Introduction à la logique des prédicats I

Robert Michels

mail@robert-michels.de

Université de Neuchâtel – semestre de printemps 2020 – 17 février 2020

Préliminaire

Les deux livres sur lesquels le cours se base :

- Lepage, F. : *Eléments de logique contemporaine*. 3e édition. PU Montréal 2010
- Halbach, V. : *The Logic Manual*. Oxford University Press 2010

Rappel – la logique des propositions

- La logique nous donne des normes du raisonnement
- Un emploi important de la logique en philosophie est l'analyse des arguments déductifs
- Analyser un argument en trois étapes :

1. Identifier les prémisses et la conclusion d'un argument dans un texte en langage naturel
2. Traduire les prémisses et la conclusion en langage formel
3. Évaluer l'argument : appliquer des méthodes formelles afin de vérifier si l'argument est valide

Rappel – la logique des propositions

- Étape 2 – nous nous sommes familiarisés avec le langage (formel) de la logique des propositions L_0, sa syntaxe et sémantique
- Étape 3 – nous nous sommes familiarisés avec deux relations différentes entre les formules de L_0 et avec trois méthodes formelles :

1. "$\vDash_{L_0}$" : la relation sémantique d'être une *conséquence logique* de – d'être tel qu'il est nécessaire que la conclusion soit vraie, si les prémisses sont vraies (ou l'équivalent : d'être tel qu'il est impossible que les prémisses soient vraies, mais la conclusion fausse) – et la méthode sémantique des tables de vérité :
   - pour vérifier si $A_1, \ldots, A_i \vDash_{\mathrm{L}_0} B$ (avec $i \in \mathbb{N}$), on construit une table de vérité de la formule de L_0 "$(A_1 \wedge \ldots \wedge A_i) \rightarrow B$" – si cette formule est une tautologie, l'affirmation est vraie, si non, elle est fausse

Rappel – la logique des propositions

2 . “$\vdash_{\mathsf{L}_0}$” : la relation syntaxique d’*être dérivable de* – d’être tel qu’on peut prouver la conclusion à partir des prémisses – et deux méthodes syntaxiques :

- la méthode des arbres : pour vérifier si $A_1, \ldots, A_i \vdash_{\mathsf{L}_0} B$, on construit un arbre avec $A_1, \ldots, A_i$ et $\neg B$ dans sa racine – si tous ses branches se terminent, B est dérivable à partir de $A_1, \ldots, A_i$, si non, ce n’est pas le cas
- la déduction naturelle : pour vérifier que $A_1, \ldots, A_i \vdash_{\mathsf{L}_0} B$, on construit une preuve en déduction naturelle de B à partir des hypothèses $A_1, \ldots, A_i$

Les limites expressives de la logique des propositions

- Le langage formel de la logique des propositions L_0 nous permet de représenter la *forme logique* des propositions ou des phrases d'un langage naturel comme le français
- Nous pouvons par exemple traduire :
    - "Anne aime les carottes ou Anne n'aime pas les carottes" par la formule de L_0 "$p \vee \neg p$"
    - "Si Bruno manque son train, il prend un taxi ou le bus." par "$p \rightarrow (q \vee r)$"

Les limites expressives de la logique des propositions

- Le concept de forme logique des propositions ou des phrases en L_0 se base sur deux suppositions fondamentales :
  1. Les propositions atomiques n'ont pas de forme logique – elles correspondent à des formules atomiques de L_0 telles que $p, q, r, \ldots$
  2. La structure d'une proposition complexe est complètement déterminée par la configuration des connecteurs logiques et les formules atomiques contenus dans la formule de L_0 qui correspond à la proposition
- Ce concept est directement lié aux concepts de vérité logique et de conséquence logique en L_0 :
- La forme logique d'une phrase détermine si elle exprime une vérité logique en L_0 et la forme logique des prémisses et de la conclusion d'un argument déterminent si l'argument est valide, soit si la formule qui correspond à sa conclusion est conséquence logique en L_0 des formules qui correspondent à ses prémisses

