CS 440/ECE448 Lecture 19: Bayes Net Inference

Including slides by Svetlana Lazebnik, 11/2016
CS440/ECE448 Lecture 19: Bayesian Networks and Bayes Net Inference

Slides by Svetlana Lazebnik, 10/2016
Modified by Mark Hasegawa-Johnson, 3/2019 and Julia Hockenmaier 3/2019
Today’s lecture

• **Bayesian Networks** (Bayes Nets)
  • A graphical representation of probabilistic models
  • Capture conditional (in)dependencies between random variables

• **Inference and Learning in Bayes Nets**
  • Inference = Reasoning
  • Learning = Parameter estimation
Review: Bayesian inference

A general scenario:

Query variables: \( X \)

Evidence (observed) variables and their values: \( E = e \)

Inference problem: answer questions about the query variables given the evidence variables

This can be done using the posterior distribution \( P(X \mid E = e) \)

Example of a useful question: **Which \( X \) is true?**

More formally: what value of \( X \) has the least probability of being wrong?

Answer: \( \text{MPE} = \text{MAP} \) (argmin \( P(\text{error}) \) = argmax \( P(X=x \mid E=e) \))
Today: What if $P(X,E)$ is complicated?

• Very, very common problem: $P(X,E)$ is complicated because both $X$ and $E$ depend on some hidden variable $Y$

• SOLUTION:
  • Represent the dependencies as a graph
  • When your algorithm performs inference, make sure it does so in the order of the graph

• FORMALISM: Bayesian Network
Bayesian Inference with Hidden Variables

- **A general scenario:**
  - Query variables: $X$
  - Evidence (observed) variables and their values: $E = e$
  - Hidden (unobserved) variables: $Y$

- **Inference problem:** answer questions about the query variables given the evidence variables
  - This can be done using the posterior distribution $P(X \mid E = e)$
  - In turn, the posterior needs to be derived from the full joint $P(X, E, Y)$

$$P(X \mid E = e) = \frac{P(X, e)}{P(e)} \propto \sum_y P(X, e, y)$$

- Bayesian networks are a tool for representing joint probability distributions efficiently
Bayesian Networks
Bayesian networks

• More commonly called *graphical models*
• A way to depict conditional independence relationships between random variables
• A compact specification of full joint distributions
Independence

- Random variables $X$ and $Y$ are **independent** ($X \perp Y$) if
  \[ P(X,Y) = P(X) \times P(Y) \]
  NB.: Since $X$ and $Y$ are R.V.s (not individual events),
  $P(X,Y) = P(X) \times P(Y)$ is an abbreviation for:
  $\forall x \forall y \, P(X=x,Y=y) = P(X=x) \times P(Y=y)$

- $X$ and $Y$ are **conditionally independent** given $Z$ ($X \perp Y \mid Z$) if
  \[ P(X,Y \mid Z) = P(X \mid Z) \times P(Y \mid Z) \]
  The value of $X$ depends on the value of $Z$, and the value of $Y$ depends on the value of $Z$, so $X$ and $Y$ are not independent.
Bayesian networks

- **Insight:** (Conditional) independence assumptions are essential for probabilistic modeling

- **Bayes Net:** a directed graph which represents the joint distribution of a number of random variables in a directed graph
  - Nodes = random variables
  - Directed edges = dependencies
Bayesian networks

- **Nodes**: random variables
- **Edges**: dependencies
  - An edge from one variable (parent) to another (child) indicates direct influence (conditional probabilities)
  - Edges must form a directed, *acyclic* graph
  - Each node is conditioned on its parents: $P(X \mid \text{Parents}(X))$
    These conditional distributions are the parameters of the network
- **Each node is conditionally independent of its non-descendants given its parent**

We have four random variables
Weather is independent of cavity, toothache and catch
Toothache and catch both depend on cavity.
Conditional independence and the joint distribution

- **Key property:** each node is conditionally independent of its non-descendants given its parents

- Suppose the nodes $X_1, ..., X_n$ are sorted in topological order of the graph (i.e. if $X_i$ is a parent of $X_j$, $i < j$)

