Introduction to the theory of computation
week 10 (Course Notes, chapter 6)

Predicates

Propositions are either true or false. A PREDICATE is true or false, depending on zero or more arguments (so a proposition is a predicate of zero arguments). For example, suppose the predicate $E(n)$ means "n is even." Then $E(n)$ is true for $n = 2$, but false for $n = 3$. We will consider predicates that come from the same DOMAIN OF DISCOURSE (denoted D), so in symbols, predicate P is a boolean-valued function:

$$P : D \times \cdots \times D \to \{0, 1\}.$$

Here's an example. Suppose D is the set of all people, and $S(x, y)$ is the predicate "x is a sibling of y." Then $S(x, y)$ is either true or false, depending on which people get assigned to x and y. Another, completely equivalent way of thinking of this is $S(x, y)$ is true if the pair (x, y) belongs in the binary RELATION "sibling" and false otherwise. Similarly, $E(n)$ is true if n belongs to the unary relation "even number" and false otherwise.

Combining predicates

You can combine predicates using the five connectives from propositional logic: $\{\wedge, \vee, \neg, \to, \leftrightarrow\}$, with the same rules relating the truth of predicates to the truth of combinations of predicates. For example, if D is the set of all people, $S(x, y)$ is the sibling relation, and $M(x)$ asserts that x is male, then

$$S(x, y) \wedge M(x)$$

is true of if x and y are siblings and x is y's brother, and false otherwise. In addition to the five logical connectives, we have quantifiers $\forall$ and $\exists$:

$\forall$ (UNIVERSAL QUANTIFIER): This can be thought of as $\wedge$ on steroids. If our domain of discourse is $D = \{x_1, x_2, \ldots\}$, then $\forall x A(x)$ asserts $A(x_1) \wedge A(x_2) \wedge \cdots$ — the predicate A is evaluated at all elements of D simultaneously, and the quantified formula is false if the unquantified formula is false for one or more elements of D.

$\exists$ (EXISTENTIAL QUANTIFIER): Symmetrically, this can be thought of as $\vee$ on steroids. If our domain of discourse is $D = \{x_1, x_2, \ldots\}$, then $\exists x A(x)$ asserts $A(x_1) \vee A(x_2) \vee \cdots$ — the predicate A is evaluated at all elements of D simultaneously, and the quantified formula is true if the unquantified formula is true for one or more elements of D.

If D is the natural numbers, and $E(n)$ means that n is even, how do you evaluate:

- $\forall n E(n)$.[1]

- $\exists n E(n)$.[2]

If D is the set of all people, and $S(x, y)$ is the sibling relation, how do you evaluate:

- $\exists x \forall y S(x, y)$.[3].
- $\forall x \exists y S(x, y)$[4]

First-order language

We use the now-familiar tool of inductive definition to define a FIRST-ORDER LANGUAGE, a set of well-formed predicate formulas. Here are the constituent parts:

- An infinite set of variables $\{x, y, z, \ldots\}$.
- A set of predicate symbols $\{P_1, P_2, \ldots\}$. Each predicate symbols has an ARITY of no less than zero, that is, the number of arguments that it takes.
- A set of CONSTANT SYMBOLS $\{c_1, c_2, \ldots\}$.
- Symbols from the set $\{\vee, \wedge, \neg, \rightarrow, \leftrightarrow, \exists, \forall, (,)\}$.

Suppose you are told which variables, predicate symbols, and constant symbols are allowed in a first-order language L. The TERMS of L are the variables and constants of L. An ATOMIC FORMULA of L is an expression of the form $P(t_1, \ldots, t_k)$, where the t_i are terms of L and P is a predicate symbol of L of arity k. The SET OF FIRST-ORDER FORMULAS of L is the smallest set such that

BASIS: Any atomic formula of L is in the set.

INDUCTION STEP: If F_1 and F_2 are in the set, and x is a variable of L, then $\neg F_1$, $(F_1 \wedge F_2)$, $(F_1 \vee F_2)$, $(F_1 \rightarrow F_2)$, $(F_1 \leftrightarrow F_2)$, $\forall x F_1$, and $\exists x F_1$ are also in the set.

This definition tells you exactly which formulas are syntactically-correct formulas of L. The structure should seem VERY familiar. You can omit parentheses following rules that are analogous to those for propositional formulas. First-order formulas can be represented as trees, in order to understand their structure better. Consider an example, the language of arithmetic, LA, which has:

- An infinite set of variables including $\{x, y, z, u, v, w\}$.
- Predicates L of arity 2, S of arity 3, P of arity 3, and $\approx$ of arity 2 (the equality predicate).
- Constant symbols 0 and 1.

LA would include formulas such as $\forall x(L(x, y) \rightarrow L(y, x))$, but not $L(x, y) \rightarrow$.

Free variables and scope

If D is the set of natural numbers, and $E(n)$ is the predicate "n is even," then $E(n)$ is a predicate of one variable — it is true or false, depending on which natural number is assigned to n. However $\forall n E(n)$ is a predicate of zero variables — its truth value doesn't depend on which natural number n we consider, since it asserts that something is true for the entire set of natural numbers.

