T-61.5140 Machine Learning: Advanced Probablistic Methods

Jaakko Hollmén

Department of Information and Computer Science
Helsinki University of Technology, Finland
e-mail: Jaakko.Hollmen@tkk.fi
Web: http://www.cis.hut.fi/Opinnot/T-61.5140/

February 13, 2008

So far on the course...

Random variables and statistical independence

- Random variables and probability distributions
- Independence and conditional independence
- Bayes's rule

Bayesian Networks

- Factorization of the joint probability distribution
- Graphical representation
- Markov Blanket and d-separation
- Inference as an application of the Bayes's rule

Other sources for the interested

Textbooks on Bayesian Networks:

- Finn V. Jensen and Thomas D. Nielsen: Bayesian Networks and Decision Graphs, Second edition, Springer-Verlag, 2007.
- Richard E. Neapolitan: Learning Bayesian Networks, Prentice-Hall, 2004.
- Learning in Graphical Models, edited by Michael Jordan, MIT Press, 1999.

Graphical models, everything in condensed form:

- Michael I. Jordan: Graphical Models, *Statistical Science*, voi 19, No. 1, pp 140–155, `http://dx.doi.org/10.1214/088342304000000026`

Inference in Bayesian Networks

$$P(A,B,C,D,E) = P(A)P(B|A)P(C|A)P(D|B,C)P(E|C)$$

What is $P(A|C=c, E=e) = \frac{\sum_B \sum_D P(A,B,c,D,e)}{P(c,e)}$?

$$\sum_B \sum_D P(A,B,c,D,e) =$$

$$\sum_B \sum_D P(e|c)P(c|A)P(D|B,c)P(A)P(B|A)$$

$$= P(e|c)P(c|A)P(A) \sum_B \sum_D P(D|B,c)P(B|A)$$

$$= P(e|c)P(c|A)P(A) \sum_B P(B|A) \sum_D P(D|B,c)$$

$$= P(e|c)P(c|A)P(A).$$

Variable elimination

Formalization of the previous inference process

- We have a set of probability tables $\mathcal{T}$
- We wish to marginalize a variable X:
    - Take all tables from $\mathcal{T}$ that include X
    - Calculate a product of them
    - Marginalize x out of it
    - Place the resulting table in $\mathcal{T}$

In the example, we marginalized first over B, then D

Variable elimination

Variable elimination

- Marginalization eliminates variables from tables
- Elimination order has a large impact on the complexity of the algorithm
- Rina Dechter: Bucket Elimination, *Artificial Intelligence*, 1999
- `http://dx.doi.org/10.1016/S0004-3702(99)00059-4`

Inference in Bayesian Networks

Inference: having some observed variables, we wish to compute the posterior probability of some other variables

- Variable elimination is one of the simplest *exact inference* algortihms
- Variable elimination works directly on variables and probability tables
- Next: Secondary representation based on the directed graph (assumptions); local message passing algorithms

Cliques and potential functions

Joint distribution may be written as a product of *potential functions* of sets of variables $\mathbf{x}_C$:

- $p(\mathbf{x}) = \frac{1}{Z} \prod_C \psi_C(\mathbf{x}_c)$
- $\psi_C(\mathbf{x}_c) \geq 0 \Rightarrow p(x) \geq 0$
- Partition function $Z = \sum_{\mathbf{x}} \prod_C \psi_C(\mathbf{x}_c)$
- Partition function is a normalization constant
- Partition function may be difficult to compute!
- In calculating local conditional distributions (a ratio of two distributions), Z cancels out
- $\mathbf{x}_C$ need to be maximal cliques of the directed graph

Clique is a subset of nodes such that there exists a link between all pairs of nodes in the subset

Hammersley-Clifford theorem

Consider

- set of distributions that are consistent with the set of conditional independence statements that can be read from the graph using graph separation
- set of distributions that can be expressed as a factorization (as on the previous slide) with respect to maximal cliques of the graph

Hammersley-Clifford theorem states the sets are identical

Example: directed and undirected chain

A Markov chain in directed and undirected form:

- $p(\mathbf{x}) = p(x_1)p(x_2|x_1)p(x_3|x_2)\dots p(x_N|x_{N-1})$
- The elements are conditional probability distributions
- Maximal cliques are the pairs of neighboring nodes
- $p(\mathbf{x}) = \frac{1}{Z}\psi_{1,2}(x_1, x_2)\psi_{2,3}(x_2, x_3)\dots\psi_{N-1,N}(x_{N-1}, x_N)$
- The elements don't necessarily have probabilistic implementation

We need operations (or a "recipe") to convert a factorization over a directed graph to that of an undirected graph!

Recipe: Constructing a Junction Tree

Take the directed acyclic graph and

- Moralize: Marry the parents with undirected edges
- Drop the direction of all arrows (moral graph)
- Triangulate the graph: find chordless cycles containing four or more nodes, add links to eliminate such cycles
- Construct a tree-structured undirected graph: nodes are maximal cliques of the triangulated graph
- Connect pairs of cliques that have variables in common

Now we have a junction tree, a secondary representation for the original Bayesian network that allows simple message passing algorithms

About the junction tree

- Multiple junction trees can be created from a given starting position
- The size of the largest clique determines the complexity of the inference procedure (treewidth)
- Treewidth is the (smallest) number of variables in the largest clique minus one
- For trees, treewidth is one (simple inference)
- What kind of models have large treewidth?