Graphical models

Review

- Graphical models (Bayes nets, Markov random fields, factor graphs)
  - graphical tests for conditional independence (e.g., d-separation for Bayes nets; Markov blanket)
  - format conversions: always possible, may lose info
  - learning (fully-observed case)
- Inference
  - variable elimination
  - today: belief propagation

Junction tree

(aka clique tree, aka join tree)

- Represents the tables that we build during elimination
  - many JTs for each graphical model
  - many-to-many correspondence w/ elimination orders
- A junction tree for a model is:
  - a tree
  - whose nodes are sets of variables (“cliques”)
  - that contains a node for each of our factors
  - that satisfies *running intersection property*

Running intersection property

- In variable elimination: once a variable X is added to our current table T, it stays in T until eliminated, then never appears again
- In JT, this means all sets containing X form a connected region of tree
  - true for all X = running intersection property

Incorporating evidence (conditioning)

- For each factor or CPT:
  - fix known/observed arguments
  - assign to some clique containing all non-fixed arguments
  - drop observed variables from the JT
- No difference from inference w/o evidence
  - we just get a junction tree over fewer variables
  - easy to check that it's still a valid JT

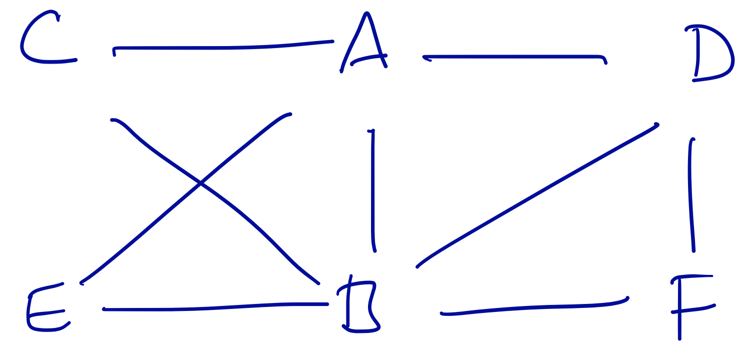
Message passing (aka BP)

- Build a junction tree (started last time)
- Instantiate evidence, pass messages (calibrate), read off answer, eliminate nuisance variables
- Main questions
  - how expensive? (what tables?)
  - what does a message represent?

Example

CEABDF

	S	T	V
0	∅		
1	AB	ABC	C
2	AB	ABE	CE
3	BD	ABD	CEA
4	DF	BDF	CEAB
5	F	DF	
6	∅	F	



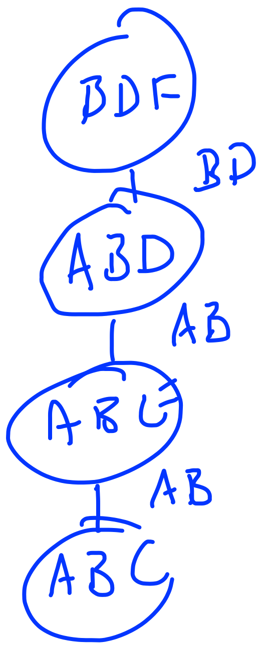
ABC — ABE — ABD — BDF

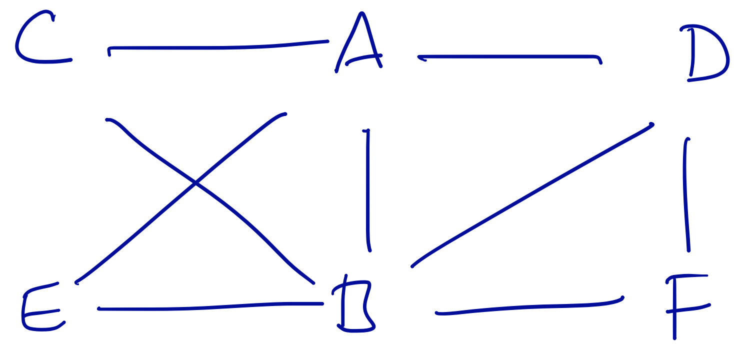
$$\sum_A \sum_B \sum_C \sum_D \sum_E \sum_F \phi_1(ABC)\,\phi_2(ABE)\,\phi_3(ABD)\,\phi_4(BDF)$$

