Predicate Logic

Predicate, or first-order, logic is a formal logical system that extends propositional logic by additional logical operators, called quantifiers, and by variables that range over domains other than Boolean values.

For example, consider the definition of reflexivity for a binary relation R:

> *A binary relation R on a set A is called* reflexive *if xRx for all x in A.*

This definition can not be formalized in propositional logic, but requires predicate logic:

$$\forall x\ [x \in A \to (x,x) \in R]$$

or, abbreviated,

$$(\forall x \in A)\ (x,x) \in R.$$

The symbol $\forall$ is called a *quantifier* or more specifically a *universal quantifier*. The letter x denotes a universally quantified variable.

Intuitive Semantics

Intuitively, the above formula is true, for a given set A and a subset $R \subseteq A \times A$, if, and only if, the set R contains all pairs (a,a), where a is an element of A.

For example, let A be the set $\{1,2,3\}$.

If R denotes the binary relation $\{(1,1),(2,2),(3,3)\}$ then the formula is true. But for $R = \{(1,2),(2,3),(3,1)\}$ it is false.

In general, the truth value of predicate logic formulas depends on how variables, such as A and R, are interpreted. In some instances a formula may be true; in other cases, false.

Example - Circularity

Consider now a slightly more complicated formula,

$$\forall x \forall y \forall z\ [(x \in A \land y \in A \land z \in A) \to ((x,y) \in R \land (y,z) \in R \to (z,x) \in R)]$$

or, abbreviated,

$$\forall x \in A\, \forall y \in A\, \forall z \in A\ [(x,y) \in R \land (y,z) \in R \to (z,x) \in R].$$

This formula expresses that the relation R is *circular*. Informally, it is true if, and only if, for all x, y, and z in A such that xRy and yRz, it is the case that zRx.

For example, if $A = \{1,2,3\}$ and $R = \{(1,1),(2,2),(3,3)\}$, then the formula is true.

The formula is also true if for R we take

$$\{(1,2),(2,3),(3,1)\}.$$

However, if we take the union of the two relations

$$\{(1,1),(2,2),(3,3),(1,2),(2,3),(3,1)\}$$

the formula is false.

Function and Predicate Symbols

Predicate logic extends the language of propositional logic by *function* and *predicate symbols*.

We use the letters $f,g,h,\ldots$ to denote function symbols, and the letters $p,q,r,\ldots$ to denote predicate symbols.

We also assume that with each function and predicate symbol a non-negative integer, called its *arity*, is associated.

Function symbols are meant to denote functions over a certain domain; predicate symbols, relations or properties. The arity indicates the number of arguments a function or relation takes.

A function or predicate symbol of arity 0 is also called a *constant*. We use the letters $a,b,c,\ldots$ to denote constant function symbols.

Finally, we use the letters $x,y,z,\ldots$ to denote *variables*, which denote elements of a given domain.

Terms and Atoms

Let $\mathcal{F}$ and $\mathcal{R}$ be sets of function and predicate symbols, respectively, and $\mathcal{X}$ be a set of variables.

The set of *terms* (over $\mathcal{F}$ and $\mathcal{X}$) is defined inductively by:

- every variable x in $\mathcal{X}$ is a term, and
- if f is a function symbol in $\mathcal{F}$ of arity n, and $t_1, \ldots, t_n$ are terms, then $f(t_1, \ldots, t_n)$ is also a term.

Note that constants are terms by this definition.

Similarly, we define the set of *atomic formulas*, or *atoms* for short, by:

- if p is a predicate symbol in $\mathcal{R}$ of arity n, and $t_1, \ldots, t_n$ are terms, then $p(t_1, \ldots, t_n)$ is an atom.

If a term or atom contains no variables, we say that it is *variable-free* or *ground*.

The parentheses are not necessary, but increase the readability.

First-Order Logic – Syntax

Predicate logic contains logical operators called *quantifiers*, more specifically a *universal* quantifier $\forall$ and an *existential* quantifier $\exists$.

