Lecture 15 - sum-product / max-product

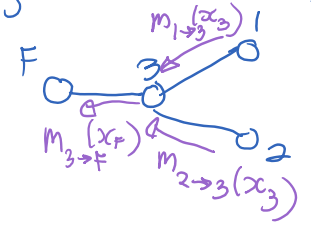
October 29, 2021 3:03 PM

today: sum-product
max-product
junction tree

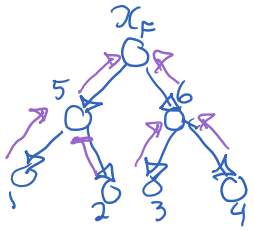
inference on trees

graph Elimination on a tree
→ good order is to eliminate leaves first

$$p(x) = \frac{1}{Z} \prod_i \psi_i(x_i) \prod_{\{i,j\} \in E} \psi_{ij}(x_i, x_j)$$



order: make a directed tree by using x_F as a root
(singleton)



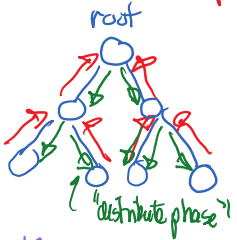
$$m_{i\to j}(x_j) = \sum_{x_i} \psi_i(x_i)\psi_{ij}(x_i, x_j) \prod_{K \in \text{children } i} m_{K\to i}(x_i)$$

i: child, j: parent

new factors containing i in the active list

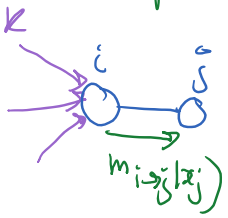
sum-product alg. (for trees)

alg. to get all node/edge marginals cheaply by storing (caching)
& re-using messages (dynamic programming)



goal: $\{i,j\} \in E$, compute $m_{i\to j}(x_j)$
$m_{j\to i}(x_i)$

rule: i can only send message to neighbor j
when it has received all messages from other neighbors



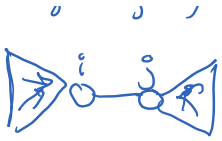
$$m_{i\to j}(x_j) = \sum_{x_i} \psi_i(x_i)\psi_{ij}(x_i, x_j) \prod_{K \in N(i)\setminus\{j\}} m_{K\to i}(x_i)$$

at end
(node marginal) $$p(x_i) \propto \prod_{j \in N(i)} m_{j\to i}(x_i)\psi_i(x_i)$$

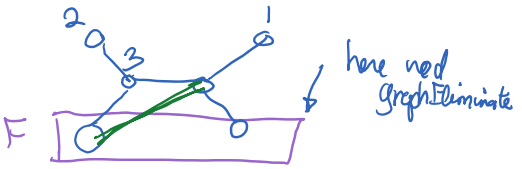
normalize: $Z = \sum_{x_i} (\ )$

(edge marginal)

$$p(x_i, x_j) = \frac{1}{Z} \psi_i(x_i)\psi_j(x_j)\psi_{ij}(x_i, x_j) \prod_{K \in N(i)\setminus\{j\}} m_{K\to i}(x_i) \cdot$$



$$p(x_i, x_j) = \frac{1}{Z} \psi_i(x_i)\psi_j(x_j)\psi_{ij}(x_i, x_j) \prod_{k \in N(i) \setminus \{j\}} m_{k \to i}(x_i) \cdot \prod_{k' \in N(j) \setminus \{i\}} m_{k' \to j}(x_j)$$



Sum-product schedules

a) above, collect / distribute schedule

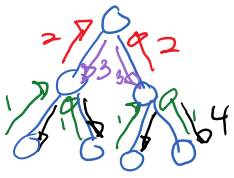
b) (flooding) parallel schedule:

1) initialize all $m_{i \to j}(x_j)$ messages to uniform dist $\forall (i,j)$ s.t. $\{i,j\} \in E$

2) at every step (in parallel) compute $m^{\text{new}}_{i \to j}(x_j)$

as if the neighbor messages were correctly computed

→ one can prove that "diameter of tree" # of steps all messages are correctly computed (for a tree) (and one fixed pt.)