- To get the joint distribution $P(X_1, ..., X_n)$, use chain rule (step 1 below) and then take advantage of independencies (step 2)

\[
P(X_1, ..., X_n) = \prod_{i=1}^{n} P(X_i \mid X_1, ..., X_{i-1})
\]

\[
= \prod_{i=1}^{n} P(X_i \mid \text{Parents}(X_i))
\]
The joint probability distribution

\[
P(X_1, \ldots, X_n) = \prod_{i=1}^{n} P(X_i | \text{Parents}(X_i))
\]

\[
P(j, m, a, \neg b, \neg e) = P(\neg b) P(\neg e) P(a | \neg b, \neg e) P(j | a) P(m | a)
\]
Example: N independent coin flips

• Complete independence: no interactions:
  $P(X_1) P(X_2) P(X_3)$
Conditional probability distributions

• To specify the full joint distribution, we need to specify a conditional distribution for each node given its parents:

\[ P(X | \text{Parents}(X)) \]
Naïve Bayes document model

• Random variables:
  • $X$: document class
  • $W_1, ..., W_n$: words in the document

• Dependencies: $P(X) P(W_1 | X) ... P(W_n | X)$
Example: Los Angeles Burglar Alarm

• I have a burglar alarm that is sometimes set off by minor earthquakes. My two neighbors, John and Mary, promised to call me at work if they hear the alarm
  • Example inference task: suppose Mary calls and John doesn’t call. What is the probability of a burglary?

• What are the random variables?
  • Burglary, Earthquake, Alarm, JohnCalls, MaryCalls

• What are the direct influence relationships?
  • A burglar can set the alarm off
  • An earthquake can set the alarm off
  • The alarm can cause Mary to call
  • The alarm can cause John to call
Example: Burglar Alarm

- A “model” is a complete specification of the dependencies.
- The conditional probability tables are the *model parameters.*

\[
P(B) \quad \text{Burglary} \quad P(E) \quad \text{Earthquake}
\]

\[
P(A | B, E) \quad \text{Alarm}
\]

\[
P(J | A) \quad \text{JohnCalls} \quad P(M | A) \quad \text{MaryCalls}
\]
Example: Burglar Alarm

- **Burglary**
  - $P(B) = 0.001$

- **Earthquake**
  - $P(E) = 0.002$

| B | E | $P(A|B,E)$ |
|---|---|------------|
| T | T | 0.95       |
| T | F | 0.94       |
| F | T | 0.29       |
| F | F | 0.001      |

- **Alarm**

- **JohnCalls**
  - $P(J|A)$
    - T: 0.90
    - F: 0.05

- **MaryCalls**
  - $P(M|A)$
    - T: 0.70
    - F: 0.01
Outline

• Review: Bayesian inference
• Bayesian network: graph semantics
• The Los Angeles burglar alarm example

• Conditional independence ≠ Independence
• Constructing a Bayesian network: Structure learning
• Constructing a Bayesian network: Hire an expert
Independence

• By saying that $X_i$ and $X_j$ are independent, we mean that
  $P(X_j, X_i) = P(X_i)P(X_j)$

• $X_i$ and $X_j$ are independent if and only if they have no common ancestors

• Example: independent coin flips

  \[X_1 \quad X_2 \quad \cdots \quad X_n\]

• Another example: Weather is independent of all other variables in this model.
Conditional independence

• By saying that $W_i$ and $W_j$ are conditionally independent given $X$, we mean that
  \[ P(W_i, W_j | X) = P(W_i | X)P(W_j | X) \]

• $W_i$ and $W_j$ are conditionally independent given $X$ if and only if they have no common ancestors other than the ancestors of $X$.

• Example: naïve Bayes model:
Conditional independence ≠ Independence

**Common cause:** Conditionally Independent

- **Y:** Project due
- **X:** Newsgroup busy
- **Z:** Lab full

Are X and Z independent? **No**

\[ P(Z, X) = \sum_Y P(Z|Y)P(X|Y)P(Y) \]

\[ P(Z)P(X) = \left( \sum_Y P(Z|Y)P(Y) \right) \left( \sum_Y P(X|Y)P(Y) \right) \]

Are they conditionally independent given Y? **Yes**

\[ P(Z, X|Y) = P(Z|Y)P(X|Y) \]

**Common effect:** Independent

- **X:** Raining
- **Z:** Ballgame
- **Y:** Traffic

Are X and Z independent? **Yes**

\[ P(X, Z) = P(X)P(Z) \]

Are they conditionally independent given Y? **No**

\[ P(Z, X|Y) = \frac{P(Y|X, Z)P(X)P(Z)}{P(Y)} ≠ P(Z|Y)P(X|Y) \]
Conditional independence ≠ Independence

Common cause: Conditionally Independent

\[ Y \leftarrow X \rightarrow Z \]

- Y: Project due
- X: Newsgroup busy
- Z: Lab full

Are X and Z independent? **No**

Knowing X tells you about Y, which tells you about Z.

