Readings:
K&F: 3.4, 14.1, 14.2

BN Semantics 3
Now it’s personal!

Parameter Learning 1

Graphical Models – 10708
Carlos Guestrin
Carnegie Mellon University

September 22nd, 2006

Building BNs from independence properties

- From d-separation we learned:
  - Start from local Markov assumptions, obtain all independence assumptions encoded by graph
  - For most *P*’s that factorize over *G*, I(*G*) = I(*P*)
  - All of this discussion was for a given *G* that is an I-map for *P*
- Now, give me a *P*, how can I get a *G*?
  - i.e., give me the independence assumptions entailed by *P*
  - Many *G* are “equivalent”, how do I represent this?
  - Most of this discussion is not about practical algorithms, but useful concepts that will be used by practical algorithms
    - Practical algs next week

Minimal I-maps

- One option:
  - *G* is an I-map for *P*
  - *G* is as simple as possible
- *G* is a **minimal I-map** for *P* if deleting any edges from *G* makes it no longer an I-map

Obtaining a minimal I-map

- Given a set of variables and conditional independence assumptions
- Choose an ordering on variables, e.g., $X_1, \ldots, X_n$
- For i = 1 to n
  - Add X_i to the network
  - Define parents of X_i, $\mathbf{Pa}_{X_i}$, in graph as the minimal subset of $\{X_1,\ldots,X_{i-1}\}$ such that local Markov assumption holds – X_i independent of rest of $\{X_1,\ldots,X_{i-1}\}$, given parents $\mathbf{Pa}_{Xi}$
  - Define/learn CPT – $P(X_i| \mathbf{Pa}_{Xi})$

Flu, Allergy, SinusInfection, Headache

Minimal I-map not unique (or minimal)

- Given a set of variables and conditional independence assumptions
- Choose an ordering on variables, e.g., $X_1, \ldots, X_n$
- For i = 1 to n
  - Add X_i to the network
  - Define parents of X_i, $\mathbf{Pa}_{X_i}$, in graph as the minimal subset of $\{X_1,\ldots,X_{i-1}\}$ such that local Markov assumption holds – X_i independent of rest of $\{X_1,\ldots,X_{i-1}\}$, given parents $\mathbf{Pa}_{Xi}$
  - Define/learn CPT – $P(X_i| \mathbf{Pa}_{Xi})$

Flu, Allergy, SinusInfection, Headache

Perfect maps (P-maps)

- I-maps are not unique and often not simple enough
- Define "simplest" *G* that is I-map for *P*
  - A BN structure *G* is a **perfect map** for a distribution *P* if I(*P*) = I(*G*)
- Our goal:
  - Find a perfect map!
  - Must address equivalent BNs

Inexistence of P-maps 1

- XOR (this is a hint for the homework)

Inexistence of P-maps 2

- (Slightly un-PC) swinging couples example

Obtaining a P-map

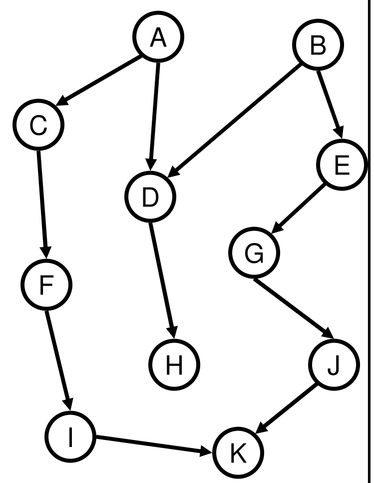
- Given the independence assertions that are true for P
- Assume that there exists a perfect map G^*
  - Want to find G^*
- Many structures may encode same independencies as G^*, when are we done?
  - Find all equivalent structures simultaneously!

I-Equivalence

- Two graphs G_1 and G_2 are **I-equivalent** if $I(G_1) = I(G_2)$
- **Equivalence class** of BN structures
  - Mutually-exclusive and exhaustive partition of graphs
- How do we characterize these equivalence classes?