Loopy belief propagation (loopy BP): approximate inference for graphs with cycles

$$m^{\text{new}}_{i \to j}(x_j) = \left(m^{\text{old}}_{i \to j}(x_j)^{\alpha} \sum_{x_i} \psi_i(x_i)\psi_{ij}(x_i, x_j) \prod_{k \in N(i) \setminus \{j\}} m^{\text{old}}_{k \to i}(x_i) \right)^{(1-\alpha)}$$

* this gives exact answer on trees (fixed pt. → yields correct marginals)
* on (not too loopy) graphs → approximate solution

$\alpha \in [0,1]$ "damping"

q step-size

Getting conditionals:

$$p(x_i \mid \bar{x}_E) \propto p(x_i, \bar{x}_E)$$

$\bar{x}_E$: indicates values we are conditioning on, $x_E = \bar{x}_E$

$x_E = \bar{x}_E$: keep this fixed during marginalization for each $j \in E$

(formal trick): redefine: $\tilde{\psi}_j(x_j) \triangleq \psi_j(x_j) \cdot \delta(x_j, \bar{x}_j)$

Kronecker-delta fct. $\delta(a,b) \triangleq \begin{cases} 1 & \text{if } a = b \\ 0 & \text{o.w.} \end{cases}$

Kronecker delta function:
$\delta(a,b) \triangleq \begin{cases} 1 & \text{if } a=b \\ 0 & \text{o.w.} \end{cases}$

Computing $m_{j\to i}(x_i)$:

$$\sum_{x_j} \tilde{\psi}_j(x_j)\, \text{stuff}(x_i, x_j)$$

$$= \psi_j(\bar{x}_j)\, \text{stuff}(x_i, \bar{x}_j)$$

at the end, result of sum-product will give $p(x_i, \bar{x}_E) = \frac{1}{Z}\psi_i(x_i) \prod_{k\to i} m_{k\to i}(x_i)$

renormalize over x_i to get $p(x_i \mid \bar{x}_E)$

16h02

max-product alg:

for sum-product, main property used was distributivity of $\oplus$ over $\odot$

all need is that $(\mathbb{R}, \oplus, \odot)$ is a semi-ring

↳ don't need additive inverses

$\circledast$ can do "sum-product" on other semi-rings

$(\mathbb{R}, \max, +)$ $\quad \max(a+b, a+c) = a + \max(b,c)$ [max-sum alg.]

$(\mathbb{R}_+, \max, \cdot)$ $\quad \max(a\cdot b, a\cdot c) = a \cdot \max(b,c)$

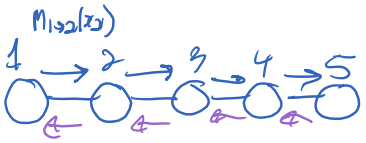
"max-product"

$$\max_{x_{1:n}} \prod_j f_j(x_j) = \prod_j \max_{x_j} f_j(x_j)$$

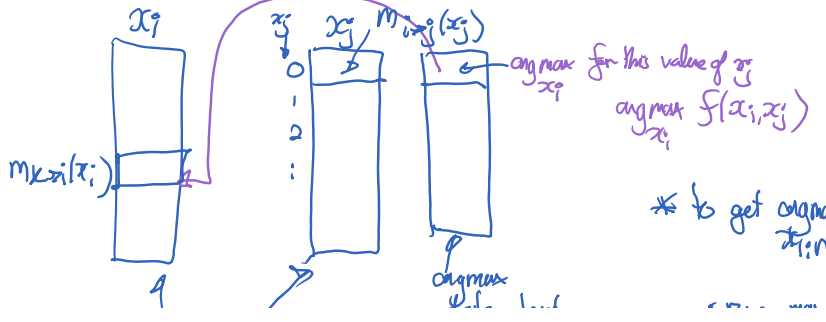
$$m_{i\to j}(x_j) = \max_{x_i} \left[\psi_i(x_i)\psi_{ij}(x_i, x_j) \prod_{k\in N(i)\setminus j} m_{k\to i}(x_i) \right]$$

for getting argmax

store argument of this max as a fct. of x_j



$$\max_{x_{1:5}} \frac{1}{Z} \prod_c \psi_c(x_c) = \frac{1}{Z} \max_{x_5} \left[m_{4\to 5}(x_5) \cdot \psi_5(x_5) \right]$$



