

Sum-Product Networks

Sum-Product Networks (SPNs)

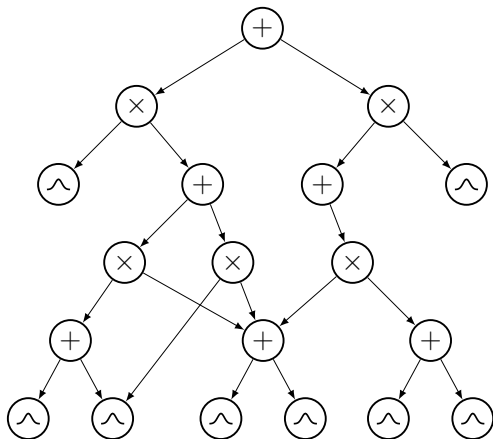
A sum-product network [Poon & Domingos 2011] over random variables \mathbf{X} is a tuple $\mathcal{S} = (\mathcal{G}, \psi, \mathbf{w}, \theta)$, consisting of a *computational graph* \mathcal{G} , a *scope-function* ψ [Trapp et al. 2019], network weights \mathbf{w} and leaf node parameters θ .

Why use SPNs? **Exact and efficient** computation of many probabilistic inference tasks, e.g. marginalisation.

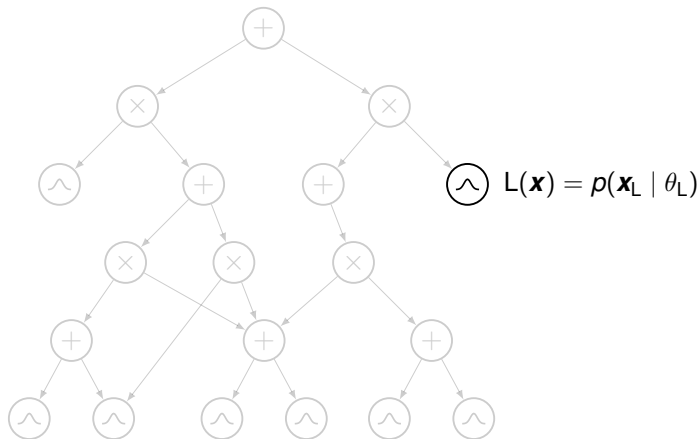
SPNs are deep structured mixture models, generalising shallow mixtures, using an architecture that is a special type of a sparse *linear* neural network with non-linear inputs.

Every SPN is a multi-linear function in its weights.

