# Boolean, Propositional, Predicate Logics

Daniel F. Simola

18 May 2005

## 1 Boolean logic

Basic expressions:

1. $x \wedge \neg x = 0$
2. $x \vee \neg x = 1$
3. $x \wedge 0 = 0$
4. $x \vee 1 = 1$
5. $\neg 1 = 0$
6. $\neg 0 = 1$
7. $\neg\neg x = x$

De Morgan's laws:

1. $\neg(a \vee b) = (\neg a) \wedge (\neg b)$
2. $\neg(a \wedge b) = (\neg a) \vee (\neg b)$

Distributive law: $(a \wedge b) \vee c = (a \vee c) \wedge (b \vee c)$
All logic can be reduced two two operators: $\vee$ and $\neg$. eg, $(a \wedge b) = \neg((\neg a) \vee (\neg b))$.

## 2 Logical Calculus

A calculus is a formal system, where deductions are made based on some atomic formulae. The calculus is composed of a set of statements (axioms) about a universe, defined by well-formed formulae (wffs). A wffs is an atomic formula built using variables and operators.

**Propositional Logic:** uses propositional variables as formulae

**Predicate Logic:** uses propositional functions as formulae

### 2.1 Propositional logic

The language consists of the following elements:

1. Variables (eg $A, B, C$): these are atomic formulae
2. Operators (connectives): NOT $\neg$, AND $\wedge$, OR $\vee$, IMPLIES $\implies$, and IFF $\iff$ (implies and is implied by)
3. Parentheses '(' and ')'

The set of wffs is recursively defined by a grammar of the following rules:

1. Variables are wffs.

2. Inductive clause I: If $A$ is wff, then $\neg A$ is wff.

3. Inductive clause II: If $A, B$ are wffs, then $(A \wedge B), (A \vee B), (A \implies B),$ and $(A \iff B)$ are wffs.

4. Closure clause: Nothing else is wff.

Applications of these wffs generate subsequent wffs using a set of inference rules. eg $(\neg A \vee B)$ is wff. Two of the more interesting rules are

- Modus ponens: Given $A$ and $(A \implies B)$, we infer $B$
- Reductio ad absurdum: Given $A$, if we can derive $A$ and $\neg A$, we infer $\neg A$

#### 2.1.1 Truth table

| $a$ | $b$ | $(\neg a)$ | $(\neg b)$ | $(a \vee b)$ | $(a \wedge b)$ | $(a \implies b)$ | $(b \implies a)$ | $(a \iff b)$ |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |

Watch out for the IMPLIES and IFF rules. eg $(a \implies b) = 1$ whether or not $a = 1$. Interpret this as "b is able to be true independent of a, but we do have a rule that says if a is true then b must be true."

### 2.2 Predicate logic

Also known as first-order logic (FOL). This logic adds quantified statements. Cute fact: FOL is powerful enough to describe set theory and thus virtually all of mathematics.

The language is composed of:

1. Variables, constants, and function variables

2. Operators (connectives): NOT $\neg$, AND $\wedge$, OR $\vee$, IMPLIES $\implies$, and IFF $\iff$ (implies and is implied by)

3. Quantifiers: universal quantifier $\forall$, existential quantifier $\exists$

4. Parentheses '(' and ')'

The set of wffs is recursively defined by a grammar of the following rules:

1. Variables are wffs. (can be generalized to a statement about predicates)

2. Inductive clause I: If $A$ is wff, then $\neg A$ is wff.

3. Inductive clause II: If $A, B$ are wffs, then $(A \wedge B), (A \vee B), (A \implies B),$ and $(A \iff B)$ are wffs.

4. Inductive clause III: If $A$ is wff containing a 'free' instance of variable $x$, then $\forall x, A$ and $\exists x, A$ are wffs. (such instances of $x$ are then 'bound'. You are able to rename bound variables since every instance belongs to the same scope.)

5. Closure clause: Nothing else is wff.

Only one inference rule is necessary: modus ponens. (See above.)