An Overview of Learning Bayes Nets From Data

Chris Meek
Microsoft Research

http://research.microsoft.com/~meek
What’s and Why’s

- What is a Bayesian network?
- Why Bayesian networks are useful?
- Why learn a Bayesian network?
What is a Bayesian Network?

also called belief networks, and (directed acyclic) graphical models

- Directed acyclic graph
  - Nodes are variables (discrete or continuous)
  - Arcs indicate dependence between variables.

- Conditional Probabilities (local distributions)

- Missing arcs implies conditional independence
- Independencies + local distributions => modular specification of a joint distribution

\[
p(x_1) \ p(x_2 \mid x_1) \ p(x_3 \mid x_1, x_2) = p(x_1, x_2, x_3)
\]
Why Bayesian Networks?

- **Expressive language**
  - Finite mixture models, Factor analysis, HMM, Kalman filter,…

- **Intuitive language**
  - Can utilize causal knowledge in constructing models
  - Domain experts comfortable building a network

- **General purpose “inference” algorithms**
  - $P(\text{Bad Battery} \mid \text{Has Gas, Won’t Start})$
    - Exact: Modular specification leads to large computational efficiencies
    - Approximate: “Loopy” belief propagation
Why Learning?

knowledge-based (expert systems)

- Answer Wizard, Office 95, 97, & 2000
- Troubleshooters, Windows 98 & 2000

data-based

- Causal discovery
- Data visualization
- Concise model of data
- Prediction
Overview

- **Learning Probabilities (local distributions)**
  - Introduction to Bayesian statistics: Learning a probability
  - Learning probabilities in a Bayes net
  - Applications

- **Learning Bayes-net structure**
  - Bayesian model selection/averaging
  - Applications
Learning Probabilities: Classical Approach

Simple case: Flipping a thumbtack

Given iid data, estimate $\theta$ using an estimator with good properties: low bias, low variance, consistent (e.g., ML estimate)
Learning Probabilities: Bayesian Approach

True probability $\theta$ is unknown

Bayesian probability density for $\theta$
Bayesian Approach: use Bayes' rule to compute a new density for $\theta$ given data

$$p(\theta \mid \text{data}) = \frac{p(\theta) p(\text{data} \mid \theta)}{\int p(\theta) p(\text{data} \mid \theta) \, d\theta} \propto p(\theta) p(\text{data} \mid \theta)$$
The Likelihood

\[ p(\text{heads} \mid \theta) = \theta \]

\[ p(\text{tails} \mid \theta) = (1 - \theta) \]

\[ p(\text{hhth...ttth} \mid \theta) = \theta^{\#h} (1 - \theta)^{\#t} \]

“binomial distribution”
Example: Application of Bayes rule to the observation of a single "heads"

\[ p(\theta) \times p(\text{heads}|\theta) = p(\theta|\text{heads}) \]
The probability of heads on the next toss

\[
p(h \mid d) = \int p(h \mid \theta, d) \ p(\theta \mid d) \ d\theta
\]

\[
= \int \theta \ p(\theta \mid d) \ d\theta
\]

\[
= E_{p(\theta \mid d)}(\theta)
\]

Note: This yields nearly identical answers to ML estimates when one uses a “flat” prior
Overview

- Learning Probabilities
  - Introduction to Bayesian statistics: Learning a probability
  - Learning probabilities in a Bayes net
  - Applications

- Learning Bayes-net structure
  - Bayesian model selection/averaging
  - Applications
From thumbtacks to Bayes nets

Thumbtack problem can be viewed as learning the probability for a very simple BN:

\[ P(X = \text{heads}) = f(\theta) \]
The next simplest Bayes net

The diagram shows two variables, $X$ and $Y$, each with two states: heads/tails and tails. The edges connecting $X$ to $Y$ indicate the conditional relationships between the variables.
The next simplest Bayes net

heads/tails $X$  

heads/tails $Y$

$i=1$ to $N$

$X_i$  

$Y_i$
The next simplest Bayes net

heads/tails $X$ heads/tails $Y$

"parameter independence"

$i=1$ to $N$

$X_i$ $Y_i$
The next simplest Bayes net

heads/tails $X$ heads/tails $Y$

"parameter independence"

↓

two separate thumbtack-like learning problems

$i=1$ to $N$

$X_i$ $Y_i$
A bit more difficult...

Three probabilities to learn:

- $\theta_{X=\text{heads}}$
- $\theta_{Y=\text{heads}|X=\text{heads}}$
- $\theta_{Y=\text{heads}|X=\text{tails}}$
A bit more difficult...