Are they conditionally independent given Y? **Yes**

If you already know Y, then X gives you no useful information about Z.

Common effect: Independent

\[ X \leftarrow Y \rightarrow Z \]

- X: Raining
- Z: Ballgame
- Y: Traffic

Are X and Z independent? **Yes**

Knowing X tells you nothing about Z.

Are they conditionally independent given Y? **No**

If Y is true, then either X or Z must be true.

Knowing that X is false means Z must be true.

We say that X “explains away” Z.
Conditional independence ≠ Independence

Being conditionally independent given X does NOT mean that $W_i$ and $W_j$ are independent. Quite the opposite. For example:

• The document topic, X, can be either “sports” or “pets”, equally probable.
• $W_1=1$ if the document contains the word “food,” otherwise $W_1=0$.
• $W_2=1$ if the document contains the word “dog,” otherwise $W_2=0$.
• Suppose you don’t know X, but you know that $W_2=1$ (the document has the word “dog”). Does that change your estimate of $p(W_1=1)$?
Conditional independence

Another example: *causal chain*

- X and Z are conditionally independent given Y, because they have no common ancestors other than the ancestors of Y.

- Being conditionally independent given Y does NOT mean that X and Z are independent. Quite the opposite. For example, suppose \( P(X) = 0.5, P(Y|X) = 0.8, P(Y|\neg X) = 0.1, P(Z|Y) = 0.7, \) and \( P(Z|\neg Y) = 0.4. \) Then we can calculate that \( P(Z|X) = 0.64, \) but \( P(Z) = 0.535 \)
Outline

• Review: Bayesian inference
• Bayesian network: graph semantics
• The Los Angeles burglar alarm example
• Conditional independence ≠ Independence
• Constructing a Bayesian network: Structure learning
• Constructing a Bayesian network: Hire an expert
Constructing a Bayes Network: Two Methods

   1. Suppose you know the variables, but you don’t know which variables depend on which others. You can learn this from data.
   2. This is an exciting new area of research in statistics, where it goes by the name of “analysis of causality.”
   3. ... but it’s almost always harder than method #2. You should know how to do this in very simple examples (like the Los Angeles burglar alarm), but you don’t need to know how to do this in the general case.

2. “Hire an Expert:”
   1. Find somebody who knows how to solve the problem.
   2. Get her to tell you what are the important variables, and which variables depend on which others.
   3. THIS IS ALMOST ALWAYS THE BEST WAY.
Constructing Bayesian networks: Structure Learning

1. Choose an ordering of variables $X_1, \ldots, X_n$
2. For $i = 1$ to $n$
   • add $X_i$ to the network
   • Check your training data. If there is any variable $X_1, \ldots, X_{i-1}$ that CHANGES the probability of $X_i=1$, then add that variable to the set $\text{Parents}(X_i)$ such that $P(X_i \mid \text{Parents}(X_i)) = P(X_i \mid X_1, \ldots, X_{i-1})$

3. Repeat the above steps for every possible ordering (complexity: $n!$).
4. Choose the graph that has the smallest number of edges.
Example

- Suppose we choose the ordering M, J, A, B, E
Example

- Suppose we choose the ordering M, J, A, B, E
Example

• Suppose we choose the ordering M, J, A, B, E
Example

• Suppose we choose the ordering M, J, A, B, E
Example

- Suppose we choose the ordering M, J, A, B, E
Example

• Suppose we choose the ordering M, J, A, B, E
Example

- Suppose we choose the ordering M, J, A, B, E
Example

• Suppose we choose the ordering M, J, A, B, E
Example contd.

