellon University

October 4th, 2006

Maximum likelihood score overfits!

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

- Information never hurts:
- Adding a parent always increases score!!!

Bayesian score

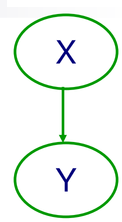
- Prior distributions:
  - Over structures
  - Over parameters of a structure
- Posterior over structures given data:

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

Bayesian score and model complexity

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

True model:



P(Y=t|X=t) = 0.5 + α
P(Y=t|X=f) = 0.5 - α

- Structure 1: X and Y independent
  - Score doesn't depend on alpha
- Structure 2: X $\rightarrow$ Y
  - Data points split between P(Y=t|X=t) and P(Y=t|X=f)
  - For fixed M, only worth it for large α
    - Because posterior over parameter will be more diffuse with less data

Bayesian, a decomposable score

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

- As with last lecture, assume:
  - Local and global parameter independence
- Also, prior satisfies **parameter modularity**:
  - If X_i has same parents in *G* and *G'*, then parameters have same prior
- Finally, structure prior P(*G*) satisfies **structure modularity**
  - Product of terms over families
  - E.g., $P(G) \propto c^{|G|}$
- Bayesian score decomposes along families!

BIC approximation of Bayesian score

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

$$\log P(D \mid \mathcal{G}) = \log P(D \mid \mathcal{G}, \theta_{\mathcal{G}}) - \frac{\log m}{2} \text{Dim}(\mathcal{G}) + O(1)$$

BIC approximation, a decomposable score

- BIC: $\text{Score}_{\text{BIC}}(\mathcal{G} : D) = \log P(D \mid \mathcal{G}, \theta_{\mathcal{G}}) - \frac{\log m}{2}\text{Dim}(\mathcal{G})$

- Using information theoretic formulation:

$$\text{Score}_{\text{BIC}}(\mathcal{G} : D) = m \sum_i \hat{I}(X_i, \mathbf{Pa}_{X_i,\mathcal{G}}) - m \sum_i \hat{H}(X_i) - \frac{\log m}{2} \sum_i \text{Dim}(P(X_i \mid \mathbf{Pa}_{X_i,\mathcal{G}}))$$

Consistency of BIC and Bayesian scores

Consistency is limiting behavior, says nothing about finite sample size!!!

- A scoring function is **consistent** if, for true model G^*, as $m \to \infty$, with probability 1
  - G^* maximizes the score
  - All structures **not I-equivalent** to G^* have strictly lower score
- **Theorem**: BIC score is consistent
- **Corollary**: the Bayesian score is consistent
- What about maximum likelihood score?

Priors for general graphs

- For finite datasets, prior is important!
- Prior over structure satisfying prior modularity
- What about prior over parameters, how do we represent it?
  - *K2 prior*: fix an α, $P(\theta_{X_i|\mathbf{Pa}_{X_i}}) = \text{Dirichlet}(\alpha,\ldots,\alpha)$
  - K2 is "inconsistent"

BDe prior

- Remember that Dirichlet parameters analogous to "fictitious samples"
- 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)
- **BDe prior**:
- Has "consistency property":

Announcements

- **Project description is out on class website**:
  - Individual or groups of two only
  - Suggested projects on the class website, or do something related to your research (preferable)
    - Must be something you started this semester
    - The semester goes really quickly, so be realistic (and ambitious ☺)
- **Project deliverables**:
  - one page proposal due next week (10/11)
  - 5-page milestone report Nov. 1st
  - Poster presentation on Dec. 1st, 3-6pm
  - Write up, 8-pages, due Dec. 8th
  - All write ups in NIPS format (see class website), page limits are strict
- **Objective**:
  - Explore and apply concepts in **probabilistic graphical models**
  - Doing a fun project!

Score equivalence

- If *G* and *G'* are I-equivalent then they have same score
- **Theorem 1**: Maximum likelihood score and BIC score satisfy score equivalence
- **Theorem 2**:
  - If P(*G*) assigns same prior to I-equivalent structures (e.g., edge counting)
  - and parameter prior is dirichlet
  - then **Bayesian score satisfies score equivalence** *if and only if* prior over parameters represented as a **BDe prior**!!!!!!

Chow-Liu for Bayesian score

- Edge weight $w_{X_j \to X_i}$ is advantage of adding X_j as parent for X_i

- Now have a directed graph, need directed spanning forest
  - Note that adding an edge can hurt Bayesian score – choose forest not tree
  - But, if score satisfies score equivalence, then $w_{X_j \to X_i} = w_{X_j \to X_i}$!
  - Simple maximum spanning forest algorithm works

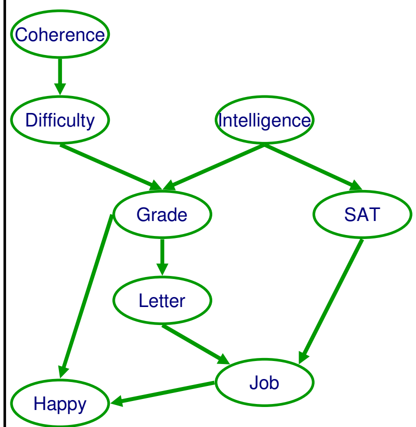
Structure learning for general graphs

- In a tree, a node only has one parent

- **Theorem**:
  - The problem of learning a BN structure with at most *d* parents 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

Understanding score decomposition



Fixed variable order 1

- Pick a variable order $\prec$
  - e.g., $X_1,\ldots,X_n$
- X_i can only pick parents in $\{X_1,\ldots,X_{i-1}\}$
  - Any subset
  - Acyclicity guaranteed!
- Total score = sum score of each node