$$\left(\sum_F \sum_D \sum_B \phi_4(BDF)\left(\sum_A \phi_3(ABD)\left(\sum_E \phi_2(ABE)\left(\sum_C \phi_1(ABC)\right)\right)\right)\right)$$

$$\left(\sum_F \sum_D \sum_B \phi_4(BDF)\left(\sum_A \phi_3(ABD)\left(\sum_E \phi_2(ABE)\right)\left(\sum_C \phi_1(ABC)\right)\right)\right)$$

What if order were FDBAEC?





Messages

- Message = smaller tables that we create by summing out some variables from a factor over a clique
  - we later multiply the message into exactly one other clique before summing out that clique
  - one message per edge (e.g., ABC — ABD)
  - arguments of message: intersection of endpoints (AB)
    - called a ***sepset*** or separating set
  - message might go in either direction over the edge depending on which side of the JT we sum out first

Belief propagation

- Idea: calculate all messages that could be passed by any elimination order consistent with our JT
- For each edge, need two runs of variable elimination: one using the edge in each direction
- Insight: that's just two runs total

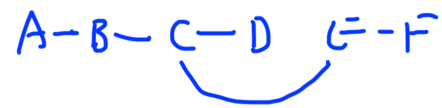
Belief propagation

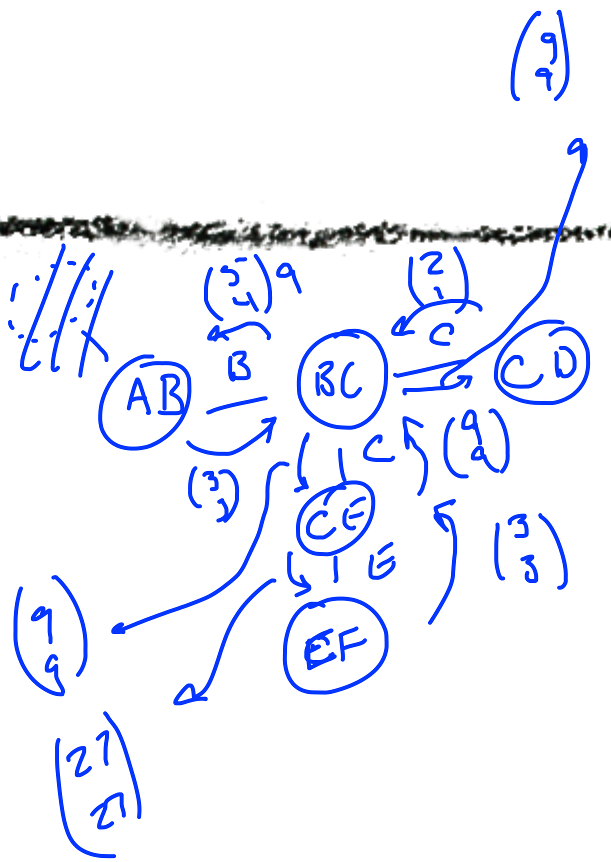
- Pick a node of JT as root arbitrarily
- Run variable elimination inward toward the root
  - any elimination order is OK as long as we do edges farther from the root first
- Run variable elimination outward from the root
  - for each child X of root R, pick an order: [all other children of R], R, X, [everything on non-root side of X]
  - pick up this run with message R→X
- Done!

All for the price of two

- Now we can simulate any order of elimination consistent with the tree:
  - orient JT edges in the direction consistent with the elimination order
  - these are the messages that elimination would compute

Example





$$\begin{array}{c|cc} & T & F \\ \hline T & 2 & 1 \\ F & 1 & 2 \end{array}$$

$D = T$

$CD \rightarrow C \begin{array}{cc} T & 2 \\ F & 1 \end{array}$

$$\begin{pmatrix} 6 & 3 \\ 3 & 6 \end{pmatrix}$$

$$\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \circ \begin{pmatrix} 2 & 1 \\ 2 & 1 \end{pmatrix} \circ \begin{pmatrix} 9 & 9 \\ 9 & 9 \end{pmatrix}$$

$$\begin{pmatrix} 4 & 1 \\ 2 & 2 \end{pmatrix} 9 \rightarrow \begin{pmatrix} 5 \\ 4 \end{pmatrix} 9$$