- Deciding conditional independence is hard in noncausal directions
  - The causal direction seems much more natural
- Network is less compact: $1 + 2 + 4 + 2 + 4 = 13$ numbers needed (vs. $1+1+4+2+2=10$ for the causal ordering)
Why store it in causal order? A: Saves memory

• Suppose we have a Boolean variable \( X_i \) with \( k \) Boolean parents. How many rows does its conditional probability table have?
  • \( 2^k \) rows for all the combinations of parent values
  • Each row requires one number for \( P(X_i = \text{true} \mid \text{parent values}) \)

• If each variable has no more than \( k \) parents, how many numbers does the complete network require?
  • \( O(n \cdot 2^k) \) numbers – vs. \( O(2^n) \) for the full joint distribution

• How many nodes for the burglary network?
  
  \[ 1 + 1 + 4 + 2 + 2 = 10 \text{ numbers (vs. } 2^5 - 1 = 31) \]
Outline

• Review: Bayesian inference
• Bayesian network: graph semantics
• The Los Angeles burglar alarm example
• Conditional independence ≠ Independence
• Constructing a Bayesian network: Structure learning

• Constructing a Bayesian network: Hire an expert
A more realistic Bayes Network: Car diagnosis

- **Initial observation**: car won’t start
- **Orange**: “broken, so fix it” nodes
- **Green**: testable evidence
- **Gray**: “hidden variables” to ensure sparse structure, reduce parameters
Car insurance
In research literature...

Causal Protein-Signaling Networks Derived from Multiparameter Single-Cell Data
In research literature...

Describing Visual Scenes Using Transformed Objects and Parts
E. Sudderth, A. Torralba, W. T. Freeman, and A. Willsky.

In research literature...

Audiovisual Speech Recognition with Articulator Positions as Hidden Variables
Mark Hasegawa-Johnson, Karen Livescu, Partha Lal and Kate Saenko

Detecting interaction links in a collaborating group using manually annotated data
S. Mathur, M.S. Poole, F. Pena-Mora, M. Hasegawa-Johnson, N. Contractor
*Social Networks* 10.1016/j.socnet.2012.04.002
Detecting interaction links in a collaborating group using manually annotated data

S. Mathur, M.S. Poole, F. Pena-Mora, M. Hasegawa-Johnson, N. Contractor

*Social Networks* 10.1016/j.socnet.2012.04.002
Summary

• Bayesian networks provide a natural representation for (causally induced) conditional independence
• Topology + conditional probability tables
• Generally easy for domain experts to construct
CS 440/ECE448 Lecture 19: Bayes Net Inference

Including slides by Svetlana Lazebnik, 11/2016
Bayes Net Inference and Learning
Bayes Network Inference & Learning

Bayes net is a **memory-efficient model** of dependencies among a set of random variables.

**Inference problem**: answer questions about the **query variables X** given the **evidence variables and their values E=e** as well as some **unobserved (hidden) variables Y**.

- We want to know the **posterior** distribution $P(X \mid E = e)$
- The posterior can be derived from the **full joint** $P(X, E, Y)$
- How do we make this **computationally efficient**?

**Learning problem**: given some training examples, how do we estimate the parameters of the model?

- Parameters $= p(\text{variable} \mid \text{parents})$, for each variable in the net
Outline

• Inference Examples
• Inference Algorithms
  • Trees: Sum-product algorithm
  • Poly-trees: Junction tree algorithm
  • Graphs: No polynomial-time algorithm
• Parameter Learning
Practice example 1

• Variables: *Cloudy, Sprinkler, Rain, Wet Grass*
Practice example 1

- Given that the grass is wet, what is the probability that it has rained?

\[ P(r \mid w) = \frac{P(r, w)}{P(w)} \]
Practice example 1

• Given that the grass is wet, what is the probability that it has rained?

\[
P(r \mid w) = \frac{P(r, w)}{P(w)} = \frac{\sum_{C=c, S=s} P(c, s, r, w)}{\sum_{C=c, S=s, R=r} P(c, s, r, w)}
\]
Practice example 1

• Given that the grass is wet, what is the probability that it has rained?

\[
P(r \mid w) = \frac{P(r, w)}{P(w)} = \frac{\sum_{C=c, S=s} P(c, s, r, w)}{\sum_{C=c, S=s, R=r} P(c, s, r, w)}
\]

\[
= \frac{\sum_{C=c, S=s} P(c) P(s \mid c) P(r \mid c) P(w \mid r, s)}{\sum_{C=c, S=s, R=r} P(c) P(s \mid c) P(r \mid c) P(w \mid r, s)}
\]
Practice Example #2

