ITCS 531: Logic 4 - First-order logic

Rob Egrot

Why first-order logic?

- Propositional logic is a significant advance on the ancient and medieval European concept of logic as the study of syllogisms.
- It provides a clear understanding of what it means for a set of propositions to imply another proposition.
- The concepts of tautology and contradiction show us how statements can be always true or false based on their form alone.
- But it's hard to say anything interesting in propositional logic.
- First-order logic addresses this limitation by adding the formal tools to create propositions as statements that can be meaningfully interpreted.

Relations

Definition 1 (Relation)

An n-ary relation between sets $X_1, \ldots X_n$ is a subset of $\prod_{i=1}^{n} X_i$. Given such a relation r, and an n-tuple $(x_1, \ldots, x_n) \in \prod_{i=1}^{n} X_i$, we say $r(x_1, \ldots, x_n)$ holds if and only if $(x_1, \ldots, x_n) \in r$.

Example 2

1. The order relation $\leq$ is a binary relation on $\mathbb{N}^2$.
2. If X is a set, and $Y \subseteq X$, then we can define a unary relation, r_Y, on X by $r_Y(x) \iff x \in Y$.
3. We can define a relation, p, on $\mathbb{N}^3$ by $p(x, y, z) \iff x^2 + y^2 = z^2$. This is a 3-ary (ternary) relation.
4. We can define a ternary relation, q, on $\mathbb{N} \times \mathbb{N} \times \mathbb{Q}$ by $q(x, y, z) \iff z = \frac{x}{y}$.

Functions

Definition 3 (Function)

An n-ary function is a well-defined map, f, from $\prod_{i=1}^{n} X_i$ to Y for some sets X_i ($i \in \{1, \dots n\}$) and Y. In this context, well-defined means that, for every $(x_1, \dots, x_n) \in \prod_{i=1}^{n} X_i$, the value of $f(x_1, \dots, x_n)$ exists and is unique.

- An n-ary function $f : \prod_{i=1}^{n} X_i \to Y$ is an $(n+1)$-ary relation on $\prod_{i=1}^{n} X_i \times Y$.
- An $(n+1)$-ary relation R can be an n-ary function, but only if it is well defined.
  - I.e. for all $(x_1, \dots, x_n)$ there is a unique y with $R(x_1, \dots, x_n, y)$.

Example 4

1. Every polynomial $a_0 + a_1 x + \ldots + a_n x^n$ defines a unary function from $\mathbb{N}$ to $\mathbb{N}$ (and from $\mathbb{R}$ to $\mathbb{R}$, or from $\mathbb{N}$ to $\mathbb{R}$ etc.)
2. Division can be thought of as a binary function d from $\mathbb{Q} \times (\mathbb{Q} \setminus \{0\})$ to $\mathbb{Q}$ by defining $d(x, y) = \frac{x}{y}$.

First-order languages

- A first-order language is a collection of symbols.
- These symbols have different roles in constructing the formulas that will represent the 'statements' in the language.
- First-order languages significantly depend on the choice of symbols, but they all have some common features.
- If we want to describe a system or object with first-order logic, we must first choose an appropriate language.
- There may be more than one language appropriate for the task.

Logical symbols

All first-order languages contain the following **logical symbols**:

1. An infinite set of variables enumerated by natural numbers,

$$V = \{x_0, x_1, \ldots\}.$$

   Note we often use other symbols e.g. x, y, z for variables, but we assume they are different names for things in V.

2. The equality symbol, $\approx$.
3. The set of logical connectives, $\{\neg, \vee, \wedge, \rightarrow\}$.
4. The set of quantifier symbols, $\{\forall, \exists\}$.
5. A set of brackets, $\{(,)\}$.

Non-logical symbols

In addition to the logical symbols, a first-order language may also contain some **non-logical symbols**:

1. A countable (possibly empty) set, $\mathcal{R}$, of *predicate* AKA *relation* symbols.
   - Every predicate symbol has an associated *arity*.

2. A countable (possibly empty) set, $\mathcal{F}$, of *function* symbols.
   - Every function symbol also has an associated *arity*.

3. A countable (possibly empty) set, $\mathcal{C}$, of *constant* symbols.
   - We can think of a constant as a 0-ary (nullary) function.

Example - arithmetic

- Suppose we want to study arithmetic in $\mathbb{N}$ with first-order logic.
- What non-logical symbols might we want?
- Probably functions $+$ and $\times$.
- Possibly constants 0 and 1.
- Maybe even something unusual like a unary predicate for saying if a number is prime.
- We are free to choose, but our choice may have consequences.

First-order formulas

- Formulas in a first-order language $\mathscr{L}$ are statements that are capable of being interpreted, in a sense to be made precise soon.
- Technically, an $\mathscr{L}$-formula is a special kind of string of symbols (logical and non-logical) from $\mathscr{L}$.
- Since we don't have basic propositions, the recursive definition of first-order formulas is more complex than in propositional logic.

