Topics of This Lecture

- Graphical Models
  - Introduction
- Directed Graphical Models (Bayesian Networks)
  - Notation
  - Conditional probabilities
  - Computing the joint probability
  - Factorization
  - Conditional Independence
  - D-Separation
  - Explaining away
- Outlook: Inference in Graphical Models

Graphical Models - What and Why?

- It’s got nothing to do with graphics!
- Probabilistic graphical models
  - Marriage between probability theory and graph theory.
  - Formalize and visualize the structure of a probabilistic model through a graph.
  - Give insights into the structure of a probabilistic model.
  - Find efficient solutions using methods from graph theory.
- Natural tool for dealing with uncertainty and complexity.
- Becoming increasingly important for the design and analysis of machine learning algorithms.
- Often seen as new and promising way to approach problems related to Artificial Intelligence.

Graphical Models

- There are two basic kinds of graphical models
  - Directed graphical models or Bayesian Networks
  - Undirected graphical models or Markov Random Fields

Key components

- Nodes
- Edges
  - Directed or undirected

Directed graphical model
Undirected graphical model

Topics of This Lecture

- Graphical Models
  - Introduction
- Directed Graphical Models (Bayesian Networks)
  - Notation
  - Conditional probabilities
  - Computing the joint probability
  - Factorization
  - Conditional Independence
  - D-Separation
  - Explaining away
- Outlook: Inference in Graphical Models
**Example: Wet Lawn**

- Mr. Holmes leaves his house.
  - He sees that the lawn in front of his house is wet.
  - This can have several reasons: Either it rained, or Holmes forgot to shut the sprinkler off.
  - Without any further information, the probability of both events (rain, sprinkler) increases (knowing that the lawn is wet).

- Now Holmes looks at his neighbor’s lawn
  - The neighbor’s lawn is also wet.
  - This information increases the probability that it rained. And it lowers the probability for the sprinkler.

⇒ How can we encode such probabilistic relationships?

**Directed Graphical Models**

- **Directed graphical model / Bayesian network:**
  - “Rain can cause both lawns to be wet.”
  - “Holmes’ lawn may be wet due to his sprinkler, but his neighbor’s lawn may not.”

**Directed Graphical Models**

- **Nodes or random variables**
  - We usually know the range of the random variables.
  - The value of a variable may be known or unknown.
  - If they are known (observed), we usually shade the node:
    - unknown
    - known

- **Examples of variable nodes**
  - Binary events: Rain (yes / no), sprinkler (yes / no)
  - Discrete variables: Ball is red, green, blue, ...
  - Continuous variables: Age of a person, ...

**Directed Graphical Models**

- **Most often, we are interested in quantitative statements**
  - i.e. the probabilities (or densities) of the variables.
  - Example: What is the probability that it rained? ...

  - These probabilities change if we have
    - more knowledge,
    - less knowledge, or
    - different knowledge
  about the other variables in the network.

- **Simplest case:**

  - This model encodes
    - The value of $b$ depends on the value of $a$.
    - This dependency is expressed through the conditional probability:
      \[ p(b|a) \]
    - Knowledge about $a$ is expressed through the prior probability:
      \[ p(a) \]
    - The whole graphical model describes the joint probability of $a$ and $b$:
      \[ p(a, b) = p(b|a)p(a) \]
If we have such a representation, we can derive all other interesting probabilities from the joint.

- E.g. marginalization

\[ p(a) = \sum_b p(a, b) = \sum_b p(b|a)p(a) \]
\[ p(b) = \sum_a p(a, b) = \sum_a p(b|a)p(a) \]

- With the marginals, we can also compute other conditional probabilities:

\[ p(a|b) = \frac{p(a, b)}{p(b)} \]

With the marginals, we can also compute other conditional probabilities:

\[ p(a|b) = \frac{p(a, b)}{p(b)} \]

Example

As before, we can compute

\[ p(a, b) = p(b|a)p(a) \]

But we can also compute the joint distribution of all three variables:

\[ p(a, b, c) = p(c|a, b)p(a, b) = p(c|b)p(b|a)p(a) \]

We can read off from the graphical representation that variable c does not depend on a, if b is known.

How? What does this mean?

Directed Graphical Models

- Chains of nodes:

\[ a \rightarrow b \rightarrow c \]

Directed Graphical Models

- Convergent connections:

\[ a \rightarrow b \rightarrow c \]

Directed Graphical Models

- Example:

\[ p(C) \]
\[ p(S|C) \]
\[ p(R|C) \]
\[ p(W|R, S) \]

Let’s see what such a Bayesian network could look like:

- Structure?
- Variable types? Binary.
- Conditional probabilities?

