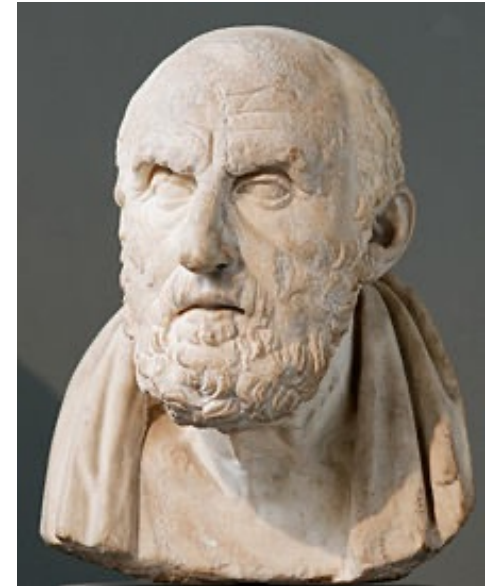


Propositional and First-Order Logic

Mark Hasegawa-Johnson

CC0 Public Domain

March 2024



By Unknown artist - Marie-Lan
Nguyen (2011), Public Domain,
[https://commons.wikimedia.org/w
/index.php?curid=1447238](https://commons.wikimedia.org/w/index.php?curid=1447238)

Outline

- Propositional Logic
- First-Order Logic
- Quantification

Propositional Logic

- “Propositions” are statements that can be either True or False
 - P=“an iguana is an animal with scales”
 - Q=“an iguana is an animal that breathes air”
 - R=“an iguana is a reptile”
- Propositional logic studies the relationships among propositions.

Symbolic Logic Functions

- Unary functions (map one proposition to another)
 - \neg (not): $\{F, T\} \rightarrow \{T, F\}$
- Binary functions (map two propositions to one)
 - \wedge (and): $\{(F, F), (F, T), (T, F), (T, T)\} \rightarrow \{F, F, F, T\}$
 - \vee (or): $\{(F, F), (F, T), (T, F), (T, T)\} \rightarrow \{F, T, T, T\}$
 - \implies (implies): $\{(F, F), (F, T), (T, F), (T, T)\} \rightarrow \{T, T, F, T\}$
 - \iff (equivalent): $\{(F, F), (F, T), (T, F), (T, T)\} \rightarrow \{T, F, F, T\}$

Using these symbols, you can combine several propositions to form one proposition. For example,

Propositional Logic

- Propositional logic is the study of how given propositions can be combined to prove new propositions.
- For example, consider the proposition $P \wedge Q \Rightarrow R$, “if an iguana has scales and breathes air, then it is a reptile.” This proposition is only false if P and Q are both true, but R is false:

P	Q	R	$P \wedge Q \Rightarrow R$
F	F	F	T
F	F	T	T
F	T	F	T
F	T	T	T
T	F	F	T
T	F	T	T
T	T	F	F
T	T	T	T

Outline

- Propositional Logic
- First-Order Logic
- Quantification

First Order Logic

- Propositional logic says that propositions can be constructed from other propositions
- First-order logic says propositions can also be constructed by applying predicates to constants

Predicates, Constants, Variables, Propositions, and Rules

- A **predicate** is like a function, that can be applied to some **variables**.
 - $BreathesAir(x)$ is true if and only if x breathes air.
- A **constant** is a particular object in the real world, which can be the value of the argument of a function:
 - $reptiles$ is a constant
- A **proposition** is a predicate applied to a constant
 - $BreathesAir(reptiles)$ is true if and only if reptiles breathes air.
- A **rule** is an implication or equivalence that's true for all values of its variable
 - $BreathesAir(x) \wedge Scales(x) \Rightarrow Reptile(x)$: everything that breathes air and has scales is a reptile.

Theorem Proving

An automatic theorem-prover uses a database of known facts and known rules to prove a theorem. For example, suppose we know that:

- Iguanas have scales: $Scales(iguanas)$
- Iguanas breathe air: $BreathesAir(iguanas)$
- Anything that breathes air and has scales is a reptile:
 $BreathesAir(x) \wedge Scales(x) \Rightarrow Reptile(x)$

And suppose we want to prove that:

- Iguanas are reptiles: $Reptile(iguanas)$

Theorem Proving by Forward-Chaining

- Forward-chaining is the process of applying rules to facts in order to prove more facts.

- For example, let's start by combining these two facts:

$$\textit{BreathesAir}(\textit{iguanas}) \wedge \textit{Scales}(\textit{iguanas})$$

- Now let's apply this rule:

$$\textit{BreathesAir}(x) \wedge \textit{Scales}(x) \Rightarrow \textit{Reptile}(x)$$

- The result: we have proven that:

$$\textit{Reptile}(\textit{iguanas})$$

Theorem-Proving by Forward-Chaining

Notice that, when we're forward-chaining, each step of the process just expands the set of available facts. If we start with the following database of facts:

$$\textit{BreathesAir}(\textit{iguanas}) \wedge \textit{Scales}(\textit{iguanas})$$

... and if we apply the rule $\textit{BreathesAir}(x) \wedge \textit{Scales}(x) \Rightarrow \textit{Reptile}(x)$, then the database can only get larger. It becomes this:

$$\textit{BreathesAir}(\textit{iguanas}) \wedge \textit{Scales}(\textit{iguanas}) \wedge \textit{Reptile}(\textit{iguanas})$$

Forward-chaining just keeps going, until the fact we want is part of the database, or until we can't prove any more facts.