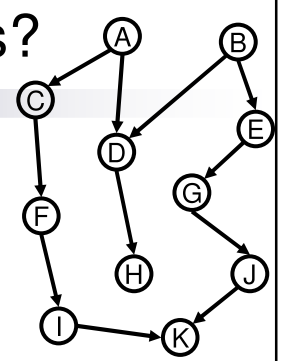
Skeleton of a BN

- **Skeleton** of a BN structure *G* is an **undirected graph** over the same variables that has an edge X–Y for every X→Y or Y→X in *G*



- (Little) **Lemma:** Two I-equivalent BN structures must have the same skeleton

What about V-structures?



- **V-structures are key property of BN structure**

- **Theorem:** If G_1 and G_2 have the same skeleton and V-structures, then G_1 and G_2 are I-equivalent

Same V-structures not necessary

- **Theorem:** If G_1 and G_2 have the same skeleton and V-structures, then G_1 and G_2 are I-equivalent

- Though sufficient, same V-structures not necessary

Immoralities & I-Equivalence

- Key concept not V-structures, but "immoralities" (unmarried parents ☺)
  - $X \rightarrow Z \leftarrow Y$, with no arrow between X and Y
  - Important pattern: X and Y independent given their parents, but not given Z
  - (If edge exists between X and Y, we have *covered* the V-structure)
- **Theorem:** G_1 and G_2 have the same skeleton and immoralities if and only if G_1 and G_2 are I-equivalent

Obtaining a P-map

- Given the independence assertions that are true for *P*
  - Obtain skeleton
  - Obtain immoralities

- From skeleton and immoralities, obtain every (and any) BN structure from the equivalence class

Identifying the skeleton 1

- When is there an edge between X and Y?

- When is there no edge between X and Y?

Identifying the skeleton 2

- Assume d is max number of parents (d could be n)

- For each X_i and X_j
  - $E_{ij} \leftarrow$ true
  - For each $\mathbf{U} \subseteq \mathbf{X} - \{X_i, X_j\}$, $|\mathbf{U}| \leq 2d$
    - Is $(X_i \perp X_j \mid \mathbf{U})$?
      - $E_{ij} \leftarrow$ true
  - If E_{ij} is true
    - Add edge X – Y to skeleton

Identifying immoralities

- Consider X – Z – Y in skeleton, when should it be an immorality?

- Must be $X \rightarrow Z \leftarrow Y$ (immorality):
  - When X and Y are **never independent** given **U,** if $Z \in \mathbf{U}$

- Must **not** be $X \rightarrow Z \leftarrow Y$ (not immorality):
  - When there exists **U** with $Z \in \mathbf{U}$, such that X and Y are **independent** given **U**

From immoralities and skeleton to BN structures

- Representing BN equivalence class as a **partially-directed acyclic graph** (PDAG)
- Immoralities force direction on other BN edges
- Full (polynomial-time) procedure described in reading

What you need to know

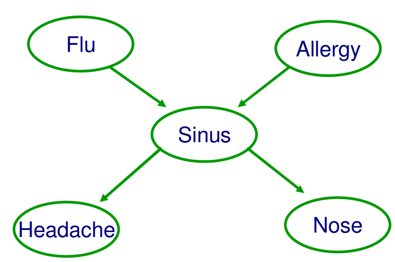
- Minimal I-map
  - every *P* has one, but usually many
- Perfect map
  - better choice for BN structure
  - not every *P* has one
  - can find one (if it exists) by considering I-equivalence
  - Two structures are I-equivalent if they have same skeleton and immoralities

Announcements

- I'll lead a special discussion session:
  - Today 2-3pm in NSH 1507
    - talk about homework, especially programming question

Review

- Bayesian Networks
  - Compact representation for probability distributions
  - Exponential reduction in number of parameters
  - Exploits independencies