Terms

Definition 5 (Term)

The set of *terms* of $\mathscr{L}$ is defined recursively.

- Every variable x is an $\mathscr{L}$-term.
- Every constant c is an $\mathscr{L}$-term.
- If f is an n-ary function symbol occurring in $\mathscr{L}$ and $t_1, \ldots, t_n$ are $\mathscr{L}$-terms then $f(t_1, \ldots, t_n)$ is also an $\mathscr{L}$-term.

- Unlike in propositional logic, the variables themselves are not propositions.
- It doesn't make sense for a variable in first-order logic to be true or false.
- The same applies to terms.

Atomic formulas

Definition 6 (Atomic formula)

The set of *atomic formulas* of $\mathscr{L}$ is defined as follows:

- If t_1 and t_2 are $\mathscr{L}$-terms, then $t_1 \approx t_2$ is an atomic $\mathscr{L}$-formula.
- If R is an n-ary relation of $\mathscr{L}$, and $t_1, \ldots, t_n$ are $\mathscr{L}$-terms, then $R(t_1, \ldots, t_n)$ is an atomic $\mathscr{L}$-formula.

- Atomic formulas are the basic propositions of first-order logic.
- They are the simplest true/false statements.
- Terms are like objects, and atomic formulas are simple statements about these objects.

Formulas

Definition 7 (Formula)

- Every atomic $\mathscr{L}$-formula is an $\mathscr{L}$-formula.
- If ϕ is an $\mathscr{L}$-formula, then $\neg\phi$ is an $\mathscr{L}$-formula.
- If ϕ and ψ are $\mathscr{L}$-formulas, then $(\phi \wedge \psi)$, $(\phi \vee \psi)$ and $(\phi \rightarrow \psi)$ are $\mathscr{L}$-formulas.
- If ϕ is an $\mathscr{L}$-formula and x is a variable symbol, then $\forall x\phi$ and $\exists x\phi$ are $\mathscr{L}$-formulas.

- Note that variables can be in the scope of multiple quantifiers.
- E.g. $\forall x \exists x R(x)$ is a formula.
- Here $\forall x$ is *null*, as it doesn't do anything.

Formulas - examples

As in propositional logic, we are sometimes loose with our use of brackets, adding them or removing them when the result makes the formulas easier for humans to read.

Example 8

Let $\mathscr{L}$ have signature $\mathcal{R} = \{R, S\}$, where R is unary and S is binary, $\mathcal{F} = \{f\}$, where f is ternary, and $\mathcal{C} = \{c, d\}$. Let x, y, z be variables.

1. $f(x, y, f(z, c, d)) \approx c$ is an atomic $\mathscr{L}$-formula.
2. $\exists z(R(f(x, z, d))) \vee S(f(x, y, x), d)$ is an $\mathscr{L}$-formula.
3. $f(x, y, z) \wedge c$ is not an $\mathscr{L}$-formula.

Definition 9 (Subformula)

If ϕ is an $\mathscr{L}$-formula, then a *subformula* of ϕ is a substring of ϕ that is also an $\mathscr{L}$-formula.

Interpreting $\mathscr{L}$-formulas

To give $\mathscr{L}$-formulas meanings we must interpret them in suitable structures.

Definition 10 ($\mathscr{L}$-structure)

Given a first-order signature, $\mathscr{L}$, an *$\mathscr{L}$-structure* is a set X, plus some additional information giving concrete meaning to the symbols in $\mathcal{R} \cup \mathcal{F} \cup \mathcal{C}$ as follows:

1. Every *n*-ary relation symbol from $\mathcal{R}$ is assigned to an *n*-ary relation on X^n.
2. Every *n*-ary function symbol from $\mathcal{F}$ is assigned to an *n*-ary function from X^n to X.
3. Every constant symbol from $\mathcal{C}$ is assigned to a specific element of X.

So an $\mathscr{L}$-structure is a pair (X, I), where X is the underlying set, and I is the function that interprets the non-logical symbols of $\mathscr{L}$ as relations, functions and constants over X.

Assignments

Definition 11 (Assignment)

An *assignment* of a first-order signature $\mathscr{L}$ to an $\mathscr{L}$-structure, $A = (X, I)$, is a function $v : V \to X$. In other words, an assignment associates every variable with an element of X.

- An $\mathscr{L}$-structure is just a set which we equip with relations, functions and constants corresponding to the symbols from $\mathscr{L}$.
- An assignment just gives a meaning to the variables of $\mathscr{L}$ as elements of the set.
- An $\mathscr{L}$-structure with an assignment turns $\mathscr{L}$-formulas into true/false statements.