Outline

- Propositional Logic
- First-Order Logic
- Quantification

Quantification

- It is sometimes useful to express compound propositions that are true for some values of their variables, but not all.
- To do this, we introduce two new symbols, called quantifiers:
- \exists (there exists)
 - Suppose P is the proposition $P = \exists x: F(x)$
 - Then $P = T$ if and only if, for at least one value of the variable x , $F(x) = T$
- \forall (for all)
 - Suppose P is the proposition $P = \forall x: F(x)$
 - Then $P = T$ if and only if, for all values of the variable x , $F(x) = T$
- “For all” negates “There exists,” and vice versa
 - $(\forall x: F(x)) \Leftrightarrow \neg(\exists x: \neg F(x))$
 - $\neg(\forall x: \neg F(x)) \Leftrightarrow (\exists x: F(x))$

Example: Colonel West

English	First-Order Logic Notation
It is a crime for Americans to sell weapons to hostile nations.	$\forall x: \exists y, z: \textit{American}(x) \wedge \textit{Weapon}(y) \wedge \textit{Sells}(x, y, z) \wedge \textit{Hostile}(z) \Rightarrow \textit{Criminal}(x)$
Colonel West sold missiles to Ganymede.	$\exists x: \textit{Sells}(\textit{west}, x, \textit{ganymede}) \wedge \textit{Missile}(x)$
Colonel West is American.	$\textit{American}(\textit{west})$
Ganymede is an enemy of America.	$\textit{Enemy}(\textit{ganymede}, \textit{america})$
Missiles are weapons.	$\forall x: \textit{Missile}(x) \Rightarrow \textit{Weapon}(x)$
An enemy of America is a hostile nation.	$\forall x: \textit{Enemy}(x, \textit{america}) \Rightarrow \textit{Hostile}(x)$

Automatic Theorem Proving

First-Order Logic Notation

$$\begin{aligned} & American(x) \wedge Weapon(y) \wedge \\ & Sells(x, y, z) \wedge Hostile(z) \\ & \Rightarrow Criminal(x) \end{aligned}$$
$$\begin{aligned} & \exists x, Missile(x) \\ & \wedge Sells(west, x, ganymede) \end{aligned}$$
$$American(west)$$
$$Enemy(ganymede, america)$$
$$Missile(x) \Rightarrow Weapon(x)$$
$$\begin{aligned} & Enemy(x, america) \\ & \Rightarrow Hostile(x) \end{aligned}$$

Can we prove the theorem:

$Criminal(west)?$

Actions that a Theorem Prover can Take

- **Universal Instantiation:**

- given the sentence $\forall x, Function(x)$,
- for any known constant C ,
- it is possible to generate the sentence $Function(C)$.

- **Existential Instantiation:**

- given the proposition $\exists x, Function(x)$,
- if no known constant A is known to satisfy $Function(A)$, then
- it is possible to define a new, otherwise unspecified constant B , and
- to generate the sentence $Function(B)$.

- **Generalized Modus Ponens:**

- Given the sentence $p_1(x_1) \wedge p_2(x_2) \wedge \dots \wedge p_n(x_n) \implies q(x_1, \dots, x_n)$, and
- given the sentences $p_1(C_1), \dots, p_n(C_n)$ for any constants C_1, \dots, C_n ,
- it is possible to generate the sentence $q(C_1, \dots, C_n)$

Automatic Theorem Proving Example

- **Existential Instantiation:**

- Input: $\exists x, \text{Missile}(x) \wedge \text{Sells}(\text{West}, x, \text{Ganymede})$
- Output: $\text{Missile}(M) \wedge \text{Sells}(\text{West}, M, \text{Ganymede})$

- **Generalized Modus Ponens:**

- Input: $\text{Missile}(M) \text{ and } \text{Missile}(x) \Rightarrow \text{Weapon}(x)$
- Output: $\text{Weapon}(M)$

- **Generalized Modus Ponens:**

- Input: $\text{Enemy}(\text{Ganymede}, \text{America}) \text{ and } \text{Enemy}(x, \text{America}) \Rightarrow \text{Hostile}(x)$
- Output: $\text{Hostile}(\text{Ganymede})$

- **Generalized Modus Ponens:**

- $\text{American}(x) \wedge \text{Weapon}(y) \wedge \text{Sells}(x, y, z) \wedge \text{Hostile}(z) \Rightarrow \text{Criminal}(x)$
- Input: $(x) \text{ and } \text{American}(\text{West}), \text{Weapon}(M), \text{Sells}(\text{West}, M, \text{Ganymede}), \text{Hostile}(\text{Ganymede})$
- Output: $\text{Criminal}(\text{West})$

Quiz

Try the quiz:

https://us.prairielearn.com/pl/course_instance/147925/assessment/2410574

Outline

- Propositional Logic
 - \neg (not) , \wedge (and) , \vee (or), \Rightarrow (implies), \Leftrightarrow (equivalent)
- First-Order Logic
 - A proposition is a predicate applied to a constant
 - A rule is an implication or equivalence that's true for all values of its variables
- Quantification
 - $\exists x: F(x)$ means that, for at least one value of the variable x , $F(x) = T$
 - $\forall x: F(x)$ means that, for all values of the variable x , $F(x) = T$