Logic for Knowledge Representation, Learning, and Inference

Luciano Serafini

serafini@fbk.eu

Version July 24, 2023

Contents

CHAPTER 1

Weighted model counting

Perhaps, before introducing weightd model counting, this is a good time to summarise the inference tasks on propositional logic that we have described sofa including the one that we are going to describe in this section. They are listed in Table 1.

Task Name	**Input**	**Result**	**Description**
Model checking:	$\phi, \mathcal{I}$	$\mathcal{I}(\phi)$	Compute the truth value that the interpretation $\mathcal{I}$ assigns to the formula ϕ, or equivalently check if ϕ is satisfied or not satisfied by $\mathcal{I}$.
Satisfiability:	ϕ	$\max_{\mathcal{I}} \mathcal{I}(\phi)$	Search for an assignment $\mathcal{I}$ to the propositional variables of ϕ that satisfies ϕ. If such an assignment does not exist ϕ is unsatisfiable otherwise it is satisfiable.
Maximum Satisfiability:	ϕ, w	$\max_{\mathcal{I}} \mathcal{I}(\phi) \cdot w(\mathcal{I})$	Search for an assignment $\mathcal{I}$ that satisfies the formula ϕ and maximize a weight function (or minimize a cost function) defined on the interpretations of the propositional variables in ϕ.
Model counting:	ϕ	$\sum_{\mathcal{I}} \mathcal{I}(\phi)$	Count how many assignments to the propositional variables of ϕ are models of (or equivalently satisfy) ϕ,
Weighted Model counting:	ϕ, w	$\sum_{\mathcal{I}} \mathcal{I}(\phi) \cdot w(\mathcal{I})$	Compute the weighted sum of the models of ϕ according to the weight function w.

1. Introduction

Weighted model counting is a generalisation of model counting where models have different weight, which is usually a positive numbers. Model counting is a special case of weighted model counting where each model weight is equal to 1. The most widespread application of weighted model counting is in probabilistic inference. Indeed, a recent and effective approach to probabilistic inference can be obtained by reducing the problem to one of weighted model counting Chavira and Darwiche 2008.

The problem of *Weighted model counting* can be formulated as follows: Given a propositional formula ϕ containing propositinal variables in $\mathcal{P}$ and a weight function w that assigns a non-negative weight to every truth assignment to propositional variables in $\mathcal{P}$, weighted model counting (henceforth WMC) concerns summing weights of the assignments that satisfy ϕ. formula. If every assignment has weight 1, the corresponding problem boils down to model counting.

DEFINITION 1.1 (Weighted model counting). *Foer every set of propositinal variables $\mathcal{P}$ and weight function $w : 2^{\mathcal{P}} \to \mathbb{R}^+$. the* weighted model counting *of a propositinal formula ϕ with propositinal variables in $\mathcal{P}$ w.r.t, w, is defined as:*

$$\textsc{wmc}(\phi, w) = \sum_{\mathcal{I}:\mathcal{P}\to\{0,1\}} \mathcal{I}(\phi) \cdot w(\mathcal{I}) \tag{1}$$

Weighted model counting and unweighted model counting (or simply model counting) share a lot. Indeed as it will be clear in this section, many of the techniques that have been developed for model counting can be equally applied or generalized for weighed model counting. There are also approaches that reduces weighted model counting to unweighted model counting Chakraborty et al. 2015. Concerning the application of weighted model counting to probabilistic reasoning we will describe them in details in this chapter.

In (unweighted) model counting each model of a formula counts 1; in weighted model counting some models are more important/probable/preferrable than others. To measure how much a model is important, it makes sense to associate a positive weight $w(\mathcal{I}) \geq 0$ to each interpretation $\mathcal{I}$. Why positive? Well this is not strictly necessary, however this is what happens. Furthermore, positive weights makes more direct the connection with probability. In weighted model counting each model $\mathcal{I}$ of a formula counts for its weight $w(\mathcal{I})$. The eight $w(\mathcal{I})$ associated to the model $\mathcal{I}$ can be interpreted in probabilistically; i.e. the weight is proportional to the likelihood of this model

EXAMPLE 1.1. *Suppose that a supermarked is selling item categories a, b, $\dots f$, g. From the fidelity cards of the customers we observe the following records:*

#	*Itemsets*						
4	a	b	c	d			
1	a	b			e	f	
7	a	b	c				
3	a		c	d		f	
2							g
1				d			
4				d			g

Every row of the above table reports the number of customers that have bought a set of items that contains at least one item for each category listed in the row, (we dont count how many items of the category he/she has bought). Notice that there are $2^7 = 128$ combinations of itemssets, and the table reports only the combinations which has been observed at least once. Therefore, if a combination is not present then, it means that no customer has ever bought items of that combination of types.

Every combination of items can be seen as an interpretation on the set of propositions $a, b, \ldots g$, where a means "the customer has bought at least one item of type a", similarly for b, dots g. We can consider the number of times we observe a combination as the weight of the corresponding interpretation. In other words we will have the follwoing weight function

			$\mathcal{I}$				
a	b	c	d	e	f	g	$w(\mathcal{I})$
1	1	1	1	0	0	0	4
1	1	0	0	1	1	0	1
1	1	1	0	0	0	0	7
1	0	1	1	0	1	0	3
0	0	0	0	0	0	1	2
0	0	0	0	1	0	0	1
0	0	0	0	1	0	1	4

We have 2^7 possible itemsets (interpretations $\mathcal{I}$), and we can assigns to each a weight equal to $w(\mathcal{I})$ where $w(\mathcal{I})$ is the number of times an itemset has been observed. Given this for every formula in the language of $a, \ldots, g$ the weighted model counting returns the number of customer that bought items with tipes that respect the formula. For instance.

$\text{WMC}(a \wedge (b \vee c)) = 4 + 1 + 7 + 3 = 15$ *The number of times that a customer buys at least an item of type a and at least one of type b or c*

$\text{WMC}(a \wedge \neg g) = 0$ *No customer buys at the same time an item of type a and an item of type g.*

$\#\text{SAT}(a \to b) = 4 + 1 + 7 + 2 + 1 + 4 = 19$ *It counts the number of times a customer buys an item of type b. Notice that in the computation we also take into account all the cases in which the customer does not buy a. This is due to the "material implication" of propositional logicm that interprets $a \to b$ as $\neg a \vee b$.*

2. From #sat to wmc

Most of the results of model counting can be generalized to weight model counting.

2.1. The partition function.

PROPOSITION 1.1. *If ϕ is valid, then* $\text{WMC}(\phi, w)$ *is equal to* $\sum_{\mathcal{I}:\mathcal{P}\to\{0,1\}} w(\mathcal{I})$

The quantity $\sum_{\mathcal{I}:\mathcal{P}\to\{0,1\}} w(\mathcal{I})$ is the sum of the weights of all the interpretations w.r.t., the weight function w. When $\mathcal{P}$ is clear from the context we use the notation $\sum_{\mathcal{I}} w(\mathcal{I})$. This quantity has an important role in many formalizations and algorithms. Therefore it has a special name: i.e., the *partition function of* w. It is usually denoted by $Z(w)$, or simply by Z, when w is clear from the context.

$$Z(w) = \sum_{\mathcal{I}} w(\mathcal{I}) \tag{2}$$

In many cases in which we have to perform weighted model counting and probabilistic reasoning we have to estimate $Z(w)$. This is usually a source of complexity. Indeed in the most general case computing (2) amounts in computing $w(\mathcal{I})$ for all the interpretations $\imath$, wich means that we have to do 2^n calculations where n is the number of propositional variables in $\mathcal{P}$. Notice thta by seeing #SAT as a special case of WMC, where $w(\mathcal{I}) = 1$, computing the partition function $Z(w)$ is not problematinc since it is equal to 2^n.

Let us now see other propoerties of WMC.

PROPOSITION 1.2.

(1) If ϕ is unsatisfiable $\text{WMC}(\phi, w)$ *is equal to* 0*;*
(2) $\#\text{SAT}(\neg\phi) = Z(w) - \text{WMC}(\phi, w)$*;*
(3) If $\phi \models \psi$ then $\text{WMC}(\phi, w) \leq \text{WMC}(\psi, w)$*;*
(4) if ϕ is equivalent to ψ, then $\text{WMC}(\phi, w) = \text{WMC}(\psi, w)$*;*

PROOF. The proof is immediate. However it is worth noticing that property (3) is tightly connected to the fact tht the weight function is positive for all the interpretations. □

2.2. wmc and conjunction. Let us see how WMC behaves w.r.t, conjunction and dijunction. The property that allow to factorize model counting of a conjunction $\phi \wedge \phi$ where ϕ and ψ do not share propositional variables can be reformulated as follows: Let $\phi \wedge \psi$ be a formula on a set of propositional variables $\mathcal{P}$ such that $\mathcal{P}(\phi) \cap \mathcal{P}(\psi) = \emptyset$, and let w be a weight function on $\mathcal{P}$. Let us consider the propoerty

$$\text{WMC}(\phi \wedge \psi, w) = \text{WMC}(\phi, w_{\mathcal{P}(\phi)}) \cdot \text{WMC}(\psi, w_{\mathcal{P}(\psi)}) \tag{3}$$

where $w_{\mathcal{Q}}$ is way to restrict w to a given subset of propositions $\mathcal{Q} \subset \mathcal{P}$. In the general case property (3) does not hold. Consider the following example.