Assignments - examples

Example 12

Let $\mathscr{L}$ have non-logical symbols $\{\leq, 0\}$, where $\leq$ is a binary relation, and 0 is a constant. We can take $\mathbb{N}$ as a $\mathscr{L}$-structure by giving these symbols their usual meanings.

1. Let ϕ be the formula $x \leq y$. Then this is true if our assignment v maps x to 1 and y to 5, for example, but false if v takes x to 465 and y to 7.
2. Let ψ be the formula $\forall x(0 \leq x)$. Then ϕ is true whatever v we choose.
3. Let χ be the formula $\exists x(x \leq y \wedge \neg(x \approx 0))$. Then χ will be true so long as $v(y) \neq 0$.

Extending assignments

An assignment v extends to $\mathscr{L}$-terms in a natural way.

Definition 13 (v^+)

Let $\mathbf{term}(\mathscr{L})$ be the set of terms of $\mathscr{L}$, and let v be an assignment for $\mathscr{L}$ to (X, I). Then define $v^+ : \mathbf{term}(\mathscr{L}) \to X$ recursively as follows:

- If x is a variable then $v^+(x) = v(x)$.
- If c is a constant then $v^+(c) = c_I$.
- If f is an n-ary function, and $t_1, \ldots, t_n$ are terms such that $v^+(t_i)$ has been defined for all $i \in \{1, \ldots, n\}$, then $v^+(f(t_1, \ldots, t_n)) = f_I(v^+(t_1), \ldots, v^+(t_n))$.

Models

- Let $\mathscr{L}$ be a first-order signature, let $A = (X, I)$ be a structure for $\mathscr{L}$, and let v be an assignment of $\mathscr{L}$ to A.
- Let ϕ be an $\mathscr{L}$-formula.
- We write $A, v \models \phi$ when A and v provide a **model** for ϕ.
- We define what this means recursively.
- Atomic formulas:
  - $A, v \models t_1 \approx t_2 \iff v^+(t_1) = v^+(t_2)$.
  - $A, v \models R(t_1, \ldots, t_n) \iff R_I(v^+(t_1), \ldots, v^+(t_n))$ holds.

Models - continued

Suppose ϕ and ψ are formulas such that whether A, u models ϕ and ψ has already been determined, for all assignments $u : V \to X$. Then:

- $A, v \models \neg\phi \iff A, v \not\models \phi$.
- $A, v \models \phi \vee \psi \iff A, v \models \phi$ or $A, v \models \psi$.
- $A, v \models \phi \wedge \psi \iff A, v \models \phi$ and $A, v \models \psi$.
- $A, v \models \phi \to \psi \iff A, v \models \neg\phi$ or $A, v \models \psi$.
- $A, v \models \forall x\phi \iff$ whenever u is an assignment of $\mathscr{L}$ to A that agrees with v on every variable except, possibly, x, we have $A, u \models \phi$.
- $A, v \models \exists x\phi \iff$ there is an assignment, u, of $\mathscr{L}$ to A that agrees with v on every variable except, possibly, x, and $A, u \models \phi$.

Free and bound variables

- If ϕ is an $\mathscr{L}$-formula, and x is a variable, then we say an occurrence of x is **free** in ϕ if there is no subformula of ϕ containing this occurrence of x that has the form $\forall x \phi'$ or $\exists x \phi'$.
- If there is a free occurrence of x in ϕ then we say that x is a **free variable** of ϕ.
- If an occurrence of x is not free in ϕ then we say it is **bound**, and that x occurs **bound** in ϕ.
- A bound occurrence of a variable is said to be in the **scope** of the corresponding quantifier.

Free and bound variables - examples

Example 14

Let $\mathscr{L}$ have signature $\mathcal{R} = \{R, S\}$, where R is unary and S is binary, $\mathcal{F} = \{f\}$, where f is ternary, and $\mathcal{C} = \{c, d\}$. Let x, y, z be variables.

1. $f(x, y, f(z, c, d)) \approx c$ has no bound variables.
2. z occurs only bound in $(\exists z(R(f(x, z, d))) \vee S(f(x, y, x), d)$, and x and y occur only free.
3. All variables in $\forall x(R(x) \vee S(x, c)) \wedge \exists x(R(f(x, x, x)))$ are bound.
4. In $\exists x R(x) \wedge S(x, y)$ the variable x occurs both free and bound. Note that x is still a free variable of this formula, even though it also occurs bound. The variable y occurs only free.

Sentences

Definition 15 (Sentence)

A sentence of $\mathscr{L}$ (an $\mathscr{L}$-sentence) is an $\mathscr{L}$-formula that contains no free variables.

- By exercise 4.4, if a sentence is true for some assignment into a model, then it is true for every assignment into the same model.
- Because bound occurrences of variables are not affected by the choice of v.
- So for sentences we can suppress v and just write, e.g. $A \models \phi$.