Fixed variable order 2

- Fix max number of parents to k
- For each *i* in order $\prec$
  - Pick $\mathbf{Pa}_{X_i} \subseteq \{X_1,\ldots,X_{i-1}\}$
    - Exhaustively search through all possible subsets
    - $\mathbf{Pa}_{X_i}$ is maximum $\mathbf{U} \subseteq \{X_1,\ldots,X_{i-1}\}$ FamScore($X_i|\mathbf{U}$: *D*)
- Optimal BN for each order!!!
- Greedy search through space of orders:
  - E.g., try switching pairs of variables in order
  - If neighboring vars in order are switch, only need to recompute score for this pair
    - O(n) speed up per iteration
    - Local moves may be worse

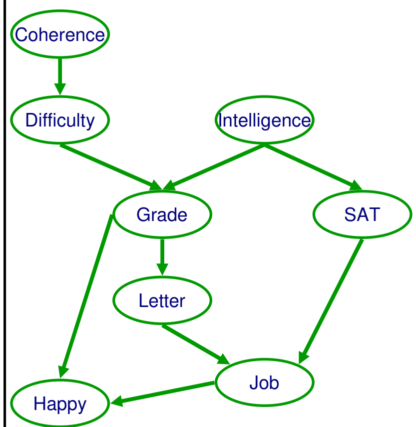
Learn BN structure using local search

Starting from Chow-Liu tree

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

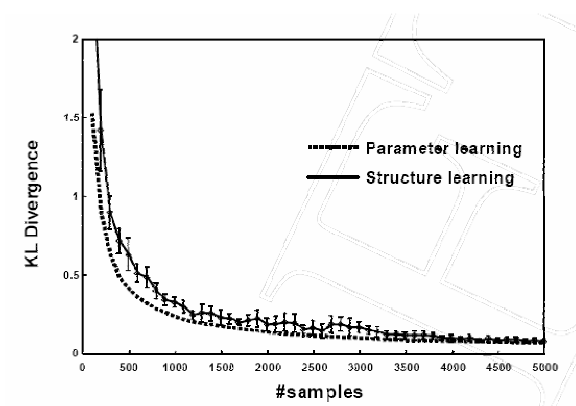
Select using favorite score

Exploit score decomposition in local search



- Add edge and delete edge:
  - Only rescore one family!
- Reverse edge
  - Rescore 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 much smaller than space of graphs
- 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

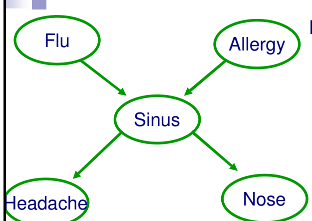
$$\log P(D \mid \mathcal{G}) = \log \int_{\theta_{\mathcal{G}}} P(D \mid \mathcal{G}, \theta_{\mathcal{G}}) P(\theta_{\mathcal{G}} | \mathcal{G}) d\theta_{\mathcal{G}}$$

- Inference for structure averaging is very hard!!!
  - Clever tricks in reading

What you need to know about learning BN structures

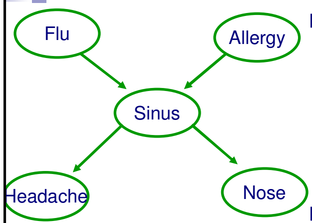
- Decomposable scores
  - Data likelihood
  - Information theoretic interpretation
  - Bayesian
  - BIC approximation
- Priors
  - Structure and parameter assumptions
  - BDe if and only if score equivalence
- Best tree (Chow-Liu)
- Best TAN
- Nearly best k-treewidth (in $O(N^{k+1})$)
- Search techniques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

Inference in graphical models: Typical queries 1



- Conditional probabilities
  - Distribution of some var(s). given evidence

Inference in graphical models: Typical queries 2 – Maximization



- Most probable explanation (MPE)
  - Most likely assignment to all hidden vars given evidence
- Maximum a posteriori (MAP)
  - Most likely assignment to some var(s) given evidence

Are MPE and MAP Consistent?

P(S=t)=0.4
P(S=f)=0.6

P(N|S)

- Most probable explanation (MPE)
  - Most likely assignment to all hidden vars given evidence
- Maximum a posteriori (MAP)
  - Most likely assignment to some var(s) given evidence

Complexity of conditional probability queries 1

- How hard is it to compute P(X|**E**=**e**)?

Reduction – 3-SAT

$$(\overline{X}_1 \vee X_2 \vee X_3) \wedge (\overline{X}_2 \vee X_3 \vee X_4) \wedge \ldots$$

Complexity of conditional probability queries 2

- How hard is it to compute P(X|**E**=**e**)?
  - At least NP-hard, but even harder!

Inference is #P-hard, hopeless?

- Exploit structure!
- Inference is hard in general, but easy for many (real-world relevant) BN structures

What about the maximization problems? First, most probable explanation (MPE)

- What's the complexity of MPE?

What about maximum a posteriori?

- At least, as hard as MPE!

- Actually, much harder!!! NP^{PP}-complete!

Can we exploit structure for maximization?

- For MPE

- For MAP

Exact inference is hard, what about approximate inference?

- Must define approximation criterion!
- Relative error of ε>0
- Absolute error of ε>0

Hardness of approximate inference

- Relative error of ε
- Absolute error of ε

What you need to know about inference

- Types of queries
  - probabilistic inference
  - most probable explanation (MPE)
  - maximum a posteriori (MAP)
    - MPE and MAP are truly different (don't give the same answer)

- Hardness of inference
  - Exact and approximate inference are NP-hard
  - MPE is NP-complete
  - MAP is much harder (NP^{PP}-complete)