

RAPPEL DES MÉTHODES À APPLIQUER DANS LES EXERCICES

INCO2: LOGIQUE PROPOSITIONNELLE

Ordre des operateurs:

L'ordre des opérations pour les connecteurs logiques est:

\neg, \wedge, \vee , et \rightarrow / \equiv

Tables de vérité:

Négation		Conjonction		
p	$\neg p$	p	q	$p \wedge q$
T	F	T	T	T
F	T	T	F	F
		F	T	F
		F	F	F

Disjonction		Implication:			
p	q	$p \vee q$	p	q	$p \rightarrow q$
T	T	T	T	T	T
T	F	T	T	F	F
F	T	T	F	T	T
F	F	F	F	F	T

Equivalence:

p	q	$p \equiv q$
T	T	T
T	F	F
F	T	F
F	F	T

Exemple d'évaluation dans *une* situation (ici p:fausse, q:vraie):

$$V: \quad (\neg p) \wedge q \quad V \models (\neg p) \wedge q$$

Exemple d'évaluation dans *toutes* les situations:

$$(p \wedge (p \rightarrow q)) \rightarrow q$$

V:

1 1 1 1 1 1 1
 1 0 1 0 0 1 0
 0 0 0 1 1 1 1
 0 0 0 1 0 1 0

$$V \models (p \wedge (p \rightarrow q)) \rightarrow q$$

(la formule est *valide*, c'est une tautologie)

(autres possibilités: satisfaisable, contradictoire)

Exemple de conséquence logique:

$p \mid (p \rightarrow q) \parallel q$

1 1 1 1 1

1 1 0 0 0

0 0 1 1 1

0 0 1 0 0

V: $p, (p \rightarrow q) \models q$

Exemple d'un système d'axiomes correct et complet:

1. $(\varphi \rightarrow (\psi \rightarrow \varphi))$.
2. $((\varphi \rightarrow (\psi \rightarrow \chi)) \rightarrow ((\varphi \rightarrow \psi) \rightarrow (\varphi \rightarrow \chi)))$.
3. $((\neg \phi \rightarrow \neg \phi) \rightarrow (\psi \rightarrow \varphi))$.
4. Modus ponens (MP): $\varphi, \varphi \rightarrow \psi \vdash \psi$.

Par exemple, à partir de:

$p \rightarrow ((q \rightarrow p) \rightarrow p)$

on peut dériver:

$p \rightarrow p$ (c'est un théorème du système d'axiomes)

INCO3-4-5: SYLLOGISTIQUE ET LOGIQUE DES PRÉDICATS

Notation pour les situations comme diagrammes de Venn:

Trois situations, représenté par trois ensembles avec chevauchements entre eux.

Tous ... sont ... -> faites ombrage des ensembles vides

Quelque ... est ... -> dessinez un "x" dans l'ensemble

Il n'y a de ... -> faites ombrage des ensembles vides

Quelque ... n'est pas ... -> dessinez un "x" dans l'ensemble complement

Methode:

- a) dessiner la squelette (deux ou trois ensembles) dans une domaine (rectangle)
- b) premises: faire ombrage des ensembles vides
- c) premises: remplir les ensembles avec des éléments spécifiques
- d) verifier qu'il y a au moins un ensemble pour la situation correspondante à la conclusion
- e) donner une response (vrai ou fausse) sur la validité de la conclusion

Logique des prédicats:

Ingrédients:

Symboles pour les constants: a,b,c,...

Symboles pour les variables: x,y,z,...

Symboles pour les prédicats: A,B,C,...P,Q,R,...

