15.3 Predicate Calculus

Predicate logic is a richer system than propositional logic, and it allows complex facts about the world to be represented. It allows new facts about the world to be derived in a way that guarantees that if the initial facts are true then the conclusions are true. Predicate calculus includes predicates, variables, constants and quantifiers.

A *predicate* is a characteristic or property that an object can have, and we are predicating some property of the object. For example, "*Socrates is a Greek*" could be expressed as $G(s)$, with capital letters standing for predicates and small letters standing for objects. A predicate may include variables, and a statement with a variable becomes a proposition once the variables are assigned values. For example, $G(x)$ states that the variable x is a Greek, whereas $G(s)$ is an assignment of values to x. The set of values that the variables may take is termed the universe of discourse, and the variables take values from this set.

Predicate calculus employs quantifiers to express properties such as all members of the domain have a particular property: e.g., $(\forall x)P(x)$, or that there is at least one member that has a particular property: e.g. $(\exists x)P(x)$. These are referred to as the *universal and existential quantifiers.*

The syllogism 'All Greeks are mortal; Socrates is a Greek; therefore Socrates is mortal' may be easily expressed in predicate calculus by

$$
\begin{array}{l}
(\forall x)(G(x) \rightarrow M(x)) \\
G(s) \\
-----------------\\
M(s)
\end{array}
$$

In this example, the predicate $G(x)$ stands for x is a Greek and the predicate $M(x)$ stands for x is mortal. The formula $G(x)\rightarrow M(x)$ states that if x is a Greek then x is mortal, and the formula $(\forall x)(G(x)\rightarrow M(x))$ states for any x that if x is a Greek then x is mortal. The formula $G(s)$ states that Socrates is a Greek and the formula $M(s)$ states that Socrates is mortal.

Example 15.6 (**Predicates**) A predicate may have one or more variables. A predicate that has only one variable (i.e. a unary or 1-place predicate) is often related to sets; a predicate with two variables (a 2-place predicate) is a relation; and a predicate with n variables (a n-place predicate) is a n-ary relation. Propositions do not contain variables and so they are 0-place predicates. The following are examples of predicates:

(i) The predicate *Prime*(x) states that x is a prime number (with the natural numbers being the universe of discourse).
(ii) *Lawyer*(a) may stand for a is a lawyer.
(iii) Mean(m, x, y) states that m is the mean
(iv) of x and y: i.e. $m = ½(x + y)$.
(iv) LT(x, 6) states that x is less than 6.
(v) $G(x, \pi)$ states that x is greater than π (where is the constant 3.14159)
(vi) $G(x, y)$ states that x is greater than y.
(vii) EQ(x, y) states that x is equal to y.
(viii) LE(x, y) states that x is less than or equal to y.
(ix) Real(x) states that x is a real number.

(x) Father(x, y) states that x is the father of y.
(xi) $\neg(\exists x)(Prime(x) \wedge B(x, 32, 36))$ states that there is no prime number between 32 and 36.

Universal and Existential Quantification

The universal quantifier is used to express a statement such as that all members of the domain have property P. This is written as $(\forall x)P(x)$ and expresses the statement that the property $P(x)$ is true for all x. Similarly, $(\forall x_1, x_2, \ldots, x_n)$ $P(x_1, x_2, \ldots, x_n)$ states that property $P(x_1, x_2, \ldots, x_n)$ is true for all $x_1, x_2, \ldots, x_n$. Clearly, the predicate $(\forall x)$ $P(a, b)$ is identical to $P(a, b)$ since it contains no variables, and the predicate $(\forall y \in \mathbb{N})$ $(x \leq y)$ is true if $x = 1$ and false otherwise.

The existential quantifier states that there is at least one member in the domain of discourse that has property P. This is written as $(\exists x)P(x)$ and the predicate $(\exists x_1, x_2, \ldots, x_n)$ $P(x_1, x_2, \ldots, x_n)$ states that there is at least one value of $(x_1, x_2, \ldots, x_n)$ such that $P(x_1, x_2, \ldots, x_n)$ is true.

Example 15.7 (**Quantifiers**)

(i) $(\exists p)$ (Prime(p) $\wedge$ $p > 1{,}000{,}000$) is true
It expresses the fact that there is at least one prime number greater than a million, which is true as there are an infinite number of primes.
(ii) $(\forall x)$ $(\exists y)$ $x < y$ is true
This predicate expresses the fact that given any number x we can always find a larger number: e.g. take $y = x + 1$.
(iii) $(\exists y)$ $(\forall x)$ $x < y$ is false
This predicate expresses the statement that there is a natural number y such that all natural numbers are less than y. Clearly, this statement is false since there is no largest natural number, and so the predicate $(\exists y)$ $(\forall x)$ $x < y$ is false.

Comment 15.1

It is important to be careful with the order in which quantifiers are written, as the meaning of a statement may be completely changed by the simple transposition of two quantifiers.