- Next – Learn BNs
  - parameters
  - structure

Thumbtack – Binomial Distribution

- P(Heads) = θ, P(Tails) = 1-θ
- Flips are i.i.d.:
  - Independent events
  - Identically distributed according to Binomial distribution
- Sequence *D* of α_H Heads and α_T Tails

$$P(\mathcal{D} \mid \theta) = \theta^{\alpha_H}(1-\theta)^{\alpha_T}$$

Maximum Likelihood Estimation

- **Data:** Observed set *D* of α_H Heads and α_T Tails
- **Hypothesis:** Binomial distribution
- Learning θ is an optimization problem
  - What's the objective function?
- MLE: Choose θ that maximizes the probability of observed data:

$$\begin{aligned}\widehat{\theta} &= \arg\max_{\theta} \; P(\mathcal{D} \mid \theta) \\ &= \arg\max_{\theta} \; \ln P(\mathcal{D} \mid \theta)\end{aligned}$$

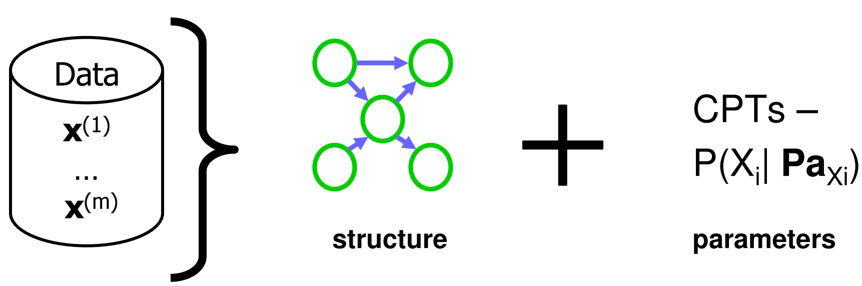
Your first learning algorithm

$$\begin{aligned}\widehat{\theta} &= \arg\max_{\theta} \ \ln P(\mathcal{D} \mid \theta) \\ &= \arg\max_{\theta} \ \ln \theta^{\alpha_H}(1-\theta)^{\alpha_T}\end{aligned}$$

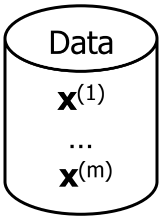
- Set derivative to zero: $\boxed{\frac{d}{d\theta} \ln P(\mathcal{D} \mid \theta) = 0}$

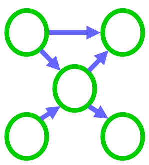
Learning Bayes nets

	Known structure	Unknown structure
Fully observable data		
Missing data		



Learning the CPTs

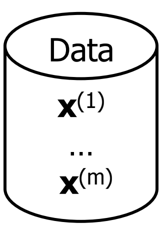


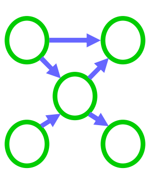


For each discrete variable X_i

MLE: $P(X_i = x_i \mid X_j = x_j) = \frac{\text{Count}(X_i = x_i, X_j = x_j)}{\text{Count}(X_j = x_j)}$

Learning the CPTs



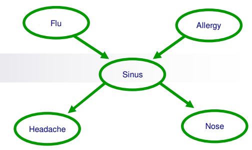


For each discrete variable X_i

MLE: $P(X_i = x_i \mid X_j = x_j) = \frac{\text{Count}(X_i = x_i, X_j = x_j)}{\text{Count}(X_j = x_j)}$

WHY??????????

Maximum likelihood estimation (MLE) of BN parameters – example



- Given structure, log likelihood of data:

$\log P(\mathcal{D} \mid \theta_{\mathcal{G}}, \mathcal{G})$

Maximum likelihood estimation (MLE) of BN parameters – General case

- Data: $\mathbf{x}^{(1)},\ldots,\mathbf{x}^{(m)}$
- Restriction: $\mathbf{x}^{(j)}[\mathbf{Pa}_{X_i}] \rightarrow$ assignment to $\mathbf{Pa}_{X_i}$ in $\mathbf{x}^{(j)}$
- Given structure, log likelihood of data:

  $\log P(\mathcal{D} \mid \theta_{\mathcal{G}}, \mathcal{G})$

Taking derivatives of MLE of BN parameters – General case

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

General MLE for a CPT

- Take a CPT: P(X|**U**)
- Log likelihood term for this CPT
- Parameter $\theta_{X=x|\mathbf{U}=\mathbf{u}}$:

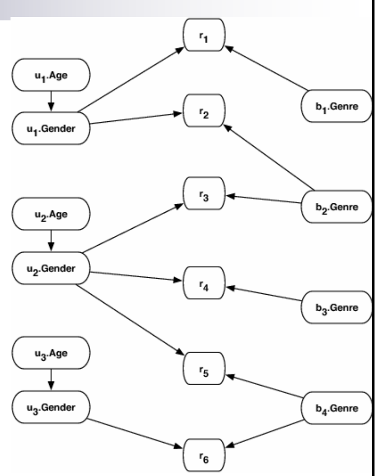
MLE: $P(X = x \mid \mathbf{U} = \mathbf{u}) = \theta_{X=x|\mathbf{U}=\mathbf{u}} = \dfrac{\text{Count}(X = x, \mathbf{U} = \mathbf{u})}{\text{Count}(\mathbf{U} = \mathbf{u})}$

Parameter sharing
(basics now, more later in the semester)

- Suppose we want to model customers' rating for books
- You know:
  - features of customers, e.g., age, gender, income,…
  - features of books, e.g., genre, awards, # of pages, has pictures,…
  - ratings: each user rates a few books
- A simple BN:

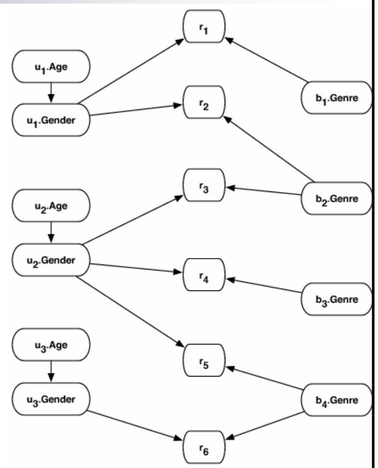
Using recommender system

- Answer probabilistic question:



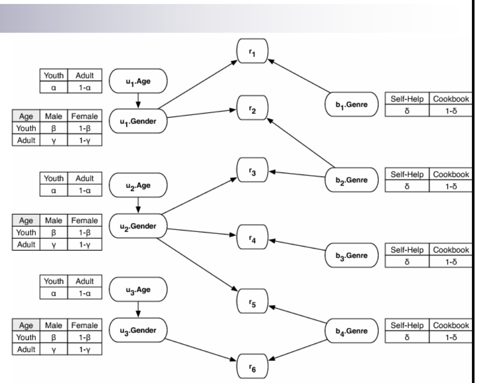
Learning parameters of recommender system BN

- How many parameters do I have to learn?
- How many samples do I have?



Parameter sharing for recommender system BN

- Use same parameters in many CPTs
- How many parameters do I have to learn?
- How many samples do I have?

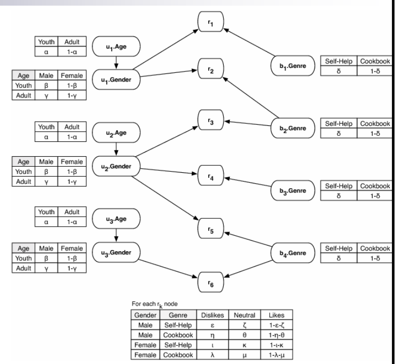


For each r_k node

Gender	Genre	Dislikes	Neutral	Likes
Male	Self-Help	ε	ζ	1-ε-ζ
Male	Cookbook	η	θ	1-η-θ
Female	Self-Help	ι	κ	1-ι-κ
Female	Cookbook	λ	μ	1-λ-μ

MLE with simple parameter sharing

- Estimating α:



- Estimating β:

- Estimating ε:

What you need to know about learning BNs thus far

- Maximum likelihood estimation
  - decomposition of score
  - computing CPTs
- Simple parameter sharing
  - why share parameters?
  - computing MLE for shared parameters