EXAMPLE 1.2. *Let $\mathcal{P} = \{p, q\}$. Consider the following two weight functions:*

$$w_1(\mathcal{I}) = 2 \cdot \mathcal{I}(p) + 3 \cdot \mathcal{I}(q)$$
$$w_3(\mathcal{I}) = 2^{\mathcal{I}(p)} \cdot 3^{\mathcal{I}(q)}$$

In the following two tables we show the weight funcitons on the left, and the weights of the formulas on the right.

p	q	$w_1(\mathcal{I})$	$w_2(\mathcal{I}))$
0	0	1	1
0	1	2	3
1	0	3	2
1	1	4	6

	p	q	$p \wedge q$
w_1	7	6	4
w_2	8	9	6

Let us now see how we can generalize the property of decomposition for model counting to weighted model counting:

PROPOSITION 1.3. *Let w be a weight function defined on a set of propositional variables $\mathcal{P} = \{p_1, \dots, p_n\}$. For every $\mathcal{Q} \subseteq \mathcal{P}$ let $w_{\mathcal{Q}}$ be a weight function defined on the set $\mathcal{Q}$, such that $w_{\mathcal{P}} = w$. If, for every pair of formulas ϕ and ψ that do not share propositionla variables, it is true that:*

$$\text{WMC}(\phi \wedge \psi, w) = \text{WMC}(\phi, w_{\mathcal{P}(\phi)}) \cdot \text{WMC}(\psi, w_{\mathcal{P}(\psi)}) \tag{4}$$

if and only if w can be specified in the following form:

$$w(\mathcal{I}) = \begin{cases} \exp\left(\sum_{i=1}^{n}\left(v_i^+ \cdot \mathcal{I}(p_i) + v_i^- \cdot \mathcal{I}(\neg p_i)\right)\right) & \text{If } \mathcal{I} \models \phi \\ 0 & \text{Otherwise} \end{cases} \tag{5}$$

for some formula ϕ and $v_i^+, v_i^- \in \mathbb{R}$

PROOF. Let $\mathbb{I}$ be the set of interpretations such that $w(\mathcal{I}) \neq 0$, then let ϕ be be a propositional formula that is true if and only if $\mathcal{I} \in \mathbb{I}$. For every $\mathcal{I} \in \mathcal{I}$, let $\mathcal{P}^+$ the set of proposiitional variables true in $\mathcal{I}$ and $\mathcal{P}^-$ those that are false. Let $\phi_{\mathcal{I}} = \bigwedge \mathcal{P}^+ \wedge \bigwedge \neg P^-$. Since $\mathcal{I}$ is the ontly interpretation that satisfies $\phi_{\mathcal{I}}$ we have that $w(\mathcal{I}) = w(\phi_{\mathcal{I}})$. From (4) we have that

$$w(\mathcal{I}) = \prod_{p \in \mathcal{P}^+} w_{\{p\}}(p) \cdot \prod_{p \in \mathcal{P}^-} w_{\{p\}}(\neg p)$$

If we define $v_i^+ = \log(w_{\{p_i\}}(p_i)$ and $v_i^- = \log(w_{\{p_i\}}(\neg p_i)$, the previous expression can be rewritten as

$$\exp\left(\sum_{i=1}^{n}\left(v_i^+ \cdot \mathcal{I}(p_i) + v_i^- \cdot \mathcal{I}(\neg p_i)\right)\right)$$

The proof of the opposite direction is left by exercise. □

Not all the weight function can be expressed in the exponential form (4). Consider for instance the weight function w_1 of Example 1.1. If w_i were expressible in exponential form there should exists four values v_p^+, v_p^-, v_q^+, and v_q^- which are solutions of the following syste of equations:

$$\begin{cases} v_p^- + v_q^- = 0 \\ v_p^- + v_q^+ = \log 2 \\ v_p^+ + v_q^- = \log 3 \\ v_p^+ + v_q^+ = \log 4 \end{cases} \tag{6}$$

However, the system does not have a solution. Indeed, by summing the first and the fourth equation and subtracting the other two we obtain that $\log 4 = -\log 3 - \log 2$ which is false.

2.3. wmc and disjunction. Let us now see how weighted model counting behaves with disjunction. The key property for decomposing model counting of disjunction is determinism. By this property if $\phi \wedge \psi$ is unsatisfiable we have that $\#\text{SAT}(\phi \vee \psi) = \#\text{SAT}(\phi) \cdot 2^m + \#\text{SAT}(\psi) \cdot 2^n$ where m (resp. n) isthe number of propositional variables that occours in ψ but not in ϕ (resp, occour in ϕ but not in ψ). When $n = m = 0$, i.e., ϕ and ψ contains the same set of propositional variables, then we have that $\#\text{SAT}(\phi \vee \psi) = \#\text{SAT}(\phi) + \#\text{SAT}(\psi)$. This property is also true in weighted model counting

A disjunctive formula $\phi \vee \psi$ is called *smooth* if ϕ and ψ contains the same set of propositional variables. A formula ϕ is in sd-DNNF (smooth deterministic decomposable negated normal form) if it is in d-DNNF and every disjunction is smooth.

Proposition 1.4. *If $\phi \vee \psi$ is deterministic and smooth then*

$$\textsc{wmc}(\phi \vee \psi, w) = \textsc{wmc}(\phi, w_{\mathcal{P}(\phi)}) + \textsc{wmc}(\psi, w_{\mathcal{P}(\psi)}) \tag{7}$$

Proof.

$$\begin{aligned}
\textsc{wmc}(\phi \vee \psi, w) &= \sum_{\mathcal{I} \models \phi \vee \psi} w(\mathcal{I}) \\
&= \sum_{\mathcal{I} \models \phi} w(\mathcal{I}) + \sum_{\mathcal{I} \models \psi} w(\mathcal{I}) && \text{By determinism} \\
&= \textsc{wmc}(\phi, w) + \textsc{wmc}(\psi, w) \\
&= \textsc{wmc}(\phi, w_{\mathcal{P}(\phi)}) + \textsc{wmc}(\psi, w_{\mathcal{P}(\psi)}) && \text{By smoothness}
\end{aligned}$$

□

To transform a d-DNNF formula into an sd-NNF formula we can apply the following rule:

- Smoothing left: For subformula $\phi \vee \psi$ with $p \in \mathcal{P}(\psi) \setminus \mathcal{P}(\phi)$ apply this transformation

$$\phi \wedge (p \vee \neg p) \vee \psi$$

- Smoothing right: For subformula $\phi \vee \psi$ with $p \in \mathcal{P}(\phi) \setminus \mathcal{P}(\psi)$ apply this transformation

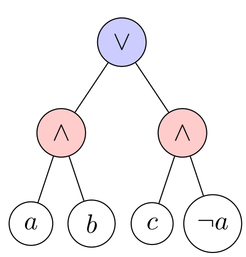
$$\phi \vee \psi \wedge (p \vee \neg p)$$

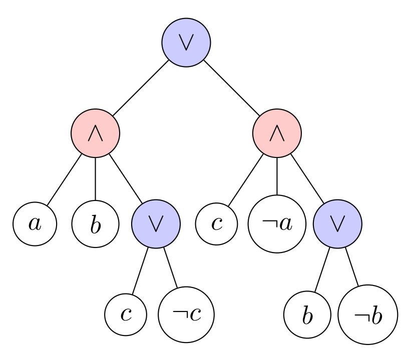
This results in:

$$\left(\phi \wedge \bigwedge_{p \in \mathcal{P}(\psi) \setminus \mathcal{P}(\phi)} (p \vee \neg p) \right) \vee \left(\psi \wedge \bigwedge_{q \in \mathcal{P}(\phi) \setminus \mathcal{P}(\psi)} (q \vee \neg q) \right)$$

Example 1.3. *Smoothing $(a \wedge b) \vee (c \wedge \neg a)$ results in*

$$(a \wedge b \wedge (c \vee \neg c)) \vee ((c \wedge \neg a) \wedge (b \vee \neg b))$$





Notice that the conjunctions that are introduced by the smoothing rule are decomposable. Furthermore the disjunctions introduced by the smoothing rules are deterministic. Therefore by smooting an d-DNNF formula we will not loos the fact the d-DNNF'ness. However, if we want to transform a generic formula in sd-NNF it is better to first transform it into d-DNNF and then apply smoothing in order to obtain an sd-DNNF formula.

Proceeding in the opposite direction would not be optimal. Consider the following example

EXAMPLE 1.4. *consider the formula $(a \wedge b) \vee c$, This formula is neither smooth nor deterministic. Should we try to first smooth it and then make it deterministic by applying Shannon's expansion? or should we proceed in the opposite direction? Let's analize the two cases:*

Smooth *then* determinism

$(a \wedge b) \vee c$	*Smoothing*
$((a \wedge b) \wedge (c \vee \neg c)) \vee (c \wedge (a \vee \neg a) \wedge (b \vee \neg b))$	*Shannon's expansion*
$(c \wedge ((a \wedge b) \vee (a \vee \neg a) \wedge (b \vee \neg b))) \vee (\neg c \wedge (a \wedge b))$	*the red disjunction is not deterministic. So we should apply again Shannon's expansion.*

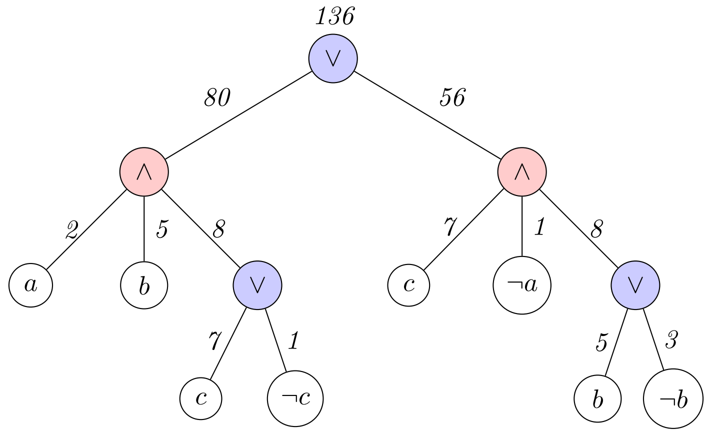
Instead if we apply first Shannon's expansion for determinism and then smoothing we proveed as follows:

