Bayesian Networks – Representation

Machine Learning – CSE546
Carlos Guestrin
University of Washington
November 25, 2013

Handwriting recognition

Character recognition, e.g., kernel SVMs
Webpage classification

- Company home page vs Personal home page vs University home page vs...

Handwriting recognition 2
Webpage classification 2

Today – Bayesian networks

- One of the most exciting advancements in statistical AI in the last decades
- Generalizes naïve Bayes and logistic regression classifiers
- Compact representation for exponentially-large probability distributions
- Exploit conditional independencies
Causal structure

Suppose we know the following:
- The flu causes sinus inflammation
- Allergies cause sinus inflammation
- Sinus inflammation causes a runny nose
- Sinus inflammation causes headaches

How are these connected?

Possible queries

- Inference
- Most probable explanation
- Active data collection
Car starts BN

- 18 binary attributes
- Inference
  - $P(BatteryAge|Starts=f)$

$2^{16}$ terms, why so fast?
Not impressed?
- HailFinder BN – more than $3^{54} = 58149737003040059690390169$ terms

Factored joint distribution - Preview

Flu, Allergy, Sinus, Headache, Nose
What about probabilities?
Conditional probability tables (CPTs)

Flu  Allergy
Sinus
Headache  Nose

Number of parameters

Flu  Allergy
Sinus
Headache  Nose
Key: Independence assumptions

Knowing sinus separates the variables from each other

(Marginal) Independence

- Flu and Allergy are (marginally) independent

<table>
<thead>
<tr>
<th></th>
<th>Flu = t</th>
<th>Flu = f</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allergy = t</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Allergy = f</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Marginally independent random variables

- **Sets** of variables $X$, $Y$
- $X$ is independent of $Y$ if
  - $P (X=x \perp Y=y), \forall x \in \text{Val}(X), y \in \text{Val}(Y)$

  - **Shorthand**:  
    - **Marginal independence**: $P (X \perp Y)$

- **Proposition**: $P$ satisfies $(X \perp Y)$ if and only if
  - $P(X,Y) = P(X)P(Y)$

---

Conditional independence

- Flu and Headache are not (marginally) independent
- Flu and Headache are independent given Sinus infection
- More Generally:
Conditionally independent random variables

- **Sets** of variables $X$, $Y$, $Z$
- $X$ is independent of $Y$ given $Z$ if
  - $P \vdash (X=x \perp Y=y | Z=z), \forall x \in \text{Val}(X), y \in \text{Val}(Y), z \in \text{Val}(Z)$

- **Shorthand**: $P \vdash (X \perp Y | Z)$
  - For $P \vdash (X \perp Y | \emptyset)$, write $P \vdash (X \perp Y)$

- **Proposition**: $P$ satisfies $(X \perp Y | Z)$ if and only if
  - $P(X,Y|Z) = P(X|Z) P(Y|Z)$

---

**The independence assumption**

Local Markov Assumption: A variable $X$ is independent of its non-descendants given its parents
Explaining away

Local Markov Assumption:
A variable X is independent of its non-descendants given its parents

Naïve Bayes revisited

Local Markov Assumption:
A variable X is independent of its non-descendants given its parents
Joint distribution

Why can we decompose? Markov Assumption!

The chain rule of probabilities

- $P(A, B) = P(A)P(B|A)$

- More generally:
  - $P(X_1, \ldots, X_n) = P(X_1)P(X_2|X_1) \ldots P(X_n|X_1, \ldots, X_{n-1})$
Chain rule & Joint distribution

Local Markov Assumption:
A variable $X$ is independent of its non-descendants given its parents

The Representation Theorem – Joint Distribution to BN

If conditional independencies in BN are subset of conditional independencies in $P$

Joint probability distribution:
$$P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i \mid Pa_{X_i})$$
Two (trivial) special cases

