

Readings:

K&F: 9.1, 9.2, 9.3, 9.4

K&F: 5.1, 5.2, 5.3, 5.4, 5.5, 5.6

Clique Trees 3

Let's get BP right

Undirected Graphical Models

Here the couples get to swing!

Graphical Models – 10708

Carlos Guestrin

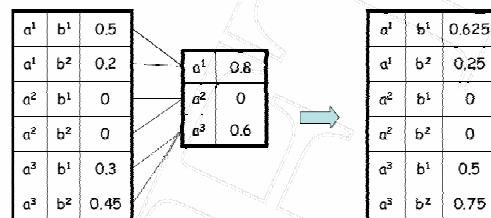
Carnegie Mellon University

October 25th, 2006

1

Factor division

- Let \mathbf{X} and \mathbf{Y} be disjoint set of variables
- Consider two factors:
 $\phi_1(\mathbf{X}, \mathbf{Y})$ and $\phi_2(\mathbf{Y})$
- Factor $\psi = \phi_1 / \phi_2$
 - $0/0=0$



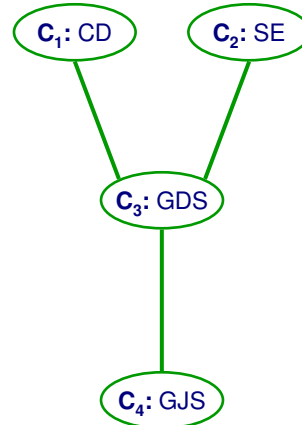
Introducing message passing with division

- Variable elimination (message passing with multiplication)

- message:
- belief:

- Message passing with division:

- message:
- belief update:



10.708 - ©Carlos Guestrin 2006

3

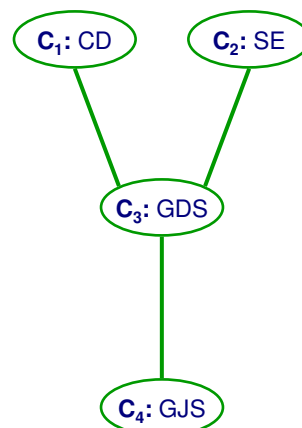
Lauritzen-Spiegelhalter Algorithm (a.k.a. belief propagation)

- Separator potentials μ_{ij}

- one per edge (same both directions)
- holds "last message"
- initialized to 1

- Message $i \rightarrow j$

- what does i think the separator potential should be?
 - $\sigma_{i \rightarrow j}$
- update belief for j :
 - pushing j to what i thinks about separator
- replace separator potential:



10.708 - ©Carlos Guestrin 2006

4

Clique tree invariant

- **Clique tree potential:**

- Product of clique potentials divided by separators potentials

- **Clique tree invariant:**

- $P(\mathbf{X}) = \pi_T(\mathbf{X})$

10.708 – ©Carlos Guestrin 2006

5

Belief propagation and clique tree invariant

- **Theorem:** Invariant is maintained by BP algorithm!

- BP reparameterizes clique potentials and separator potentials

- At convergence, potentials and messages are marginal distributions

10.708 – ©Carlos Guestrin 2006

6

Subtree correctness

- **Informed message** from i to j , if all messages into i (other than from j) are informed
 - Recursive definition (leaves always send informed messages)
- **Informed subtree:**
 - All incoming messages informed
- **Theorem:**
 - Potential of connected informed subtree T' is marginal over $\text{scope}[T']$
- **Corollary:**
 - At convergence, clique tree is *calibrated*
 - $\pi_i = P(\text{scope}[\pi_i])$
 - $\mu_{ij} = P(\text{scope}[\mu_{ij}])$

10.708 – ©Carlos Guestrin 2006

7

Clique trees versus VE

- **Clique tree advantages**
 - Multi-query settings
 - Incremental updates
 - Pre-computation makes complexity explicit
- **Clique tree disadvantages**
 - Space requirements – no factors are “deleted”
 - Slower for single query
 - Local structure in factors may be lost when they are multiplied together into initial clique potential

10.708 – ©Carlos Guestrin 2006

8

Clique tree summary

- Solve marginal queries for all variables in only twice the cost of query for one variable
- Cliques correspond to maximal cliques in induced graph
- Two message passing approaches
 - VE (the one that multiplies messages)
 - BP (the one that divides by old message)
- Clique tree invariant
 - Clique tree potential is always the same
 - We are only reparameterizing clique potentials
- Constructing clique tree for a BN
 - from elimination order
 - from triangulated (chordal) graph
- Running time (only) exponential in size of largest clique
 - Solve **exactly** problems with thousands (or millions, or more) of variables, and cliques with tens of nodes (or less)