$(a \wedge b) \vee c$	*Shannon's exp. on a*
$((b \vee c) \wedge a) \vee (c \wedge \neg a)$	*Shannon's exp. on b*
$((b \vee (c \wedge \neg b)) \wedge a) \vee (c \wedge \neg a)$	*Smoothing*
$((b \vee (c \wedge \neg b)) \wedge a) \vee (c \wedge \neg a \wedge (b \vee \neg b)$	*Smoothing*
$(((b \wedge (c \vee \neg c)) \vee (c \wedge \neg b)) \wedge a) \vee (c \wedge \neg a \wedge (b \vee \neg b))$	

2.4. Weighted model counting by knowledge compilation. When weight function has the exponential form, i.e., the weights are associated with literals weighted model counting can be performed on sd-DNNF formulas by transforming into a sum-product circuit. In particular, an sd-DNNF formula can be transformed a sum-product circuit as follows:

- Every leaf (literal) is associated with its weight;
- at every $\wedge$-node we perform the product of the child nodes;
- at every $\vee$-node we perform the sum of the child nodes.

EXAMPLE 1.5. *Consider the following weighted literals: $w(a) = 2$, $w(\neg a) = 1$, $w(b) = 5$, $w(\neg b) = 3$, $w(c) = 7$ and $w(\neg c) = 1$.*



3. Weights and probabilities

In this section we recall the formal relationship between weighted model coungint in propositional logic and probability. Let us first introduce the basic definition of probability measure.

3.1. Basic probability. A probability[1] space or event space is a set Ω together with a probability measure P on it. Ω is called the *sample space* and every element of ω is the otucome of some esperiment. Any subset of the sample space is called *event*.

Example 1.6.

1. *Tossing a coin. The sample space is* $\Omega = \{H, T\}$, $= \{H\}$ *is an event.*
2. *Tossing a die. The sample space is* $\Omega = \{1, 2, \ldots, 6\}$, $A = \{2, 4, 6\}$ *is an event, which can be described in words as "the result of the draw is an even number".*
3. *Tossing a coin twice. The sample space is* $\Omega = \{HH, HT, TH, TT\}$. $A = \{HH, HT\}$ *is an event, which can be described in words as "the frst toss results in a Head".*
4. *Tossing a die twice. The sample space is* $\Omega = \{(1, 1), (1, 2), \ldots, (6, 6)\}$, *which contains 36 elements. "The sum of the results of the two toss is equal to 10" is an event* $A = \{(i, j) \mid i + j = 10\}$.

Example 1.7. *The set* $\mathbb{I} = \{\mathcal{I} \mid \mathcal{I} : \mathcal{P} \to \{0, 1\}\}$ *of assignments of a set of propositional variable* $\mathcal{P}$ *is a sample space, and the set of interpretations that satisfies a formula* ϕ *is an event.*

P associates a number in $[0, 1]$ to every subset of Ω. A subset A of Ω is also called an *event*. This means that Pr associates the probability to each event $A \subseteq \Omega$

$$\Pr(A) = \text{probability of } A$$

with

$$0 \leq \Pr(A) \leq 1 \tag{8}$$

[1]This is not the most general definition of probability space, but it is sufficiend for our purposes.

The probability of the whole space Ω is normalized to be $\Pr(\Omega) = 1$ and the probability of the empty set is 0 i.e, $\Pr(\emptyset) = 0$. For an element $\omega \in \Omega$ we may call $\{\omega\}$ an atomic event, and write $\Pr(\omega)$ instead of the more precise notation $\Pr(\{\omega\})$ to denote it's probability.

For two disjoint subsets A and B of Ω

$$\Pr(A \cup B) = \Pr(A) + \Pr(B) \tag{9}$$

In this case we say that A and B are disjoint events.

When Ω is finite, every event A is equal to the union of the atomic events that are contained in A. More precisely, if $A = \{\omega_1, \ldots, \omega_k\}$, then A can be expressed the union of the atomic events,

$$A = \{\omega_1\} \cup \{\omega_2\} \cup \cdots \cup \{\omega_k\}$$

Notice that every pair $\{\omega_i\}$ and $\{\omega_j\}$ with $i \neq j$ $\{\omega_i\} \cap \{\omega_j\} = \emptyset$. This implies, that we can apply property (9), obtaining:

$$\Pr(A) = \Pr(\omega_1) + \Pr(\omega_2) + \cdots + \Pr(\omega_k)$$

In conclusion, if Ω is finite, the probability of every event can be obtained by summing the probability the atomic events that belong to the event A. In other words, if we know $\Pr(\omega)$ for every $\omega \in \Omega$ then we can compute the probability of every event A by

$$\Pr(A) = \sum_{\omega \in A} \Pr(\omega)$$

The requirment that $\Pr(\Omega) = 1$ imposes also the restution that

$$\sum_{\omega \in \Omega} \Pr(\omega) = 1$$

i.e, that Pr is normalized to 1.

An important notion in probability theory is that of *conditional probability.* Given two event A and B in Ω the probability of A conditioned by B, denoted by $\Pr(A \mid B)$ is defined as:

$$\Pr(A \mid B) = \frac{\Pr(A \cap B)}{\Pr(B)} \tag{10}$$

Example 1.8. *Suppose that you draw a dice and you know that the result is larger than* 3*, what is the probability that it is an odd number? Conditional probability $P(A \mid B)$ provide the answer to this question. In this case, the conditioning event is "the result of the toss is* > 3 *(B in the formula) and the conditioned event is "the result of the toss is odd", A in the formula. If we are in presence of a fair dice, we have that*

$$Pr(A \mid B) = \frac{Pr(A \cap B)}{Pr(B)} = \frac{1}{3}$$

3.2. From weights to probabilities. As shown in Example 1.7, one can see the set of all interpretations $\mathbb{I}$ of a set of propositional variables $\mathcal{P}$ as a sample space, where each interpretation is a single outcome of an experiment. This sample space contains $2^{|\mathcal{P}|}$ elements, and therefore it is finite. The weight function w maps every interpretation of $\mathcal{P}$ into a positive integer, we could therefore define the probability of an interpretation as the normalized weight:

$$\mathrm{Pr}(\mathcal{I}) = \frac{w(\mathcal{I})}{\sum_{\mathcal{I}\in\mathbb{I}} w(\mathcal{I})} \tag{11}$$

Notice that if $w(\mathcal{I}) \geq 0$ for every $\mathcal{I}$ then Pr is a probability measure. Indeed we have that The probability of the set of all interpretations is equal to 1:

$$\mathrm{Pr}(\mathbb{I}) = \sum_{\mathcal{I}\in\mathbb{I}} \mathrm{Pr}(\mathcal{I}) = \sum_{\mathcal{I}\in\mathbb{I}} \frac{w(\mathcal{I})}{\sum_{\mathcal{I}\in\mathbb{I}} w(\mathcal{J})} = \frac{\sum_{\mathcal{I}\in\mathbb{I}} w(\mathcal{I})}{\sum_{\mathcal{I}\in\mathbb{I}} w(\mathcal{I})} = 1$$

Every formula ϕ defines a set $\mathbb{I}_\phi = \{\mathcal{I} \in \mathbb{I} \mid \mathcal{I} \models \phi\}$ of interpretations, which contains all the interpretations that satisfies ϕ. This means that every ϕ corresponds to the event $\mathbb{I}_\phi \subseteq \mathbb{I}$. Given this corrispondence, we can identify the event $\mathbb{I}_\phi$ with a formula ϕ, which allows us to talk about a "probability of a formula ϕ" denoted by $\mathrm{Pr}(\phi)$, and defined by the following equation:

$$\mathrm{Pr}(\phi) = \sum_{\mathcal{I}\models\phi} \mathrm{Pr}(\mathcal{I}) \tag{12}$$

if we plug in (12) the definition of $\mathrm{Pr}(\mathcal{I})$ given in (11) we obtain

$$\mathrm{Pr}(\phi) = \sum_{\mathcal{I}\models\phi} \frac{w(\mathcal{I})}{\sum_{\mathcal{I}\in\mathbb{I}} w(\mathcal{I})} = \frac{\sum_{\mathcal{I}\models\phi} w(\mathcal{I})}{\sum_{\mathcal{I}\models\top} w(\mathcal{I})}$$

which is equal to

$$\mathrm{Pr}(\phi \mid w) = \frac{1}{Z(w)} \textsc{wmc}(\phi, w) \tag{13}$$

Weighted model counting allows also to define conditional probability. WHich is $\mathrm{Pr}(\phi \mid \psi)$. Let us apply the definition:

$$\mathrm{Pr}(\phi \mid \psi) = \frac{\mathrm{Pr}(\phi \cap \psi)}{\mathrm{Pr}(\psi)}$$

What is the event $\phi \cap \psi$?, In our convention ϕ denotes the event $\mathbb{I}_\phi$ of the propositional assignments that satisfies ϕ and similarly $\mathbb{I}_\psi$. Therefore $\phi \cap \psi$ denotes $\mathbb{I}_\phi \cap \mathbb{I}_\psi$, which is the set of propositional interpretations that satisfy both ϕ and ψ. This coincides with the event $\mathbb{I}_{\psi\wedge\psi}$, also written as $\phi \wedge \psi$. We therefore have that

$$\mathrm{Pr}(\phi \mid \psi) = \frac{\mathrm{Pr}(\phi \wedge \psi)}{\mathrm{Pr}(\psi)}$$

If we replace the definition of probability of a formula in terms of weighted model counting provided in (13) we obtain

$$\mathrm{Pr}(\phi \mid \psi) = \frac{\frac{\textsc{wmc}(\phi\wedge\psi)}{\textsc{wmc}(\top)}}{\frac{\textsc{wmc}(\psi)}{\textsc{wmc}(\top)}} = \frac{\textsc{wmc}(\phi \wedge \psi)}{\textsc{wmc}(\psi)} \tag{14}$$

4. wmc for inference in Bayesian Networks

Bayesian networks (BNs) are graphical models (graphical in the sense that they are based on a graph) for probabilistic reasoning. The basic structure of a Bayesian networks is a directed acyclic graph where the nodes represent variables (discrete or continuous) and edges represent direct probabilistic connections between them. These direct connections are often causal connections. In addition, BNs model the quantitative strength of the connections between variables, allowing probabilistic beliefs about them to be updated automatically as new information becomes available.

