Next Week

Monday: PS 5 discussion and Bayesian classifiers (read Autoclass paper)
Wednesday: Context specific independence and local structure.
Friday: Normal-Wishart priors
Progress report #2
You should be 80% done with the technical portion of your project.

23 April: Papers due (No late projects accepted)

Structure Estimation with Incomplete Data
Where are we?

<table>
<thead>
<tr>
<th>Known Structure</th>
<th>Unknown Structure</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complete Data</td>
<td>Statistical parameter estimation</td>
</tr>
<tr>
<td>Incomplete Data</td>
<td>Parameter optimization</td>
</tr>
</tbody>
</table>

Issues

Parameter estimation
- Done

Model selection
- Before we needed to worry only about finding the best prediction (MAP): \( \hat{\Theta} = \arg \max P(\Theta | D) \)
- Now we need to approximate \( \log P(D | \Theta, G) \) to select the correct model.

‘Obvious’ Algorithm
1. Generate DAG.
2. Score DAG using “parametric EM”

Structural EM
Review: Parametric EM

Case 1: T T F
Fill in missing data using inference

Case 2: ? F T

Case 3: F ? F

Case 4: ? ? T

$\Theta_{k+1} = \arg \max_{\Theta} L[E[S_k]; G, \Theta]$

$E[S_k(D, H) | \Theta_k, G]$

E-Step                  M-Step

Optimization over Structure and Parameters (Incomplete Data)

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

P(A|h)  
P(B|A)  
P(C|h)  
P(D|A,C)  
P(D|A)  
P(E|D)  
P(h)
Approximating $P\{D|\Theta, G\}$

Why?
Select models based on how well they fit the data
Bayesian score: $\log P\{D|\Theta, G\} P\{\Theta, G\}$

Approaches
- MCMC (exact if you simulate long enough)
- Laplace
  - Full
  - Block diagonal
  - Diagonal
- BIC-MAP and BIC-ML
- Cheeseman-Stutz CS-MAP, CS-ML

Source:
Chickering and Heckerman, Efficient Approximations for the Marginal Likelihood of Bayesian Networks with Hidden Variables, MSR-TR-96-08

Assumptions
Multinomial variables with Dirichlet priors.

Dirichlet priors. $P\{x_i^j \mid pa_i^j\} = \theta_{ij}$ \hspace{1cm} $\sum_{j} \theta_{ij} = 1$
Large-scale approximations

As \( M \to \infty \) (\( M \) is # of samples)
approximate \( P\{\Phi \mid D, G\} \approx P\{D \mid \Phi, G\}\{\Phi \mid G\} \)
as a multi-variate Gaussian distribution.

Define \( g(\Phi) = \log(P\{D \mid \Phi, G\}\{\Phi \mid G\}) \)

Define \( \tilde{\Phi} \equiv \arg\max_{\phi} g(\Phi) \) (MAP configuration)

Taylor approximation

Approximate (2nd order Taylor) about MAP:
\[
g(\Phi) = g(\tilde{\Phi}) - \frac{1}{2} (\Phi - \tilde{\Phi})^\top A(\Phi - \tilde{\Phi})
\]
\[
A = \frac{\partial^2}{\partial \phi_{ik} \partial \phi_{abc}} (-g(\Phi)) \quad \text{(Hessian of } -g)\]
Laplace approximation

\[ P\{D \mid \Phi, G\} \{G \mid G\} = \exp(g(\Phi)) \]

\[ = P\{D \mid \Phi, G\} \{\Phi \mid G\} \exp\left(-\frac{1}{2}(\Phi - \Phi)^T A(\Phi - \Phi)\right) \]

Laplacian approximation

\[ P\{D \mid G\} = \int P\{D \mid \Phi, G\} \{\Phi \mid G\} d\Phi \approx P\{D \mid \Phi, G\} \{\Phi \mid G\} (2\pi)^{d/2} |A|^{-1/2} \]

\[ \log P\{D \mid G\} = \log P\{D \mid \Phi, G\} + \log P\{\Phi \mid G\} + \frac{d}{2} \log(2\pi) - \frac{1}{2} \log|A| \]

Computing the Hessian

Hessian can be computed by Bayes net propagation
(Thiessen, UAI-97)

Hessian can also be computed by likelihood ratio tests
(Rafferty, 95)
Limitations of Laplacian approximation

[Kass et al, 1988]
Relative accuracy $O(1/M)$, when
Unique MAP
(No unique MAP when aliasing, or reduced dimensionality)

$\Phi$ cannot lie on boundary of parameter space.

Possible for Hessian to be negative meaning that we cannot compute $\log|A|$!

Natural parameter set
If you were alert... you may have noticed that we used phi instead of theta...
Natural parameter set is a log transform of the normal parameters.

$$\phi_{ijk} = \log \frac{\theta_{ijk}}{\theta_{ij1}}$$

The coordinate system used has a strong effect on the accuracy of the approximation.

