Formule logique : Sémantique

Quentin Fortier

April 12, 2022

Évaluation de formule

Définition

Une **valuation** sur un ensemble V de variables est une fonction de V vers $\{0, 1\}$.

0 est parfois noté Faux ou $\bot$. 1 est parfois noté Vrai ou $\top$.

Évaluation de formule

Définition

Une **valuation** sur un ensemble V de variables est une fonction de V vers $\{0, 1\}$.

0 est parfois noté Faux ou $\perp$. 1 est parfois noté Vrai ou $\top$.

Définition

Soit v une valuation sur V.

L'**évaluation** $[\![\varphi]\!]_v$ d'une formule φ sur v est définie inductivement :

- $[\![T]\!]_v = 1$, $[\![F]\!]_v = 0$
- $[\![x]\!]_v = v(x)$ si $x \in V$
- $[\![\neg\varphi]\!]_v = 1 - [\![\varphi]\!]_v$
- $[\![\varphi \wedge \psi]\!]_v = \min([\![\varphi]\!]_v, [\![\psi]\!]_v)$
- $[\![\varphi \vee \psi]\!]_v = \max([\![\varphi]\!]_v, [\![\psi]\!]_v)$

Si $[\![\varphi]\!]_v = 1$, on dit que v est un **modèle** pour φ.

Évaluation de formule

```
let rec eval d = function
    | T -> true
    | F -> false
    | Var(x) -> d x
    | Not(p) -> not (eval p)
    | And(p, q) -> (eval p) && (eval q)
    | Or(p, q) -> (eval p) || (eval q)
```

Ici une valuation v à valeur booléenne est utilisée.

Évaluation de formule

Définition

Deux formules φ et ψ sur V sont **équivalentes** (et on note $\varphi \equiv \psi$) si, pour toute valuation $v : V \to \{0,\ 1\}$:

$$[\![\varphi]\!]_v = [\![\psi]\!]_v$$

Évaluation de formule

Définition

Deux formules φ et ψ sur V sont **équivalentes** (et on note $\varphi \equiv \psi$) si, pour toute valuation $v : V \to \{0,\ 1\}$:

$$[\![\varphi]\!]_v = [\![\psi]\!]_v$$

Lois de de Morgan

Pour toutes formules φ, ψ :

$$\neg(\varphi \vee \psi) \equiv \neg\varphi \wedge \neg\psi$$
$$\neg(\varphi \wedge \psi) \equiv \neg\varphi \vee \neg\psi$$

Évaluation de formule

Définition

Deux formules φ et ψ sur V sont **équivalentes** (et on note $\varphi \equiv \psi$) si, pour toute valuation $v : V \to \{0,\ 1\}$:

$$[\![\varphi]\!]_v = [\![\psi]\!]_v$$

Lois de de Morgan

Pour toutes formules φ, ψ :

$$\neg(\varphi \vee \psi) \equiv \neg\varphi \wedge \neg\psi$$
$$\neg(\varphi \wedge \psi) \equiv \neg\varphi \vee \neg\psi$$

Définition

Une formule toujours évaluée à 1 est une **tautologie**.
Une formule toujours évaluée à 0 est une **antilogie**.
Une formule qui possède au moins une évaluation à 1 est **satisfiable**.

Évaluation de formule

Soit $G = (V, E)$ un graphe.

Exercice

Définir une formule logique satisfiable si et seulement si G est biparti (c'est-à-dire : $\exists A \subseteq V$ tel que les seules arêtes de G soient entre un sommet de A et un sommet de ${}^cA$).
Écrire une fonction OCaml pour effectuer cette transformation.

Évaluation de formule

Quelques équivalences importantes :

$$\neg\neg\varphi \equiv \varphi$$

$$\varphi \wedge \varphi \equiv \varphi$$

$$\varphi \vee \varphi \equiv \varphi$$

$$\varphi_1 \wedge (\varphi_2 \wedge \varphi_3) \equiv (\varphi_1 \wedge \varphi_2) \wedge \varphi_3$$

$$\varphi_1 \vee (\varphi_2 \vee \varphi_3) \equiv (\varphi_1 \vee \varphi_2) \vee \varphi_3$$

$$\boxed{\varphi_1 \vee (\varphi_2 \wedge \varphi_3) \equiv (\varphi_1 \vee \varphi_2) \wedge (\varphi_1 \vee \varphi_3)}$$

$$\boxed{\varphi_1 \wedge (\varphi_2 \vee \varphi_3) \equiv (\varphi_1 \wedge \varphi_2) \vee (\varphi_1 \wedge \varphi_3)}$$

$$\varphi_1 \longrightarrow \varphi_2 \equiv \neg\varphi_2 \longrightarrow \neg\varphi_1$$

