Logique classique
Cours 2 : Logique propositionnelle

Odile PAPINI

POLYTECH
Université d'Aix-Marseille
odile.papini@univ-amu.fr
http://odile.papini.perso.luminy.univ-amu.fr/sources/LOG.html

Plan du cours

Bibliographie I

- DELAHAYE J. P., *Outils logiques pour l'intelligence artificielle.* ,
  Eyrolles, Paris, 1986.
- GOCHET P.& GRIBOMONT P., *Logique : méthodes pour l'informatique fondamentale.*
  Langue, Raisonnement, Calcul, Hermes, Paris, 1990.
- KLEENE S. C., *Logique mathématique.* ,
  Epistémologie, Jacques Gabay, Paris, 1987.
- THAYSE A.& AL., *Approche logique de l'intelligence artificielle, Tome 1.* ,
  Informatique, DUNOD, Paris, 1990.

Bibliographie II

Alliot J.-M., Scheix T., Brisset P. & Garcia F., *Intelligence artificielle et Informatique théorique.*, CEPADUES EDITIONS, Toulouse, 2002.

Bibliographie 2 I

Support de cours logique propositionnelle

http ://www.irit.fr/ Andreas.Herzig/C/prop.html
http ://www.grappa.univ-lille3.fr/ champavere/Enseignement/0607/l2miashs /ia/logique.pdf
http ://www-lipn.univ-paris13.fr/ levy/pdf/CoursLogMod.pdf

Exercices
http ://home.etu.unige.ch/ guigong3/TPdeLogique.html
http ://users.info.unicaen.fr/ zanutti/logique/
http ://liris.cnrs.fr/ amille/enseignements/emiage/ supports−IA/logique/logique_propositions.pdf

Intoduction

proposition

concept de proposition :

information atomique **contingente**

ce qui est ou ce qui n'est pas, un **fait**, une **assertion**

exemple de propositions

$$2 + 2 = 4$$

$$1 + 1 = 0$$

Le soleil brille

Il a les yeux rouges

un carré est un polygone

tout homme est mortel $\cdots$

Le langage propositionnel $\mathcal{L}$

Vocabulaire

- un ensemble infini dénombrable de variables propositionnelles ou **propositions** $\mathcal{P}$
- les constantes : 0 (*Faux*, F ou $\perp$) et 1 (*Vrai*, V, ou $\top$)
- les connecteurs : $\neg \quad \wedge \quad \vee \quad \rightarrow \quad \leftrightarrow$
- les parenthèses

Le langage propositionnel

Procédé de formation des formules de $\mathcal{L}$

- $\perp$ (ou 0 ou F) : est une formule
- $\top$ (ou 1 ou V) : est une formule
- p : une variable propositionnelle est une formule
- si P et Q sont des formules alors
  $\neg P, \quad P \wedge Q, \quad P \vee Q, \quad P \rightarrow Q, \quad P \leftrightarrow Q$
  sont des formules

Le langage propositionnel

Exemples de formules propositionnelles

- $(\neg P \wedge Q) \vee (P \wedge \neg Q)$
- $P \vee (Q \wedge R)$
- $(P \rightarrow (Q \rightarrow P))$
- $(A \rightarrow B) \rightarrow ((A \rightarrow (B \rightarrow C)) \rightarrow (A \rightarrow C))$
- $\neg\neg A \rightarrow A$

Le langage propositionnel

exercice : représentation d'énoncé

- Si Pierre vient, on joue aux cartes ;
- Si Pierre et Jean viennent, il y a des disputes ;
- Si on ne joue pas aux cartes, il n'y a pas de dispute ;
- Pierre ne vient pas.