MacKay (1996)
The natural parameter set typically leads to more accurate approximations of “this type” of approximation (Taylor?)

Observation
NPS has a fewer number of parameters (1/2 for binomial problems). Helps eliminate reduced dimensionality problem?
Approximations to Laplace Approximation

Computing $|A|$ requires $O(d^2)$ expensive operations

Approximate $A$ as
- Block diagonal (Buntine, 94)
  Blocks are parameters for each variable.
- Diagonal (Becker + LeCun, 89)

Laplace:
$$\log P\{D | G\} = \log P\{D | \Phi, G\} + \log P\{\Phi | G\} + \frac{d}{2} \log (2\pi) - \frac{1}{2} \log |A|$$
Select terms that increase in $M$:
- $\log |A|$ increases as $\log M$
- $P\{D | \Phi\}$ increases as $M$
- MAP $\hat{\Phi}$ approaches $\text{ML}(\Phi \equiv \arg \max_{\Phi} P\{D | \Phi, G\})$ estimate

BIC/MDL:
$$\log P\{D | G\} = \log P\{D | \Phi, G\} - \frac{d}{2} \log M$$

Kass+Wasserman(95) and Rafferty(95)
For “particular priors”, relative error is $O(M^{-1/2})$
Cheeseman-Stutz Approximation

Approximation

\( P(D|G) \) can be computed efficiently for complete data.

Assume \( D' \) is any completion of \( D \)

\[
P(D|G) = P(D'|G) \int \frac{P(D|\Phi,G)d\Phi}{P(D'|\Phi,G)d\Phi}
\]

Approximation is best if \( P(D|\Phi,G) \) and \( P(D'|\Phi,G) \) have same shape.

Select \( D' \) so that its sufficient statistics match the expected sufficient statistics given \( D \) and \( G \).

Cheeseman-Stutz Approximation (cont'd)

\[
P(D|G) = P(D'|G) \int \frac{P(D|\Phi,G)d\Phi}{P(D'|\Phi,G)d\Phi}
\]

One approximation (not CS): Apply Laplace approximation to denominator and numerator.

\[
\log P(D|G) \\
\approx \log P(D'|G) - \log P(D|\bar{\Phi},G) + \frac{1}{2}\log|A| + \log P(D|\bar{\Phi},G) - \frac{1}{2}\log|\bar{A}|
\]
Cheeseman-Stutz

The CS approximation:
Use BIC for numerator and denominator

\[
\log P\{D|G\} = \log P\{D'|G\} - \log P\{D'|\Phi, G\} + \frac{d'}{2} \log M + \log P\{D|\Phi, G\} - \frac{d}{2} \log M
\]

Geiger (96) argues that \(d'\) and \(d\) are equal.

\[
\log P\{D|G\} = \log P\{D'|G\} - \log P\{D'|\Phi, G\} + \log P\{D|\Phi, G\}
\]

Correction for Hidden Variables

Assume that there is a hidden variable with \(k\) states.

There will be \(k!\) peaks in the distribution (unique ways to relabel the hidden variable)

Approximation: Multiply \(P(D|G)\) by \(k!\)

Applies to Laplace, BIC, MDL, and CS approximations.

Doesn’t apply to MCMC

\textit{MCMC can linger around a single peak}....
Experiments

“Unsupervised” Naïve Bayes classification problem

C is not observed (Aliasing)

Want to identify the optimal class size, \(|C| = r\)

\(r\) is incremented until the marginal likelihood begins to decrease for all approximations.

X’s are binary with Beta(1,1) distributions.

Scores compared

\(P(D|G)\) computed with

MCMC (gold standard)

Laplace, Block, Diagonal

BIC

BIC-MAP

BIC-ML (MDL)

CS (Cheeseman Stutz)

CS-MAP

CS-ML
Synthetic data generation

Forward sampling:

Sample the prior probabilities for hidden node \( C \) from a dirichlet\((1,1,1,...,1)\).
Sample the conditional probabilities in the \( n \) observation nodes from a beta\((1,1)\).

For \( m = 1 \) to \( M \)
- Sample \( C \)
- Sample \( X[1:n] \)

Keep the \( X \)'s and discard all of the other parameters.

Computation time vs model dimension.

Computation time vs \( d = nr \). Note that the time for a single EM iteration is plotted.
Laplace and MCMC (Candidate) significantly increase learning time.
Separation.

If clusters are well-separated, \( P\{C \mid X\} \approx 1 \) for most \( X \).

Technique for measuring separation:

\[
Sep(G, \Phi, X) = 1 - \frac{1}{M \log r} \sum_{m=1}^{M} \sum_{x} P\{X[m] \mid G, \Phi\} H(C \mid X[m], \Phi, G)
\]

Technique for establishing correlation:

\[
\phi(x_i^k \mid c') = \phi(x_i^k \mid c') + N(0, \eta)
\]
Notes

Typically

MCMC, Laplace, Block, Diagonal, and CS-MAP peak at the same value of $r$.