Algèbre de Bool

En notant $\overline{a}$ au lieu de $\neg a$, $a + b$ au lieu de $a \vee b$, ab au lieu de $a \wedge b$, les équivalences précédentes deviennent :

$$\overline{\overline{a}} \equiv a$$

$$aa \equiv a$$

$$a + a \equiv a$$

$$a(bc) \equiv (ab)c$$

$$a + (b + c) \equiv (a + b) + c$$

$$a + bc \equiv (a + b)(a + c)$$

$$a(b + c) \equiv ab + ac$$

Et les lois de De Morgan :

$$\overline{a + b} \equiv \overline{a}\overline{b}$$

$$\overline{ab} \equiv \overline{a} + \overline{b}$$

Algèbre de Bool

Exercice

Comment peut-on réécrire $(\bigvee_i \varphi_i) \wedge (\bigvee_j \psi_j)$?

Et $(\bigwedge_i \varphi_i) \vee (\bigwedge_j \psi_j)$?

Algèbre de Bool

Théorème

Soit φ une formule possédant des $\neg$ uniquement sur des variables.
Alors $\neg\varphi$ équivaut à :

1. inverser les $\vee$ et $\wedge$
2. inverser les variables avec leurs négations

Algèbre de Bool

Théorème

Soit φ une formule possédant des $\neg$ uniquement sur des variables.
Alors $\neg\varphi$ équivaut à :

1. inverser les $\vee$ et $\wedge$
2. inverser les variables avec leurs négations

Preuve : Par induction structurelle.

Algèbre de Bool

Théorème

Soit φ une formule possédant des $\neg$ uniquement sur des variables.
Alors $\neg\varphi$ équivaut à :

1. inverser les $\vee$ et $\wedge$
2. inverser les variables avec leurs négations

Preuve : Par induction structurelle.
Par exemple si $\varphi = (x \vee y) \wedge ((\neg x \wedge z) \vee \neg y) \vee \neg z$ alors :

$$\neg\varphi \equiv (\neg x \wedge \neg y) \vee ((x \vee \neg z) \wedge y) \wedge z$$

Algèbre de Bool

Théorème

Soit φ une formule possédant des $\neg$ uniquement sur des variables.
Alors $\neg\varphi$ équivaut à :

1. inverser les $\vee$ et $\wedge$
2. inverser les variables avec leurs négations

Preuve : Par induction structurelle.
Par exemple si $\varphi = (x \vee y) \wedge ((\neg x \wedge z) \vee \neg y) \vee \neg z$ alors :

$$\neg\varphi \equiv (\neg x \wedge \neg y) \vee ((x \vee \neg z) \wedge y) \wedge z$$

On peut calculer sur des formules un peu comme sur les réels.
Par exemple, comme $(a+b)(c+d)e = ace + ade + bce + bde$:

$$(a \vee b) \wedge (c \vee d) \wedge e \equiv (a \wedge c \wedge e) \vee (a \wedge d \wedge e) \vee (b \wedge c \wedge e) \vee (b \wedge d \wedge e)$$

Algèbre de Bool

Soit $V = \{x_0, ..., x_{n-1}\}$. Pour savoir si une formule est une tautologie, une méthode naïve est d'énumérer les 2^n distributions de vérité $v : V \to \{0, 1\}$.

Algèbre de Bool

Soit $V = \{x_0, ..., x_{n-1}\}$. Pour savoir si une formule est une tautologie, une méthode naïve est d'énumérer les 2^n distributions de vérité $v : V \to \{0, 1\}$.

On peut représenter v par un entier dont le $i$ème bit est $v(x_i)$ (*bitset*). On énumère alors tous les entiers de 0 à $2^n - 1$.

Algèbre de Bool

Soit $V = \{x_0, ..., x_{n-1}\}$. Pour savoir si une formule est une tautologie, une méthode naïve est d'énumérer les 2^n distributions de vérité $v : V \to \{0, 1\}$.

On peut représenter v par un entier dont le $i$ème bit est $v(x_i)$ (*bitset*). On énumère alors tous les entiers de 0 à $2^n - 1$.