Directed Graphical Models

- A general directed graphical model (Bayesian network) consists of

  - A set of variables: \( U = \{x_1, \ldots, x_n\} \)
  - A set of directed edges between the variable nodes.
  - The variables and the directed edges define an acyclic graph.
  - Acyclic means that there is no directed cycle in the graph.
  - For each variable \( x_i \) with parent nodes \( p_a_i \) in the graph, we require knowledge of a conditional probability:

\[ p(x_i|\{x_j|j \in p_a_i\}) \]
**Directed Graphical Models**

- Given
  - Variables: \( U = \{x_1, \ldots, x_n\} \)
  - Directed acyclic graph: \( G = (V,E) \)
    - \( V \): nodes = variables, \( E \): directed edges
  - We can express / compute the joint probability as
    \[
    p(x_1, \ldots, x_n) = \prod_{i=1}^{n} p(x_i|\{x_j | j \in \text{pa}_i\})
    \]
    where \( \text{pa}_i \) denotes the parent nodes of \( x_i \).
  - We can express the joint as a product of all the conditional distributions from the parent-child relations in the graph.
  - We obtain a factorized representation of the joint.

**Exercise: Computing the joint probability**

\[
p(x_1, \ldots, x_7) = ?
\]

\[
p(x_1)p(x_2)p(x_3)p(x_4|x_1, x_2, x_3)
p(x_5|x_1, x_3)p(x_6|x_4) \ldots
\]
Exercise: Computing the joint probability

\[ p(x_1, \ldots, x_T) = p(x_1)p(x_2)p(x_3|x_1, x_2, x_3) \]
\[ p(x_5|x_1, x_3)p(x_6|x_4)p(x_7|x_4, x_5) \]

General factorization

\[ p(x) = \prod_{k=1}^{N} p(x_k | pa_k) \]

We can directly read off the factorization of the joint from the network structure!

Example: Classifier Learning

- Bayesian classifier learning
  - Given \( N \) training examples \( x = (x_0, \ldots, x_N) \) with target values \( t \)
  - We want to optimize the classifier \( y \) with parameters \( w \).
  - We can express the joint probability of \( t \) and \( w \):
    \[ p(t, w) = p(w) \prod_{n=1}^{N} p(t_n|w, x_n) \]
  - Corresponding Bayesian network:

```
  w
  \[ \sim \text{"Plate"} \]
  \[ \text{(short notation for } N \text{ copies)} \]
```

Conditional Independence

- Suppose we have a joint density with 4 variables.
  
  \[ p(x_0, x_1, x_2, x_3) \]

  - For example, 4 subsequent words in a sentence:
    \( x_0 = \text{"Machine"}, \ x_1 = \text{"learning"}, \ x_2 = \text{"is"}, \ x_3 = \text{"fun"} \)
  - The product rule tells us that we can rewrite the joint density:
    
    \[ p(x_0, x_1, x_2, x_3) = p(x_0)p(x_1|x_0)p(x_2|x_0, x_1)p(x_3|x_0, x_1, x_2) \]
    
    \[ = p(x_0)p(x_1|x_0)p(x_2|x_1)p(x_3|x_2) \]
    
    \[ = p(x_0)p(x_1|x_0)p(x_2|x_1)p(x_3|x_2) \]

Conditional Independence

- The notion of conditional independence means that
  - Given a certain variable, other variables become independent.
    
    - More concretely here:
      
      \[ p(x_5|x_0, x_1, x_2) = p(x_5|x_2) \]
      
      - This means that \( x_5 \) is conditionally independent from \( x_0 \) and \( x_1 \) given \( x_2 \).
      
      \[ p(x_7|x_4, x_5) = p(x_7|x_4) \]
      
      - This means that \( x_7 \) is conditionally independent from \( x_5 \), given \( x_4 \).
      
      - Why is this?
        
        \[ p(x_7|x_4, x_5) = p(x_7|x_4)p(x_5|x_1) \]
        
        - \( p(x_7|x_4) = p(x_7|x_4) \)
        
        - \( p(x_5|x_1) \)
        
        - Independent given \( x_4 \).
Conditional Independence - Notation

- \( X \) is conditionally independent of \( Y \) given \( V \)
  - Equivalence: \( X \perp Y | V \Leftrightarrow p(X|Y,V) = p(X|V) \)
  - Also: \( X \perp Y | V \Leftrightarrow p(X,Y|V) = p(X|V)p(Y|V) \)
  - Special case: Marginal Independence
    \( X \perp Y \Leftrightarrow X \perp Y | \emptyset \Leftrightarrow p(X,Y) = p(X)p(Y) \)
  - Often, we are interested in conditional independence between sets of variables:
    \( X \perp Y | V \Leftrightarrow \{X \perp Y | V, \forall X \in X \text{ and } \forall Y \in Y \} \)

Conditional Independence

- Directed graphical models are not only useful...
  - Because the joint probability is factorized into a product of simpler conditional distributions.
  - But also, because we can read off the conditional independence of variables.
- Let’s discuss this in more detail...