• Suppose you have an observation, for example, “Jack called” (J=1)
• You want to know: was there a burglary?
• You need

\[ P(B = 1|J = 1) = \frac{P(B, J = 1)}{\sum_b P(B = b, J = 1)} \]

• So you need to compute the table \( P(B, J) \) for all possible settings of \( (B, J) \)
Bayes Net Inference: The Hard Way

1. \[ P(B, E, A, J, M) = P(B) \cdot P(E) \cdot P(A|B,E) \cdot P(J|A) \cdot P(M|A) \]

2. \[ P(B, J) = \sum_{E} \sum_{A} \sum_{M} P(B, E, A, J, M) \]

Exponential complexity (#P-hard, actually): N variables, each of which has K possible values \( \Rightarrow O(K^N) \) time complexity
Is there an easier way?

• **Tree-structured Bayes nets: the sum-product algorithm**
  • Quadratic complexity, $O\{ NK^3 \}$

• **Polytrees: the junction tree algorithm**
  • Pseudo-polynomial complexity, $O\{ NK^M \}$, for $M<N$

• **Arbitrary Bayes nets: #P complete, $O\{ K^N \}$**
  • The SAT problem is a Bayes net!
1. Tree-Structured Bayes Nets

- Suppose these are all binary variables.
- We observe $E=1$
- We want to find $P(H=1|E=1)$
- Means that we need to find both $P(H=0,E=1)$ and $P(H=1,E=1)$ because

$$P(H = 1|E = 1) = \frac{P(H = 1, E = 1)}{\sum_h P(H = h, E = 1)}$$
The Sum-Product Algorithm (Belief Propagation)

- Find the **only undirected path from the evidence variable to the query variable** (E-D-B-F-G-I-H)
- Find the **directed root of this path** $P(F)$
- Find the **joint probabilities of root and evidence**: $P(F=0,E=1)$ and $P(F=1,E=1)$
- Find the **joint probabilities of query and evidence**: $P(H=0,E=1)$ and $P(H=1,E=1)$
- Find the **conditional probability** $P(H=1|E=1)$
The Sum-Product Algorithm (Belief Propagation)

Starting with the root $P(F)$, we find $P(F,E)$ by alternating **product steps and sum steps**:

1. **Product**: $P(B,D,F) = P(F)P(B|F)P(D|B)$
2. **Sum**: $P(D,F) = \sum_{B=0}^{1} P(B,D,F)$
3. **Product**: $P(D,E,F) = P(D,F)P(E|D)$
4. **Sum**: $P(E,F) = \sum_{D=0}^{1} P(D,E,F)$
Starting with the root $P(E,F)$, we find $P(E,H)$ by alternating product steps and sum steps:

1. Product: $P(E,F,G) = P(E,F)P(G|F)$
2. Sum: $P(E,G) = \sum_{F=0}^{1} P(E,F,G)$
4. Sum: $P(E,I) = \sum_{G=0}^{1} P(E,G,I)$
6. Sum: $P(E,H) = \sum_{I=0}^{1} P(E,H,I)$
Time Complexity of Belief Propagation

- Each **product** step generates a table with 3 variables
- Each **sum** step reduces that to a table with 2 variables
- If each variable has K values, and if there are $O\{N\}$ variables on the path from evidence to query, then time complexity is $O\{NK^3\}$
Time Complexity of Bayes Net Inference

• **Tree-structured Bayes nets: the sum-product algorithm**
  • Quadratic complexity, $O\{NK^3\}$

• **Polytrees: the junction tree algorithm**
  • Pseudo-polynomial complexity, $O\{NK^M\}$, for $M<N$

• **Arbitrary Bayes nets: \#P complete, $O\{K^N\}$**
  • The SAT problem is a Bayes net!
2. The Junction Tree Algorithm

a. **Moralize** the graph (identify each variable’s Markov blanket)
b. **Triangulate** the graph (eliminate undirected cycles)
c. Create the **junction tree** (form cliques)
d. **Run the sum-product algorithm** on the junction tree
2.a. Markov Blanket

• Suppose there is a Bayes net with variables A, B, C, D, E, F, G, H
• The “Markov blanket” of variable F is D, E, G if
  \[ P(F|A, B, C, D, E, G, H) = P(F|D, E, G) \]
2.a. Markov Blanket

