Advanced Machine Learning
Summer 2019

Part 7 – Graphical Models I
25.04.2019

Prof. Dr. Bastian Leibe
RWTH Aachen University, Computer Vision Group
http://www.vision.rwth-aachen.de

Many slides adapted from B. Schiele, S. Roth

Course Outline

• Regression Techniques
  - Linear Regression
  - Regularization (Ridge, Lasso)
  - Kernels (Kernel Ridge Regression)

• Deep Reinforcement Learning

• Probabilistic Graphical Models
  - Bayesian Networks
  - Markov Random Fields
  - Inference (exact & approximate)

• Deep Generative Models
  - Generative Adversarial Networks
  - Variational Autoencoders

Topics of This Lecture

• Probabilistic Graphical Models
  - Introduction

• Directed Graphical Models (Bayesian Networks)
  - Notation
  - Conditional probabilities
  - Computing the joint probability
  - Factorization
  - Conditional Independence
  - D-Separation
  - Explaining away

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.
  - Has become an important way of designing and analyzing machine learning algorithms.

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

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
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

• or Bayesian networks
  – Are based on a directed graph.
  – The nodes correspond to the random variables.
  – The directed edges correspond to the (causal) dependencies among the variables.
  – The notion of a causal nature of the dependencies is somewhat hard to grasp.
  – We will typically ignore the notion of causality here...
  – The structure of the network qualitatively describes the dependencies of the random variables.

• 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.

Directed Graphical Models

• Simplest case:
  \[ p(a, b) = p(b|a)p(a) \]

• 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 \( b \):
Directed Graphical Models

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

- We start with the simple product rule:
  
  \[
  p(x_1, x_2, x_3) = p(x_1)p(x_2|x_1)p(x_3|x_1, x_2)
  \]

Directed Graphical Models

- Chains of nodes:
  
  \[
  a \rightarrow b \rightarrow c
  \]

  - As before, we can compute
    
    \[
    p(a, b) = p(b) \cdot p(a)
    \]

  - 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

- Convergent connections:
  
  \[
  a \rightarrow b \rightarrow c
  \]

  - Here the value of \(c\) depends on both variables \(a\) and \(b\).
  - This is modeled with the conditional probability:
    
    \[
    p(c|a, b)
    \]

  - Therefore, the joint probability of all three variables is given as:
    
    \[
    p(a, b, c) = p(c|a, b)p(a, b) = p(c|a, b)p(a)p(b)
    \]

Example

- Evaluating the Bayesian network:
  
  \[
  p(S) = \begin{cases} 0.5 & \text{Cloudy} \\ 0.5 & \text{Rain} \end{cases}
  \]

  \[
  p(R) = \begin{cases} 0.8 & \text{Sprinkler} \\ 0.2 & \text{Wet grass} \end{cases}
  \]

  \[
  p(W|S, R) = \begin{cases} 1.0 & \text{F} \\ 0.0 & \text{T} \end{cases}
  \]

  \[
  p(C) = \begin{cases} 0.5 & \text{F} \end{cases}
  \]

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

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 \(p_a\), in the graph, we require knowledge of a conditional probability:
    
    \[
    p(x_i|\{x_j|j \in p_a\})
    \]
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

$$p(x_1, \ldots, x_n) = \prod_{i=1}^{n} p(x_i|\text{pa}_i)$$

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:

$$U = f_{x_1; \ldots; x_n}$$

$G = (V; E)$

Exercise: Computing the joint probability

$$p(x_1, \ldots, x_7) = ?$$

$p(x_1; x_2; x_3) \cdot p(x_4; x_1; x_2; x_3) \cdot p(x_5; x_1; x_3) \cdot p(x_6; x_4) \cdot \ldots$