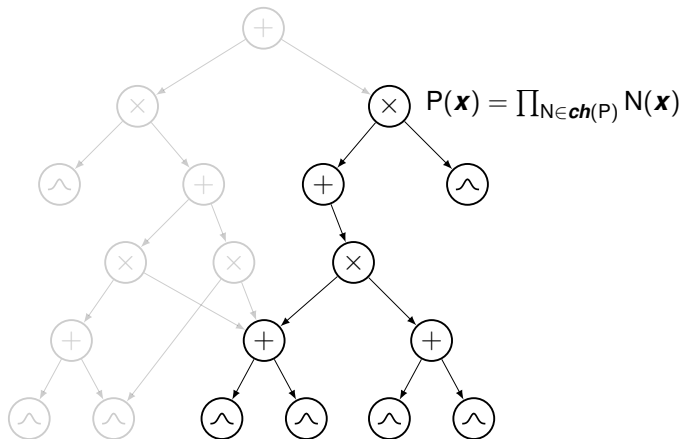
Computational Graph: \mathcal{G}



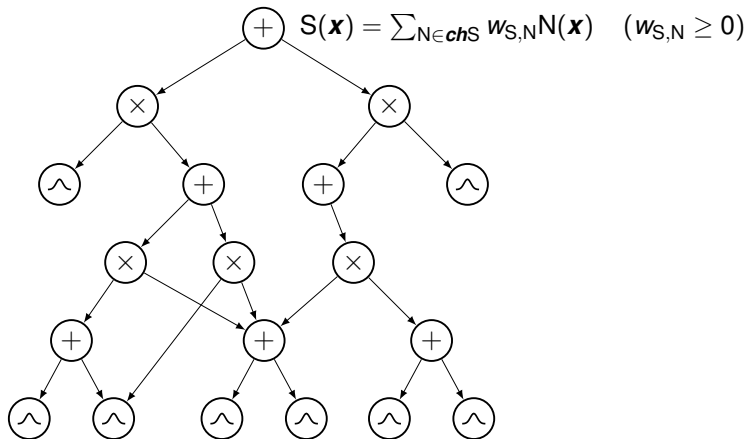
Computational Graph: \mathcal{G}



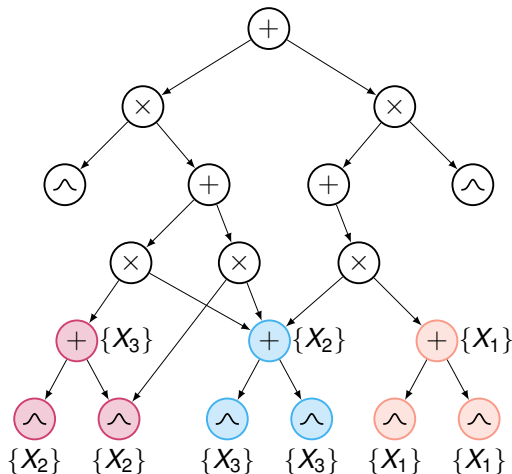
Computational Graph: \mathcal{G}



Computational Graph: \mathcal{G}

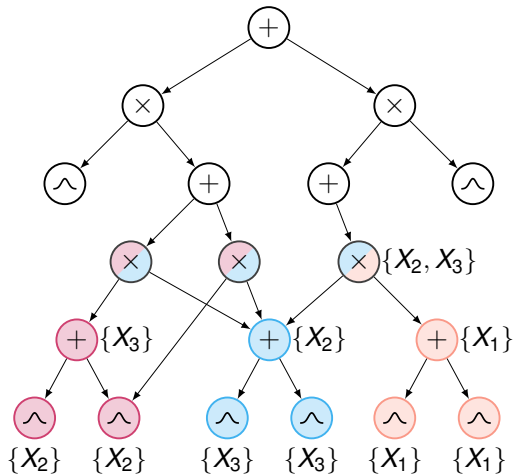


Scope Function: $\psi: \mathbf{N} \mapsto \mathcal{P}(\mathbf{X})$



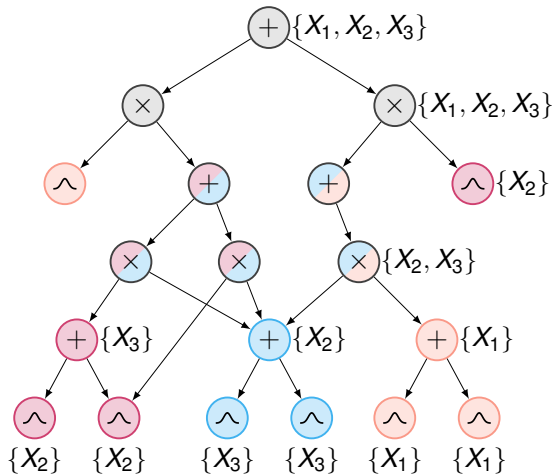
completeness /
smoothness

Scope Function: $\psi: \mathbf{N} \mapsto \mathcal{P}(\mathbf{X})$

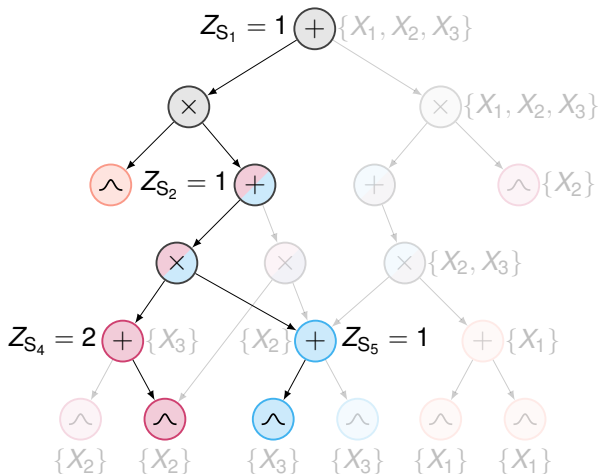


decomposability