• Suppose there is a Bayes net with variables A, B, C, D, E, F, G, H

• The “Markov blanket” of variable F is D, E, G if

\[
P(F|A, B, C, D, E, G, H) = P(F|D, E, G)
\]
2.a. Markov Blanket

• The “Markov blanket” of variable F is D,E,G if
  \[ P(F|A,B,C,D,E,G,H) = P(F|D,E,G) \]
  • How can we prove that?
• \[ P(A,\ldots,H) = P(A)P(B|A) \ldots \]
• Which of those terms include F?
2.a. Markov Blanket

• Which of those terms include F?
• Only these two:
  \[ P(F|D) \]
  and
  \[ P(G|E,F) \]
2.a. Markov Blanket

The Markov Blanket of variable F includes only its immediate family members:

- Its parent, D
- Its child, G
- The other parent of its child, E

Because \( P(F|A,B,C,D,E,G,H) = P(F|D,E,G) \)
2.a. Moralization

“Moralization” =
1. If two variables have a child together, force them to get married.
2. Get rid of the arrows (not necessary any more).

Result: Markov blanket = the set of variables to which a variable is connected.
2.b. Triangulation

**Triangulation** = draw edges so that there is no unbroken cycle of length $> 3$.

There are usually many different ways to do this. For example, here's one:
2.c. Form Cliques

Clique = a group of variables, all of whom are members of each other’s immediate family.

Junction Tree = a tree in which
- Each node is a clique from the original graph,
- Each edge is an “intersection set,” naming the variables that overlap between the two cliques.
2.d. Sum-Product

Suppose we need $P(B,G)$:

1. Product: $P(B,C,D,F)=P(B)P(C|B)P(D|B)P(F|D)$
2. Sum: $P(B, C, F) = \sum_D P(B, C, D, F)$
4. Sum: $P(B, E, F) = \sum_C P(B, C, E, F)$
5. Product: $P(B,E,F,G) = P(B,E,F)P(G|E,F)$
6. Sum: $P(B, G) = \sum_E \sum_F P(B, E, F, G)$

Complexity: $O\{NK^M\}$, where $N$=# cliques, $K$=# values for each variable, $M = 1 +$ # variables in the largest clique
Consider the burglar alarm example.

a. Moralize this graph

b. Is it already triangulated? If not, triangulate it.

c. Draw the junction tree
Solution
Solution

a. Moralize this graph
b. Is it already triangulated?

Answer: yes. There is no unbroken cycle of length > 3.
Solution

c. Draw the junction tree
Time Complexity of Bayes Net Inference

• Tree-structured Bayes nets: the sum-product algorithm
  • Quadratic complexity, $O\{NK^3\}$

• Polytrees: the junction tree algorithm
  • Pseudo-polynomial complexity, $O\{NK^M\}$, for $M<N$

• Arbitrary Bayes nets: #P complete, $O\{K^N\}$
  • The SAT problem is a Bayes net!
Bayesian network inference

• In full generality, NP-hard
  • More precisely, #P-hard: equivalent to counting satisfying assignments

• We can reduce satisfiability to Bayesian network inference
  • Decision problem: is $P(Y) > 0$?

$$Y = (U_1 \lor U_2 \lor U_3) \land (\neg U_1 \lor \neg U_2 \lor U_3) \land (U_2 \lor \neg U_3 \lor U_4)$$
Bayesian network inference

• In full generality, NP-hard
  • More precisely, #P-hard: equivalent to counting satisfying assignments
• We can reduce satisfiability to Bayesian network inference
  • Decision problem: is \( P(Y) > 0? \)

\[ Y = (U_1 \lor U_2 \lor U_3) \land (\neg U_1 \lor \neg U_2 \lor U_3) \land (U_2 \lor \neg U_3 \lor U_4) \]

G. Cooper, 1990
Bayesian network inference

\[
P(U_1, U_2, U_3, U_4, C_1, C_2, C_3, D_1, D_2, Y) =
\]
\[
P(U_1)P(U_2)P(U_3)P(U_4)P(C_1 | U_1, U_2, U_3)P(C_2 | U_1, U_2, U_3)P(C_3 | U_2, U_3, U_4)P(D_1 | C_1)P(D_2 | D_1, C_2)P(Y | D_2, C_3)
\]
Why can’t we use the junction tree algorithm to efficiently compute $\Pr(Y)$?
Bayesian network inference