Les limites expressives de la logique des propositions

- Bien que le concept de forme logique s'applique strictement aux phrases (ou propositions), on peut parler aussi de la forme logique d'un argument ; la forme logique d'un argument consiste simplement en la forme logique de ses prémisses et de sa conclusion
- Une conséquence de cette conception de la forme, la vérité et la conséquence logique est que la structure interne des phrases qui correspondent à des formules atomiques de L_0 n'a pas d'incidence sur la validité des arguments

Les limites expressives de la logique des propositions

- Selon la logique des propositions, les trois propositions suivantes ont la même forme logique, à savoir la forme d'une formule atomique :
  1. Claude est une personne.
  2. Il y a quelque chose qui est à l'est de Neuchâtel.
  3. Tous les objets qui sont bordeaux sont rouges.
- Pourtant, il existe des arguments qui semblent être valides, mais qui ne le sont pas dans le sens défini par les concepts de conséquence logique et de dérivabilité en L_0
- Leur validité ne peut être expliquée sans prendre en compte la structure interne des propositions

Les limites expressives de la logique des propositions

Exemples

a1 Toute personne a des souvenirs des épisodes passés de sa vie.

a2 Claude est une personne.

a3 $\therefore$ Claude a des souvenirs des épisodes passés de sa vie.

b1 Berne est à l'est de Neuchâtel.

b2 $\therefore$ Alors, il y a quelque chose qui est à l'est de Neuchâtel.

c1 Tous les objets qui sont bordeaux sont rouges.

c2 Tous les objets qui sont rouges sont colorés.

c3 $\therefore$ Tous les objets qui sont bordeaux sont colorés.

Les limites expressives de la logique des propositions

Invalidité des exemples en L_0

- Chacun de ces trois arguments remplit la condition nécessaire et suffisante pour être valide (au sens sémantique) – il est impossible pour sa conclusion d'être fausse si ses prémisses sont vraies
- Néanmoins, $a1, a2 \not\models_{\mathrm{L}_0} a3$, $b1 \not\models_{\mathrm{L}_0} b2$ et $c1, c2 \not\models_{\mathrm{L}_0} c3$
- Pourquoi ?
- Essayez de traduire les prémisses et les conclusions de ces arguments en L_0
- Comme ces propositions ne contiennent pas des connecteurs logiques de L_0, les formalisations des trois arguments qui s'imposent sont les suivantes :

Les limites expressives de la logique des propositions

Invalidité des exemples en L_0

a1* p

a2* q

a3* $\therefore r$

b1* s

b2* $\therefore t$

c1* u

c2* v

c3* $\therefore w$

Les limites expressives de la logique des propositions

Invalidité des exemples en L_0

- Évidemment, ce ne sont pas des formes d'argument valides
- Quel est le problème ?
- La validité de chacun de ces arguments dépend d'une structure logique interne des propositions qui ne peut pas être représentée par le langage de la logique des propositions
- Pour résoudre ce problème, il nous faut un langage formel qui puisse représenter cette structure interne et une logique pour ce langage qui nous permette d'expliquer la validité évidente de ces trois arguments et d'une grande classe d'arguments similaires
- Cette logique est la logique des prédicats

Le langage de la logique des prédicats L_1 – la syntaxe

Introduction informelle au langage L_1

Les termes de L_1

- Le langage L_1 est une extension du langage L_0 : elle contient les mêmes connecteurs logiques, mais elle contient aussi quatre nouveaux types de symboles logiques qui nous permettent de représenter la structure logique interne des propositions
    1. constantes
    2. prédicats
    3. variables
    4. quantificateurs

Le langage de la logique des prédicats L_1 – la syntaxe

Introduction informelle au langage L_1

Les constantes

- Les constantes sont désignées par des lettres minuscules $a, b, c, \ldots$
- Une constante (ou constante d'individu) est un symbole qui correspond au nom d'un objet en langage naturel
- P. ex. la prémisse a2 contient le nom "Claude" et la prémisse b1 contient les noms "Berne" et "Neuchâtel"
- Dans une traduction en langage L_1, ces noms sont représentés par des constantes (p. ex. c, b et n) qui dénotent les objets Claude, Berne et Neuchâtel