Le langage propositionnel

exercice : représentation d'énoncé

- Si Didier est l'auteur de ce bruit, il est stupide ou dépourvu de principes.
- Didier n'est ni stupide ni dépourvu de principes.
- Didier n'est pas l'auteur de ce bruit.

système formel

Axiomes

$\forall P,\ Q,\ R \in \mathcal{L}$

A1) $(P \rightarrow (Q \rightarrow P))$

A2) $((P \rightarrow (Q \rightarrow R)) \rightarrow ((P \rightarrow Q) \rightarrow (P \rightarrow R)))$

A3) $((\neg P \rightarrow \neg Q) \rightarrow (Q \rightarrow P))$

système formel

Règles de déduction

$\forall P,\ Q,\ R \in \mathcal{L}$

- **modus ponens**

$$\frac{\vdash P,\ \vdash P \to Q}{\vdash Q}$$

- **règle de substitution**

  remplacer dans un théorème une variable propositionnelle, partout où elle figure, par :

  - une autre variable propositionnelle
  - ou une formule bien formée

système formel

Définitions

$\forall P,\ Q \in \mathcal{L}$

D1) $P \to Q =_{def} \neg P \vee Q$

D2) $P \wedge Q =_{def} \neg(\neg P \vee \neg Q)$

D3) $P \leftrightarrow Q =_{def} (\neg P \vee Q) \wedge (P \vee \neg Q)$

système formel

Déduction

une **déduction** à partir d'hypothèses $H_1, H_2, \cdots, H_m$ est une suite de formules bien formées $F_1, F_2, \cdots, F_p$ où chaque F_i est soit :

- une hypothèse
- un axiome
- ou une formule obtenue à partir des règles d'inférence (substitution ou modus ponens) appliquées aux formules placées avant F_i dans la déduction

notation

$$H_1, H_2, \cdots, H_m \vdash F_p$$

théorème : déduction sans hypothèse $\vdash F$

système formel

exemple : déduction

Donner une déduction de C à partir de A, B et $A \to (B \to C)$, plus formellement, montrer que :

$$A,\ B,\ A \to (B \to C) \quad \vdash \quad C$$

système formel

Proposition :

$$\forall P \in \mathcal{L} \quad \vdash (P \rightarrow P)$$

Proposition :

$\forall P_1, \cdots, P_{n-1} \in \mathcal{L}$

si $\quad P_1, \cdots, P_{n-1} \vdash (P_n \rightarrow Q) \quad$ alors $\quad P_1, \cdots, P_n \vdash Q$

Théorème de déduction :

Soient $P_1, \cdots, P_n, Q \in \mathcal{L}$

si $\quad P_1, \cdots, P_n \vdash Q \quad$ alors $\quad P_1, \cdots, P_{n-1} \vdash (P_n \rightarrow Q)$

système formel

Quelques théorèmes utiles :

$\forall P, Q, R \in \mathcal{L}$, toutes les formules suivantes sont des théorèmes :

- $\vdash ((P \to Q) \to ((Q \to R) \to (P \to R)))$
- $\vdash (P \to ((P \to Q) \to Q))$
- $\vdash (\neg P \to (P \to Q))$
- $\vdash (\neg\neg P \to P)$
- $\vdash (P \to \neg\neg P)$
- $\vdash ((P \to Q) \to (\neg Q \to \neg P))$
- $\vdash (P \to (\neg Q \to \neg(P \to Q)))$
- $\vdash ((Q \to P) \to ((\neg Q \to P) \to P))$

système formel

exercices : déduction

- Montrer que $\forall P \in \mathcal{L}, \vdash (P \rightarrow P)$
- En utilisant le théorème de déduction montrer que :
$\vdash (A \rightarrow (B \rightarrow C)) \rightarrow (B \rightarrow (A \rightarrow C))$

sémantique de la logique propositionnelle

Interprétation

toute application σ de $\mathcal{P}$ (ensemble des propositions) dans $\{0, 1\}$ telle que :