Exercice

En déduire des fonctions OCaml `tautologie` et `satisfiable`.
On pourra utiliser `Int.logand`, `Int.logor`, `Int.shift_left` pour les opérations bit à bit.

Complexité :

Algèbre de Bool

Soit $V = \{x_0, ..., x_{n-1}\}$. Pour savoir si une formule est une tautologie, une méthode naïve est d'énumérer les 2^n distributions de vérité $v : V \to \{0, 1\}$.

On peut représenter v par un entier dont le $i$ème bit est $v(x_i)$ (*bitset*). On énumère alors tous les entiers de 0 à $2^n - 1$.

Exercice

En déduire des fonctions OCaml `tautologie` et `satisfiable`.
On pourra utiliser `Int.logand`, `Int.logor`, `Int.shift_left` pour les opérations bit à bit.

Complexité : $\geq 2^n$.

Table de vérité

Soit φ une formule sur V. On peut représenter les différentes valeurs des évaluations de φ par une **table de vérité**.

Table de vérité

Soit φ une formule sur V. On peut représenter les différentes valeurs des évaluations de φ par une **table de vérité**.

Table de vérité de $(x \wedge y) \vee (\neg x \wedge \neg y)$:

x	y	$(x \wedge y) \vee (\neg x \wedge \neg y)$
0	0	1
0	1	0
1	0	0
1	1	1

Chaque ligne correspond à une valuation v possible et $[\![\varphi]\!]_v$.

Table de vérité

Soit φ une formule sur V. On peut représenter les différentes valeurs des évaluations de φ par une **table de vérité**.

Table de vérité de $(x \wedge y) \vee (\neg x \wedge \neg y)$:

x	y	$(x \wedge y) \vee (\neg x \wedge \neg y)$
0	0	1
0	1	0
1	0	0
1	1	1

Chaque ligne correspond à une valuation v possible et $[\![\varphi]\!]_v$.

Deux formules sont équivalentes ssi elles ont la même table de vérité.

Table de vérité

Vous êtes perdus dans le désert et vous avez le choix entre 2 chemins, gardés par 2 sphinx.
Le premier vous dit : « au moins un des chemins conduit à une oasis. »
Le second ajoute : « le chemin de droite se perd dans le désert. »
Sachant que les deux sphinx disent tous deux la vérité, ou bien mentent tous deux, que faites vous ?

Table de vérité

Vous êtes perdus dans le désert et vous avez le choix entre 2 chemins, gardés par 2 sphinx.
Le premier vous dit : « au moins un des chemins conduit à une oasis. »
Le second ajoute : « le chemin de droite se perd dans le désert. »
Sachant que les deux sphinx disent tous deux la vérité, ou bien mentent tous deux, que faites vous ?

Soient $x =$ « le chemin de gauche conduit à une oasis » et $y =$ « le chemin de droite conduit à une oasis ».

Table de vérité

Vous êtes perdus dans le désert et vous avez le choix entre 2 chemins, gardés par 2 sphinx.
Le premier vous dit : « au moins un des chemins conduit à une oasis. »
Le second ajoute : « le chemin de droite se perd dans le désert. »
Sachant que les deux sphinx disent tous deux la vérité, ou bien mentent tous deux, que faites vous ?

Soient $x =$ « le chemin de gauche conduit à une oasis » et $y =$ « le chemin de droite conduit à une oasis ».
D'après l'hypothèse, la formule $\varphi = ((x \vee y) \wedge \neg y) \vee (\neg(x \vee y) \wedge y)$ doit être vraie.

Table de vérité

Vous êtes perdus dans le désert et vous avez le choix entre 2 chemins, gardés par 2 sphinx.
Le premier vous dit : « au moins un des chemins conduit à une oasis. »
Le second ajoute : « le chemin de droite se perd dans le désert. »
Sachant que les deux sphinx disent tous deux la vérité, ou bien mentent tous deux, que faites vous ?

Soient $x =$ « le chemin de gauche conduit à une oasis » et $y =$ « le chemin de droite conduit à une oasis ».
D'après l'hypothèse, la formule $\varphi = ((x \vee y) \wedge \neg y) \vee (\neg(x \vee y) \wedge y)$ doit être vraie.
En écrivant la table de vérité de φ ou en utilisant notre fonction Caml, on trouve que la seule solution est $x = 1$ et $y = 0$: il faut donc prendre le chemin de gauche.

Table de vérité

Nombre de tables de vérités différentes sur n variables :

Table de vérité

Nombre de tables de vérités différentes sur n variables : 2^{2^n}
(2 choix pour chacune des 2^n distributions de vérité).