4.1. Boolean Bayesian networks. In this chapter we consider only bayesian network which are based only on boolean random variables, though bayesian networks are defined on any type of random variables.

A directed acyclic graph $C = (V, E)$ is a graph on the set of vertices V and with directed edges $E \subseteq V \times V$ that does not contains cycles. A cycle is a sequence $(v_0, v_1), (v_1, v_2), \ldots, (v_{n-1}, v_n)$ where $v_0 = v_n$ and for all $0 \leq i < n$ $(v_i, v_{i+1}) \in E$. Given a directed graph $G = (V, E)$ for every vertex $v \in V$, we define $\text{par}(v) = \{v' \mid (v', v) \in E\}$.

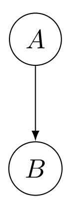
Definition 1.2 (Bayesian Network). *A* Bayesian network *on a set of random variables $\boldsymbol{X} = \{X_1, \ldots, X_n\}$ is a pair $\mathcal{B} = (G, Pr)$ is a pair composed of a directed acyclic graph $G = ([n], E)$ (where $[n] = \{1, \ldots, n\}$) and Pr specifies the conditional probababilities*

$$Pr(X_i = x_i \mid \boldsymbol{X}_{par(i)} = \boldsymbol{x}_{par(i)})$$

for every $X_i \in \boldsymbol{X}$. $\mathcal{B}$ uniquely define the join distribution on $\boldsymbol{X}$

$$Pr(\boldsymbol{X} = \boldsymbol{x}) = \prod_{i=1}^{n} Pr(X_i = x_i \mid \boldsymbol{X}_{par(i)} = \boldsymbol{x}_{par(i)}) \tag{15}$$

Example 1.9. *The following simple Bayesian Netsork*



$Pr(A) = 1$
0.3

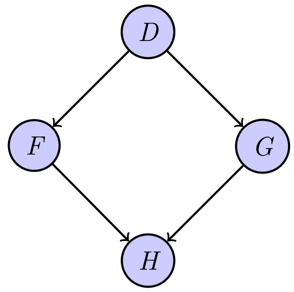
a	$Pr(B = 1 \mid A = a)$
0	0.4
1	0.9

specifies the joint probability distribution $P(A, B) = P(A) \cdot P(B \mid A)$ shown in the following table

a	b	$P(A = a, B = b)$
0	0	0.42
0	1	0.28
1	0	0.03
1	1	0.27

Example 1.10. *As a second example consider the Bayesian network shown in this picture taken from Sang, Beame, and Kautz 2005*

P(D=1)
0.5



d	$P(F = 1 \mid D = d)$
1	*0.6*
0	*0.1*

D	$P(G = 1 \mid D = d)$
1	*0.7*
0	*0.2*

f	*g*	$P(H = 1 \mid F = f, G = g)$
1	*1*	*1.0*
1	*0*	*0.5*
0	*1*	*0.4*
0	*0*	*0.0*

The above Bayesian Network specifies a probability distribution on four boolean random variables, D, F, G, and H. Since they are boolean, they can take two values 0 *and* 1*. As in propositional logic, boolean variables are used to express propositions, In this example we have that the propositional/random variables D, E, F and G stand for the following propositions:*

D :	*John is Doing some work*
F :	*John has Finished his work*
G :	*John is Getting tired*
H :	*John Has a rest*

Every node of the graph is associated with a table that expresses the probability of the variables conditioned to the values of the parents. Theset tables are called conditional probabuility table (CPT). For instance the table associated to the node F states that:

$$Pr(F = 1 \mid D = 1) = 0.6$$
$$P(F = 1 \mid D = 0) = 0.1$$

The table specifies only the conditional probability for one of the two values of the boolean variable, since the value of the other can be obtained by difference.

$$Pr(F = 0 \mid D = 1) = 1 - Pr(F = 1 \mid D = 1) = 0.5$$
$$Pr(F = 0 \mid D = 1\text{‘}) = 1 - Pr(F = 1 \mid D = 0) = 0.9$$

The above Bayesian Network specifies the following joint distribution on D, F, G, H

d	f	g	h	$Pr(D, F, G, H = d, f, g, h)$
0	0	0	0	$0.5 \cdot 0.9 \cdot 0.8 \cdot 1.0 = 0.360$
0	0	0	1	$0.5 \cdot 0.9 \cdot 0.8 \cdot 0.0 = 0.000$
0	0	1	0	$0.5 \cdot 0.9 \cdot 0.2 \cdot 0.6 = 0.054$
0	0	1	1	$0.5 \cdot 0.9 \cdot 0.2 \cdot 0.4 = 0.036$
0	1	0	0	$0.5 \cdot 0.1 \cdot 0.8 \cdot 0.6 = 0.024$
0	1	0	1	$0.5 \cdot 0.1 \cdot 0.8 \cdot 0.4 = 0.016$
0	1	1	0	$0.5 \cdot 0.1 \cdot 0.2 \cdot 0.0 = 0.000$
0	1	1	1	$0.5 \cdot 0.1 \cdot 0.2 \cdot 1.0 = 0.010$
1	0	0	0	$0.5 \cdot 0.4 \cdot 0.3 \cdot 1.0 = 0.060$
1	0	0	1	$0.5 \cdot 0.4 \cdot 0.3 \cdot 0.0 = 0.000$
1	0	1	0	$0.5 \cdot 0.4 \cdot 0.7 \cdot 0.6 = 0.084$
1	0	1	1	$0.5 \cdot 0.4 \cdot 0.7 \cdot 0.4 = 0.056$
1	1	0	0	$0.5 \cdot 0.6 \cdot 0.3 \cdot 0.5 = 0.045$
1	1	0	1	$0.5 \cdot 0.6 \cdot 0.3 \cdot 0.5 = 0.045$
1	1	1	0	$0.5 \cdot 0.6 \cdot 0.7 \cdot 0.0 = 0.000$
1	1	1	1	$0.5 \cdot 0.6 \cdot 0.7 \cdot 1.0 = 0.210$
				1.000

From the previous example it should be clear that a Bayesian Network $\mathcal{B}$ with boolean variables expresses a probility distribution on a set of inbterpretations on the propositional variables corresponding to the random variables of $\mathcal{B}$. Therefore, An assignment to the random variables corresponds to a propositional interpretation. To highlight this fact from now on we use $\mathcal{P}$ to denot the set of random boolean variables (propositional variables) and $\mathcal{I}$ to denote an assignment to them.

There are many possible tasks that can be done with a given Bayesian network $\mathcal{B} = (G, Pr)$ on a set of propositional variables $\mathcal{P}$. The task we consider here is *Conditional Probability Queries.*

DEFINITION 1.3 (Conditional Probability Queries). *Given a Bayesian Network $\mathcal{B}$ on a set of propositional variables $\mathcal{P}$ a conditional probability queries is an expression of the form $Pr_{\mathcal{B}}(\phi \mid \psi)$, where ψ is called the evidence, and ϕ the query. The anwer to this query is*

$$\frac{\Pr_{\mathcal{B}}(\phi \wedge \psi)}{\Pr_{\mathcal{B}}(\phi)} = \frac{\sum_{\mathcal{I} \models \phi \wedge \psi} \Pr_{\mathcal{B}}(\mathcal{I})}{\sum_{\mathcal{I} \models \psi} \Pr_{\mathcal{B}}(\mathcal{I})} \tag{16}$$

In the previous example, If we want to know the probability that the work is finished (F) given the fact that we see John at the sun having a rest (H), we have to evaluate $Pr(F \mid H)$; in this case F is the query and H is the evidence.

4.2. Answering Conditional Probability Queries via wmc. Weighted model counting can be used to compute the value $\Pr_{\mathcal{B}} \phi \mid \psi$. To this purpose we exploit the relation between weighted model counting and probability, i.e., that

$$Pr(\phi|\psi) = \frac{\text{WMC}(\phi \wedge \psi, w)}{\text{WMC}(\psi, w)}$$

Therefore, the problem become to define a proper weight function w corresponding to a given Bayesian Network. The method is based on the following steps

(1) Extend the set of propositional variables $\mathcal{P}$ of the Bayesian Network, which are called *state variables* with the following new variables called *chance variables* For every state variable p that has parents $k > 0$ parent variables introduce 2^k new propositional variables $p_{\boldsymbol{b}}$ for every $\boldsymbol{b} \in \{0,1\}^k$. In the previous example we introduce G_0 and G_1, F_0 and F_1, and H_{00}, H_{01}, H_{10}, and H_{11}.