The symbols of a *first-order (predicate) logic* are thus the following:

logical connectives: $\wedge, \vee, \neg, \rightarrow$
quantifiers: $\forall, \exists$
function symbols: $f, g, h, \ldots$
predicate symbols: $p, q, r, \ldots$
variables: $x, y, z, \ldots$

A *first-order language* $\mathcal{L}$ is specified by its sets of function and predicate symbols and variables.

Syntactically well-formed *formulas* are expressions constructed from atomic formulas and the logical operators. The syntax rules for propositional connectives are the same as for propositional logic (except that atomic formulas are used instead of propositional variables).

Quantified formulas are of the form

$$(\forall x\ F) \text{ or } (\exists x\ F)$$

where F is a formula and x a variable.

The formula F is called the *scope* of the quantifier; and we say that the quantifier *binds* the variable x.

Examples of Predicate Logic Formulas

Some students do not satisfy the prerequisites for CSE-213.

$$\exists x(Student(x) \wedge \neg PrereqCSE213(x)$$

All students who satisfy the prerequisites for CSE-213 may take the course.

$$\forall x[Student(x) \wedge PrereqCSE213(x) \rightarrow TakesCSE213(x)]$$

Some students in CSE-213 drink.

$$\exists x[Student(x) \wedge TakesCSE213(x) \wedge Drinks(x)]$$

If some students drink then all students drink.

$$\exists x(Student(x) \wedge Drinks(x)) \rightarrow \forall x(Student(x) \rightarrow Drinks(x))$$

More Examples

Only dogs bark.

Rephrase: *It barks only if it is a dog.*

Or equivalently: *If it barks, then it is a dog.*

$$\forall x\,[Barks(x) \rightarrow Dog(x)]$$

Everyone has a father.

$$\forall x\,[Person(x) \rightarrow \exists y\,(Person(y) \wedge Father(y, x))]$$

Nobody is infallible.

$$\neg \exists x\,[Person(x) \wedge \neg Fallible(x)]$$

Free and Bound Variables

There is an important distinction between *bound* and *free* occurrences of variables.

All occurrences of the variable x in a subformula $\forall x\, F$ or $\exists x\, F$ are said to be *bound*. Occurrences of x that are not bound are said to be *free*.

In other words, bound occurrences of a variable are those that occur in the scope of a quantifier binding that variable. The same variable may have both free and bound occurrences in a formula.

Formulas without free occurrences of variables are called *sentences*.

Example. The formula

$$\forall x\ ((r(x,y) \rightarrow (\forall y((\exists x\ r(z,x)) \wedge r(x,z)))$$

is not a sentence. It contains both free and bound occurrences of variables.

In general, the truth value of a formula depends on the assignment of values to its free variables. Semantically, sentences are formulas that are either true or false.

Substitution

An important operation in predicate logic is the substitution of terms (or values) for variables, more specifically for *free* occurrences of variables.

Definition.

> If F is a predicate logic formula, t is a term, and x is a variable, then we denote by $F(x/t)$ the formula obtained by (simultaneously) replacing all free occurrences of x in F by t.
>
> The expression x/t is called a *substitution*.

If the variable x is not free in F, i.e., if F contains only bound occurrences of x or no occurrences of x at all, then $F(x/t)$ is identical to F.

For example, let F be the formula $\forall x\, p(x,y)$ and let t be a term. Then $F(x/t) = F$ and $F(y/t) = \forall x\, p(x,t)$.

Replacements of different (free) occurrences of the same variable take place simultaneously.

For instance, if F is the disjunction $p(x,x) \vee \exists y\, q(x,y)$, then $F(x/h(x))$ is $p(h(x),h(x)) \vee \exists y\, q(h(x),y)$.

Substitution (cont.)

Unfortunately, substitutions may have undesired side effects semantically, and hence their application is usually done subject to certain restrictions.

Definition.

> We say that a term t is *free to replace* a variable x in a formula F if no free occurrence of x is in the scope of a quantifier that binds a variable y occurring in t.