A bit more difficult...

heads/tails $X$ → $Y$ heads/tails

$\Theta X \rightarrow \Theta_{Y|X=\text{heads}} \rightarrow \Theta_{Y|X=\text{tails}}$

$X_1 \rightarrow Y_1$

$X_2 \rightarrow Y_2$

$\neg$ $\neg$

case 1

case 2
A bit more difficult...

heads/tails $X$ heads/tails

$\Theta_{X}$ $\Theta_{Y|X=\text{heads}}$ $\Theta_{Y|X=\text{tails}}$

$X_{1}$ $Y_{1}$ $X_{2}$ $Y_{2}$

case 1

case 2
A bit more difficult...

3 separate thumbtack-like problems
In general...

Learning probabilities in a BN is straightforward if:
- Likelihoods from the exponential family
  (multinomial, poisson, gamma, ...)
- Parameter independence
- Conjugate priors
- Complete data
Incomplete data makes parameters dependent
Incomplete data makes parameters dependent

Parameter Learning for incomplete data

- Monte-Carlo integration
  - Investigate properties of the posterior and perform prediction

- Large-sample Approx. (Laplace/Gaussian approx.)
  - Expectation-maximization (EM) algorithm and inference to compute mean and variance.

Variational methods
Overview

- **Learning Probabilities**
  - Introduction to Bayesian statistics: Learning a probability
  - Learning probabilities in a Bayes net
  - Applications

- **Learning Bayes-net structure**
  - Bayesian model selection/averaging
  - Applications
Example: Audio-video fusion
Beal, Attias, & Jojic 2002

Video scenario

Audio scenario

Goal: detect and track speaker

Slide courtesy Beal, Attias and Jojic
Separate audio-video models

\[ \pi_r \quad r \quad \pi_l \quad l_x \quad l_y \quad \pi_S \quad s \quad \mu_S, \phi_s \]

\[ \lambda_2, \nu_2 \quad \lambda_1, \nu_1 \quad \lambda \quad \psi \]

Frame n=1,…,N

audio data video data

Slide courtesy Beal, Attias and Jojic
Combined model

Frame n=1,…,N

audio data

video data

Slide courtesy Beal, Attias and Jojic
Tracking Demo

Slide courtesy Beal, Attias and Jojic
Overview

- **Learning Probabilities**
  - Introduction to Bayesian statistics: Learning a probability
  - Learning probabilities in a Bayes net
  - Applications

- **Learning Bayes-net structure**
  - Bayesian model selection/averaging
  - Applications
Two Types of Methods for Learning BNs

- **Constraint based**
  - Finds a Bayesian network structure whose implied independence constraints “match” those found in the data.

- **Scoring methods (Bayesian, MDL, MML)**
  - Find the Bayesian network structure that can represent distributions that “match” the data (i.e. could have generated the data).
Learning Bayes-net structure

Given data, which model is correct?

model 1: $X \quad Y$

model 2: $\rightarrow$
Bayesian approach

Given data, which model is correct? more likely?

model 1: \( \begin{align*}
&X \\
&Y
\end{align*} \)
\[ p(m_1) = 0.7 \] \[ p(m_1 \mid d) = 0.1 \]

Data \( d \)

model 2: \( \begin{align*}
&X \\
&\rightarrow Y
\end{align*} \)
\[ p(m_2) = 0.3 \] \[ p(m_2 \mid d) = 0.9 \]
Bayesian approach: Model Averaging

Given data, which model is correct? more likely?

model 1: \( X \) \( Y \) \( p(m_1) = 0.7 \) \( p(m_1 | d) = 0.1 \)

model 2: \( X \rightarrow Y \) \( p(m_2) = 0.3 \) \( p(m_2 | d) = 0.9 \)

average predictions
Bayesian approach: Model Selection

Given data, which model is correct? more likely?

model 1: \( X \) \( Y \) \( p(m_1) = 0.7 \)  
Data \( d \)  
\( p(m_1 | d) = 0.1 \)

model 2: \( X \) \( Y \) \( p(m_2) = 0.3 \)  
\( p(m_2 | d) = 0.9 \)

Keep the best model:  
- Explanation  
- Understanding  
- Tractability
To score a model, use Bayes rule

Given data $d$:

$$p(m | d) \propto p(m) p(d | m)$$

"marginal likelihood"

$$p(d | m) = \int p(d | \theta, m) p(\theta | m) d \theta$$
The Bayesian approach and Occam’s Razor

\[ p(d \mid m) = \int p(d \mid \theta_m, m) p(\theta_m \mid m) d\theta_m \]

- True distribution
- Simple model
- Just right
- Complicated model
- All distributions
Computation of Marginal Likelihood

Efficient closed form if
- Likelihoods from the exponential family (binomial, poisson, gamma, ...)
- Parameter independence
- Conjugate priors
- No missing data, including no hidden variables

Else use approximations
- Monte-Carlo integration
- Large-sample approximations
- Variational methods
Practical considerations

The number of possible BN structures is super exponential in the number of variables.

