First Order (Predicate) Logic: Syntax and Natural Deduction [1]

A Reminder of Our Plot

- I wish to provide some historical and intellectual context to the formal tools that logicians developed to study the semantics of artificial languages.

- For this reason, I'm beginning with a *purely syntactic* presentation of two key logical systems: Propositional Logic (PL) and First Order (Predicate) Logic (FOL).

- In our last notes, we covered PL. Now, we'll get a (syntactic) introduction to FOL.

1. A Review of First Order (Predicate) Logic (FOL): Syntax and Informal Semantics [2]

The system of First Order Logic (FOL) is intended to capture the inferences that depend upon:
(i) the meaning of the so-called 'sentential connectives': *and*, *or*, *if…then*, and *not*
(ii) the meaning of the quantifiers 'every' and 'some'

(1) **The Vocabulary of Symbols**

a. The Logical Constants:

(i)	~	Negation	'It is not the case that…'
(ii)	&	Conjunction	'and'
(iii)	v	Disjunction	'or' (inclusive)
(iv)	→	(Material) Implication	'if…then'
(v)	∀	Universal Quantifier	'for all…'
(vi)	∃	Existential Quantifier	'there is an…'

b. Syntactic Symbols: (,)

a. The Non-Logical Constants (a.k.a 'The Logical Variables')

(i) An infinite set of *predicate letters*: {P, Q, R, B, … P_1, P_2, P_3, P_4, …}
- Each predicate letter has an associated 'arity' (unary, binary, *etc.*)
- Each predicate letter 'stands for' a property or a relation

(ii) An infinite set of *individual constants*: {a, b, c, …, a_1, a_2, a_3 , … }
- The individual constants 'stand for' proper names (Bill, John, etc.)

(iii) An infinite set of *variables*: {x, y, z, …, x_1, x_2, x_3, … }

---

[1] These notes are based upon material in the following required reading: Gamut (1991), Chapter 3 pp. 65-83, Chapter 4 pp. 128-148; Partee *et al.* (1993), Chapter 7 pp. 135-140.
[2] My discussion here will assume prior familiarity with the overall system of First Order Logic. Students are referred to Partee *et al.* (1993), Chapter 7 for crucial background.

(2) **The Definition of a 'Well-Formed Formula' (WFF) in FOL**

The set of 'well-formed formulae' of PL, WFF, is the smallest set such that:

a. If φ is an n-ary predicate letter and each of $\alpha_1, \ldots, \alpha_n$ is either an individual constant or a variable, then $\varphi\alpha_1\ldots\alpha_n \in$ WFF

b. If $\varphi, \psi \in$ WFF, then
   1. $\sim\varphi \in$ WFF
   2. $(\varphi \,\&\, \psi) \in$ WFF
   3. $(\varphi \vee \psi) \in$ WFF
   4. $(\varphi \rightarrow \psi) \in$ WFF

c. If $\varphi \in$ WFF and υ is a variable, then
   1. $\forall\upsilon\varphi \in$ WFF
   2. $\exists\upsilon\varphi \in$ WFF

<u>Notes:</u>

- The clause in (2a) creates the *atomic formulae* of FOL. The clause in (2c) creates the *universal formulae* and *existential formulae*.

- The set WFF includes formulae with 'free variables' and 'vacuous quantification' (defined properly later)

Hxb	(free variables)
∃y∀xHxb	(vacuous quantification)
∃yHxb	(free variables and vacuous quantification)

(3) **Using FOL To Encode Sentences of English**

We can use the syntactic rules in (2) and the informal semantics in (1) to write FOL formulae that 'encode' certain statements of English:

a. *Sentence:* 'If Bill or John is leaving, then Mary and Sue aren't happy.'
   Encoding: $((Lb \vee Lj) \rightarrow (\sim Hm \,\&\, \sim Hs))$

b. *Sentence:* 'Every cat gave a book to Bill.'
   Encoding: $\forall x(Cx \rightarrow \exists y(By \,\&\, Gxyb))$

In setting up such encodings, it is critical to supply a 'key', indicating what the predicate letters and individual constants 'stand for':

Key		
	Lx: x is leaving	b: Bill
	Hx: x is happy	j: John
	Cx: x is a cat	m: Mary
	Bx: x is a book	s: Sue
	Gxyz: x gave y to z	

<u>An Important Note:</u>

- In the key above, formulae of the form 'Gxyz' are interpreted so that x is the 'subject' of *gave*, while 'y' is the direct object, and 'z' is the indirect object.
- *Nothing forces this however.* We could just have easily had the following in our key:
  Gxyz: z gave y to x
- **Such 'right-to-left' readings of atomic formulae will be useful to us later, when we're mechanically translating sentences of English into sentences of FOL…**

(4) **Key Definition: Scope**
If ∀υψ is a subformula of φ, then ψ is the *scope* of (this occurrence of) '∀υ' in φ
If ∃υψ is a subformula of φ, then ψ is the *scope* of (this occurrence of) '∃υ' in φ

<u>Illustration:</u> In the formula '∃x(Qx & ∀y(Py → ∃zSxyz))'

(i) The scope of ∃x = (Qx & ∀y(Py → ∃zSxyz))
(ii) The scope of ∀y = (Py → ∃zSxyz)
(iii) The scope of ∃z = Sxyz

(5) **Key Definition: Free and Bound Variables**

a An occurrence of the variable υ in the formula φ is *free in* φ if (i) and (ii) hold:
(i) that occurrence of υ does not occur directly to the right of either ∃ or ∀
(ii) that occurrence of υ is not in the scope of any occurrence of ∃υ or ∀υ in φ

b. An occurrence of the variable υ is *bound by ∀υ (∃υ) in* φ if (i) and (ii) hold:
(i) ∀υψ (∃υψ) is a subformula of φ
(ii) That occurrence of υ is free in ψ

<u>Illustration:</u> In the formula '∀x(Px & ∃xBx)'

(i) The occurrence of 'x' in 'Px' is free in '(Px & ∃xBx)'
(ii) The occurrence of 'x' in 'Bx' is free in 'Bx'
(iii) The occurrence of 'x' in 'Bx' is *not* free in '(Px & ∃xBx)'

(iv) The first occurrence of 'x' in '(Px & ∃xBx)' is bound by '∀x'
(v) The occurrence of 'x' in 'Bx' is bound by '∃x'
(vi) The occurrence of 'x' in 'Bx' is *not* bound by '∀x'

(6) **Key Definition: Sentence**
φ is a *sentence* of FOL if (i) φ ∈ WFF, and (ii) there are no free variables in φ

2. A Review of First Order Logic (FOL): Natural Deduction

(7) **Major Goal of This Section**
Let's provide a *purely syntactic* characterization of 'valid inference' in the FOL notation.

- This syntactic characterization will be embodied in a *proof system* (natural deduction)

- We're going to lay out some rules – stated entirely in *syntactic terms* – for deriving formulae in FOL from other formulae.
  - As we'll see, these syntactic rules intuitively capture certain key aspects of the everyday meaning of the English logical words 'every' and 'some'

(8) **Features Inherited from PL Natural Deduction**
All the following will directly imported from our system of natural deduction for PL:

a. Definition of 'derivation'
b. Turnstyle notation '⊢'
c. The rules I&, E&, Repetition, I→, E→, I∨, E∨, I~, E~, EFSQ, ~~

(9) **Special Feature of FOL Natural Deduction**
φ can be an assumption in an FOL derivation *iff* φ is a sentence.

Our natural deduction system for FOL adds four new rules.

(10) **Special Notation for Statement of Deduction Rules for FOL**

If φ ∈ WFF, α is an individual constant, and υ is a variable, then '[α/υ]φ' is the formula just like φ, except that every *free* instance of υ is replaced with an instance of α:

[b/x](Pcx & Dabx) = (Pcb & Dabb)

(11) **The Rule of '∃-Introduction' (I∃)**

1.	…	
…	…	
n	[α/υ]φ	
…	…	
m	∃υφ	I∃ n

Intuitive Motivation:
If we can (in English) assert for a particular thing α that 'φ' is true of α, then we can assert that there is something that 'φ' is true of.

(12) **The Rule of '∀-Elimination' (E∀)**

1.	…	
…	…	
n	$\forall \upsilon \varphi$	
…	…	
m	$[\alpha/\upsilon]\varphi$	E∀ n

<u>Intuitive Motivation</u>
If we can (in English) assert that 'φ' is true of everything, then for any particular thing α, we can assert that 'φ' is true of α,

<u>Illustration of I∃ and E∀:</u> $\forall x Px \vdash \exists x Px$

1.	∀xPx	Assumption
2.	Pb	E∀ 1
3.	∃xPx	I∃ 3

(13) **The Rule of '∀-Introduction' (I∀)**

<u>Intuitive Motivation:</u>
If we can show that 'φ' is true of an arbitrary entity α ('arbitrary' = we've not assumed anything about α whatsoever), then we can assert that 'φ' is true of *everything*.

<u>Key Definition:</u>
If '$[\alpha/\upsilon]\varphi$' appears in a derivation at line n, then α *is arbitrary at line n* if (i) and (ii) hold
- (i) α does not appear in any (non-dropped) assumptions in the derivation
- (ii) α does not appear in φ

<u>The Rule:</u>
The following is an acceptable derivation, as long as α is arbitrary at line n.

1.	…	
…	…	
n	$[\alpha/\upsilon]\varphi$	
…	…	
m	$\forall \upsilon \varphi$	I∀ n

<u>Illustration:</u> $\forall x \forall y Pxy \vdash \forall x Pxx$

1.	∀x∀yPxy	Assumption
2.	∀yPay	E∀ 1
3.	Paa	E∀ 2
4.	∀xPxx	I∀ 3

- <u>Note:</u> 'Paa' = '[a/x]Pxx', and a is arbitrary (in '[a/x]Pxx') in at line 3.

(14) **The Rule of '∃-Elimination' (E∃)**

<u>Intuitive Motivation:</u>
If we can assert (i) that 'φ' is true of something, *and* (ii) that if 'φ' is true of an arbitrary entity α, then ψ must be true ('arbitrary' = we've not assumed anything about α whatsoever), then we can assert that 'ψ' is true.

<u>Key Definition:</u>
If '$[\alpha/\upsilon]\varphi \rightarrow \psi$' appears in a derivation at line n, then α *is arbitrary at line n* if (i)-(iii):

(i) α does not appear in any (non-dropped) assumptions in the derivation
(ii) α does not appear in φ
(iii) α does not appear ψ

<u>The Rule:</u>
The following is an acceptable derivation, as long as α is arbitrary at line n.

1.	…	
…	…	
n_1	$\exists\upsilon\varphi$	
…	…	
n_2	$[\alpha/\upsilon]\varphi \rightarrow \psi$	
…	…	
m	ψ	E∃ n_1, n_2

<u>Illustration:</u> $\exists x \forall y Pxy \vdash \forall y \exists x Pxy$

1.	$\exists x \forall y Pxy$	Assumption
2.	| $\forall y Pby$	Assumption
3.	| Pba	E∀ 2
4.	| $\exists x Pxa$	I∃ 3
5.	$\forall y Pby \rightarrow \exists x Pxa$	I→
6.	$\exists x Pxa$	E∃ 1,5
7.	$\forall y \exists x Pxy$	I∀ 6

- <u>Note:</u>
'$\forall y Pby$' = '$[b/x]\forall y Pxy$', and b is arbitrary (in '$[b/x]\forall y Pxy \rightarrow \exists x Pxa$') at line 5

- <u>Note:</u> '$\exists x Pxa$' = '$[a/y]\exists x Pxy$', and a is arbitrary (in '$[a/y]\exists x Pxy$') at line 6.

<u>Side Comment:</u>
- The rules I∀ and E∃ add significantly to the complexity of our natural deduction system.
- However, they are crucial for system to capture *all* the valid inferences in FOL
- They also greatly complicate the proof that our natural deduction system for FOL is 'complete'

3. The Power of Our Natural Deduction System for FOL

Although we our system for FOL has just 15 (really, 14) relatively simple rules, it can capture a great many intuitively valid inferences!

(15) **Derivation of Quantifier Negation, Part 1** $\sim\exists x Px \vdash \forall x \sim Px$

1.	~∃xPx		Assumption
2.	│	Pa	Assumption
3.	│	∃xPx	I∃ 2
4.	│	⊥	E~ 1, 3
5.	~Pa		I~
6.	∀x~Px		I∀ 6

- Note: '~Pa' = '[a/x]~Px' and a is arbitrary (in '[a/x]~Px') at line 5

(16) **Derivation of Quantifier Negation, Part 2** $\forall x \sim Px \vdash \sim\exists x Px$

1.	∀x~Px			Assumption
2.	│	∃xPx		Assumption
3.	│	│	Pa	Assumption
4.	│	│	~Pa	E∀ 1
5.	│		⊥	E~ 3,4
6.	│	(Pa → ⊥)		I→
7.	│	⊥		E∃ 2, 6
8.	~∃xPx			I~

- Note: '(Pa →⊥)' = '[a/x](Px→⊥)' and a is arbitrary (in '[a/x](Px→⊥)' at line 6

(17) **The Big Question**
Does our system offer a *perfect* syntactic characterization of 'validity' for FOL?
- Does every derivation correspond to a valid inference?
- Does every valid inference in FOL correspond to a derivation?

How would we even show this?...