First Case: Divergent (“Tail-to-Tail”)

- Divergent model
  
  \[
  p(a, b) = \sum_c p(a, b, c) = \sum_c p(a,c)p(b|c)p(c)
  \]
  - In general, this is not equal to \( p(a)p(b) \).
  - The variables are not independent.

Second Case: Chain (“Head-to-Tail”)

- Let us consider a slightly different graphical model:
  
  \[
  p(a, b, c) = \sum_c p(a, c)p(b|c)p(c) = p(a)p(b|c)
  \]
  - If \( c \) becomes known, are \( a \) and \( b \) conditionally independent? Yes!
  
  \[
  p(a, b|c) = \frac{p(a, b, c)}{p(c)} = \frac{p(a,c)p(b|c)p(c)}{p(c)} = p(a)p(b|c)
  \]
Perceptual and Sensory Augmented Computing
Machine Learning, WS 13/14

Third Case: Convergent (“Head-to-Head”)
• Let’s look at a final case: Convergent graph
  > Are a and b independent? **YES**!
  > $p(a, b) = \sum_c p(a, b, c) = \sum_c p(c|a, b)p(a)p(b) = p(a)p(b)$
  > This is very different from the previous cases.
  > Even though a and b are connected, they are independent.

Summary: Conditional Independence
• Three cases
  > Divergent (“Tail-to-Tail”)
    > Conditional independence when c is observed.
  > Chain (“Head-to-Tail”)
    > Conditional independence when c is observed.
  > Convergent (“Head-to-Head”)
    > Conditional independence when neither c, nor any of its descendants are observed.

D-Separation
• Definition
  > Let A, B, and C be non-intersecting subsets of nodes in a directed graph.
  > A path from A to B is blocked if it contains a node such that either
    > The arrows on the path meet either head-to-tail or tail-to-tail at the node, and the node is in the set C, or
    > The arrows meet head-to-head at the node, and neither the node, nor any of its descendants, are in the set C.
  > If all paths from A to B are blocked, A is said to be d-separated from B by C.
• If A is d-separated from B by C, the joint distribution over all variables in the graph satisfies $A \perp B \mid C$.
  > Read: “A is conditionally independent of B given C.”

D-Separation: Example
• Exercise: What is the relationship between a and b?

Explaining Away
• Let’s look at Holmes’ example again:

 – Observation “Holmes’ lawn is wet” increases the probability of both “Rain” and “Sprinkler.”
Explaining Away

- Let’s look at Holmes’ example again:

  - Observation “Holmes’ lawn is wet” increases the probability of both “Rain” and “Sprinkler”.
  - Also observing “Neighbor’s lawn is wet” decreases the probability for “Sprinkler”. (They’re conditionally dependent!)
  - The “Sprinkler” is explained away.

The “Bayes Ball” Algorithm

- Game rules
  - An unobserved node ($W \notin V$) passes through balls from parents, but also bounces back balls from children.
  - An observed node ($W \in V$) bounces back balls from parents, but blocks balls from children.
  - The Bayes Ball algorithm determines those nodes that are $d$-separated from the query node.

Example: Bayes Ball

- Which nodes are $d$-separated from $G$ given $C$ and $D$?

Intuitive View: The “Bayes Ball” Algorithm

- Game
  - Can you get a ball from $X$ to $Y$ without being blocked by $\bar{Y}$?
  - Depending on its direction and the previous node, the ball can
    - Pass through (from parent to all children, from child to all parents)
    - Bounce back (from any parent/child to all parents/children)
    - Be blocked

Example: Bayes Ball

- Which nodes are $d$-separated from $G$ given $C$ and $D$?
Example: Bayes Ball

- Which nodes are d-separated from \( G \) given \( C \) and \( D \)?

Rule:

- \( F \) is d-separated from \( G \) given \( C \) and \( D \).

The Markov Blanket

- Markov blanket of a node \( x_i \):
  - Minimal set of nodes that isolates \( x_i \) from the rest of the graph.
  - This comprises the set of:
    - Parents,
    - Children, and
    - Co-parents of \( x_i \).

This is what we have to watch out for!

Summary

- Graphical models:
  - Marriage between probability theory and graph theory.
  - Give insights into the structure of a probabilistic model.
    - Direct dependencies between variables.
    - Conditional independence
  - Allow for efficient factorization of the joint.
    - Factorization can be read off directly from the graph.
    - Capability to explain away hypotheses by new evidence.

- Next lecture:
  - Undirected graphical models (Markov Random Fields)
  - Efficient methods for performing exact inference.

References and Further Reading

- A thorough introduction to Graphical Models in general and Bayesian Networks in particular can be found in Chapter 8 of Bishop’s book.

Christopher M. Bishop
Pattern Recognition and Machine Learning
Springer, 2006