

# Logic

So far we have described truth conditions through translation to English, trying to be precise.

**Today:** Propositional logic, connectives, predicate logic with variables

## 1 Propositional logic<sup>1</sup>

Propositional logic gives us a way to precisely describe *relationships of meaning* between individual propositions.

- (1) a. Cheryl ate the noodles.  
b. Cheryl ate the salad and the noodles.
- (2) a. If tomorrow is a public holiday, there will be no class tomorrow.  
b. Tomorrow is a public holiday.  
c. There will be no class tomorrow.

Propositional logic can be thought of as its own toy language, with a toy syntax and a toy semantics. We will call “grammatical sentences” in propositional logic *well-formed formulas (wff)*.

### (3) The syntax of propositional logic:

- a. Atomic formulas:  $p, q, r, \dots$ <sup>2</sup> are all wffs.

The atomic formulas function as the “lexicon” for this “language.”

- b. Negation: If  $\varphi$  is a wff,  $\neg\varphi$  is also a wff.

- c. Binary connectives: if  $\varphi$  and  $\psi$  are wffs, then so are:

- i.  $[\varphi \wedge \psi]$
- ii.  $[\varphi \vee \psi]$
- iii.  $[\varphi \rightarrow \psi]$
- iv.  $[\varphi \leftrightarrow \psi]$

- d. Nothing else is a wff.

I’ll be sloppy and drop the [ ] except where they’re necessary to make the “constituency” clear.

---

<sup>1</sup>Much of today’s handout is inspired by materials by Masha Esipova & Lucas Champollion and Liz Coppock.

<sup>2</sup>*IFS* uses capital letters  $P, Q, R$ , etc. But mathematics is invariant under change of notation.

**Exercise:** \* the formulas in (4) which are not wffs:

- (4) a.  $p \wedge p$                       d.  $\neg p \vee q$                       g.  $q \neg p$   
 b.  $\neg \neg p$                               e.  $p q \wedge$                       h.  $q \rightarrow q$   
 c.  $\neg \wedge p$                               f.  $p \neg$                       i.  $q \leftarrow p$

We then need a way to interpret wffs:

- Each atomic proposition is either true or false in a given model: e.g.  $\llbracket p \rrbracket^M = 1$  or  $\llbracket q \rrbracket^M = 0$
- We then define a semantics for each of the operators in (3b–c) above, which we can express using *truth tables*.

Negation  $\neg$ : “not”

In any model  $M$ ,  $\llbracket \neg \varphi \rrbracket^M = \begin{cases} 0 & \text{if } \llbracket \varphi \rrbracket^M = 1 \\ 1 & \text{if } \llbracket \varphi \rrbracket^M = 0 \end{cases}$

$p$	$\neg p$
1	0
0	1

Conjunction  $\wedge$ : “and”

$p$	$q$	$p \wedge q$
1	1	1
1	0	0
0	1	0
0	0	0

**Exercise:** (5) is ambiguous. Describe the two readings in terms of  $p$  = “Keith had beer” and  $q$  = “Keith had durian.”

- (5) Keith didn’t have beer and durian.  
 a.  
 b.

What is the relationship between (5a) and (5b)? Use the truth table to help guide your thinking.

$p$	$q$	$p \wedge q$	$\neg p$	$\neg q$	(5a)	(5b)
1	1	1	0	0	1	1
1	0	0	0	1	0	0
0	1	0	1	0	1	0
0	0	0	1	1	0	0

Disjunction  $\vee$ : We want this (roughly) to correspond to natural language “or,” but here there’s a challenge. Consider:

- (6) We can meet today or tomorrow.
- (7) Your taxes will be lower if you are over 65 or blind.

We define  $\vee$  as inclusive disjunction and refer to exclusive disjunction as  $\text{xor}$ :

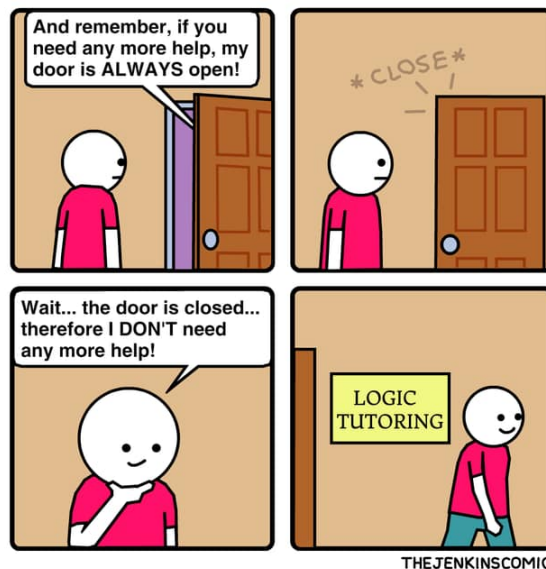
$p$	$q$	$p \vee q$
1	1	1
1	0	1
0	1	1
0	0	0

$p$	$q$	$p \text{ XOR } q$
1	1	0
1	0	1
0	1	1
0	0	0

Conditional  $\rightarrow$  and biconditional  $\leftrightarrow$ : We want  $\rightarrow$  to (very roughly) correspond to “if...then,” but the behavior of natural language conditionals turns out to be quite complicated. The meaning adopted for  $\rightarrow$  is traditionally called the *material conditional*.