Table de vérité

Nombre de tables de vérités différentes sur n variables : 2^{2^n}
(2 choix pour chacune des 2^n distributions de vérité).

Question

Est-ce que toutes les tables de vérités possibles peuvent être obtenues par une formule logique?

Table de vérité

Nombre de tables de vérités différentes sur n variables : 2^{2^n}
(2 choix pour chacune des 2^n distributions de vérité).

Question

Est-ce que toutes les tables de vérités possibles peuvent être obtenues par une formule logique?

Exemple : comment obtenir la table suivante?

x	y	$?$
0	0	1
0	1	1
1	0	0
1	1	1

Table de vérité

Nombre de tables de vérités différentes sur n variables : 2^{2^n}
(2 choix pour chacune des 2^n distributions de vérité).

Question

Est-ce que toutes les tables de vérités possibles peuvent être obtenues par une formule logique?

Exemple : comment obtenir la table suivante?

x	y	?
0	0	1
0	1	1
1	0	0
1	1	1

Avec la formule $\neg x \vee y$, qu'on note aussi $x \longrightarrow y$.

Table de vérité

2ème exemple :

x	y	z	?
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

Table de vérité

Cette méthode marche tout le temps, et permet de prouver :

Théorème

Toute table de vérité peut être obtenue avec une formule logique.
Il existe donc exactement 2^{2^n} formules logiques à n variables, à équivalence près.

Table de vérité

Cette méthode marche tout le temps, et permet de prouver :

Théorème

Toute table de vérité peut être obtenue avec une formule logique. Il existe donc exactement 2^{2^n} formules logiques à n variables, à équivalence près.

De plus, la forme de la formule obtenue est bien particulière.

Définition

- Un **littéral** est une variable ou sa négation.
- Une **clause** est une conjonction de littéraux (c'est à dire de la forme $\ell_1 \wedge \ell_1 \wedge ... \wedge \ell_p$ où ℓ_i est un littéral).

Table de vérité

Cette méthode marche tout le temps, et permet de prouver :

Théorème

Toute table de vérité peut être obtenue avec une formule logique.
Il existe donc exactement 2^{2^n} formules logiques à n variables, à équivalence près.

De plus, la forme de la formule obtenue est bien particulière.

Définition

- Un **littéral** est une variable ou sa négation.
- Une **clause** est une conjonction de littéraux (c'est à dire de la forme $\ell_1 \wedge \ell_1 \wedge \ldots \wedge \ell_p$ où ℓ_i est un littéral).

Théorème

Toute formule logique est équivalente à une formule sous **forme normale disjonctive**, c'est à dire de la forme $c_1 \vee \ldots \vee c_k$ où c_i est une clause.

Conséquence

Définition

On dit que ψ est une **conséquence** de φ, et on note $\varphi \models \psi$, si tout modèle de φ est un modèle de ψ (c'est-à-dire : pour toute valuation v, $[\![\varphi]\!]_v \leq [\![\psi]\!]_v$).

Conséquence

Définition

On dit que ψ est une **conséquence** de φ, et on note $\varphi \models \psi$, si tout modèle de φ est un modèle de ψ (c'est-à-dire : pour toute valuation v, $[\![\varphi]\!]_v \leq [\![\psi]\!]_v$).

Exemples :

- $x \models x \vee (y \wedge z)$
- $y \models x \longrightarrow y$

Conséquence

Lemme

Soient φ et ψ deux formules. Alors :

$$\varphi \models \psi \text{ si et seulement si } \models \varphi \longrightarrow \psi$$

Ensemble de formules

On peut généraliser les définitions à un ensemble Γ de formules :

Définition

- Un modèle pour Γ est une valuation v telle que :

$$\forall \varphi \in \Gamma,\ [\![\varphi]\!]_v = 1$$

- Une formule φ est une conséquence logique de Γ si tout modèle de Γ est un modèle de φ.
  On note alors $\Gamma \models \varphi$.

Ensemble de formules

On peut généraliser les définitions à un ensemble Γ de formules :

Définition

- Un modèle pour Γ est une valuation v telle que :

$$\forall \varphi \in \Gamma,\ [\![\varphi]\!]_v = 1$$

- Une formule φ est une conséquence logique de Γ si tout modèle de Γ est un modèle de φ.
  On note alors $\Gamma \models \varphi$.

<u>Exemple</u> :

$$\{x, x \longrightarrow y\} \models y$$