Kschischang Loeliger Frey

Using it

- Want: P(A, B | D=T)
  - i.e., $\sum_C \sum_E \sum_F P(A\,B\,C\,E\,F \mid D=T)$

    $= \sum_C \sum_E \sum_F \prod \cdots \mid D=T$

    $\sum_A \sum_B \sum_C \sum_E \sum_F \cdots$
- Variable elimination:

  $\begin{pmatrix}5\\4\end{pmatrix} 9$

  $\begin{pmatrix}2&1\\1&2\end{pmatrix} \circ \begin{pmatrix}5&4\\5&4\end{pmatrix} \cdot 9 = \begin{pmatrix}10&4\\5&8\end{pmatrix} \cdot 9 \Big/ Z$

Marginals

- More generally, marginal over any subtree:
  - product of all incoming messages and all local factors
  - normalize
- Special case: clique marginals

Read off answer

- Find some subtree that mentions all variables of interest
- Compute distribution over variables mentioned in this subtree
  - product of all messages into subtree and all factors inside subtree / normalizing constant
- Marginalize (sum out) nuisance variables

Inference—recap

- Build junction tree (e.g., by looking at tables built for a particular elimination order)
- Instantiate evidence
- Pass messages
- Pick a subtree containing desired variables, read off its distribution, and sum out nuisance variables

Calibration

$ABC \overset{}{\underset{BC}{—}} BCD$

$P(ABCD) = P(ABC)\, P(\cancel{BC}D \mid ABC)$

$\to P(\cancel{BC}D \mid BC)$

- After BP, easy to get all clique marginals
  - also all sepset marginals (sum out from clique on either side)
- Bayes rule: P(clique \ sepset | sepset) =

$$P(\text{clique}) / P(\text{sepset})$$

$c_1 \overset{s}{—} c_2$

- So, joint P(clique$_1$ ∪ clique$_2$) =

$$P(c_1)P(c_2 \mid c_1) = P(c_1)\, P(c_2) / P(\text{sepset})$$

sepset

- Continue over entire tree: P(everything) =

$$\prod_{\text{clique } i} P(C_i) \Big/ \prod_{\text{sepset } j} P(\text{sep}_j)$$

Hard v. soft factors

Hard

X

Y		0	1	2
	0	0	0	0
	1	0	0	1
	2	0	1	1

Soft

X

Y		0	1	2
	0	1	1	1
	1	1	1	3
	2	1	3	3

Moralize & triangulate (to build JT)

- Moralize:
  - for factor graphs: a clique for every factor
  - for Bayes nets: "marry the parents" of each node
- Triangulate: find a chordless 4-or-more-cycle, add a chord, repeat
- Find all maximal cliques
- Connect maximal cliques w/ edges in any way that satisfies RIP

Continuous variables

- Graphical models can have continuous variables too
  - CPTs → conditional probability densities (or measures)
  - potential tables → potential functions
  - message tables → message functions
  - sums → integrals
- Q: how do we represent the functions?
  - A: any way we want…
  - mixtures of Gaussians, sets of samples, Gaussian processes
  - and in a few minutes: exponential family distributions

Loopy BP

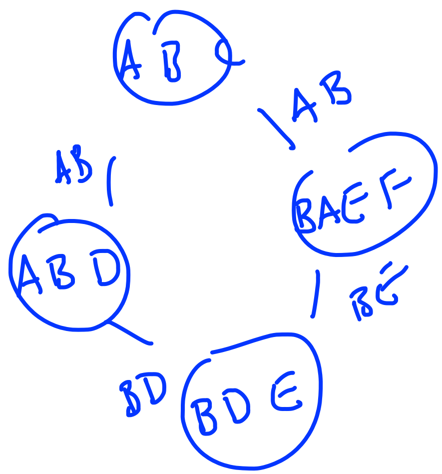


Plate models