* to get $\arg\max_{x_{1:n}} p(x_{1:n})$ "decoding"

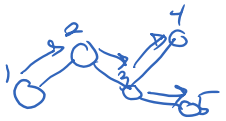
- run max-product alg. (only forward messages)
- backtrack the argmax pointers to get full argmax

aka Viterbi algorithm

property of tree UGM

$p \in \mathcal{G}(\text{tree})$ with non-zero marginals $\Rightarrow$

$$p(x) = \prod_{i \in V} \psi_i(x_i) \prod_{\{i,j\} \in E} \frac{\psi_{ij}(x_i,x_j)}{\psi_i(x_i)\psi_j(x_j)}$$



$$p(x) = p(x_5|x_3)\,p(x_4|x_3)\,p(x_3|x_2)\,p(x_2|x_1)\,p(x_1)$$

$$= p(x_5)\,p(x_3)\,p(x_2)\,p(x_1,x_2)$$

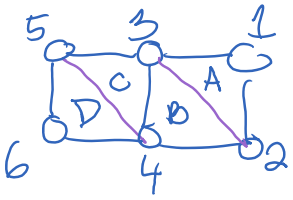
⊛ similar to DGM, for any set of factors $\{f_{ij}(x_i,x_j)\}, \{f_i(x_i)\}$ $f_{ij} \geq 0$, $f_i \geq 0$ s.t. "local consistency property"

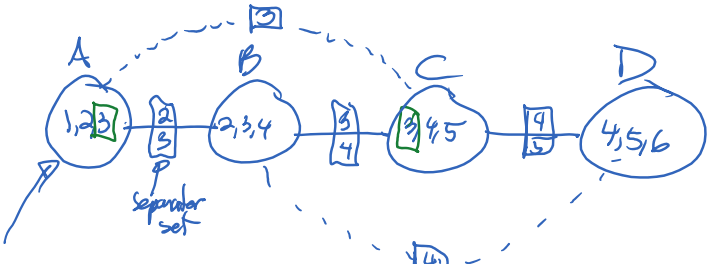
$$\begin{cases} \sum_{x_j} f_{ij}(x_i,x_j) = f_i(x_i) & \forall x_i \\ \sum_{x_i} f_{ij}(x_i,x_j) = f_j(x_j) & \forall x_j \\ \sum_{x_i} f_i(x_i) = 1 & \forall i \end{cases}$$

then if defines a joint $p(x) = \prod_i f_i(x_i) \prod_{\{i,j\} \in E} \frac{f_{ij}(x_i,x_j)}{f_i(x_i) f_j(x_j)}$ (for a tree)

(we could show) that we get correct marginals ie $p(x_i) = f_i(x_i)$ etc. //

junction tree algorithm: generalization of sum-product to a clique tree (with J.T. property)





above is a clique tree with the "running intersection property" } "junction tree"

↳ if $j \in C_1 \cap C_2$, then

$j \in C_k \; \forall C_k$ along path from C_1 to C_2

to build a J.T. on a Δ-graph:

- use maximum weight spanning tree alg. on clique graph (with size of separator set as weight on edges) $\Rightarrow$ running intersection property

$\exists$ a J.T. $\Leftrightarrow$ triangulated graph / decomposable graph $\leftrightarrow$ running graphEliminate

(*) when have a J.T., one can show

$$p(x_V) = \frac{\prod_C p(x_C)}{\prod_S p(x_S)}$$

S: separator sets in the J.T.

J.T. alg.: reconstruct the above formulation by starting with $p(x_V) = \frac{1}{Z} \frac{\prod_C \psi_C(x_C)}{\prod_S \varphi_S(x_S)}$ where $\varphi_S(x_S) = 1$ at initialization

alg message passing on J.T. to update

ψ_C^{new}, φ_S^{new} $\longrightarrow$ at the end $p(x_C)$, $p(x_S)$