Scope Function: $\psi: \mathbf{N} \mapsto \mathcal{P}(\mathbf{X})$

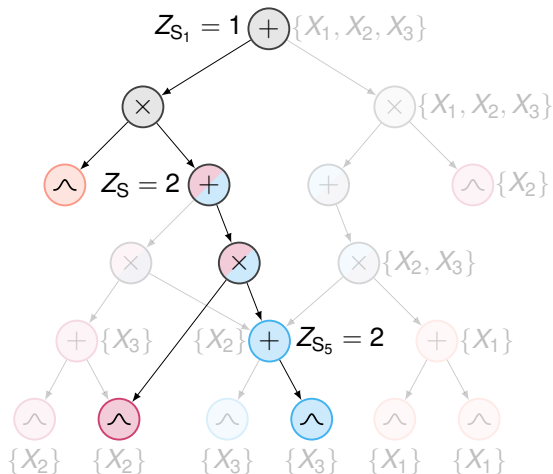


SPNs as a Mixture of Induced Trees



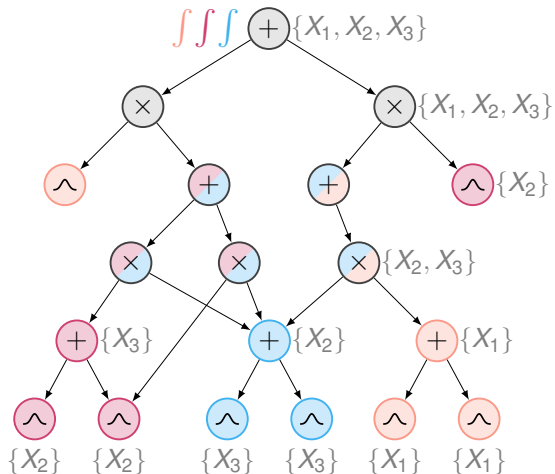
$$\mathcal{S}(\mathbf{x}) = \sum_{i=1}^K \prod_{(S,N) \in \mathcal{T}_{i,E}} w_{S,N} \prod_{L \in \mathcal{T}_{i,V}} p(\mathbf{x}_L \mid \theta_L) \quad (1)$$

SPNs as a Mixture of Induced Trees

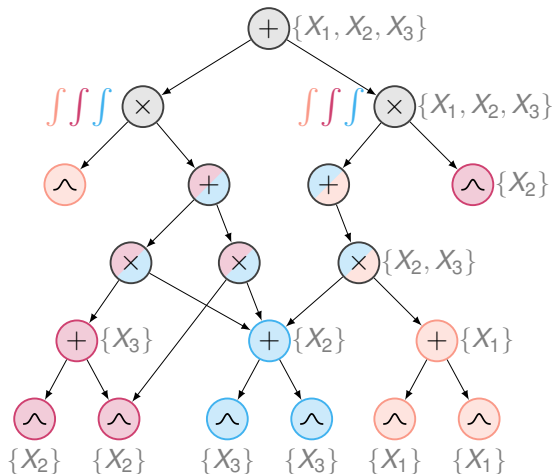


$$\mathcal{S}(\mathbf{x}) = \sum_{i=1}^K \prod_{(S,N) \in \mathcal{T}_{i,E}} w_{S,N} \prod_{L \in \mathcal{T}_{i,V}} p(\mathbf{x}_L \mid \theta_L) \quad (1)$$