- Edgeless graph
- Fully-connected graph

Bayesian Networks – (Structure) Learning

Machine Learning – CSE546
Carlos Guestrin
University of Washington
November 25, 2013
Review

- Bayesian Networks
  - Compact representation for probability distributions
  - Exponential reduction in number of parameters
- Fast probabilistic inference
  - As shown in demo examples
  - Compute $P(X|e)$
- Today
  - Learn BN structure

Learning Bayes nets

Data $x^{(1)}$, ..., $x^{(m)}$ + structure + CPTs – $P(X_i| Pa_{x_i})$
Learning the CPTs

For each discrete variable $X_i$

Data

$x^{(1)}$

$\ldots$

$x^{(m)}$

MLE:

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

Information-theoretic interpretation of maximum likelihood

- Given structure, log likelihood of data:

$$\log P(\mathcal{D} \mid \theta_{\mathcal{G}}, \mathcal{G})$$
Information-theoretic interpretation of maximum likelihood 2

Given structure, log likelihood of data:

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

Information-theoretic interpretation of maximum likelihood 3

Given structure, log likelihood of data:

$$\log \hat{P}(D \mid \theta, \mathcal{G}) = m \sum_{i} \sum_{x_i, \mathbf{Pa}_X_i, \mathcal{G}} \hat{P}(x_i, \mathbf{Pa}_X_i, \mathcal{G}) \log \hat{P}(x_i \mid \mathbf{Pa}_X_i, \mathcal{G})$$
Decomposable score

- Log data likelihood
\[
\log \hat{P}(D | \theta, G) = m \sum_i \hat{I}(X_i, Pa_{X_i} | G) - m \sum_i \hat{H}(X_i)
\]

- Decomposable score:
  - Decomposes over families in BN (node and its parents)
  - Will lead to significant computational efficiency!!!
  - Score\((G : D) = \sum_i \text{FamScore}(X_i|Pa_{X_i} : D)\)

How many trees are there?

Nonetheless – Efficient optimal algorithm finds best tree
Scoring a tree 1: equivalent trees

\[ \log P(D \mid \theta, G) = m \sum_i I(X_i, Pa_{X_i, G}) - m \sum_i H(X_i) \]

Scoring a tree 2: similar trees

\[ \log P(D \mid \theta, G) = m \sum_i I(X_i, Pa_{X_i, G}) - m \sum_i H(X_i) \]
Chow-Liu tree learning algorithm 1

- 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:
    $$I(X_i, X_j) = \sum_{x_i, x_j} \hat{P}(x_i, x_j) \log \frac{\hat{P}(x_i, x_j)}{\hat{P}(x_i)\hat{P}(x_j)}$$
- Define a graph
  - Nodes $X_1, \ldots, X_n$
  - Edge $(i,j)$ gets weight $I(X_i, X_j)$

Chow-Liu tree learning algorithm 2

- $\log \hat{P}(\mathcal{D} \mid \theta, G) = m \sum_i I(X_i, \text{Pa}_{X_i}, G) - m \sum_i \bar{H}(X_i)$
- Optimal tree BN
  - Compute maximum weight spanning tree
  - Directions in BN: pick any node as root, breadth-first-search defines directions
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) \( d > 1 \)**

- Most structure learning approaches use heuristics
- (Quickly) Describe the two simplest heuristic

Learn BN structure using local search

**Starting from Chow-Liu tree**

**Local search**, possible moves:
- Add edge
- Delete edge
- Invert edge

**Score using BIC**
Learn Graphical Model Structure using LASSO

- Graph structure is about selecting parents:
  - If no independence assumptions, then CPTs depend on all parents:
  - With independence assumptions, depend on key variables:
  - One approach for structure learning, sparse logistic regression!

What you need to know about learning BN structures

- Decomposable scores
  - Maximum likelihood
  - Information theoretic interpretation
- Best tree (Chow-Liu)
- Beyond tree-like models is NP-hard
- Use heuristics, such as:
  - Local search
  - LASSO