Estudo sobre Sum-Product Networks e Aprendizagem Profunda

Renato Lui Geh

Instituto de Matemática e Estatística
Universidade de São Paulo

16 de junho de 2016

Índice

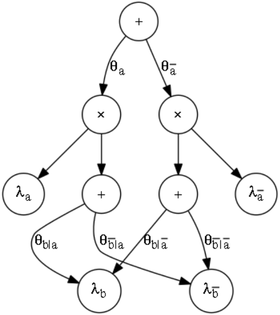
Relembrando...

Definição

Uma SPN S é um DAG com três tipos de nós: soma, produto e indicadores. Todo nó indicador é uma folha. Todo nó soma tem pais produto, e todo nó produto tem pais soma. Toda aresta com destino a um nó soma tem uma aresta com um peso associado. O valor de um nó soma i é $\sum_{j \in Ch(i)} w_{ij} v_j$ e o valor de um nó produto i é $\prod_{j \in Ch(i)} v_j$, onde $Ch(i)$ é o conjunto de filhos de i, v_i é o valor do nó i e w_{ij} é o peso associado a aresta $i \to j$. Uma SPN representa um *network polynomial* de uma distribuição de probabilidade, e os indicadores da função são as folhas da SPN. O valor de uma SPN é o valor do nó raíz.

[PD11]

Relembrando. . .



Sub-SPNs

Proposição

Seja um nó arbitrário i de uma SPN S, então S_i é uma sub-SPN que tem nó raíz em i.

Prova sub-SPNs

Demonstração.

Considere o caso base em que i é um nó indicador. Um nó indicador é uma distribuição de probabilidade monovariável. Portanto i é uma distribuição de probabilidade e pode ser representada por uma SPN, que no caso possue apenas um nó.

Se i é um nó soma, então o valor de i é $v_i = \sum_{j \in Ch(i)} w_{ij} v_j$. A soma de várias distribuições de probabilidade é uma distribuição de probabilidade. Portanto um nó soma é representável por uma SPN.

Caso i seja um nó produto, então o valor de i é $v_i = \sum_{j \in Ch(i)} v_j$. A multiplicação de distribuições de probabilidade é bem definida e é uma distribuição de probabilidade. Um nó produto é uma SPN. $\square$

Completude

Definição (Completude)

Uma SPN S é completa se e somente se, para todo nó soma i, o escopo de i é igual par-a-par $Sc(S_i) = Sc(S_j)$ ao escopo de cada filho $j \in Ch(i)$.

Consistência

Definição (Consistência)

Uma SPN S é consistente se e somente se, para todo nó produto i, nenhum filho de i tem valor diferente dos outros filhos.

Validade

Definição (Validade)

Uma SPN S é válida se S é consistente e completa.

Decomponibilidade

Definição (Decomponibilidade)

Uma SPN S é decomponível se e somente se, para cada par $c_1, c_2 \in Ch(i)$ para qualquer i nó produto em S, $Sc(c_1) \cap Sc(c_2) = \emptyset$.

Uma definição alternativa

Definição

Uma SPN tem uma definição recursiva. Definimos que uma SPN S_i pode ser apenas:

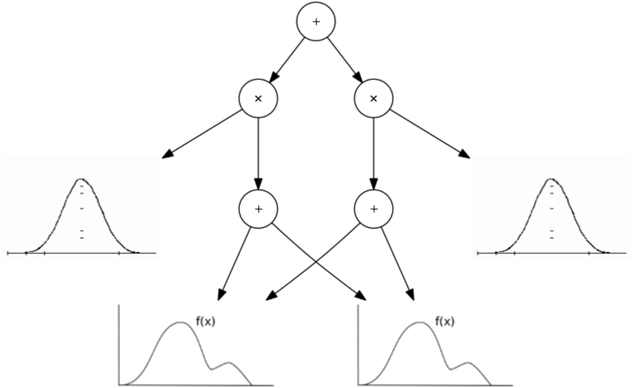
(i) Uma distribuição monovariável $p(\mathbf{X})$ ou;

(ii) Um nó soma tal que $S_i = \sum_{j \in Ch(i)} w_{ij} v_j$ onde para cada filho $j, k \in Ch(i)$, $Sc(S_j) = Sc(S_k)$ ou;

(iii) Um nó produto tal que $S_i = \prod_{j \in Ch(i)} v_j$ onde para cada filho $j, k \in Ch(i)$, $Sc(S_j) \cap Sc(S_k) = \emptyset$.