Each of the following conditions implies that t is free to replace x in F:

1. $t = x$
2. t is a constant.
3. The variables of t do not occur in F.
4. The variables of t do not occur within the scope of a quantifier in F.
5. The formula F contains no quantifiers.
6. The variable x does not occur free within the scope of a quantifier in F.

Is $f(x,y)$ free for x in $(\forall x\ p(x) \wedge q(y)) \rightarrow (\neg p(x) \vee q(y))$?

Is $f(x,y)$ free for y in $\forall x\ (p(x) \wedge q(y)) \rightarrow (\neg p(x) \vee q(y))$?

Interpretations

Let $\mathcal{F}$ be a set of function symbols, $\mathcal{R}$ a set of predicate symbols, and $\mathcal{X}$ a set of variables.

An *interpretation* $\mathcal{I}$ for the predicate logic based on $\mathcal{F}$, $\mathcal{R}$, and $\mathcal{X}$ consists of the following components:

1. a nonempty set D, which is called the *domain* of the interpretation,
2. for each n-ary function symbol $f \in \mathcal{F}$ an n-ary function $f^{\mathcal{I}} : D^n \rightarrow D$,
3. for each n-ary predicate symbol $p \in \mathcal{R}$ an n-ary relation $p^{\mathcal{I}} \subseteq D^n$, and
4. for each variable x in $\mathcal{X}$ an element $x^{\mathcal{I}}$ in D.

For example, let $\mathcal{F}$ be the set $\{0, 1, +, *, -\}$ and $\mathcal{R}$ the set $\{=, \leq, <\}$. We may take the set of real numbers as domain and define $0^{\mathcal{I}}$ as the real number 0, $1^{\mathcal{I}}$ as the real number 1, $+^{\mathcal{I}}$ as addition, $*^{\mathcal{I}}$ as multiplication, $-^{\mathcal{I}}$ as subtraction, $=^{\mathcal{I}}$ as the equality predicate, $\leq^{\mathcal{I}}$ as the less-than-or-equal-to relation, and $<^{\mathcal{I}}$ as the less-than relation.

Interpretation of Terms

Let $\mathcal{T}(\mathcal{F}, \mathcal{X})$ denote the set of terms built from function symbols in $\mathcal{F}$ and variables in $\mathcal{X}$.

Intuitively, terms denote elements of the domain D. More specifically, each interpretation $\mathcal{I}$ with domain D induces a mapping

$$t^{\mathcal{I}} : \mathcal{T}(\mathcal{F}, \mathcal{X}) \to D$$

defined recursively by:

$$t^{\mathcal{I}} = \begin{cases} x^{\mathcal{I}} & \text{if } t \text{ is a variable } x \\ f^{\mathcal{I}}(t_1^{\mathcal{I}}, \ldots, t_k^{\mathcal{I}}) & \text{if } t = f(t_1, \ldots, t_k) \end{cases}$$

For example, let t be the term $x * (y + (x + 1) * z)$. If $\mathcal{I}$ is an interpretation as indicated above, with $x^{\mathcal{I}} = 3$, $y^{\mathcal{I}} = 2$, and $z^{\mathcal{I}} = 1$, then $t^{\mathcal{I}} = 18$.

Notation

> If $\mathcal{I}$ is an interpretation with domain D and $d \in D$, we denote by $\mathcal{I}(x/d)$ the interpretation that is the same as $\mathcal{I}$ except that $x^{\mathcal{I}(x/d)} = d$.

Semantics of Predicate Logic

Let $\mathcal{I}$ be an interpretation with domain D.

We define the semantics of predicate logic formulas via a binary relation denoted by the symbol $\models$. Informally, $\mathcal{I} \models F$ holds if the formula F is true under interpretation $\mathcal{I}$.

The formal definition is by structural induction on formulas:

(i) If $F = p(t_1, \ldots, t_k)$, then $\mathcal{I} \models F$ iff $(t_1^{\mathcal{I}}, \ldots, t_k^{\mathcal{I}}) \in p^{\mathcal{I}}$.