If D is the set of all people, and $S(x, y)$ is the predicate "x is a sibling of y," then what is the arity of $\exists x S(x, y)$?[5] We say that x is a DUMMY variable that is BOUND by the quantifier $\exists x$. A predicate is not

ABOUT its dummy variables, and dummy variables don't contribute to its arity. Things can get a little involved:

$$(\forall x P(x) \wedge \exists y Q(x, y)).$$

In this case, the instance of x in $P(x)$ is bound, the instance of x in $Q(x, y)$ is FREE (not bound), and the instance of y is bound. Now consider

$$\forall x(P(x, y) \to \exists x Q(x, y)),$$

where every occurrence of x is bound, and every occurrence of y is free. Here are the rules (Course Notes page 148), for arbitrary formulas F_1 and F_2:

- $\neg F_1$ has the same free variables as F_1.
- $(F_1 \oplus F_2)$ has the union of the free variables in F_1 and F_2, if $\oplus$ is one of the four binary connectives.
- $\forall x F_1$ and $\exists x F_1$ have the free variables of F_1, minus x.

Interpretations and truth

To decide the truth value of a propositional formula, it is enough to know the truth value of each propositional variable (the truth assignment). A lot more information is needed to decide the truth value of predicate $P(x, y, z, c)$

1. We need to know the domain of discourse, D, from which x, y, and z can be chosen, and to which c belongs.
2. We need to know the meaning of the predicate to which P corresponds. Alternatively, we need to know which quadruples (x, y, z, w) are members of the subset P of D^4 that defines P.
3. We need to know which element of D corresponds to c
4. We need to know which domain elements free variables x, y, and z correspond to.

Suppose we only knew three out of four of the above for a given language L. Then we have a STRUCTURE S for language L, and we know D^S, c^S, and P^S (the domain, the mapping of the constant(s) to elements of the domain, and the mapping of predicate symbols(s) to predicates).

Although this is insufficient to determine the truth of every formula in L, it is certainly enough to evaluate every SENTENCE (formula with no free variables) in L.

You can have two different structures for the same language. Suppose L has predicate symbols P (of arity 3), $\approx$ (of arity 2, the equality predicate), and M (of arity 2). In addition we have an infinite set of variables that include $\{x, y, z\}$, and constant symbol c. Here are two completely different structures for L:

1. In structure S_1, L is a language for describing family relationships
   (a) D is the set of all people
   (b) $P(x, y, z)$ expresses the predicate "x and y are parents of z," while $M(x, y)$ expresses the predicate "x is the mother of y."
   (c) Constant symbol c refers to Madonna.
2. In structure S_2, L is a language for describing facts about arithmetic
   (a) D is the natural numbers

(b) $P(x, y, z)$ expresses the predicate "x plus y equals z," while $M(x, y)$ expresses the predicate "x is greater than y."

(c) Constant symbol c refers to 0.

Given a structure S for language L, a VALUATION of S is a function σ that maps each variable of L to an element of D:

$$\sigma(x, y, z) = (\text{Frank}, \text{Fatimah}, \text{Frank}).$$

Notice that there is no need for σ to be 1–1 (we could map EVERY variable to Madonna). Given structure S_1 and valuation σ, $P(x, y, z)$ expresses the predicate "Frank and Fatimah are Frank's parents." If you already know σ, then $\sigma|_v^x$ maps x to v and is identical to σ for every other variable y.

$$\sigma|_{\text{Fred}}^{z}(x, y, z) = (\text{Frank}, \text{Fatimah}, \text{Fred}).$$

An interpretation $I = (S, \sigma)$ combines a structure S for language L with a valuation σ. This gives enough information to define the truth value of a formula F in L in interpretation $I = (S, \sigma)$. Let's use the notation for the value of a term, t_k in interpretation $I = (S, \sigma)$ is denoted

$$t_k^I = \begin{cases} \sigma(t_i), & t_i \text{ is a variable} \\ c_k^K, & t_k = c_k \text{ a constant symbol} \end{cases}$$

BASIS: If $F(t_1, \ldots, t_n)$ is an atomic formula corresponding to n-ary predicate F^S, then $F(t_1, \ldots, t_n)$ is true exactly when $(t_1^I, \ldots, t_n^I) \in F^S$. If $n = 0$, then F is a proposition whose truth value is given by S.

INDUCTION STEP: If F is constructed from atomic formulas F_1 and F_2 with known truth values in $I = (S, \sigma)$, using a connective or a quantifier, then the truth value of F in $I = (S, \sigma)$ is determined by the five logical connectives exactly as it is for propositional formulas. For the quantifiers:

$F = \forall x F_1$ is true in (S, σ) if and only if F_1 is true in $(S, \sigma|_v^x)$ for every v in D.

$F = \exists x F_1$ is true in (S, σ) if and only if F_1 is true in $(S, \sigma|_v^x)$ for some v in D.

If F is true in I, then I SATISFIES F. Otherwise, I FALSIFIES F.

Notes

[1] Think $(E(0) \wedge E(1) \wedge E(2) \wedge \cdots)$, clearly false.

[2] Think $(E(0) \vee E(1) \vee \cdots)$, clearly true.

[3] There is somebody who is everybody's sibling, false

[4] Everybody has a sibling. False, but a little more plausible.

[5] This has arity 1, since it only depends on what person is assigned to y.