Laplace, Block, Diagonal, and CS-MAP agree with MCMC for $r \leq r^*$ but falls below MCMC for $r > r^*$

Why?

- Many MAP configurations when $r > r^*$?
- Tested by finding 100's of local maxima and summing—no improvement
- Possible explanation: When $r > r^*$ there are too many classes to fit the data and it is likely that some of the classes will be empty. The parameters for these empty classes will be superfluous, thus the maximum of $P(\Phi|D,G)$ will be a ridge, not a peak.

CS-MAP is better than CS-ML.
BIC-MAP is better than BIC-ML.

Sensitivity of results wrt observed variables.
Sensitivity to priors.

Results

Insensitive to
  sample size,
  number of classes in generator,
  separation,
  number of observed variables.

Some sensitivity to prior.
BIC and MDL seem to peak early
  under fits the data.
Laplace approximation should fail at a boundary
10% of $P(X|C)$ are set to zero.
CS MAP is more robust.

<table>
<thead>
<tr>
<th>Data set</th>
<th>n</th>
<th>$r^*$</th>
<th>M</th>
<th>MCMC</th>
<th>Laplace</th>
<th>Block</th>
<th>Diagonal</th>
<th>CS MAP</th>
<th>BIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small Soybean</td>
<td>35</td>
<td>4</td>
<td>47</td>
<td>4</td>
<td>3-4</td>
<td>3-4</td>
<td>4</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Audiology</td>
<td>70</td>
<td>24</td>
<td>226</td>
<td>2</td>
<td>-H</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Lung Cancer</td>
<td>56</td>
<td>3</td>
<td>32</td>
<td>3</td>
<td>-H</td>
<td>-H</td>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>
Approximation Conclusions

Model selection
  All except for BIC/MDL are accurate for model selection.

Sensitivity to priors
  All except BIC/MDL are sensitive to priors

MAP/ML
  CS is more accurate with MAP
  BIC/MDL is more accurate with ML

Accuracy
  Cheeseman Stutz tends to be more accurate than other approximations
  CS best when MAP is near a boundary.

Structural EM (Friedman, 97)
**Standard EM**

**Standard greedy structure search with incomplete data:**

Find all of the networks $C_1, C_2, \ldots, C_n$ that are adjacent to $G_n$

- Note that there are $O(n^2)$ successors
- “flip the bit on any arc”

Optimize parameters for $C_1, C_2, \ldots, C_n$ using “Parametric EM.”

- Have to run enough iterations to guarantee good model selection.

- Score $P[D|G]$ for $C_1, C_2, \ldots, C_n$ using BIC, Laplace, MCMC or CS.

Let $G_{n+1} = \arg \max_{G' \in C \cup \{G_n\}} \text{Score}(G')$

It is only practical to use this algorithm for VERY small problems

---

**Structural EM**

Optimize the parameters $\Theta_n$ for $G_n$ using some number of steps of EM.

Complete the data using the expected sufficient statistics given $\Theta_n$

Pretend that the data is complete and search some number of steps (say $k$) to find $G_{n+k}$

Why is this a win?

- We only have to run EM a few times.
- The same "completion" of the data is used to score several networks.

Say that $C_B(D)$ is a completion of the data using B

$$\text{Score}_{\text{MDL}}(B'; D) - \text{Score}_{\text{MDL}}(B'; D) \geq \text{Score}_{\text{MDL}}(B'; C_B(D)) - \text{Score}_{\text{MDL}}(B'; C_B(D))$$
Structure learning with incomplete data

Big changes:
- Need to use EM to optimize the parameters.
- Need to approximate \( P(\Theta | G) \) in order to compute \( P(D | G) \)

Approximations to the likelihood function \( P(D | G) \) (for clustering)
- All approximations underestimate when the problem has reduced dimensionality.
- Don’t use BIC or MDL: Underfits the data.
- Of the “cheap” approximations, Cheeseman-Stutz is best.
- Open issue: How well do these work for non-clustering problems?

Structural EM:
- It is a good idea to reuse the “completion” derived from one run of EM to score many adjacent candidate graphs.
- The technique described as paring search time down from years to hours on large problems (3-4 orders of magnitude).

Next Week

Next Week
- Monday: PS 5 discussion and Bayesian classifiers (read Autoclass paper)
- Wednesday: Context specific independence and local structure.
- Friday: Normal-Wishart priors
  - Progress report #2
  - You should be 80% done with the technical portion of your project.

Week After Next
- 12 April: Learning Dynamic Belief Networks.
- **14 April: We have no class**
- 16 April: Gaussian/Discrete networks

The Last Week
- 19 April: Summary
- 21 + 23 April: Student presentations
- 23 April: Papers due (No late projects accepted)