Le langage de la logique des prédicats L_1 – la syntaxe

Introduction informelle au langage L_1

Les prédicats I

- Les prédicats sont désignés par des lettres majuscules $P, Q, R, \ldots$
- Un prédicat est un symbole qui correspond à une expression ou une partie de phrase en langage naturel qui correspond à son tour à une propriété ou une relation – p. ex. "…est verte" ou "…habite à …"
- Les prédicats sont des expressions essentiellement incomplètes : ils ont un nombre d'arguments qui doivent être comblés par des termes (constantes ou variables) pour former une formule atomique
- P. ex. a2 contient l'expression "…est une personne" et b1 contient l'expression "…est à l'est de …" qui correspondent à des prédicats

Le langage de la logique des prédicats L_1 – la syntaxe

Introduction informelle au langage L_1

Les prédicats II

- Chaque prédicat a une *arité*, un nombre d'arguments qui peuvent être comblés par des constantes ou des variables
- Un prédicat qui correspond à "... est une personne" a une arité de 1, un prédicat qui correspond à "... est à l'est de ... " a l'arité 2, un prédicat qui correspond à "... est entre ... et ... " a l'arité 3, etc.
- L'arité d'un prédicat est indiquée par un index en hauteur : "P^n" : "P^n" est un prédicat n-aire/est d'une arité de n
- On appelle les prédicats d'une arité supérieure à 1 *prédicats relationnels* ou simplement *relations* (on utilise souvent la lettre "R" pour ce type de prédicat)

Le langage de la logique des prédicats L_1 – la syntaxe

Introduction informelle au langage L_1

Les prédicats III

- Les arguments d'un prédicat ont un ordre fixe :
- Dans une formule, l'ordre des arguments d'un prédicat est indiqué par l'ordre des termes auxquels le prédicat est appliqué : P. ex. "*Rab*" – "*R*" est une relation d'arité 2 et, dans cette formule, "*a*" est le premier et "*b*" le deuxième argument
- Un exemple qui montre que l'ordre des arguments d'une relation est important : p. ex. si "*A*" signifie la relation "... aime ...", "*Axy*" n'exprime pas le même état de choses que "*Ayx*" – si Claude aime Alex, ça n'implique pas qu'Alex aime aussi Claude (malheureusement ou heureusement...)

Le langage de la logique des prédicats L_1 – la syntaxe

Introduction informelle au langage L_1

Les prédicats IV

- La notation officielle en langage L_1 est une notation préfixée : dans une formule bien formée de L_1, le prédicat est toujours préfixé à ces arguments (il précède les arguments)
- Remarque : si on parle seulement de relations binaires (d'arité 2), on utilise parfois une notation infixée et on écrit "aRb" au lieu de "Rab"

Le langage de la logique des prédicats L_1 – la syntaxe

Introduction informelle au langage L_1

Les variables

- Les variables sont désignées par des lettres minuscules $\ldots, x, y, z$ de la fin de l'alphabet
- Comme les constantes, elles se combinent avec un prédicat pour former une formule atomique de L_1
- La différence entre les constantes et les variables est que les variables n'ont pas une valeur fixe – une variable dans une formule atomique indique que cette formule se réfère à un objet arbitraire ; on peut dire qu'elle se réfère à un objet, mais pas à un objet particulier tel que p. ex. Neuchâtel, moi, ou la Tour Eiffel
- Une formule atomique qui contient seulement des variables, p. ex. "Px", ne correspond pas à une proposition (ou à une phrase du langage naturel), comme elle ne détermine pas à quel objet particulier la propriété qui correspond au prédicat "P" est attribuée

Le langage de la logique des prédicats L_1 – la syntaxe

Introduction informelle au langage L_1

Les quantificateurs I

- L'alphabet du langage contient deux quantificateurs signifiés par "$\forall$" et "$\exists$"
- Dans une formule, un quantificateur est toujours suivi d'une variable ; chaque occurrence de cette même variable dans la portée du quantificateur est *liée à ce quantificateur* (mais notez la discussion à suivre des occurrences libres et liées des variables pour des exceptions !)
- "$\forall x$" signifie "tous/toutes les x" (ou aussi "chaque x") – "$\forall$" est appelé le quantificateur universel
- "$\exists x$" signifie "il y a (au moins) un x" (ou aussi "il existe (au moins) un x") – "$\exists$" est appelé le quantificateur existentiel ou parfois aussi le quantificateur particulier