10.708 – ©Carlos Guestrin 2006

9

Announcements

- Recitation tomorrow, don't miss it!!!
 - Khalid on Undirected Models

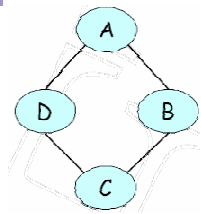
10.708 – ©Carlos Guestrin 2006

10

Swinging Couples revisited

- This is no perfect map in BNs
- But, an undirected model will be a perfect map

Potentials (or Factors) in Swinging Couples



$\pi_1[A, B]$			$\pi_2[B, C]$			$\pi_3[C, D]$			$\pi_4[D, A]$		
a^0	b^0	30	b^0	c^0	100	c^0	d^0	1	d^0	a^0	100
a^0	b^1	5	b^0	c^1	1	c^0	d^1	100	d^0	a^1	1
a^1	b^0	1	b^1	c^0	1	c^1	d^0	100	d^1	a^0	1
a^1	b^1	10	b^1	c^1	100	c^1	d^1	1	d^1	a^1	100

Computing probabilities in Markov networks v. BNs

- In a BN, can compute prob. of an instantiation by multiplying CPTs

- In an Markov networks, can only compute ratio of probabilities directly

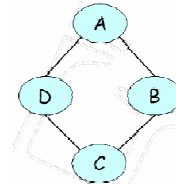
$\pi_1[A, B]$			$\pi_2[B, C]$			$\pi_3[C, D]$			$\pi_4[D, A]$		
a^0	b^0	30	b^0	c^0	100	c^0	d^0	1	d^0	a^0	100
a^0	b^1	5	b^0	c^1	1	c^0	d^1	100	d^0	a^1	1
a^1	b^0	1	b^1	c^0	1	c^1	d^0	100	d^1	a^0	1
a^1	b^1	10	b^1	c^1	100	c^1	d^1	1	d^1	a^1	100

Normalization for computing probabilities

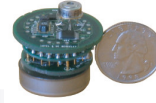
- To compute actual probabilities, must compute normalization constant (also called partition function)

Assignment				Unnormalized	Normalized
a^0	b^0	c^0	d^0	300000	0.01
a^0	b^0	c^0	d^1	300000	0.04
a^0	b^0	c^1	d^0	300000	0.04
a^0	b^0	c^1	d^1	30	$4.1 \cdot 10^{-6}$
a^0	b^1	c^0	d^0	500	$6.9 \cdot 10^{-5}$
a^0	b^1	c^0	d^1	500	$6.9 \cdot 10^{-5}$
a^0	b^1	c^1	d^0	5000000	0.69
a^0	b^1	c^1	d^1	500	$6.9 \cdot 10^{-5}$
a^1	b^0	c^0	d^0	100	$1.4 \cdot 10^{-5}$
a^1	b^0	c^0	d^1	1000000	0.14
a^1	b^0	c^1	d^0	100	$1.4 \cdot 10^{-5}$
a^1	b^0	c^1	d^1	100	$1.4 \cdot 10^{-5}$
a^1	b^1	c^0	d^0	10	$1.4 \cdot 10^{-6}$
a^1	b^1	c^0	d^1	100000	0.014
a^1	b^1	c^1	d^0	100000	0.014
a^1	b^1	c^1	d^1	100000	0.014

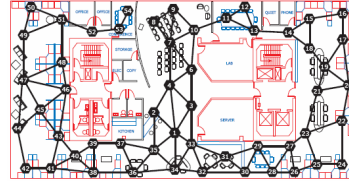
- Computing partition function is hard! → Must sum over all possible assignments



Factorization in Markov networks



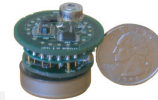
- Given an undirected graph H over variables $\mathbf{X}=\{X_1, \dots, X_n\}$
- A distribution P **factorizes** over H if \exists
 - subsets of variables $\mathbf{D}_1 \subseteq \mathbf{X}, \dots, \mathbf{D}_m \subseteq \mathbf{X}$, such that the \mathbf{D}_i are *fully connected* in H
 - *non-negative potentials* (or factors) $\pi_1(\mathbf{D}_1), \dots, \pi_m(\mathbf{D}_m)$
 - also known as clique potentials
 - such that
- Also called Markov random field H , or Gibbs distribution over H