- $\sigma(\bot) = 0$ et $\sigma(\top) = 1$

$\forall P, Q \in \mathcal{L}$,

- $\sigma(\neg P) = 1 - \sigma(P)$
- $\sigma(P \vee Q) = max(\sigma(P), \sigma(Q))$
- $\sigma(P \wedge Q) = min(\sigma(P), \sigma(Q))$
- $\sigma(P \rightarrow Q) = max((1 - \sigma(P)), \sigma(Q))$
- $\sigma(P \leftrightarrow Q) = min(max((1 - \sigma(P)), \sigma(Q)), max(\sigma(P), (1 - \sigma(Q))))$.

sémantique de la logique propositionnelle

Table de vérité

P	Q	$P \vee Q$	$P \wedge Q$	$P \rightarrow Q$	$P \leftrightarrow Q$
1	1	1	1	1	1
1	0	1	0	0	0
0	1	1	0	1	0
0	0	0	0	1	1

sémantique de la logique propositionnelle

exercice

- Quelles sont les interprétations qui rendent vraie la formule :

$$(A \rightarrow (B \rightarrow C)) \rightarrow (B \rightarrow (A \rightarrow C)) \ ?$$

sémantique de la logique propositionnelle

quelques définitions

$\forall P, Q \in \mathcal{L}$ et $\mathcal{F} \subset \mathcal{L}$

P est une **tautologie** si pour toute interprétation σ, $\sigma(P) = 1$, on écrit $\models P$

Q est une **conséquence logique** de P si pour toute interprétation σ, si $\sigma(P) = 1$ alors $\sigma(Q) = 1$, on écrit $P \models Q$

Q est une **conséquence logique** de $\mathcal{F}$ si pour toute interprétation σ, tq $\forall P \in \mathcal{F}$, si $\sigma(P) = 1$ alors $\sigma(Q) = 1$, on écrit $\mathcal{F} \models Q$

sémantique de la logique propositionnelle

exercices

- Monter que les axiomes A1), A2), A3) sont des tautologies
- Est-ce que $(P \rightarrow R)$ est une tautologie ?
- Est -ce que ψ est une conséquence logique de ϕ ?
  - $\phi = (P \vee Q) \wedge (\neg P \vee R)$, $\psi = Q \wedge R$
  - $\phi = (P \rightarrow Q) \wedge (P \rightarrow \neg Q)$, $\psi = \neg P$

sémantique de la logique propositionnelle

quelques définitions

$\forall P, Q \in \mathcal{L}$ et $\mathcal{F} \subset \mathcal{L}$

- P est **satisfaisable** ou **cohérente** s'il existe une interprétation σ tq $\sigma(P) = 1$
- $\mathcal{F}$ est **satisfaisable** s'il existe une interprétation σ tq $\forall P \in \mathcal{F}$, $\sigma(P) = 1$
- P est **insatisfaisable** ou **incohérente** si pour toute interprétation σ, $\sigma(P) = 0$
- $\mathcal{F}$ est **insatisfaisable** si pour toute interprétation σ, $\exists P \in \mathcal{F}$ tq $\sigma(P) = 0$

sémantique de la logique propositionnelle

exercices

- les formules suivantes sont-elles cohérentes ?
  - $(a \vee \neg b) \wedge (\neg a \vee b) \wedge \neg(a \leftrightarrow b)$
  - $b \rightarrow (\neg c \rightarrow \neg(b \rightarrow c))$
- les ensembles de formules suivants sont-ils satisfaisables ?
  - $\mathcal{F} = \{a \vee b \vee c, \neg a \vee b, \neg a \vee c, \neg b, \neg c\}$
  - $\mathcal{G} = \{a \vee b, \neg a, \neg b\}$

sémantique de la logique propositionnelle

Quelques propriétés

$\forall P, Q \in \mathcal{L}$,

- $\models (P \rightarrow Q)$ ssi $P \models Q$
- $\models (P \leftrightarrow Q)$ ssi $P \equiv Q$
- si $\models P$ et $\models (P \rightarrow Q)$ alors $\models Q$
- $\models (P \wedge Q)$ ssi $\models P$ et $\models Q$
- si $\models P$ ou $\models Q$ alors $\models (P \vee Q)$