(2) To each of the introduced variables associate the weight specified in the corresponding line of the CPT. I.e., $w(p_{\boldsymbol{b}}) = \Pr(p = 1 \mid \text{par}(p) = \boldsymbol{b})$. In the above example we have the following weights:

$$
\begin{aligned}
w(D) &= 0.5 & & \\
w(F_0) &= 0.1 & w(F_1) &= 0.5 \\
w(G_0) &= 0.2 & w(G_1) &= 0.7 \\
w(H_{00}) &= 0.0 & w(H_{01}) &= 0.4 \\
w(H_{10}) &= 0.5 & w(H_{11}) &= 1.0
\end{aligned}
$$

The weight of $\neg p_{\boldsymbol{b}}$ is set to $1 - w(p_{\boldsymbol{b}})$, and the weights of all the other literals are set to 1.

(3) The intuition of the choice variable $p_{\boldsymbol{b}}$ is that it will be true of the parent variables of p will take the values $\boldsymbol{b}$. So for instance F_0 it means that the only parent of F, which is D will be false. The third step is to connect the new variables to the variables in the graph; For every change variable $p_{\boldsymbol{b}}$ correspinding to a line $Pr(p = 1 \mid \text{par}(p) = \boldsymbol{b})$, we add the following formula, where $\text{par}(p) = \{p_1, \ldots, p_k\}$

$$
p_{\boldsymbol{b}} \leftrightarrow p \wedge \left(\bigwedge_{\substack{i=1 \\ b_i=1}}^{k} p_i \wedge \bigwedge_{\substack{i=1 \\ b_i=0}}^{k} \neg p_i \right)
$$

In our example we add the following formulas:

$$
\begin{array}{llll}
F_0 \leftrightarrow F \wedge \neg D & F_1 \leftrightarrow F \wedge D & G_0 \leftrightarrow G \wedge \neg D & G_1 \leftrightarrow G \wedge D \\
H_{00} \leftrightarrow H \wedge \neg F \wedge \neg G & H_{01} \leftrightarrow H \wedge \neg F \wedge G & H_{10} \leftrightarrow H \wedge F \wedge \neg G & H_{11} \leftrightarrow H \wedge F \wedge G
\end{array}
$$

(4) A further obtimization consistes in replacing literals with weight equal to 0 with formulas. In particular, if $w(l) = 0$ add the unweighted formula $\neg l$. and remove the weight of l. In the above example for instance we remove the weights for H_{00} and add $\neg H_{00}$.

Let $\Phi_{\mathcal{B}}$ and $w_{\mathcal{B}}$ be the conjunction of the formulas and the weight function obtained by applying the previous steps to Bayesian network $\mathcal{B}$, We then define the weight function

$$
w_{\mathcal{B}}(\mathcal{I}) = \begin{cases} \prod_{\mathcal{I} \models p_{\boldsymbol{b}}} w(p_{\boldsymbol{b}}) \cdot \prod_{\mathcal{I} \not\models p_{\boldsymbol{b}}} w(\neg p_{\boldsymbol{b}}) & \text{if } \mathcal{I} \models \Phi_{\mathcal{B}} \\ 0 & \text{Otherwise} \end{cases}
$$

Notice that w is more than a weight function. Indeed it is a probability distribution on the models of $\Phi_{\mathcal{B}}$, since the partition function $Z(w_{\mathcal{B}}) = \sum_{\mathcal{I}} w(\mathcal{I}) = \sum_{\mathcal{I} \models \Phi_{\mathcal{B}}} w(\mathcal{I}) = 1$, i.e., the weight of all the models of $\Phi_{\mathcal{B}}$ sum up to 1. Indeed we have the following proposition:

PROPOSITION 1.5. *Let $\mathcal{B}$ be a Bayesian networks on the boolean random variables $X_1, \ldots, X_n$ that defines the joint probability distribution $Pr(X_1, \ldots, X_n)$.*

- *for every assignment $\boldsymbol{x} = (x_1, \dots, x_n)$ to the variables $X_1, \dots, X_n$. there is a unique interpretaiton $\mathcal{I}_{\boldsymbol{x}}$ that satisfies $\Phi_{\mathcal{B}}$ and such that $\mathcal{I}(X_i) = x_i$*
- *For every $\mathcal{I}$ that satisfies $\Phi_{\mathcal{B}}$*

$$w_{\mathcal{B}}(\mathcal{I}) = Pr(X_1 = \mathcal{I}(X_1), \dots, X_n = \mathcal{I}(X_n))$$

The answer to the conditional probability query $Pr(\phi \mid \psi)$ w.r.t, the Bayesian network $\mathcal{B}$ can be computed via weighted model counting.

$$\frac{\text{WMC}(\Phi_{\mathcal{B}} \wedge \phi \wedge \psi, w_{\mathcal{B}})}{\text{WMC}(\Phi_{\mathcal{B}} \wedge \psi, w_{\mathcal{B}})} \tag{17}$$

To this purpose we can use for instance knowledge compilation of $\Phi_{\mathcal{B}}$ in an sd-DNNF formula in order to compute the circuit. Notice that if the evidence ψ is a conjunction of literals we can set the weights of the opposite literal to 0 in the circuit of $\Phi_{\mathcal{B}}$ and we immediately obtain a circuit for $\Phi \wedge \psi$ Similarly, if the query ϕ is a conjunction of literals. THerefore in case of conjunctive conditional probability queryies (i.e., queries in which the evidence and the query are conjunctions of literals) once we have computed the circuit for $\Phi_{\mathcal{B}}$ we can easily computeall the answers of conditional probability conjunctive queries.

EXAMPLE 1.11. *The sd-DNNF of the $\Phi_{\mathcal{B}}$ for the previous example is*

$$\begin{array}{r} D \wedge (F \wedge F_1 \wedge (G \wedge G_1 \wedge (H \wedge H_{11} \vee \neg H \wedge \neg H_{11})) \vee \\ (\neg G \wedge \neg G_1 \wedge (H \wedge H_{10} \vee \neg H \wedge \neg H_{10}))) \vee \\ (\neg F \wedge F_1 \wedge (G \wedge G_1 \wedge (H \wedge H_{01} \vee \neg H \wedge \neg H_{01})) \vee \\ (\neg G \wedge \neg G_1 \wedge (H \wedge H_{00} \vee \neg H \wedge \neg H_{00}))) \vee \\ \neg D \wedge (F \wedge F_0 \ \wedge (G \wedge G_0 \wedge (H \wedge H_{11} \vee \neg H \wedge \neg H_{11})) \vee \\ (\neg G \wedge \neg G_0 \wedge (H \wedge H_{10} \vee \neg H \wedge \neg H_{10}))) \vee \\ (\neg F \wedge F_0 \ \wedge (G \wedge G_0 \wedge (H \wedge H_{01} \vee \neg H \wedge \neg H_{01})) \vee \\ (\neg G \wedge \neg G_0 \wedge (H \wedge H_{00} \vee \neg H \wedge \neg H_{00}))) \end{array} \tag{18}$$

Formula (18) *can be converted in a sum-product circuit. $C_{\mathcal{B}}$ that cam be used to compute the probability of any conjunctive formula. For instance to compute the probability of $H \wedge \neg G$, it is sufficient to set the weights of the literals $\neg H$ and G to* 0. *and compute the circuit. Therefore if we want to answer the conditional conjunctive query $Pr(D \mid H \wedge \neg G)$ we use the circuit to compute $Pr(D \wedge H \wedge \neg G)$ and $Pr(H \wedge \neg G)$. The anwer will be the fraction of the two results i.e.,*

$$\frac{Pr(D \wedge H \wedge \neg G)}{Pr(H \wedge \neg G)}$$

5. Learning weights

Until now, we have assumed that the weights associated to literals are given. In this last section, we describe a basic method to automatically learn the weights from a set of observations.

First, we have to define what is an observation. Since we want to learn a weight function (or a probability distribution) on the set of interpretations of a set of propositional variables $\mathcal{P}$ then the observations must be instances of such assugnments. Suppose we have a set of observations which are represented as a multi-set (i.e., a set with repeated objects) of interpretations $\mathbb{I} = \mathcal{I}^{(1)}, \mathcal{I}^{(2)}, \dots, \mathcal{I}^{(d)}$ where d the the size of the observations. The criteria used to learn the weights is that of the *maximum likelihood* that requires to maximize the probability of observing the data. Formally, we want to find a set of parameters (weights) $\boldsymbol{w}$ such that

$$\text{Likelihood}(\mathbb{I} \mid \boldsymbol{w}) = \Pr(\mathbb{I} \mid \boldsymbol{w}) \tag{19}$$

is maximal. We make the following simplifying hypothesis .The first one is that the observations are i.i.d. (independent and identically distributed). This hypothesis allows us to factorize the probability of all the observations as a product of the probability of each single observation.

$$\text{Likelihood}(\mathbb{I} \mid \boldsymbol{w}) = \prod_{i=1}^{d} \Pr(\mathcal{I} \mid \boldsymbol{w})$$

The second hypothesis concerns the form of $Pr(\cdot \mid \boldsymbol{w})$. We assume that this probability is specified via weighted model counting, i.e., that

$$Pr(\mathcal{I} \mid \boldsymbol{w}) = \frac{1}{Z(\boldsymbol{w})} w(\mathcal{I} \mid \boldsymbol{w})$$