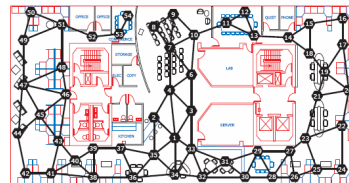
10.708 - ©Carlos Guestrin 2006

15

Global Markov assumption in Markov networks



- A path $X_1 - \dots - X_k$ is **active** when set of variables \mathbf{Z} are observed if none of $X_i \in \{X_1, \dots, X_k\}$ are observed (are part of \mathbf{Z})
- Variables \mathbf{X} are **separated** from \mathbf{Y} given \mathbf{Z} in graph H , $sep_H(\mathbf{X}; \mathbf{Y} | \mathbf{Z})$, if there is no active path between any $X \in \mathbf{X}$ and any $Y \in \mathbf{Y}$ given \mathbf{Z}
- The **global Markov assumption** for a Markov network H is



10.708 - ©Carlos Guestrin 2006

16

The BN Representation Theorem

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

Obtain

Joint probability distribution:

$$P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i | \text{Pa}_{X_i})$$

Important because:
Independencies are sufficient to obtain BN structure G

If joint probability distribution:

$$P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i | \text{Pa}_{X_i})$$

Obtain

Then conditional independencies in BN are subset of conditional independencies in P

Important because:
Read independencies of P from BN structure G

10.708 - ©Carlos Guestrin 2006

17

Markov networks representation Theorem 1

If joint probability distribution P :

$$P(X_1, \dots, X_n) = \frac{1}{Z} \prod_{i=1}^m \pi_i(D_i)$$

Then


H is an I-map for P

- If you can write distribution as a normalized product of factors \Rightarrow Can read independencies from graph

10.708 - ©Carlos Guestrin 2006

18

What about the other direction for Markov networks ?

If H is an I-map for P  Then **joint probability distribution P :**

$$P(X_1, \dots, X_n) = \frac{1}{Z} \prod_{i=1}^m \pi_i(D_i)$$


- Counter-example: X_1, \dots, X_4 are binary, and only eight assignments have positive probability:

(0,0,0,0)	(1,0,0,0)	(1,1,0,0)	(1,1,1,0)
(0,0,0,1)	(0,0,1,1)	(0,1,1,1)	(1,1,1,1)
- For example, $X_1 \perp X_3 | X_2, X_4$:
- But distribution doesn't factorize!!!

10.708 - ©Carlos Guestrin 2006

19

Markov networks representation Theorem 2 (Hammersley-Clifford Theorem)

If H is an I-map for P and P is a positive distribution  Then **joint probability distribution P :**

$$P(X_1, \dots, X_n) = \frac{1}{Z} \prod_{i=1}^m \pi_i(D_i)$$

- Positive distribution and independencies $\Rightarrow P$ factorizes over graph

10.708 - ©Carlos Guestrin 2006

20

Representation Theorem for Markov Networks

If joint probability distribution P :

$$P(X_1, \dots, X_n) = \frac{1}{Z} \prod_{i=1}^m \pi_i(D_i)$$

Then

H is an I-map for P

If H is an I-map for P
and
 P is a positive distribution

Then

joint probability distribution P :

$$P(X_1, \dots, X_n) = \frac{1}{Z} \prod_{i=1}^m \pi_i(D_i)$$

10.708 – ©Carlos Guestrin 2006

21

Completeness of separation in Markov networks

■ Theorem: Completeness of separation

- For “almost all” distributions that P factorize over Markov network H , we have that $I(H) = I(P)$
- “almost all” distributions: except for a set of measure zero of parameterizations of the Potentials (assuming no finite set of parameterizations has positive measure)

■ Analogous to BNs

10.708 – ©Carlos Guestrin 2006

22

What are the “local” independence assumptions for a Markov network?

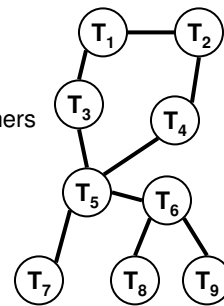
- In a BN G :
 - local Markov assumption: variable independent of non-descendants given parents
 - d-separation defines global independence
 - Soundness: For all distributions:
- In a Markov net H :
 - **Separation** defines global independencies
 - What are the notions of local independencies?

10.708 – ©Carlos Guestrin 2006

23

Local independence assumptions for a Markov network

- **Separation** defines global independencies
- **Pairwise Markov Independence:**
 - Pairs of non-adjacent variables are independent given all others
- **Markov Blanket:**
 - Variable independent of rest given its neighbors



10.708 – ©Carlos Guestrin 2006

24

Equivalence of independencies in Markov networks

- **Soundness Theorem:** For all positive distributions P , the following three statements are equivalent:
 - P entails the global Markov assumptions
 - P entails the pairwise Markov assumptions
 - P entails the local Markov assumptions (Markov blanket)

10-708 – ©Carlos Guestrin 2006

25

Minimal I-maps and Markov Networks

- A fully connected graph is an I-map
- Remember minimal I-maps?
 - A “simplest” I-map → Deleting an edge makes it no longer an I-map
- In a BN, there is no unique minimal I-map
- Theorem: **In a Markov network, minimal I-map is unique!!**
- Many ways to find minimal I-map, e.g.,
 - Take pairwise Markov assumption:
 - If P doesn't entail it, add edge:

10-708 – ©Carlos Guestrin 2006

26

How about a perfect map?