(ii) If $F = \neg G$, then $\mathcal{I} \models F$ holds iff $\mathcal{I} \models G$ does not hold.

(iii) If $F = G_1 \wedge G_2$, then $\mathcal{I} \models F$ holds iff both $\mathcal{I} \models G_1$ and $\mathcal{I} \models G_2$ hold.

(iv) If $F = G_1 \vee G_2$, then $\mathcal{I} \models F$ holds iff at least one of $\mathcal{I} \models G_1$ and $\mathcal{I} \models G_2$ holds.

(v) If $F = \forall x G$, then $\mathcal{I} \models F$ holds iff $\mathcal{J} \models G$ holds for all interpretations $\mathcal{J} = \mathcal{I}(x/d)$, where $d \in D$.

(vi) If $F = \exists x G$, then $\mathcal{I} \models F$ holds iff $\mathcal{J} \models G$ holds for some interpretation $\mathcal{J} = \mathcal{I}(x/d)$, where $d \in D$.

Models and Countermodels

An interpretation $\mathcal{I}$ is said to be a *model* of a formula F if F is true with respect to $\mathcal{I}$. Otherwise, the interpretation is called a *countermodel* for F.

For example, Let F be the formula

$$\forall x\, [p(f(x, x), x) \to p(x, y)]$$

Consider an interpretation $\mathcal{I}$ where the domain D is the set of natural numbers; the function $f^{\mathcal{I}}$ is defined by $f^{\mathcal{I}}(i, j) = (i + j) \bmod 3$; $p^{\mathcal{I}}$ is the equality relation; and $y^{\mathcal{I}} = 0$. This interpretation is a model for F. (Informally, F expresses the fact that for all natural numbers k, if $2k \bmod 3 = k$, then $k = 0$.)

On the other hand, if $\mathcal{I}$ is an interpretation where the domain D is the set $\{a, b\}$, $f^{\mathcal{I}}$ is any binary function such that $f^{\mathcal{I}}(a, a) = a$ and $f^{\mathcal{I}}(b, b) = b$, $p^{\mathcal{I}}$ is the equality relation, and $y^{\mathcal{I}} = a$; then $\mathcal{I}$ is a countermodel for F.

Validity

Some formulas are true for every possible interpretation. In other words, they are true based on their logical structure. Examples of such formulas are (propositional) tautologies.

In general, we say that a predicate logic formula is *valid* if every interpretation is a model. A formula that is not valid is also called *invalid*.

A formula is said to be *satisfiable* if it has a model; and *unsatisfiable*, otherwise. Thus, a formula is unsatisfiable if every interpretation is a countermodel.

A valid formula is satisfiable, whereas invalid formulas may be satisfiable or unsatisfiable.

For example, the formula $\exists x\, (p(x) \wedge q(x)) \to \exists x\, p(x) \wedge \exists x\, q(x)$ is valid.

On the other hand, $\exists x\, p(x) \wedge \exists x\, q(x) \to \exists x\, (p(x) \wedge q(x))$ is satisfiable, but not valid.

Closures

In many applications formulas with free variables are transformed into (or implicitly interpreted as) sentences.

Let F be a formula with free variables $x_1, \ldots, x_n$. By the *universal closure* of F we mean the sentence

$$\forall x_1 \cdots \forall x_n F$$

and by the *existential closure* of F, the sentence

$$\exists x_1 \cdots \exists x_n F.$$

For example, if F is the formula $\forall x\, p(x, y)$, then its universal closure is $\forall y \forall x\, p(x, y)$ and its existential closure is $\exists y \forall x\, p(x, y)$.

Theorem [Closure Properties]

1. A formula is valid if, and only if, its universal closure is valid.

2. A formula is satisfiable if, and only if, its existential closure is satisfiable.

Logical Equivalence

Two predicate logic formulas F and G are said to be *(logically) equivalent* if, and only if, they have the same models. We write $F \equiv G$ to indicate that F and G are equivalent.