How do we find the best graph(s)?
Model 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
Learning the correct model

- True graph $G$ and $P$ is the generative distribution
- Markov Assumption: $P$ satisfies the independencies implied by $G$
- Faithfulness Assumption: $P$ satisfies only the independencies implied by $G$

Theorem: Under Markov and Faithfulness, with enough data generated from $P$ one can recover $G$ (up to equivalence). Even with the greedy method!
Learning Bayes Nets From Data

data

<table>
<thead>
<tr>
<th>$X_1$</th>
<th>$X_2$</th>
<th>$X_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>true</td>
<td>1</td>
<td>Red</td>
</tr>
<tr>
<td>false</td>
<td>5</td>
<td>Blue</td>
</tr>
<tr>
<td>false</td>
<td>3</td>
<td>Green</td>
</tr>
<tr>
<td>true</td>
<td>2</td>
<td>Red</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

+ prior/expert information

Bayes-net learner

Bayes net(s)
Overview

- Learning Probabilities
  - Introduction to Bayesian statistics: Learning a probability
  - Learning probabilities in a Bayes net
  - Applications

- Learning Bayes-net structure
  - Bayesian model selection/averaging
  - Applications
Preference Prediction (a.k.a. Collaborative Filtering)

- **Example:** Predict what products a user will likely purchase given items in their shopping basket.
- **Basic idea:** use other people’s preferences to help predict a new user’s preferences.

- **Numerous applications**
  - Tell people about books or web-pages of interest
  - Movies
  - TV shows
Example: TV viewing

Nielsen data: 2/6/95-2/19/95

<table>
<thead>
<tr>
<th></th>
<th>Show1</th>
<th>Show2</th>
<th>Show3</th>
</tr>
</thead>
<tbody>
<tr>
<td>viewer 1</td>
<td>y</td>
<td>n</td>
<td>n</td>
</tr>
<tr>
<td>viewer 2</td>
<td>n</td>
<td>y</td>
<td>y</td>
</tr>
<tr>
<td>viewer 3</td>
<td>n</td>
<td>n</td>
<td>n</td>
</tr>
</tbody>
</table>

~200 shows, ~3000 viewers

Goal: For each viewer, recommend shows they haven’t watched that they are likely to watch
Making predictions

\[ \text{infer: } p(\text{watched 90210} \mid \text{everything else we know about the user}) \]
Making predictions

infer: \( p \) (watched 90210 | everything else we know about the user)
Making predictions

infer $p(\text{watched Melrose place} \mid \text{everything else we know about the user})$
Recommendation list

- p=.67 Seinfeld
- p=.51 NBC Monday night movies
- p=.17 Beverly hills 90210
- p=.06 Melrose place
Software Packages

- **BUGS**: [http://www.mrc-bsu.cam.ac.uk/bugs](http://www.mrc-bsu.cam.ac.uk/bugs)
  parameter learning, hierarchical models, MCMC

- **Hugin**: [http://www.hugin.dk](http://www.hugin.dk)
  Inference and model construction

- **xBaies**: [http://www.city.ac.uk/~rgc](http://www.city.ac.uk/~rgc)
  chain graphs, discrete only

- **Bayesian Knowledge Discoverer**: [http://kmi.open.ac.uk/projects/bkd](http://kmi.open.ac.uk/projects/bkd)
  commercial

- **MIM**: [http://inet.uni-c.dk/~edwards/miminfo.html](http://inet.uni-c.dk/~edwards/miminfo.html)

  classification

- **BN Power Constructor**: BN PowerConstructor

- **Microsoft Research: WinMine**
  [http://research.microsoft.com/~dmax/WinMine/Tooldoc.htm](http://research.microsoft.com/~dmax/WinMine/Tooldoc.htm)
For more information...

Tutorials:
K. Murphy (2001)
http://www.cs.berkeley.edu/~murphyk/Bayes/bayes.html


Books: