CS 245 - Logic and Computation **Fall 2016**

Lecture 8, 9, 10, 11, and 12: October 4 - 20, 2016

Lecturer: Kevin Lanctot *Notes By: Harsh Mistry*

8.1 Natural Deduction Basic Rules Continued

- ¬-Elimination Rule : $\sum, \varphi, \neg\varphi \vdash \perp$
  - $$\frac{\varphi \quad \neg\varphi}{\perp}$$
- ¬-Introduction Rule : If $\sum$, $\varphi \vdash \perp$, then $\sum \vdash \neg\varphi$
  - $$\frac{\begin{array}{c}\varphi\\ \vdots\\ \perp\end{array}}{\neg\varphi}$$
- ¬¬-Elimination Rule : If $\sum \vdash \neg\neg\varphi$, then $\sum \vdash \varphi$
  - $$\frac{\neg\neg\varphi}{\varphi}$$
- ⊥-Elimination Rule : If $\sum \vdash \perp$, then $\sum \vdash \varphi$
  - $$\frac{\perp}{\varphi}$$

8.2 Natural Deduction Derived rules

- Modus tollens
$$\frac{\varphi \to \alpha \quad \neg\alpha}{\neg\varphi}$$
- Proof by contradiction (Reductio ad absurdum)
$$\text{if } \sum, \neg\varphi \vdash \perp, \text{ then } \sum \vdash \varphi$$
- Law of Excluded Middle
$$\vdash \varphi \vee \neg\varphi$$
- Double-Negation Introduction
$$\text{if } \sum \vdash \varphi, \text{ then } \sum \neg\neg\varphi$$

8.3 Soundness and Completeness of Natural Deduction

8.3.1 Soundness

Soundness of natural deduction means that the conclusion of the proof is always a logical consequence of the premises. That is,

$$\text{if } \sum \vdash_{ND} \varphi, \text{ then } \sum \models \varphi$$

8.3.2 Completeness

Completeness of natural deduction means that all logical consequences in propositional logic are provable in Natural Deduction. That is,

$$\text{if } \sum \models \varphi, \text{ then } \sum \vdash_{ND} \varphi$$

8.4 First Order Predicate Logic

8.4.1 Ingredients for Predicate Logic

- Domains : The set of objects; also called the universe
  - A domain is a non-empty set. In principle, any non empty set can be a domain.
- Constants : objects with specific names
  - A constant symbol refers to an object in the domain
- Relations : properties of objects, alone or in combination
  - A predicate, or relation, represents a property that an individual, or collection of individuals, may or may not have
  - In English we might express a predicate as "__________ is a student"
  - In symbolic logic we write "S(x)" to mean "x has property S"
- Functions : Association of objects to others

8.4.2 Quantifiers

- "For All" is denoted by $\forall$, the universal quantifier symbol
- "There exists" is denoted by $\exists$, the existential quantifier symbol.

8.5 Syntax of Predicate Logic

There are 7 kinds of symbols

1. Constant Symbols : $c, d, c_1, c_2, \ldots$

2. Variables : $x, y, z, \ldots$
3. Function Symbols : $f, g, h, \ldots$
4. Predicate Symbols : $P, Q, \ldots$
5. Connectives : $\neg, \wedge, \vee, \rightarrow$
6. Quantifiers : $\forall$ and $\exists$
7. Punctuation : "(" , ")" , "." and ","

8.5.1 Terms

The set of terms is defined inductively as follows

1. Each constant symbol is a term, and each variable is a term. Such terms are called atomic terms
2. If $t_1, \ldots, t_n$ are terms and f is an n-ary function symbol, then $f(t_1, \ldots, t_n)$ is a term. If n =2, we may write $(t_1 f t_2)$ instead of $f(t_1, t_2)$
3. Nothing else is a term

8.5.2 Atomic Formulas

An atomic formula (or atom) is an expression of the form

$$P(t_1, \ldots, t_n)$$

where P is an n-ary relation symbol and each t_i is a term $1 \leq i \leq n$.

8.5.3 General Formulas

We define the set of well formed formulas of first-order logic inductively as follows

1. An atomic formula is a formula
2. If α is a formula, then $(\neg\alpha)$ is a formula
3. If α and β are formulas, and $\star$ is a binary connective symbol, then $\alpha \star \beta$ is a formula.
4. If α is a formula and x is a variable, then each of $(\forall x\alpha)$ and $(\exists x\alpha)$ is a formula
5. Nothing else is a formula

Note : Case 4 is referred to as the scope of the quantifier.

8.5.4 Parse Trees

Parse trees for FOL formulas are similar to parse trees for propositional formulas

- Quantifiers $\forall x$ and $\exists x$ form nodes in the same way as negation
- A predicate $P(t_1, \ldots, t_n)$ has a node labelled P with a sub-tree for each of the terms $t_1, \ldots, t_n$

8.6 Interpretations

Fix a set γ of constants, functions symbols, and relation symbols.

An interpretation I (for the set γ) consists of

- A non-empty set $dom(I)$, called the domain (or universe) of I
- For each constant symbol c, a member c^I of $dom(I)$
- For each function symbol $f^{(i)}$, an i-ary function f^I
- For each relation symbol $R^{(i)}$, an i-ary relation R^I

8.6.1 Values of Variables-Free Terms

for terms and formulas that contain no variables or quantifiers, an interpretation suffices to specify their meaning. The meaning arises in the obvious(?) fashion from the syntax of the term or formula.

Definition 8.1 *Fix an interpretation I. For each term t containing no variables, the value of t under interpretation I, denoted t^I, is as follows*

- *If t is s constant c, the value t^I is c^I*
- *If t is $f(t_1, \ldots, t_n)$, the value t^I is $f^I(t_1^I, \ldots, t_n^I)$*

8.6.2 Formulas with Variable-Free Terms

Formulas get values in much the same fashion as terms, except that values of formulas lie `{F,T}`

Definition 8.2 *Fix an interpretation I. For each term α containing no variables, the value of α under interpretation I, denoted α^I, is as follows*

- *If α is $R(t_1, \ldots, t_n)$ then*
$$\alpha^I = \begin{cases} T & \text{if } (t_q^I, \ldots, t_n^I) \in R^I \\ F & \text{otherwise} \end{cases}$$
- *If α is $\neg\beta$ or $\beta \star \gamma$, then α^I is determined by β^I and Y^I in the same way as for propositional logic.*

8.6.3 Overlooked Points

1. There is NO default meaning for relations, function, or constant symbols
2. Functions must be defined at every point in the domain

8.7 Variables

An occurrence of a variable in a formula is bound if it lies in the scope of some quantifier of the same variable; otherwise it is free. In other words, a quantifier binds its variable within its scope.

Formally, a variable occurs free in a formula α if and only if it is a member of FV(α). Free variables in α are defined as follows

1. If α is $P(t_1, \ldots, t_k)$, then $\text{FV}(\alpha) = \{x \mid x \text{ appears in some } t_i\}$
2. If α is $\neg\beta$, then $\text{FV}(\alpha) = \text{FV}(\beta)$
3. if α is $\beta \star \gamma$, then $\text{FV}(\alpha) = FV(\beta) \cup FV(\gamma)$
4. If α is Q x B (for Q $\in \{\forall, \exists\}$) then $\text{FV}(\alpha) = FV(\beta) - \{x\}$

In summary: a formula has the same free variables as its parts, except that a quantified variable becomes bound. A formula with no free variables is called a closed formula, or a sentence.

8.7.1 Substitutions

The notation $\alpha\left[t/x\right]$, for a variable x, a term t, and formula α, denotes the formula obtained from α by replacing each free occurrence of x with t.

8.8 Semantics of Predicate Logic

From Kevin's Slides

FOL includes more ingredients (i.e., predicates, functions, variables, terms, constants, etc.) and, hence, the semantics for FOL must account for all of the ingredients. We already saw the concept of an interpretation, which specifies the domain and the identities of the constants, relations and functions. Formulas that include variables, and perhaps quantifiers, require additional information, known as an environment (or assignment).

Environment : A first order environment is a function that assigns values in the domain to each variable.

8.8.1 Meaning of terms

The combination of an interpretation and an environment supplies a value for every term.

Definition 8.3 *Fix an interpretation I an environment E. for each term t, the value of t under I and E, denoted $t^{(I,E)}$, is as follows*

- *If t is a constant c, the value is c^I*
- *if t is a variable x, the value is x^E*
- *if t is $f(t_1, \ldots, t_n)$, the value is $F^I(t_1^{(I,E)}, \ldots, t_n^{(I,E)})$*

8.8.2 Quantified Formulas

For any environment E and domain element d, the environment E with x re-assigned to d, denoted E[x $\rightarrow$ d], is given by

$$E\left[x \rightarrow d\right](y) \begin{cases} d & \text{if y is x} \\ E(y) & \text{if y is not x} \end{cases}$$

8.8.2.1 Values of Quantified Formulas

- $$(\forall x\alpha)^{(I,E)} = \begin{cases} T & \text{if } \alpha^{(I,E\left[x \rightarrow d\right])} = T \text{ for every d } \in dom(I) \\ F & otherwise \end{cases}$$
- $$(\exists x\alpha)^{(I,E)} = \begin{cases} T & \text{if } \alpha^{(I,E\left[x \rightarrow d\right])} = T \text{ for every d } \in dom(I) \\ F & otherwise \end{cases}$$

8.8.3 Satisfaction

Note : Refer to Course Resources for full condition chart

In general, If $I \models_E \alpha$ for every E, then I satisfies $alpha$, denoted $I \models \alpha$

8.8.4 Validity and Satisfiability

Validity and Satisfiability of FOL formulas have definitions analogous to tautology for propositional logic

Definition 8.4 *A formula α is*

- *Valid : if every interpretation and environment satisfy α; that is if $I \models_E \alpha$ for every I and E*
- *Satisfiable : if some interpretation and environment satisfy α; that is, if $I \models_E \alpha$ for some I and E*
- *Unsatisfiable : is no interpretation and environment satisfy α; that is if $I \not\models_E \alpha$ for every I and E*

Note : *The term tautology is not used in predicate logic.*

8.8.5 Relevance Lemma

Lemma 8.5 *Let α be a first-order formula, I be an interpretation, and E_2 and E_2 be two environments such that*

$$E_1(x) = E_2(x) \textit{ for every x that occurs free in } \alpha$$

Then

$$I \models_{E_1} \alpha \textit{ if and only if } I \models_{E_2} \alpha$$

8.8.6 Logical Consequence

Suppose $\sum$ is a set of formulas and α is a formula. We say that α is a *logical consequence* of *sum*, written as $\sum \models \alpha$, iff for any interpretation I and environment E, we have $I \models_E \sum$ implies $I \models_E \alpha$

Note : $\models \alpha$ means α is valid.