Exercise: Computing the joint probability

$$p(x_1, \ldots, x_7) = p(x_1)p(x_2)p(x_3)p(x_4|x_1; x_2; x_3)\ldots$$

Exercise: Computing the joint probability

$$p(x_1, \ldots, x_7) = p(x_1)p(x_2)p(x_3)p(x_4|x_1; x_2; x_3)\ldots$$

Exercise: Computing the joint probability

$$p(x_1, \ldots, x_7) = p(x_1)p(x_2)p(x_3)p(x_4|x_1; x_2; x_3)\ldots$$
This means that 

\[ \text{Schiele} \]

\[ \text{given} \]

\[ \text{or} \]

\[ \text{We can express the joint probability of} \]

\[ \text{The product rule tells us that we can rewrite the joint density:} \]

\[ \text{For example, 4 subsequent words in a sentence:} \]

\[ \text{We want to optimize the classifier} \]

\[ \text{More concretely here:} \]

\[ \text{is conditionally independent from} \]

\[ \text{E.g.} \]

\[ \text{Given} \]

\[ \text{They are directly reflected in the structure of the graphical model.} \]

\[ \text{Example: Classifier Learning} \]

\[ \text{Bayesian classifier learning} \]

\[ \text{– Given} \]

\[ \text{– We want to optimize the classifier} \]

\[ \text{– We can express the joint probability of} \]

\[ \text{– Corresponding Bayesian network:} \]

\[ \text{Short notation:} \]

\[ \text{N} \]

\[ \text{– “Plate” for} \]

\[ \text{– “Plate” copies} \]

\[ \text{Conditional Independence} \]

\[ \text{Suppose we have a joint density with 4 variables.} \]

\[ \text{– For example, 4 subsequent words in a sentence:} \]

\[ \text{– The product rule tells us that we can rewrite the joint density:} \]

\[ \text{The notion of conditional independence means that} \]

\[ \text{– Given a certain variable, other variables become independent.} \]

\[ \text{– More concretely here:} \]

\[ \text{• This means that} \]

\[ \text{• This means that} \]

\[ \text{Why is this?} \]
Because the joint probability is factorized into a product of simpler distributions:

\[ p(X) = \prod_i p(a_i; b_i; c_i) \]

Are \( a \) and \( b \) independent?

- Marginalize out \( c \):
  \[
  p(a, b) = \sum_c p(a, b, c) = \sum_c p(a|c)p(b|c)p(c)
  \]

- In general, this is not equal to \( p(a)p(b) \).

\( \Rightarrow \) The variables are not independent.

First Case: Divergent ("Tail-to-Tail")

- Let's return to the original graph, but now assume that we observe the value of \( c \).
  \[
  p(a, b|c) = \frac{p(a, b, c)}{p(c)} = \frac{p(a|c)p(b|c)p(c)}{p(c)} = p(a|c)p(b|c)
  \]

\( \Rightarrow \) If \( c \) becomes known, the variables \( a \) and \( b \) become conditionally independent.

Second Case: Chain ("Head-to-Tail")

- Let us consider a slightly different graphical model:
  \[
  p(a, b|c) = \frac{p(a, b, c)}{p(c)} = \frac{p(a)p(c|a)p(b|c)}{p(c)} = p(a)p(c|a)p(b|c)
  \]

- If \( c \) becomes known, are \( a \) and \( b \) conditionally independent?

  \( \Rightarrow \) Yes!
Third Case: Convergent ("Head-to-Head")

• Let's look at a final case: Convergent graph
  - Are a and b independent?
  
  \[
p(a, b) = \sum_c p(a, b, c) = \sum_c p(c|a, b)p(a)p(b) = p(a)p(b)
  \]
  - This is very different from the previous cases.
  - Even though a and b are connected, they are independent.

Third Case: Convergent ("Head-to-Head")

• Now we assume that c is observed
  - Are a and b independent?
  
  \[
p(a, b|c) = \frac{p(a, b, c)}{p(c)} = \frac{p(a)p(b)p(c|a, b)}{p(c)} = p(a)p(b)
  \]
  - In general, they are not conditionally independent.
  - This also holds when any of c's descendants is observed.
  - This case is the opposite of the previous cases!

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.

Intuitive View: The "Bayes Ball" Algorithm

- Game
  - Can you get a ball from X to Y without being blocked by V?
  - 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

  R.D. Shachter, Bayes-Ball: The Rational Pastime (for Determining Irrelevance and Requisite Information in Belief Networks and Influence Diagrams), UAI98, 1998

The "Bayes Ball" Algorithm

- Game rules
  - An unobserved node (W  V) passes through balls from parents, but also bounces back balls from children.
  - An observed node (W ∈ 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?

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:

$\bullet$ Query

Example: Bayes Ball

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

Rule:

$\bullet$ Query

Example: Bayes Ball

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

Rule:

$\bullet$ Query

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!

The Markov Blanket

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.
    - We will use this for efficient inference algorithms!
    - 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