The third and final hypothesis is that $w(\mathcal{I} \mid \boldsymbol{w})$ is specified by an exponential form w.r.t a given set of formulas $\phi_1, \ldots, \phi_n$. This means that

$$w(\mathcal{I} \mid \boldsymbol{w}) = \exp\left(\sum_{i=1}^{n} w_i \mathcal{I}(\phi_i)\right)$$

The problem consists in finding a tuple of weights $\boldsymbol{w} = (w_1, \ldots, w_n)$ that best fits the observed data. Putting everything toghether we are interested in the vector of real number $\boldsymbol{w} = (w_1, \ldots, w_n)$ that meximizes

$$\text{Likelihood}(\mathbb{I} \mid \boldsymbol{w}) = \frac{1}{Z(\boldsymbol{w})^d} \exp\left(\sum_{i=1}^{d}\sum_{j=1}^{n} w_j \mathcal{I}^{(i)}(\phi_j)\right)$$

$$Z(\boldsymbol{w}) = \sum_{\mathcal{I}} \exp\left(\sum_{j=1}^{n} w_j \mathcal{I}(\phi_j)\right)$$

It is convenient to pass to the logarithmic space. Indeed due to the monotonicity of the logarithmic function, we have that maximinzing a function $f(x)$ is equivalent to maximizing the logarithm of the function i.e. maximizing $\log(f(x))$. We therefore want to maximize the logarithm of the likelihood, also known as loglikelihood

$$\begin{aligned}
\text{LogLik}(\mathbb{I} \mid \boldsymbol{w}) &= \log(\text{Likelihood}(\mathbb{I} \mid \boldsymbol{w})) \\
&= \sum_{i=1}^{d}\sum_{j=1}^{n} w_j \cdot \mathcal{I}^{(i)}(\phi_j) - d \cdot \log(Z(\boldsymbol{w})) \qquad (20) \\
&= \sum_{j=1}^{n}\sum_{i=1}^{d} \; w_j \cdot \mathcal{I}^{(i)}(\phi_j) - d \cdot \log(Z(\boldsymbol{w})) \\
&= \sum_{j=1}^{n} w_j \cdot n_j - d \cdot \log(Z(\boldsymbol{w})) \qquad (21)
\end{aligned}$$

where n_j is the number of observations $\mathcal{I}^{(i)}$ for which the fornula ϕ_j is true. We can try to maximize (21) with gradient ascent approach, by putting to zeros the partial derivatives of the log likelihood w.r.t. the parameters w_j.

$$
\begin{aligned}
\frac{\partial \mathrm{logLik}(\mathbb{I} \mid \boldsymbol{w})}{\partial w_i} &= 0 \\
\frac{\partial \left(\sum_{j=1}^{n} w_j \cdot n_j - d \cdot \log(Z(\boldsymbol{w}))\right)}{\partial w_j} &= 0 \\
n_j - \frac{d \cdot \sum_{\mathcal{I}} \mathcal{I}(\phi_j) \cdot \exp\left(\sum_{j=1}^{n} w_j \cdot \mathcal{I}(\phi_j)\right)}{Z(\boldsymbol{w})} &= 0
\end{aligned}
$$

requires exponential amount of time. We can use an approximation by learning the weight of each formulas separately. I.e., we assume that it is the only formula that we use to cmpute the weight function. We therefore consider the case where we have a unique formula ϕ_1 and compute the derivative w.r.t. the only parameter w_1.

$$
\begin{aligned}
n_1 - \frac{d \cdot \sum_{\mathcal{I}} \mathcal{I}(\phi_1) \cdot \exp(w_1 \cdot \mathcal{I}(\phi_1))}{Z(\boldsymbol{w})} &= 0 \\
n_1 - \frac{d \cdot \sum_{\mathcal{I} \models \phi_1} \cdot \exp(w_1)}{Z(\boldsymbol{w})} &= 0 \\
n_1 - \frac{d \cdot \#\textsc{sat}(\phi_1) \cdot \exp(w_1)}{Z(\boldsymbol{w})} &= 0 \\
n_1 - \frac{d \cdot \#\textsc{sat}(\phi_1) \cdot \exp(w_1)}{\sum_{\mathcal{I} \models \phi_1} \exp(w_1) + \sum_{\mathcal{I} \not\models \phi_1} \exp(0)} &= 0 \\
\frac{d \cdot \#\textsc{sat}(\phi_1) \cdot \exp(w_1)}{\#\textsc{sat}(\phi_1) \cdot \exp(w_1) + \#\textsc{sat}(\neg\phi_1)} &= n_1 \\
n_1 \cdot \exp(w_1) \cdot \#\textsc{sat}(\phi_1) + n \cdot \#\textsc{sat}(\neg\phi_1) &= d \cdot \exp(w_1) \cdot \#\textsc{sat}(\phi_1) \\
\frac{n_1 \cdot \#\textsc{sat}(\neg\phi_1)}{(d - n_1)\#\textsc{sat}(\phi_1)} &= \exp(w_1) \\
w_1 &= \log\left(\frac{n_1 \cdot \#\textsc{sat}(\neg\phi_1)}{(d - n_1) \cdot \#\textsc{sat}(\phi_1)}\right)
\end{aligned}
$$

IN the following we summarize how weighted model counting can be used to perform probabilistic prediction starting from a set of observations. Suppose that you are interested in doing some predictions which are expressible with a propositional formula Q starting from a set of evidences which are expressed in terms of a propositional formula E. In synthesis you want learn a set of parameters $\boldsymbol{w}$ that allows you to answer the query $Pr(Q \mid E, \boldsymbol{w})$.

(1) collect a set of d observations, i.e., $\mathcal{I}^{(1)}, \ldots, \mathcal{I}^{(d)}$ from which you want to extract knowledge that can be used to answer your query.
(2) select a set set of propositional formulae $\phi_1, \ldots, \phi_n$ that you want to use to describe the properties of your data. How to choose this formula is a matter of design. A possible criteria is to conisder formulas such that their truth value has some impact on the answer to the query $\Pr(Q \mid E, \boldsymbol{w})$.
(3) learn the weights $w_1, \ldots, w_k$, separately using formula:

$$
w_j = \log\left(\frac{n_j \cdot \#\textsc{sat}(\neg\phi_j)}{(d - n_j) \cdot \#\textsc{sat}(\phi_j)}\right) \tag{22}
$$

(4) apply inference i.e., compute $\Pr(Q \mid E, \boldsymbol{w})$ via weighted model counting. i.e.

$$\Pr(Q \mid E, \boldsymbol{w}) = \frac{\text{WMC}(Q \wedge E, \boldsymbol{w})}{\text{WMC}(E, \boldsymbol{w})}$$

EXAMPLE 1.12. *Suppose that we have* $\mathbb{I} = \mathcal{I}^{(1)}, \ldots, \mathcal{I}^{(22)}$ *are summarized in the following table:*

#	Itemsets						
4	a	b	c	d			
1	a	b			e	f	
7	a	b	c				
3	a		c	d		f	
2							g
1				d			
4				d			g

Suppose that we are interested in answering query of the form $Pr(x \mid y \wedge \neg z)$ for every $x, y, z \in \{a, b, c, d, e, f, g\}$. We can learn the weights of a set of formulas. See for instance the following (randomly choosen)

$$
\begin{array}{rl}
a & w = \log\left(\dfrac{15 \cdot 2^6}{7 \cdot 2^6}\right) \approx 0.76 \\
\neg a & w = \log\left(\dfrac{7 \cdot 2^6}{15 \cdot 2^6}\right) \approx -0.76 \\
e & w = \log\left(\dfrac{1 \cdot 2^6}{21 \cdot 2^6}\right) \approx -3.04 \\
\neg e & w = \log\left(\dfrac{21 \cdot 2^6}{1 \cdot 2^6}\right) \approx 3.04 \\
a \wedge b & w = \log \dfrac{12 \cdot (2^7 - 2^5)}{10 \cdot 2^5} \approx 8.21 \\
c \wedge d & w = \log \dfrac{7 \cdot (2^7 - 2^5)}{15 \cdot 2^5} \approx 7.27 \\
e \wedge f & w = \log \dfrac{1 \cdot (2^7 - 2^5)}{21 \cdot 2^5} \approx 4.99 \\
a \rightarrow b & w = \log \dfrac{19 \cdot (2^7 - 3 \cdot 2^5)}{3 \cdot 3 \cdot 2^5} \approx 0.75 \\
a \wedge b \wedge c \wedge \neg e \wedge \neg f \rightarrow g & w = \log\left(\dfrac{11}{11 \cdot (2^7 - 2)}\right) \approx -4.84 \\
a \wedge b \wedge \neg c \wedge \neg d \wedge e \wedge f \wedge \neg g & w = \log(21 \cdot (2^7 - 1)) \approx 7.89
\end{array}
$$

And then estimate for instance $Pr(a \mid b, \neg c)$ via weighted model counting.

6. Exercises

Exercise 1:

Given a set of propositional variable $\mathcal{P}$, let $w(\mathcal{I}) = |\{p \in \mathcal{P} \mid \mathcal{I} \models p\}|$, i.e., $w(\mathcal{I})$ is the number of propositional variables that are true in $\mathcal{I}$. Compute WMC($\top$).

Exercise 2:

Given a set of propositional variable $\mathcal{P}$, let $w(\mathcal{I}) = x^{|\{p \in \mathcal{P} | \mathcal{I} \models p\}|}$ for some $x \neq 0$. Compute WMC($\top$).

Exercise 3:

Let $\mathcal{P} = \{p, q\}$ be a set of propositional variables. Check if the following two weight functions on the interpretations of $\mathcal{P}$ can be expressed in terms of a weight function on the literals of $\mathcal{P}$.