Le langage de la logique des prédicats L_1 – la syntaxe

Introduction informelle au langage L_1

Les quantificateurs II

- Syntaxiquement, les quantificateurs fonctionnent comme la négation : ils se combinent avec une formule pour former une autre formule comme p. ex. “$\forall x Px$”, “$\exists x(Px \vee \neg Px)$” ou “$\exists x \forall y(\neg Rxy)$”
- Comme les connecteurs logiques, les quantificateurs ont une portée qui est marquée par des parenthèses ; exemples :
  - $\forall x(Px \rightarrow \neg Qx)$
  - $Pa \wedge \exists x Qx$
  - $\forall x Px \vee Qy$

Le langage de la logique des prédicats L_1 – la syntaxe

Introduction informelle au langage L_1

Les quantificateurs III

- Quelques formules et leur significations expliqués :
  - "$\exists x(Px)$" dit qu'il ya un x qui est P (l'occurrence de la variable x dans "Px" est liée à $\exists$)
  - "$\forall x(Rxx)$" dit que tous les x sont en relation R avec eux-mêmes (les deux occurrences de la variable x sont liées à $\forall$)
  - "$\forall x(Px \wedge \exists x(Qx))$" dit que tous les x sont P et il y a un x qui est Q (la première occurrence de la variable x (dans la sous-formule "Px") est lié à "$\forall x$", la deuxième occurrence de la même variable (dans la sous-formule "Qx") est lié à "$\exists x$", mais pas à "$\forall x$")
- Commentaire sur le denier exemple : comme la portée des quantificateurs détermine toujours quelle variable est liée à quel quantificateur, on peut dans la même formule utiliser une seule variable qui est liée à différents quantificateurs dans différentes parties de la formule
- Néanmoins, il est préférable d'utiliser une variable différente pour chaque quantificateur pour des raisons de clarté

Le langage de la logique des prédicats L_1 – la syntaxe

Introduction informelle au langage L_1

Les quantificateurs IV

- Notez que les expressions du langage naturel "tous/toutes","chaque" et "il y a", "il existe" sont moins précises que les occurrences des quantificateurs qu'on utilise en L_1 pour les traduire, car elles ne spécifient pas de variables
- Cette différence nous permet d'éliminer des ambiguïtés du langage naturel
- P. ex. alors que la phrase française "Tous les enfants reçoivent un cadeau." est ambigüe, L_1 nous permet de (en fait, nous force à !) distinguer clairement entre ses deux sens possibles :
  1. Premier sens : Chaque enfant reçoit un cadeau – $\forall x(Ex \to \exists y(Cy \wedge Rxy)$ ("Pour tous les x, si x est un enfant (E), il y a un objet y qui est un cadeau (C) et qui est reçu (R) par x.")
  2. Deuxième sens : Tous les enfants reçoivent le même cadeau – $\exists x(Cx \wedge \forall y(Ey \to Ryx))$ ("Il y a un x qui est un cadeau et pour tous les y, si y est un enfant, y reçoit x.")

La syntaxe de L_1

L'alphabet de L_1

Définition : l'alphabet de L_1 L'alphabet de L_1 consiste en :

- un ensemble dénombrable des prédicats des différentes arités : $\{P_0^1, \ldots, P_n^1, \ldots, P_n^k\}$, $n, k \in \mathbb{N}$
- un ensemble dénombrable des variables : $\{x_0, \ldots, x_n\}$
- un ensemble dénombrable des constantes : $\{c_0, \ldots, c_n\}$
- les connecteurs logiques : $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$
- les quantificateurs : $\forall$, $\exists$
- les parenthèses : (,)