sémantique de la logique propositionnelle

exercice

- Montrer :
  - $\models (P \rightarrow Q)$ ssi $P \models Q$
  - $\models (P \leftrightarrow Q)$ ssi $P \equiv Q$

la logique propositionnelle

Quelques théorèmes

théorème (d'adéquation) :
$\forall P \in \mathcal{L} \quad$ si $\vdash P$ alors $\models P$
(les formules qui sont des théorèmes sont des tautologies)

théorème (de complétude faible) :
$\forall P \in \mathcal{L} \quad$ si $\models P$ alors $\vdash P$
(les formules qui sont des tautologies sont des théorèmes)

théorème (de complétude forte) :
Soit $\mathcal{F}$ un ensemble de formules de $\mathcal{L}$, $\quad \forall P \in \mathcal{L}$
si $\mathcal{F} \models P$ alors $\mathcal{F} \vdash P$

logique propositionnelle

Quelques théorèmes

théorème (de compacité) :
Soit $\mathcal{F}$ un ensemble de formules de $\mathcal{L}$.
Si toute famille finie $\mathcal{F}' \subset \mathcal{F}$ est satisfaisable alors $\mathcal{F}$ est aussi satisfaisable.

théorème (de finitude) :
Soit $\mathcal{F}$ un ensemble de formules de $\mathcal{L}$. Soit $Q \in \mathcal{L}$
si $\mathcal{F} \models Q$ alors $\exists \mathcal{F}' \subset \mathcal{F}$ fini tq $\mathcal{F}' \models Q$

théorème (de décidabilité) :
$\forall P \in \mathcal{L}$, il existe un programme qui, pour toute formule P, indique en un temps fini si oui ou non $\vdash P$

logique pour l'informatique

formes normales

littéral : une proposition ou la négation d'une proposition
clause : disjonction de littéraux
cube : conjonction de littéraux
forme conjonctive normale : une conjonction de clauses
forme disjonctive normale : une disjonction de cubes

théorème (de normalisation) :

- $\forall P \in \mathcal{L}$, P admet une forme conjonctive normale qui lui est équivalente
- $\forall P \in \mathcal{L}$, P admet une forme disjonctive normale qui lui est équivalente

logique pour l'informatique : algorithme de normalisation

début

- **élimination des connecteurs $\rightarrow$ et $\leftrightarrow$**

  `remplacer` $P \leftrightarrow Q$ `par` $(P \rightarrow Q) \wedge (Q \rightarrow P)$

  `puis remplacer` $P \rightarrow Q$ `par` $\neg P \vee Q$

- **application des lois de Morgan**

  `remplacer` $\neg(P \wedge Q)$ `par` $\neg P \vee \neg Q$ `et` $\neg(P \vee Q)$ `par` $\neg P \wedge \neg Q$

- **élimination des doubles négations**

  `remplacer` $\neg\neg P$ `par` P

- **application des règles de distributivité**

  `remplacer` $P \vee (Q \wedge R)$ `par` $(P \vee Q) \wedge (P \vee R)$

  `et` $(P \wedge Q) \vee R$ `par` $(P \vee R) \wedge (Q \vee R)$

fin

formes normales

exercice

- Mettre sous forme CNF :
  - $\neg(a \vee b \rightarrow c)$
  - $P \rightarrow (Q \rightarrow R)$
  - $\neg((P \rightarrow Q) \wedge (R \rightarrow S))$

formes normales

exercice

- Mettre sous forme DNF :
  - $\neg(a \vee b \rightarrow c)$
  - $P \rightarrow (Q \rightarrow R)$
  - $\neg((P \rightarrow Q) \wedge (R \rightarrow S))$