(23)

$\mathcal{I}$	p	q	$w(\mathcal{I})$
$\mathcal{I}_1$	0	0	0
$\mathcal{I}_2$	0	1	1
$\mathcal{I}_3$	1	0	2
$\mathcal{I}_4$	1	1	3

Solution A weight funciton on the interpretation can be expressed in terms of a weight function on the literals, if the weight of the interpretation is equal to the product of the weight of the literals that are true in the interpretation, i.e., if

$$w(\mathcal{I}) = \prod_{p \in \mathcal{P}} w(p)^{\mathcal{I}(p)} w(\neg p)^{\mathcal{I}(\neg p)}$$

Therefore, to be expressed in term of a weight funciton on literals the weight function (23) should be such that

$$\begin{aligned} w(\neg p) \cdot w(\neg q) &= 0 \\ w(\neg p) \cdot w(q) &= 1 \\ w(p) \cdot w(\neg q) &= 2 \\ w(p) \cdot w(q) &= 3 \end{aligned}$$

THe first equation implies that either $w(\neg p) = 0$ or $w(\neg q) = 0$; but the second equation implies that $w(\neg p) \neq 0$ and the third equation implies that $w(\neq q) \neq 0$, which is impossible. Therefore the weight function (23) cnanot be expressed in terms of weight funciton on literals. $\square$

Exercise 4:

Suppose you have three coins: the faces of the first coin are black and white, the faces of the second coin are yellow and green, and the faces of the third coin are red and green. In an experiment you toss the first coin; if you obtain a black you toss the second coin otherwise you toss the third coin.

(1) Model this experiment in propositional logic and
(2) use model counting to determine what are the number of possible outcomes?
(3) Let p, q and r be the probability of obtaining a black, yellow, and red faces when tossing the first, second and third coin respectively. Compute the probability of obtaining an outcome which is either red or green.

Solution We can use the language B, W, R, G, Y to state that in the outcome there is a coin with a black, white, red, green, and yellow faces respectively. Notice that this is possible since there is no possibility to have outcomes with two coins with the same color face. We can now formalize the constraints of the game in terms of the following formulas:

$B + W = 1$	The toss of the first coin can have only one result among black and white
$R + Y + G = 1$	The toss of the second or third coin can have only one result since only one coin among the two is tossed
$B \to Y \vee G$	If you have a black then you can have only one among yellow and green since you toss the second coin
$W \to R \vee G$	If you have a white then you can have only one among red and green since you toss the third coin

The models that satisfies all the formulas are 4

$$\{B, Y\} \qquad \{B, G\} \qquad \{W, R\} \qquad \{W, G\}$$

If we associate the following weights:

$$w(B) = p \quad w(W) = 1 - p \quad w(Y) = q \quad w(G) = 1 - q \quad w(R) = r \quad w(G) = 1 - r$$

In this wey however we assign two weights to the same atom G. Indeed we have to distinguish when G is obtained by the second or by the third coin. To this purpose we introduce two new atoms

$$G_2 \leftrightarrow B \wedge G \qquad G_3 \leftrightarrow W \wedge G$$

Adding these new propositions does not change the number of models since they are fully defined in terms of the previous propositions. We still have 4 models by they are:

$$\{B, Y\} \qquad \{B, G, G_2\} \qquad \{W, R\} \qquad \{W, G, G_3\}$$

and update the weights for G to

$$w(G_2) = 1 - q \qquad w(G_3) = 1 - r$$

$$w(\{B, Y\}) = pq \quad w(\{B, G, G_2\}) = p(1 - q) \quad w(\{W, R\}) = (1 - p)r \quad w(\{W, G, G_3\}) = (1 - p))(1 - r)$$

Notice that the weight of all the models sum up to 1, and therefore they can be considered to be probabilities of the outcomes. Finally, the probability of in the result we have either red or green, is equal to the probability of the formula $R \vee G$ which can be computed by the sum of the probabilities of the models that satisfies $R \vee G$, i.e., $p(1 - q) + (1 - p)r + (1 - p)(1 - r) = p(1 - q) + 1 - p = 1 - pq$. $\square$

Exercise 5:

Consider the set $\mathcal{P} = \{p, q\}$ of propositional variables and the following weight functions:

$\mathcal{I}$	p	q	$w_1(\mathcal{I})$
$\mathcal{I}_1$	0	0	1
$\mathcal{I}_2$	0	1	2
$\mathcal{I}_3$	1	0	3
$\mathcal{I}_4$	1	1	4

$\mathcal{I}$	p	q	$w_2(\mathcal{I})$
$\mathcal{I}_1$	0	0	4
$\mathcal{I}_2$	0	1	3
$\mathcal{I}_3$	1	0	8
$\mathcal{I}_4$	1	1	6

Check if they can be expressed in terms of a weight function $w : \mathcal{P} \to \mathbb{R}^+$.

Solution A weight function maps every interpretation of the propositional variables into a positive real number. This mapping can be specified explicilty, by providing the explicit weight for every interpretation (we have to specify 2^n numbers, where

n is the number of propositional variables) This is how we have specified the two weight functions w_1 and w_2 of the exercise. Alternatively, and more compactly, the weight function can be specified indirectly, by associating a weight to every literal, and then define the weight of an assignment as the product of the weight of the literals that are satisfied by the assignment. i.e.,

$$w(\mathcal{I}) = \prod_{\substack{l \in Literals \\ \mathcal{I} \models l}} w(l)$$

With this method instead of specifying 2^n numbers we have to provide at most $2n$ parameters (corresponding to the number of literals). However, this is not always possible, i.e., there are weight functions that cannot be specified in terms of a weight function on the literals. The exercise asks if the weight functions w_1 and w_2 can or cannot be specified by a weight function defined on literals.

Let's start with w_1. If w_1 can be specified with the weight of the literals, then we have that

$$\begin{cases} w_1(\mathcal{I}_1) = 1 = w(\neg p) \cdot w(\neg q) \\ w_1(\mathcal{I}_2) = 2 = w(\neg p) \cdot w(q) \\ w_1(\mathcal{I}_3) = 3 = w(p) \cdot w(\neg q) \\ w_1(\mathcal{I}_4) = 4 = w(p) \cdot w(q) \end{cases}$$

To find the weight function for the literals, we have therefore to solve the following system of equations, where we have replaced $w(\neg p)$, $w(\neg q)$, $w(q)$, and $w(p)$, with a, b, c and d.

$$\begin{cases} a \cdot b = 1 \\ a \cdot c = 2 \\ d \cdot b = 3 \\ d \cdot c = 4 \end{cases} \Rightarrow \begin{cases} a = \frac{1}{b} \\ c = 2b \\ d = \frac{3}{b} \\ d = \frac{2}{b} \end{cases}$$

which does not have a solution. This implies that the weight funciton w_1 cannot be expressed in terms of weight function on literals. Let us now consider w_2. We proceed in the same way:

$$\begin{cases} a \cdot b = 4 \\ a \cdot c = 3 \\ d \cdot b = 8 \\ d \cdot c = 6 \end{cases} \Rightarrow \begin{cases} b = \frac{4}{a} \\ c = \frac{3}{a} \\ d = 2a \end{cases}$$

which has infinite solution for $a \neq 0$, for instance:

$$\begin{array}{llll} w(\neg p) = \dfrac{1}{2} & w(\neg q) = 8 & w(p) = 1 & w(q) = 6 \\ w(\neg p) = 1 & w(\neg q) = 4 & w(p) = 2 & w(q) = 3 \end{array}$$

□

Exercise 6:

Consider the following weighted formulas

literal	A	$\neg A$	B	$\neg B$	C	$\neg C$
weight	1	2	1	2	1	2

Compute the weight and the probabilities of the formulas

$$(A \vee B) \to (B \vee C)$$

.

Solution Let us compute the weights of all the interpretations.

	A	B	C	$w(\mathcal{I})$
$\mathcal{I}_1$	0	0	0	8
$\mathcal{I}_2$	0	0	1	4
$\mathcal{I}_3$	0	1	0	4
$\mathcal{I}_4$	0	1	1	2
$\mathcal{I}_5$	1	0	0	4
$\mathcal{I}_6$	1	0	1	2
$\mathcal{I}_7$	1	1	0	2
$\mathcal{I}_8$	1	1	1	1

Notice that the models of $\phi = (A \vee B) \to (B \vee C)$ are all but $\mathcal{I}_5$. therefore the weight of the fomrula is the total weight (i.e., the weight of $\top$) minus the weight of $\mathcal{I}_5$. IN summary

$$\begin{aligned} w(\top) &= \sum_{i=1}^{8} w(\mathcal{I}_i) = 27 \\ w(\phi) &= w(\top) - w(\mathcal{I}_5) = 23 \\ Pr(\phi) &= \frac{w(\phi)}{w(\top)} = \frac{23}{27} \end{aligned}$$

□

Exercise 7:
Consider the following weighted formulas

$$\begin{array}{ccc} weight & : & literal \\ 1 & : & A \\ 2 & : & \neg A \\ 1 & : & B \\ 2 & : & \neg B \\ 1 & : & C \\ 2 & : & \neg C \end{array}$$

Compute the weight and the probabilities of the formulas

$$(A \vee B) \to (B \vee C)$$

Remember that in weighted model counting you multiply the weights of the literal that are true. **Solution**

A	B	C	w	$A \vee B \to B \vee C$
0	0	0	$2^3 = 8$	1
0	0	1	$2^2 = 4$	1
0	1	0	$2^3 = 4$	1
0	1	1	$2^1 = 2$	1
1	0	0	$2^2 = 4$	0
1	0	1	$2^1 = 2$	1
1	1	0	$2^1 = 2$	1
1	1	1	$2^0 = 1$	1

the weighted model counting of the $A \vee B \to B \vee C$ is equal to 23, and the probability is $\frac{23}{27} \approx 0.85$. $\square$

Exercise 8:

Provide a weight function $W : \mathcal{P} \to \mathbb{R}^+$, that is equivalent to weight function defined in the previous exercise. Explain why such a weight function does not exist for the weight defined in Exercise 6.