The well-formed formulae in the predicate calculus are built from terms and predicates, and the rules for building the formulae are described briefly in Sect. 15.3.1. Examples of well-formed formulae include

$$(\forall x)(x > 2)$$
$$(\exists x)x^2 = 2$$
$$(\forall x)\ (x > 2 \wedge x < 10)$$
$$(\exists y)x^2 = y$$
$$(\forall x)\ (\forall y)\ Love(y, x) \qquad \text{(everyone is loved by someone)}$$
$$(\exists y)\ (\forall x) Love(y, x) \qquad \text{(someone loves everyone)}$$

The formula $(\forall x)(x > 2)$ states that every x is greater than the constant 2; $(\exists x)\ x^2 = 2$ states that there is an x that is the square root of 2; $(\forall x)\ (\exists y)\ x^2 = y$ states that for every x there is a y such that the square of x is y.

15.3.1 Sketch of Formalization of Predicate Calculus

The formalization of predicate calculus includes the definition of an alphabet of symbols (including constants and variables), the definition of function and predicate letters, logical connectives and quantifiers. This leads to the definitions of the terms and well-formed formulae of the calculus.

The predicate calculus is built from an alphabet of constants, variables, function letters, predicate letters and logical connectives (including the logical connectives discussed in propositional logic, and universal and existential quantifiers).

The definition of terms and well-formed formulae specifies the syntax of the predicate calculus, and the set of well-formed formulae gives the language of the predicate calculus. The terms and well-formed formulae are built from the symbols, and these symbols are not given meaning in the formal definition of the syntax.

The language defined by the calculus needs to be given an *interpretation* in order to give a meaning to the terms and formulae of the calculus. The interpretation needs to define the domain of values of the constants and variables, provide meaning to the function letters, the predicate letters and the logical connectives.

Terms are built from constants, variables and function letters. A constant or variable is a term, and if $t_1, t_2, \ldots, t_k$ are terms, then $f_i^k(t_1, t_2, \ldots, t_k)$ is a term (where f_i^k is a k-ary function letter). Examples of terms include

$$x^2 \qquad \text{where } x \text{ is a variable and square is a } 1-\text{ary function letter}$$
$$x^2 + y^2 \qquad \text{where } x^2 + y^2 \text{ is shorthand for the function add(square}(x),\ \text{square}(y))$$
$$\text{where add is a } 2-\text{ary function letter and square is a } 1-\text{ary function}$$

The well-formed formulae are built from terms as follows. If P_i^k is a k-ary predicate letter, $t_1, t_2, \ldots, t_k$ are terms, then $P_i^k\ (t_1, t_2, \ldots, t_k)$ is a well-formed formula. If A and B are well-formed formulae then so are $\neg A$, $A \wedge B$, $A \vee B$, $A \rightarrow B$, $A \leftrightarrow B$, $(\forall x)A$ and $(\exists x)A$.

There is a set of axioms for predicate calculus and two rules of inference used for the deduction of new formulae from the existing axioms and previously deduced formulae. The deduction of a new formula Q is via a sequence of well-formed formulae P_1, P_2, ... P_n (where $P_n = Q$) such that each P_i is either an axiom, a hypothesis or deducible from one or more of the earlier formulae in the sequence.

The two rules of inference are *modus ponens* and *generalization*. Modus ponens is a rule of inference that states that given predicate formulae A, and $A \Rightarrow B$ then the predicate formula B may be deduced. Generalization is a rule of inference that states that given predicate formula A, then the formula $(\forall x)$A may be deduced where x is any variable.

The deduction of a formula Q from a set of hypothesis H is denoted by $H \vdash Q$, and where Q is deducible from the axioms alone this is denoted by $\vdash$ Q. The *deduction theorem* states that if $H \cup \{P\} \vdash$ Q then H $\vdash$ P → Q[3] and the converse of the theorem is also true: i.e. if H $\vdash$ P → Q then H ∪ {P} $\vdash$ Q.

The approach allows reasoning about symbols according to rules, and to derive theorems from formulae irrespective of the meanings of the symbols and formulae. Predicate calculus is *sound*: i.e. any theorem derived using the approach is true, and the calculus is also *complete*.

Scope of Quantifiers

The scope of the quantifier $(\forall x)$ in the well-formed formula $(\forall x)A$ is A. Similarly, the scope of the quantifier $(\exists x)$ in the well-formed formula $(\exists x)B$ is B. The variable x that occurs within the scope of the quantifier is said to be a *bound variable*. If a variable is not within the scope of a quantifier it is *free*.

Example 15.8 (**Scope of Quantifiers**)

(i) x is free in the well-formed formula $\forall y\ (x^2 + y > 5)$
(ii) x is bound in the well-formed formula $\forall x\ (x^2 > 2)$

A well-formed formula is *closed* if it has no free variables. The substitution of a term t for x in A can only take place only when no free variable in t will become bound by a quantifier in A through the substitution. Otherwise, the interpretation of A would be altered by the substitution.