Opérateurs logiques: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$

Quantificateurs: $\forall x$ ("tous les x") and $\exists x$ ("au moins un x")

Exemples:

Jean regarde Marie: $\text{regarde}(j,m)$

Tout le monde regarde quelqu'un: $\forall x \exists y(\text{regarde}(x,y))$

Si on regarde quelqu'un, il nous regarde: $\forall x \forall y(\text{regarde}(x,y) \rightarrow \text{regarde}(y,x))$

Évaluation diagrammatique d'une formule de logique des prédicats:

a) assigner des formes pour chaque prédicat unaire et des flèches pour chaque prédicat binaire

b) dans un domaine (rectangle), dessiner les formes et les flèches correspondantes à les formules

Interprétation sémantique:

Un modèle est un n-uplet $M = \langle D, I, g \rangle$ où:

D est le domaine: une collection non vide d'objets;

I est la fonction d'interprétation, en attribuant:

à chaque symbole de constante un objet D

à chaque symbole de prédicat une relation sur les objets dans D

g est l'assignation de variable, en attribuant à chaque variable un objet dans D.

Évaluation sémantique d'une formule de logique des prédicats:

vérifier que chaque formule soit satisfaisable par sa interprétation.

INCO6: RDF, SPARQL

Exemples de triplets RDF(S) (syntaxe Turtle):

@prefix rdf: <<http://www.w3.org/1999/02/22-rdf-syntax-ns#>>

@prefix rdfs: <<http://www.w3.org/2000/01/rdf-schema#>>

@prefix dbpedia: <<http://dbpedia.org/resource/>>

@prefix mononto: <<http://www.monontologie.org#>>

mononto:Neuromancer rdf:type mononto:LivreDeChevet .

mononto:Neuromancer mononto:auteur dbpedia:William_Gibson .

mononto:Neuromancer rdfs:label "Gibson's Neuromancer" .

_:id1 mononto:inclut mononto:Neuromancer .

mononto:LivreDeChevet rdfs:subClassOf mononto:Livre .

mononto:auteurScienceFiction rdfs:subPropertyOf mononto:auteur .

mononto:auteur rdfs:domain mononto:Oeuvre .

mononto:auteur rdfs:range mononto:Personne .

Logique des prédicats et RDF(S), exemples de correspondances:

$\text{Duck}(\text{Donald_Duck}) \Leftrightarrow \text{mononto:Donald_Duck} \text{ rdf:type mononto:Duck}$
 $\text{nièce}(\text{Donald_Duck}, \text{Oncle_Picsou}) \Leftrightarrow \text{mononto:Donald_Duck} \text{ mononto:nièce mononto:Oncle_Picsou}$
 $\forall x(\text{Vieu_canard}(x) \rightarrow \text{Canard}(x)) \Leftrightarrow \text{mononto:Vieu_canard} \text{ rdf:subClassOf mononto:Canard}$
 $\forall xy(\text{nièceMasculin}(c,y) \rightarrow \text{nièce}(x,y)) \Leftrightarrow \text{mononto:nièceMasculin} \text{ rdfs:subPropertyOf mononto:nièce}$

Interprétation est comme en logique des prédicats, sauf que:

- subClassOf et subPropertyOf sont reflexive (différemment de l'implication)
- rdfs:Resource et rdfs:Literal sont disjoints

Exemples de requêtes SPARQL:

```

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX type: <http://dbpedia.org/class/yago/>
PREFIX prop: <http://dbpedia.org/property/>
SELECT ?country_name ?population
WHERE {
?country a type:LandlockedCountries ;
          rdfs:label ?country_name ;
          prop:populationEstimate ?population .
FILTER (?population > 15000000) .
}

```

```

PREFIX myont: <http://example.org/myont.owl#>
PREFIX type: <http://dbpedia.org/class/yago/>
PREFIX dbpo: <http://dbpedia.org/ontology/>
CONSTRUCT { ?country myont:hasWaterPercentage ?water }
WHERE {
?country a type:LandlockedCountries ;
          dbpo:percentageOfAreaWater ?water }
} LIMIT 100

```

```

select distinct ?thing ?p
where {
{ ?thing ?p dbpedia:Lake_Victoria }
UNION
{ dbpedia:Lake_Victoria ?p ?thing }
UNION
{ ?thing dbpedia-owl:abstract ?abstract .
  FILTER regex(?abstract, "Lake Victoria") }
} LIMIT 100

```