[GD13]

Uma definição alternativa



Shallow vs. Deep SPNs

Sejam p uma distribuição suficientemente complexa, S uma SPN que representa p e $m = |Sc(p)|$.

$$n \text{ camadas ocultas: Computa-se } S \text{ em } \mathcal{O}(\exp(m)).$$

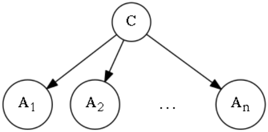
$$n+1 \text{ camadas ocultas: Computa-se } S \text{ em } \mathcal{O}(m^k), k <<< m.$$

Prova disso em *Shallow vs. Deep Sum-Product Networks*, Delalleau e Bengio [DB11].

Classificação por Naive Bayes

$$C : \text{ variável classe}$$
$$\mathbf{A} = \{A_1, \ldots, A_n\} : \text{ variáveis atributos}$$

$$A_i \perp A_j \equiv A_i \perp_d A_j, \text{ para } 1 \leq i, j \leq n \text{ e } i \neq j$$



Classificação por Naive Bayes

Pelo Teorema da Fatorização:

$$\Pr(C, A_1, \ldots, A_n) = \Pr(C) \prod_{i=1}^{n} \Pr(A_i | C)$$

Classificação resume-se a encontrar um máximo c:

$$\arg\max_c \left(\Pr(C = c | A_1 = a_1, \ldots, A_k = a_k) = \frac{\Pr(C = c, A_1, \ldots, A_k = a_k)}{\Pr(A_1 = a_1, \ldots, A_k = a_k)} \right)$$

Aprendizado de Naive Bayes por MLE

MLE (Maximum Likelihood Estimation) $\equiv$ Máxima verossimilhança

Aprender uma Naive Bayes:

Variável classe $\Pr(C = c) = \frac{N[C=c]}{N}$

i-**ésimo atributo** $\Pr(A_i = a_i | C = c) = \frac{N[A_i=a_i, C=c]}{N[C=c]}$

Uma visão mais intuitiva

Nós internos: variáveis latentes – camadas ocultas:

- $+$**:** mistura de distribuições – "semelhança entre instâncias";
- $\times$**:** independência entre variáveis.

Nós folhas: Valoração/instanciação das variáveis.

Uma visão mais intuitiva

Nós internos: variáveis latentes – camadas ocultas:

- $+$**:** mistura de distribuições – "semelhança entre instâncias";
- $\times$**:** independência entre variáveis.

Nós folhas: Valoração/instanciação das variáveis.

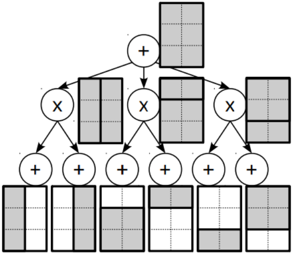


Figura: *Learning the Structure of Sum-Product Networks Using Clustering on Variables*, Dennis e Ventura [DV12]

Aprendizado estrutural de SPNs

Algoritmo 1 LearnSPN [GD13]

Input Conjunto $\mathbf{X}$ de variáveis, conjunto $\mathbf{I}$ de instâncias
Output Uma SPN resultante do aprendizado estrutural

if $|\mathbf{X}| = 1$ **then**
    Retorna uma distribuição monovariável de $\mathbf{X}$
end if
Tente dividir as variáveis $\mathbf{X}$ em duas partições $\mathbf{X}_1$ e $\mathbf{X}_2$ onde $\mathbf{X}_1$ é (aproximadamente) independente de $\mathbf{X}_2$
if dá para dividir **then**
    return $\prod_{i=1}^{2}$ LearnSPN($\mathbf{X}_i$, $\mathbf{I}$)
else
    Divida as instâncias $\mathbf{I}$ em partições $\mathbf{I}_1$ e $\mathbf{I}_2$ tal que $\mathbf{I}_1$ e $\mathbf{I}_2$ sejam o mais similares possíveis.
    return $\sum_{i=1}^{2} \frac{|\mathbf{I}_i|}{|\mathbf{I}|}$ LearnSPN($\mathbf{X}$, $\mathbf{I}_i$)
end if

Referências e Bibliografia I

Olivier Delalleau e Yoshua Bengio. "Shallow vs. Deep Sum-Product Networks". Em: *Advances in Neural Information Processing Systems 24 (NIPS 1011)* (2011).

Aaron Dennis e Dan Ventura. "Learning the Architecture of Sum-Product Networks Using Clustering on Variables". Em: *Advances in Neural Information Processing Systems* 25 (2012).

Robert Gens e Pedro Domingos. "Learning the Structure of Sum-Product Networks". Em: *International Conference on Machine Learning* 30 (2013).

Hoifung Poon e Pedro Domingos. "Sum-Product Networks: A New Deep Architecture". Em: *Uncertainty in Artificial Intelligence* 27 (2011).