A term t is free for x in A if no free occurrence of x occurs within the scope of a quantifier $(\forall y)$ or $(\exists y)$ where y is free in t. This means that the term t may be substituted for x without altering the interpretation of the well-formed formula A.

For example, suppose A is $\forall y\ (x^2 + y^2 > 2)$ and the term t is y, then t is not free for x in A as the substitution of t for x in A will cause the free variable y in t to become bound by the quantifier $\forall y$ in A, thereby altering the meaning of the formula to $\forall y\ (y^2 + y^2 > 2)$.

[3]This is stated more formally that if H ∪ {P} $\vdash$ Q by a deduction containing no application of generalization to a variable that occurs free in P then H $\vdash$ P → Q.

15.3.2 Interpretation and Valuation Functions

An *interpretation* gives meaning to a formula and it consists of a *domain of discourse* and a *valuation function*. If the formula is a sentence (i.e. does not contain any free variables) then the given interpretation of the formula is either true or false. If a formula has free variables, then the truth or falsity of the formula depends on the values given to the free variables. A formula with free variables essentially describes a relation say, $R(x_1, x_2, \ldots x_n)$ such that $R(x_1, x_2, \ldots x_n)$ is true if $(x_1, x_2, \ldots x_n)$ is in relation R. If the formula is true irrespective of the values given to the free variables, then the formula is true in the interpretation.

A *valuation* (meaning) *function gives meaning to the logical symbols and connectives*. Thus, associated with each constant c is a constant c_Σ in some universe of values Σ; with each function symbol f of arity k, we have a function symbol f_Σ in Σ and $f_\Sigma: \Sigma^k \rightarrow \Sigma$; and for each predicate symbol P of arity k a relation $P_\Sigma \subseteq \Sigma^k$. *The valuation function, in effect, gives the semantics of the language of the predicate calculus L.*

The truth of a predicate P is then defined in terms of the meanings of the terms, the meanings of the functions, predicate symbols, and the normal meanings of the connectives.

Mendelson [3] provides a technical definition of truth in terms of *satisfaction* (with respect to an interpretation M). Intuitively a formula F is *satisfiable* if it is *true* (in the intuitive sense) for some assignment of the free variables in the formula F. If a formula F is satisfied for every possible assignment to the free variables in F, then it is *true* (in the technical sense) for the interpretation M. An analogous definition is provided for *false* in the interpretation M.

A formula is *valid* if it is true in every interpretation; however, as there may be an uncountable number of interpretations, it may not be possible to check this requirement in practice. M is said to be a model for a set of formulae if and only if every formula is true in M.

There is a distinction between proof theoretic and model theoretic approaches in predicate calculus. *Proof theoretic* is essentially syntactic, and there is a list of axioms with rules of inference. The theorems of the calculus are logically derived (i.e. $\vdash A$) and the logical truths are as a result of the syntax or form of the formulae, rather than the *meaning* of the formulae. *Model theoretical*, in contrast is essentially semantic. The truth derives from the meaning of the symbols and connectives, rather than the logical structure of the formulae. This is written as $\vdash_M A$.

A calculus is *sound* if all of the logically valid theorems are true in the interpretation, i.e. proof theoretic $\Rightarrow$ model theoretic. A calculus is *complete* if all the truths in an interpretation are provable in the calculus, i.e. model theoretic $\Rightarrow$ proof theoretic. A calculus is *consistent* if there is no formula A such that $\vdash A$ and $\vdash \neg A$.

The predicate calculus is sound, complete and consistent. *Predicate calculus is not decidable*: i.e. there is no algorithm to determine for any well-formed formula A whether A is a theorem of the formal system. The undecidability of the predicate calculus may be demonstrated by showing that if the predicate calculus is decidable then the halting problem (of Turing machines) is solvable. We discussed the halting problem in Chap. 13.

15.3.3 Properties of Predicate Calculus

The following are properties of the predicate calculus.

$$
\begin{array}{rl}
\text{(i)} & (\forall x)P(x) \equiv (\forall y)P(y) \\
\text{(ii)} & (\forall x)P(x) \equiv \neg(\exists x)\neg P(x) \\
\text{(iii)} & (\exists x)P(x) \equiv \neg(\forall x)\neg P(x) \\
\text{(iv)} & (\exists x)P(x) \equiv (\exists y)P(y) \\
\text{(v)} & (\forall x)(\forall y)P(x,y) \equiv (\forall y)\,(\forall x)P(x,y) \\
\text{(vi)} & (\exists x)(P(x) \vee Q(x)) \equiv (\exists x)P(x) \vee (\exists y)Q(y) \\
\text{(vii)} & (\forall x)P(x) \wedge Q(x)) \equiv (\forall x)\,P(x) \wedge (\forall y)Q(y)
\end{array}
$$