These notes are intended to give a summary of relevant concepts from the lectures which are helpful to complete the exercises. It is not intended to cover the lectures thoroughly. Learning this content is not a replacement for working through the lecture material and the exercises.

Sum-product algorithm — Variable elimination for factor trees reformulated with "messages" which allows for re-use of computations already done. See table on following page.

Max-product algorithm — Same as the sum-product algorithm, but max replaces $\sum$.

Max-sum algorithm — Max-product algorithm in the log-domain. See table on following page.

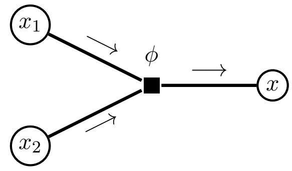
Sum-product algorithm

$\mu_{\phi \to x}(x)$

Factor to variable

$\mu_{\phi \to x}(x) = \sum_{x_1,\ldots,x_j} \phi(x_1, \ldots, x_j, x) \prod_{i=1}^{j} \mu_{x_i \to \phi}(x_i)$

where $\{x_1, \ldots, x_j\} = \mathrm{ne}(\phi) \setminus \{x\}$

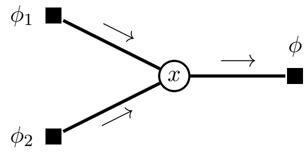


$\mu_{x \to \phi}(x)$

Variable to factor

$\mu_{x \to \phi}(x) = \prod_{i=1}^{j} \mu_{\phi_i \to x}(x)$

where $\{\phi_1, \ldots, \phi_j\} = \mathrm{ne}(x) \setminus \{\phi\}$

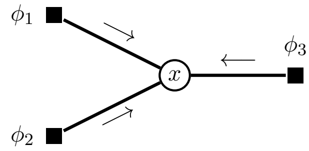


$\tilde{p}(x)$

Univariate marginals – unnormalised

$p(x) \propto \prod_{i=1}^{j} \mu_{\phi_i \to x}(x)$

where $\{\phi_1, \ldots, \phi_j\} = \mathrm{ne}(x)$

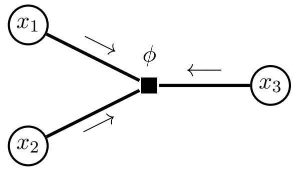


$\tilde{p}(x_1, \ldots, x_j)$

Joint marginals of variables sharing a factor– unnormalised

$p(x_1, \ldots, x_j) \propto \phi(x_1, \ldots, x_j) \prod_{i=1}^{j} \mu_{x_i \to \phi}(x_i)$

where $\{x_1, \ldots, x_j\} = \mathrm{ne}(\phi)$



Max-sum algorithm

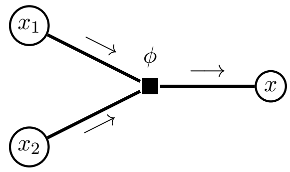
$\gamma_{\phi \to x}(x)$

Factor to variable

$\gamma_{\phi \to x}(x) = \max_{x_1,\ldots,x_j} \log \phi(x_1, \ldots, x_j, x) + \sum_{i=1}^{j} \gamma_{x_i \to \phi}(x_i)$

$\gamma^*_{\phi \to x}(x) = \mathrm{argmax}_{x_1,\ldots,x_j} \log \phi(x_1, \ldots, x_j, x) + \sum_{i=1}^{j} \gamma_{x_i \to \phi}(x_i)$

where $\{x_1, \ldots, x_j\} = \mathrm{ne}(\phi) \setminus \{x\}$

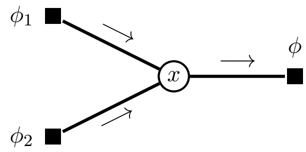


$\gamma_{x \to \phi}(x)$

Variable to factor

$\gamma_{x \to \phi}(x) = \sum_{i=1}^{j} \gamma_{\phi_i \to x}(x)$

where $\{\phi_1, \ldots, \phi_j\} = \mathrm{ne}(x) \setminus \{\phi\}$

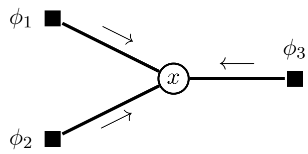


$\log p_{\max}$

Maximum probability

$\log p_{\max} = \max_x \gamma^*(x), \quad \gamma^*(x) = -\log Z + \sum_{i=1}^{j} \gamma_{\phi_i \to x}(x)$

where $\{\phi_1, \ldots, \phi_j\} = \mathrm{ne}(x)$



$\mathrm{argmax}_{\mathbf{x}}\, \tilde{p}(\mathbf{x})$

Maximum probability states – no need for normalisation

Init: $\hat{x} = \mathrm{argmax}_x \gamma^*(x) = \mathrm{argmax}_x \sum_{i=1}^{j} \gamma_{\phi_i \to x}(x)$

Backtrack to leaves via $\gamma^*_{\phi \to x}(x)$