La syntaxe de L_1

L'alphabet de L_1

Remarques sur l'alphabet

- Les connecteurs logiques sont exactement les mêmes que dans L_0
- Conventions : pour simplifier notre notation, nous utilisons souvent
  - les lettres $P, Q, R, \ldots$ au lieu des prédicats $P^1_0, P^1_1, P^1_2, \ldots, P^2_0, P^2_1, P^2_2, \ldots$, etc. si l'arité de ces prédicats est déterminée par le contexte (p. ex. si "P" traduit "... est une personne", il est clair que "P" est un prédicat d'arité 1, si "Q" traduit "... est à l'est de ...", il est clair que "Q" est d'arité 2, etc.)
  - les lettres $\ldots, x, y, z$ au lieu des variables $x_0, x_1, x_2, \ldots$
  - les lettres $a, b, c, \ldots$ au lieu des constantes $c_0, c_1, c_2, \ldots$
- (Évidemment, ces conventions ont été précédemment utilisées dans la discussion informelle du langage L_1)

La syntaxe de L_1

Les formules de L_1

Définition : les formules de L_1

1. Pour chaque prédicat P^i_j d'arité i, si chaque $t_k \in \{t_0, \ldots, t_i\}$ est une constante ou une variable, $P^i_j t_0 \ldots t_i$ est une formule de L_1.
2. Si A est une formule, $\neg(A)$ est une formule.
3. Si A et B sont des formules, $(A \wedge B)$ est une formule.
4. Si A et B sont des formules, $(A \vee B)$ est une formule.
5. Si A et B sont des formules, $(A \rightarrow B)$ est une formule.
6. Si A et B sont des formules, $(A \leftrightarrow B)$ est une formule.
7. Si A est une formule, $\forall x_i A$ est une formule. $(x_i \in \{x_0, \ldots, x_n\})$
8. Si A est une formule, $\exists x_i A$ est une formule. $(x_i \in \{x_0, \ldots, x_n\})$
9. Rien d'autre n'est une formule de L_1.

La syntaxe de L_1

Les formules de L_1

Remarques sur les règles de formation

- On appelle les formules qui contiennent des connecteurs logiques ou des quantificateurs, c'est-à-dire les formules construites au moyen des règles de formation 2-8, des formules *moléculaires* ou *formules complexes*
- Les formules qui ne contiennent ni connecteurs logiques, ni quantificateurs, c'est-à-dire qui sont construites seulement grâce à la règle 1, sont appelées *formules atomiques* ou *formules simples*
- Les règles contiennent des métavariables (A, B) qui sont utilisées pour référer à des formules **atomiques ou complexes** de L_1 – chaque formule de L_1 peut être construite par une série d'applications des règles de formation 1-8
- Conventions de notation : Pour simplifier, les parenthèses extérieures, telles que les parenthèses qui contiennent des formules atomiques, peuvent être omises ; p.ex. on peut écrire "$(Pa \wedge \exists x Qx)$" au lieu de "$Pa \wedge \exists x Qx$"

La syntaxe de L_1

Les formules de L_1

Exemples : formules bien-formées de L_1 – et leur simplifications conventionnelles

- $P^1_1 a$ – Pa
- $P^2_3 xy$ – Rxy
- $P^2_3 ay$ – Ray
- $\exists x_0((P^2_0 x_0 x_1 \wedge (P^1_0 c_0 \vee \forall x_2((P^2_0 x_2 x_0 \leftrightarrow P^2_0 x_0 x_2)))))$ – $\exists x(Pxy \wedge (Qa \vee \forall z(Pzx \leftrightarrow Pxz)))$
- et chaque formule bien-formée de L_0 dans laquelle chaque variable propositionnelle est remplacée par une formule atomique de L_1

La syntaxe de L_1

Les formules de L_1

Exemples : formules mal-formées de L_1 (simplifications conventionnelles)

- PQ
- QPx
- xP
- $\forall(Px \rightarrow Qx)$
- $Qx\forall x$
- $\exists a(Pa)$ (ou "a" est une constante)
- et chaque formule mal-formée de L_0 dans laquelle chaque variable propositionnelle est remplacée par une formule atomique de L_1