- Remember perfect maps?
 - independencies in the graph are exactly the same as those in P
- For BNs, doesn't always exist
 - counter example: Swinging Couples
- How about for Markov networks?

10.708 – ©Carlos Guestrin 2006

27

Unifying properties of BNs and MNs

- BNs:
 - give you: V-structures, CPTs are conditional probabilities, can directly compute probability of full instantiation
 - but: require acyclicity, and thus no perfect map for swinging couples
- MNs:
 - give you: cycles, and perfect maps for swinging couples
 - but: don't have V-structures, cannot interpret potentials as probabilities, requires partition function
- Remember PDAGS???
 - skeleton + immoralities
 - provides a (somewhat) unified representation
 - see book for details

10.708 – ©Carlos Guestrin 2006

28

What you need to know so far about Markov networks

- Markov network representation:
 - undirected graph
 - potentials over cliques (or sub-cliques)
 - normalize to obtain probabilities
 - need partition function
- Representation Theorem for Markov networks
 - if P factorizes, then it's an I-map
 - if P is an I-map, only factorizes for positive distributions
- Independence in Markov nets:
 - active paths and separation
 - pairwise Markov and Markov blanket assumptions
 - equivalence for positive distributions
- Minimal I-maps in MNs are unique
- Perfect maps don't always exist

10.708 – ©Carlos Guestrin 2006

29

Some common Markov networks and generalizations

- Pairwise Markov networks
- A very simple application in computer vision
- Logarithmic representation
- Log-linear models
- Factor graphs

10.708 – ©Carlos Guestrin 2006

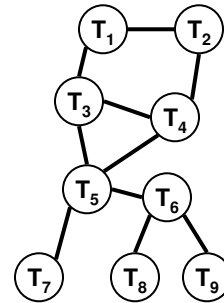
30

Pairwise Markov Networks

- All factors are over single variables or pairs of variables:

- Node potentials
- Edge potentials

- Factorization:



- Note that there may be bigger cliques in the graph, but only consider pairwise potentials

10.708 - ©Carlos Guestrin 2006

31

A very simple vision application

- Image segmentation: separate foreground from background

- Graph structure:

- pairwise Markov net
- grid with one node per pixel



- Node potential:

- "background color" v. "foreground color"

- Edge potential:

- neighbors like to be of the same class

10.708 - ©Carlos Guestrin 2006

32

Logarithmic representation

- Standard model: $P(X_1, \dots, X_n) = \frac{1}{Z} \prod_{i=1}^m \pi_i(\mathbf{D}_i)$
- Log representation of potential (assuming positive potential):
 - also called the energy function
- Log representation of Markov net:

10.708 – ©Carlos Guestrin 2006

33

Log-linear Markov network (most common representation)

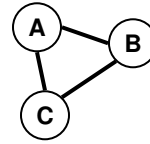
- **Feature** is some function $\phi[\mathbf{D}]$ for some subset of variables \mathbf{D}
 - e.g., indicator function
- **Log-linear model** over a Markov network H :
 - a set of features $\phi_1[\mathbf{D}_1], \dots, \phi_k[\mathbf{D}_k]$
 - each \mathbf{D}_i is a subset of a clique in H
 - two ϕ 's can be over the same variables
 - a set of weights w_1, \dots, w_k
 - usually learned from data
 - $P(X_1, \dots, X_n) = \frac{1}{Z} \exp \left[\sum_{i=1}^k w_i \phi_i(\mathbf{D}_i) \right]$

10.708 – ©Carlos Guestrin 2006

34

Structure in cliques

- Possible potentials for this graph:

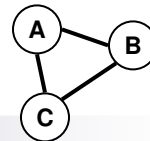


10.708 – ©Carlos Guestrin 2006

35

Factor graphs

- Very useful for approximate inference
 - Make factor dependency explicit
- Bipartite graph:
 - variable nodes (ovals) for X_1, \dots, X_n
 - factor nodes (squares) for ϕ_1, \dots, ϕ_m
 - edge $X_i - \phi_j$ if $X_i \in \text{Scope}[\phi_j]$



10.708 – ©Carlos Guestrin 2006

36

Summary of types of Markov nets

- Pairwise Markov networks

- very common
- potentials over nodes and edges

- Log-linear models

- log representation of potentials
- linear coefficients learned from data
- most common for learning MNs

- Factor graphs

- explicit representation of factors
 - you know exactly what factors you have
- very useful for approximate inference