Exercise 9:

Given the following observations on the items bought by people.

#	*Itemsets*						
4	a	b	c	d			
1		b			e	f	g
7	a	b	c				
3	a		c		e	f	
2							g
1		b			e		
4	a		c	d			g

Learn the weights of the following formulas:

(1) $a \wedge b \to c$
(2) $b \wedge c \to d \vee f$

Exercise 10:

Explain the relation between weight function and probability distribution on the set of interpretation.

Exercise 11:

Compute the weight for the following formula: $(b \to \neg c) \vee (d \leftrightarrow f)$ from the following itemset:

#	*Itemsets*						
4		b	c	d			
2	a				e	f	
6	a	b	c				
1	a		c	d		f	
3	a		c		e		g
5				d			
9		b		d	e		g

Solution We have to apply the formula for computing the weight given a set of observations, i.e.,

$$w = \ln\left(\frac{n \cdot \#SAT(\neg\phi)}{(d-n)\#SAT(\phi)}\right)$$

where n is the number of observations for which the fornula ϕ is true and d is the total number of observations. Let us first compute $\#\textsc{sat}((b \to \neg c) \vee (d \leftrightarrow f))$. We do it by truth table

b	c	d	f	(b	→	¬	c)	∨	(d	↔	f)
1	1	1	1	1	0	0	1	1	1	1	1
1	1	1	0	1	0	0	1	0	1	0	0
1	1	0	1	1	0	0	1	0	0	0	1
1	1	0	0	1	0	0	1	1	0	1	0
1	0	1	1	1	1	1	0	1	1	1	1
1	0	1	0	1	1	1	0	1	1	0	0
1	0	0	1	1	1	1	0	1	0	0	1
1	0	0	0	1	1	1	0	1	0	1	0
0	1	1	1	0	1	0	1	1	1	1	1
0	1	1	0	0	1	0	1	1	1	0	0
0	1	0	1	0	1	0	1	1	0	0	1
0	1	0	0	0	1	0	1	1	0	1	0
0	0	1	1	0	1	1	0	1	1	1	1
0	0	1	0	0	1	1	0	1	1	0	0
0	0	0	1	0	1	1	0	1	0	0	1
0	0	0	0	0	1	1	0	1	0	1	0
								14			

Since we have to take into consideration also other three propositional variables not appearing in the formula we have that we have that

$$\begin{aligned}
\#\textsc{sat}((b \to \neg c) \vee (d \leftrightarrow f)) &= 14 \\
\#\textsc{sat}(\neg((b \to \neg c) \vee (d \leftrightarrow f))) &= 2^7 - \#\textsc{sat}((b \to \neg c) \vee (d \leftrightarrow f)) = 16 - 14 = 2 \\
d &= 4 + 2 + 6 + 1 + 3 + 5 + 9 = 30 \\
n &= 26
\end{aligned}$$

By replacing this values in the formula we obtain

$$w = \ln\left(\frac{26 \cdot 2}{4 \cdot 14}\right) \approx \ln(0.9286) \approx -0.74$$

□

Exercise 12:
Prove that the formulas

$$\forall y(P(y) \wedge \exists x Q(x)) \qquad \forall y P(y) \wedge \exists x Q(x)$$

are equivalent. (Suggestion: you have to show that every interpretation satisfies the first formula if and only if it satisfies the second one).

Solution

$$
\begin{aligned}
\mathcal{I} \models \forall y(P(y) \wedge \exists x Q(x)) &\iff \text{for all } d \in \Delta^{\mathcal{I}}, \mathcal{I} \models P(x) \wedge \exists y P(y)[a_{x \leftarrow d}] \\
&\iff \text{for all } d \in \Delta^{\mathcal{I}}, \mathcal{I} \models P(x)[a_{x \leftarrow d}] \text{ and } \mathcal{I} \models \exists y P(y)[a_{x \leftarrow d}] \\
&\iff \text{for all } d \in \Delta^{\mathcal{I}}, \mathcal{I} \models P(x)[a_{x \leftarrow d}] \text{ and } \mathcal{I} \models \exists y P(y) \\
&\iff \mathcal{I} \models \forall x P(x) \text{ and } \mathcal{I} \models \exists y P(y) \\
&\iff \mathcal{I} \models \forall x P(x) \wedge \exists y P(y)
\end{aligned}
$$

□

Exercise 13:

Let $\mathcal{P} = \{p_1, \ldots, p_n\}$ be a set of propositional variables and $\prec$ be a total order[2] on the set $\mathbb{I}$ of interpretations of $\mathcal{P}$. Consider the problem of ginding a weight function $w : \mathcal{L} \to \mathbb{R}^+$, where $\mathcal{L}$ is the set of literals on $\mathcal{P}$, such that

$$\mathcal{I} \prec \mathcal{J} \quad \text{if and only if} \quad w(\mathcal{I}) \leq w(\mathcal{J})$$

(1) make a simple exempla with $|\mathcal{P}| = 2$,
(2) discusso on the fact if the problem has always a positive solution or not.
(3) Outline a method to find the solution.

Solution

(1) Let us consider the following two orders of the interpretations of $\{p, q\}$, where ij represents the interpretation $\mathcal{I}(p) = 1$ and $\mathcal{I}(q) = j$.

$$00 \prec 01 \prec 10 \prec 11 \tag{24}$$

$$00 \prec 11 \prec 10 \prec 01 \tag{25}$$

In the order (24) we can define $w(\neg p) = w(\neg q) = 1$ and $w(p) = 3$, and $w(q) = 2$, we obtain that

$$w(00) = 1 < w(01) = 2 < w(10) = 3 < w(11) = 6$$

In the order (25) instead we can write the following system of inequalities:

$$\begin{cases} w(00) < w(10) \\ w(11) < w(01) \\ w(00) < w(11) \end{cases}$$

which can be rewritten as

$$\begin{cases} w(\neg p) w(\neg q) < w(p) w(\neg q) \\ w(p) w(q) < w(\neg p) w(q) \end{cases} \Rightarrow \begin{cases} w(\neg p) < w(p) \\ w(p) < w(\neg p) \end{cases}$$

which does not have any solution.

(2) From the previous example one can see that the problem does not always have a solution.

[2]A total order $\prec$ on a set S is a binary relation on S such that (a) $s \not\prec s$, (b) if $s \neq t$ then $s \prec t$ or $t \prec s$, and (c) $s \prec t \prec u$ implies $s \prec u$. An example of total order is the usual order $<$ on integers.

(3) A method for solving this problem is to writhe explicitly the system of inequalities for every $\mathcal{I} \prec \mathcal{J}$

$$\prod_{i=1}^{n} w(p_i)^{\mathcal{I}(p_i)} \cdot w(\neg p_i)^{\mathcal{I}(\neg p_i)} < \prod_{i=1}^{n} w(p_i)^{\mathcal{J}(p_i)} \cdot w(\neg p_i)^{\mathcal{J}(\neg p_i)}$$

and try to solve it.

□

Exercise 14:

Compute the weighted model counting of the formula

$$(A \to B) \wedge (B \to (C \vee D))$$

via knowledge compilation with the following weight function:

lit	A	$\neg A$	B	$\neg B$	C	$\neg C$	D	$\neg D$
$w(lit)$	3	1	1	3	3	0.5	4	2

You can check your result by computing WMC using truth table (this is not strictly necessary for the exercize).

Solution We compute the WMC of the formula Φ by compiling it in the sd-DNNF form and then transform it in a computational circuit

$$\begin{aligned}
(A \to B) \wedge (B \to (C \vee D)) & \qquad \text{to NNF} \\
(\neg A \vee B) \wedge (\neg B \vee C \vee D) & \qquad \text{to DNNF with shannon expansion on } B \\
(B \wedge (C \vee D)) \vee (\neg B \wedge \neg A) & \qquad \text{to d-DNNF} \\
(B \wedge (C \vee (\neg C \wedge D))) \vee (\neg B \wedge \neg A) & \qquad \text{to sd-DNNF} \\
(B \wedge (C \wedge (D \vee \neg D) \vee (\neg C \wedge D)) \wedge (A \vee \neg A)) \vee & \\
(\neg B \wedge \neg A \wedge (C \vee \neg C) \wedge (D \vee \neg D)) & \qquad \text{To circuit} \\
(1 \cdot (3 \cdot (4+2) + (0.5 \cdot 4)) \cdot (3+1)) \; + & \\
(3 \cdot 1 \cdot (3+0.5) \cdot (4+2)) = & \\
80 + 63 = 143 &
\end{aligned}$$

□

Bibliography

Chakraborty, Supratik et al. (2015). "From weighted to unweighted model counting". In: *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

Chavira, Mark and Adnan Darwiche (2008). "On probabilistic inference by weighted model counting". In: *Artificial Intelligence* 172.6, pp. 772–799. ISSN: 0004-3702. DOI: https://doi.org/10.1016/j.artint.2007.11.002. URL: https://www.sciencedirect.com/science/article/pii/S0004370207001889.

Sang, Tian, Paul Beame, and Henry Kautz (2005). "Solving Bayesian networks by weighted model counting". In: *Proc.AAAI-05*. Vol. 1, pp. 475–482.