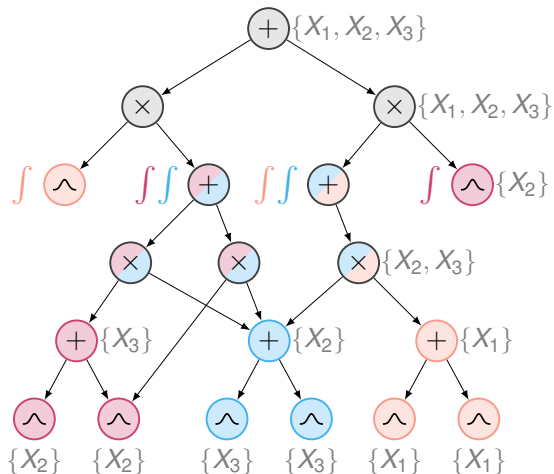
Exact Marginalisation in SPNs



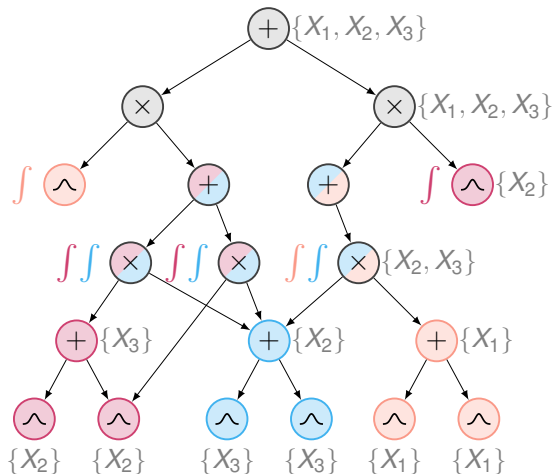
Exact Marginalisation in SPNs



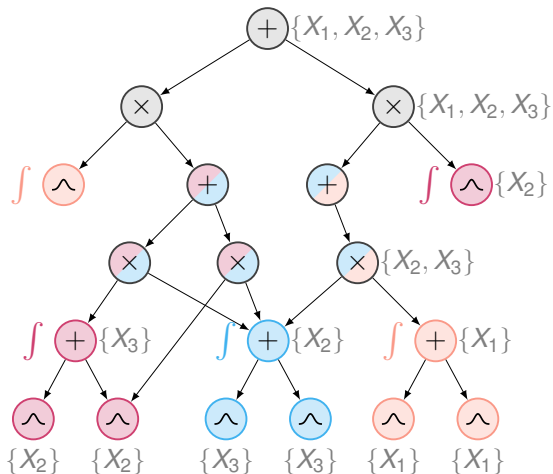
Exact Marginalisation in SPNs



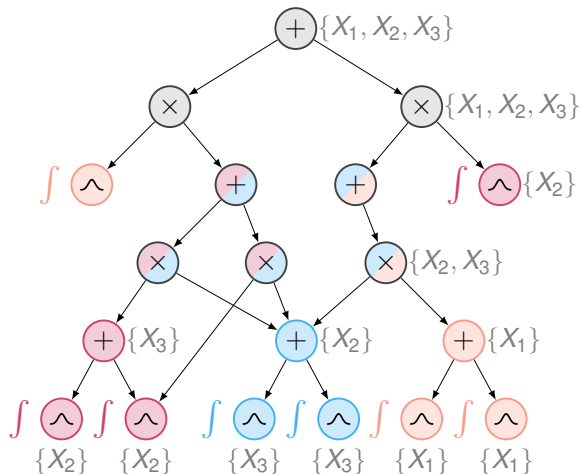
Exact Marginalisation in SPNs



Exact Marginalisation in SPNs



Exact Marginalisation in SPNs



References

[Poon & Domingos 2011] H. Poon and P. M. Domingos, 'Sum-product networks: A new deep architecture', UAI 2011, pp. 337–346.

[Trapp et al. 2019] M. Trapp, R. Peharz, H. Ge, F. Pernkopf and Z. Ghahramani, 'Bayesian learning of sum-product networks', NeurIPS 2019, pp. 6344–6355.