Why can’t we use the junction tree algorithm to efficiently compute $\Pr(Y)$?
Answer: after we moralize and triangulate, the size of the largest clique ($u_2u_3c_1c_2c_3$) is $M \approx N$, same order of magnitude as the original problem.
Time Complexity of Bayes Net Inference

- Tree-structured Bayes nets: the sum-product algorithm
  - Quadratic complexity, $O\{NK^3\}$
- Polytrees: the junction tree algorithm
  - Pseudo-polynomial complexity, $O\{NK^M\}$, for $M < N$
- Arbitrary Bayes nets: #P complete, $O\{K^N\}$
  - The SAT problem is a Bayes net!
Parameter learning

- **Inference problem**: given values of evidence variables $E = e$, answer questions about query variables $X$ using the posterior $P(X | E = e)$

- **Learning problem**: estimate the parameters of the probabilistic model $P(X | E)$ given a *training sample* $\{(x_1, e_1), \ldots, (x_n, e_n)\}$
Parameter learning: complete data

• Suppose we know the network structure (but not the parameters), and have a training set of complete observations
Parameter learning

• Suppose we know the network structure (but not the parameters), and have a training set of complete observations

• Example:

\[ P(S = T | C = T) = \frac{\text{#samples with } S = T, C = T}{\text{# samples with } C = T} = \frac{1}{4} \]

<table>
<thead>
<tr>
<th>Sample</th>
<th>C</th>
<th>S</th>
<th>R</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>2</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>3</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>4</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>5</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>6</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Parameter learning

• Suppose we know the network structure (but not the parameters), and have a training set of *complete* observations
  • $P(X \mid \text{Parents}(X))$ is given by the observed frequencies of the different values of $X$ for each combination of parent values
Parameter learning: missing data

- Suppose we know the network structure (but not the parameters), and have a training set, but the training set is *missing some observations*.

### Training set

<table>
<thead>
<tr>
<th>Sample</th>
<th>C</th>
<th>S</th>
<th>R</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>?</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>2</td>
<td>?</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>3</td>
<td>?</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>4</td>
<td>?</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>5</td>
<td>?</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>6</td>
<td>?</td>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Missing data: the EM algorithm

- The EM algorithm starts (“Expectation Maximization”) starts with an initial guess for each parameter value.
- We try to improve the initial guess, using the algorithm on the next two slides:
  - E-step
  - M-step

```
Training set

<table>
<thead>
<tr>
<th>Sample</th>
<th>C</th>
<th>S</th>
<th>R</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>?</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>2</td>
<td>?</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>3</td>
<td>?</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>4</td>
<td>?</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>5</td>
<td>?</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>6</td>
<td>?</td>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
```
Missing data: the EM algorithm

• E-Step (Expectation): Given the model parameters, replace each of the missing numbers with a probability (a number between 0 and 1) using

\[
P(C = 1|S, R, W) = \frac{P(C = 1, S, R, W)}{P(C = 1, S, R, W) + P(C = 0, S, R, W)}
\]

Training set

<table>
<thead>
<tr>
<th>Sample</th>
<th>C</th>
<th>S</th>
<th>R</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.5?</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>2</td>
<td>0.5?</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>3</td>
<td>0.5?</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>4</td>
<td>0.5?</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>5</td>
<td>0.5?</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>6</td>
<td>0.5?</td>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>…</td>
<td>…</td>
<td>…</td>
<td>…</td>
<td>…</td>
</tr>
</tbody>
</table>
Missing data: the EM algorithm

- M-Step (Maximization): Given the missing data estimates, replace each of the missing model parameters using

\[ P(\text{Variable} = T | \text{Parents} = \text{value}) = \frac{E[\# \text{ times Variable} = T, \text{Parents} = \text{value}]}{E[\# \text{times Parents} = \text{value}]} \]
Missing data: the EM algorithm

- Iterate back and forth between E-step and M-step until the model converges.

<table>
<thead>
<tr>
<th>Sample</th>
<th>C</th>
<th>S</th>
<th>R</th>
<th>W</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.5?</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>2</td>
<td>0.5?</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>3</td>
<td>0.5?</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>4</td>
<td>0.5?</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>5</td>
<td>0.5?</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>6</td>
<td>0.5?</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Summary: Bayesian networks

- Structure
- Parameters
- Inference
- Learning