Theorem

$F \equiv G$ iff $(F \to G) \wedge (G \to F)$ is valid.

Examples of equivalences are generalized versions of equivalences from propositional logic. For instance,

$$\neg\neg d(x, y) \equiv d(x, y)$$

is obtained from the propositional equivalence $\neg\neg P \equiv P$ by substituting the atomic formula $d(x, y)$ for P.

Another example is

$$\forall x\, p(x) \vee \neg\forall x\, p(x) \equiv T.$$

In general, if two propositional formulas are logically equivalent, one may replace propositional symbols by arbitrary predicate logic formulas: the resulting formulas are still equivalent.

Other equivalences derive from more subtle connections between quantifiers and propositional connectives.

Negations of Quantified Statements

The negation of a universal statement is logically equivalent to an existential statement:

$$\neg\forall x\, p(x) \equiv \exists x\, \neg p(x).$$

Not all students drink.
Some student does not drink.

The negation of an existential statement is logically equivalent to a universal statement:

$$\neg\exists x\, p(x) \equiv \forall x\, \neg p(x).$$

No student failed the course.
All students passed the course.

These equivalences indicate that semantically only one of the two quantifiers is really needed.

List of Equivalences

1. (a) $\neg\forall x\, F \equiv \exists x\, \neg F$
   (b) $\neg\exists x\, F \equiv \forall x\, \neg F$
   (c) $\forall x\, (F \wedge G) \equiv \forall x\, F \wedge \forall x\, G$
   (d) $\exists x\, (F \vee G) \equiv \exists x F \vee \exists x\, G$
   (e) $\exists x\, (F \to G) \equiv \forall x F \to \exists x\, G$

2. (a) $\forall x \forall y\, F \equiv \forall y \forall x\, F$
   (b) $\exists x \exists y\, F \equiv \exists y \exists x\, F$

3. Assuming that x is not free in G:
   (a) $\forall x\, (F \wedge G) \equiv (\forall x F) \wedge G$
   (b) $\exists x\, (F \wedge G) \equiv (\exists x F) \wedge G$
   (c) $\forall x\, (F \vee G) \equiv (\forall x F) \vee G$
   (d) $\exists x\, (F \vee G) \equiv (\exists x F) \vee G$
   (e) $\forall x\, (F \to G) \equiv (\exists x\, F) \to G$
   (f) $\exists x\, (F \to G) \equiv (\forall x\, F) \to G$
   (g) $\forall x\, (G \to F) \equiv G \to \forall x\, F$
   (h) $\exists x\, (G \to F) \equiv G \to \exists x\, F$

Logical Consequence

We say that a predicate logic formula F *logically implies* G (or that G is a *logical consequence* of F), and write $F \models G$, if every model for F is also a model for G.

Examples of logical consequences are

$$\forall x\, p(x) \models \exists x p(x) \text{ and}$$

$$\exists x\, \forall y\, p(x,y) \models \forall y\, \exists x\, p(x,y).$$

But $\forall x\, \exists y\, p(x,y) \not\models \exists y \forall x\, p(x,y)$.

Note that two formulas F and G are equivalent if each formula logically implies the other.

Are the following logical equivalences correct?

$$\begin{aligned} \forall x\, p(x) \vee \forall x\, q(x) &\equiv \forall x\, (p(x) \vee q(x)) \\ \exists x\, p(x) \wedge \exists x\, q(x) &\equiv \exists x\, (p(x) \wedge q(x)) \end{aligned}$$

If equivalence does not hold, can one prove that one side implies the other?

Dracula

By a logical argument one means a sequence of formulas, of which the last one is called the *conclusion*, while all the others are called *hypotheses*.

An argument is (logically) correct if the conclusion is a logical consequence of the (conjunction of all the) hypotheses.

Example.

(1) Everyone is afraid of Dracula.
(2) Dracula is afraid only of me.
Therefore I am Dracula.

Is this argument correct?