$p$	$q$	$p \rightarrow q$
1	1	1
1	0	0
0	1	1
0	0	1

$p$	$q$	$p \leftrightarrow q$
1	1	1
1	0	0
0	1	0
0	0	1



See discussion of conditionals in *IFS* §3, especially §3.2.3.

(8) **De Morgan's laws:**

a.  $\neg p \wedge \neg q$  iff  $\neg[p \vee q]$

b.  $\neg p \vee \neg q$  iff  $\neg[p \wedge q]$

**Exercise:** Prove (8b) using a truth table:

$p$	$q$	$\neg p$	$\neg q$	$\neg p \vee \neg q$	$p \wedge q$	$\neg[p \wedge q]$
1	1					
1	0					
0	1					
0	0					

**Q:** Can we really “prove” something by truth tables?

**A:** Yes! if done correctly. Each row in a truth table corresponds to a particular model (or class of models). By checking each possible combination of truth values for the atomic formulas, we are essentially checking across all possible models.<sup>3</sup>

---

(9) **Summary: Semantics for propositional logic**

a. Atomic formulas: Each atomic formula is true or false; e.g.  $\llbracket p \rrbracket^M = 1$  or  $\llbracket p \rrbracket^M = 0$ .

b. Negation:  $\llbracket \neg\varphi \rrbracket^M = 1$  iff  $\llbracket \varphi \rrbracket^M = 0$

c. Binary connectives:

$$\llbracket [\varphi \wedge \psi] \rrbracket^M = 1 \text{ iff } \llbracket \varphi \rrbracket^M = 1 \text{ and } \llbracket \psi \rrbracket^M = 1 \qquad \llbracket [\varphi \rightarrow \psi] \rrbracket^M = 1 \text{ iff } \llbracket \varphi \rrbracket^M = 0 \text{ or } \llbracket \psi \rrbracket^M = 1$$

$$\llbracket [\varphi \vee \psi] \rrbracket^M = 1 \text{ iff } \llbracket \varphi \rrbracket^M = 1 \text{ or } \llbracket \psi \rrbracket^M = 1 \qquad \llbracket [\varphi \leftrightarrow \psi] \rrbracket^M = 1 \text{ iff } \llbracket \varphi \rrbracket^M = \llbracket \psi \rrbracket^M$$

---

<sup>3</sup>If the relevant formulas depend on  $n$  atomic propositions, there will be  $2^n$  rows to check. Here, with just  $p$  and  $q$ , there are  $2^2 = 4$  rows to check.

## 2 Predicate logic with variables<sup>4</sup>

We now introduce a way of talking about predicates and their arguments, not just relationships between wffs (i.e. sentences).

### (10) Individuals and predicates:

- a. Individuals include John, Mary, Taro, Hanako, Pochi, Fido, etc.
- b. One-place (unary) predicates: Student, Japanese, Smart, Barks, Dog, Cat, etc.  
If  $\pi$  is a one-place predicate and  $\alpha$  is an individual,  $\pi(\alpha)$  is an atomic formula.
- c. Two-place (binary) predicates: Likes, Petted, Kissed, etc.  
If  $\pi$  is a two-place predicate and  $\alpha, \beta$  are individuals,  $\pi(\alpha, \beta)$  is an atomic formula.

Some examples of wff:

- |                                 |   |
|---------------------------------|---|
| (11) Student(John)              | (13) Petted(Mary,Pochi) $\leftrightarrow$ Likes(Mary,Pochi) |
| (12) $\neg\neg$ Japanese(Pochi) | (14) Happy(Fido) $\rightarrow$ Barks(Fido)                  |

To this, we can add *variables*:

### (15) Variables and two quantifiers:

- a. Variables:  $x, y, z$  etc. These variables stand in for individuals.
- b. Quantification: If  $\varphi$  is a wff and  $u$  is a variable,  $[\forall u . \varphi]$  and  $[\exists u . \varphi]$  are wffs.

They are interpreted as follows:

- i.  $[\forall u . \varphi]^M = 1$  iff for all individuals  $k$ ,  $[\varphi]^M = 1$  when  $u = k$
  - ii.  $[\exists u . \varphi]^M = 1$  iff there is at least one individual  $k$  such that  $[\varphi]^M = 1$  when  $u = k$
- (16)  $\forall x . [\text{Student}(x) \rightarrow \text{Smart}(x)]$
- (17)  $\exists y . [\text{Dog}(y) \wedge \text{Smart}(y)]$

---

<sup>4</sup>My presentation here is somewhat informal, to try to keep things very digestible. *IFS* includes a much more formal presentation of predicate logic with variables, in the section beyond what is assigned.

## Notes on variables

### (18) Some math “sentences”:

- a.  $1 = 2 - 1$  a sentence with no variables; not context-sensitive  
b.  $n = 2 - 1$  a sentence with a variable; context-sensitive  
c.  $\forall n (2(n + 1) = 2n + 2)$  a sentence with a variable; *not* context-sensitive

- We say (18b) contains a *free variable* because the truth of the sentence depends on the context. In particular, the sentence is true iff the variable “ $n$ ” is interpreted as 1.
- The truth of sentence (18c), like (18a), does not depend on the context at all.

### (19) Some terminology, using (18c) as an example:

$$\underbrace{\forall n \left( 2 \underbrace{(n + 1)}_{\text{bound}} = 2 \underbrace{n + 2}_{\text{bound}} \right)}_{\text{scope}}$$

- *Binders* control the interpretation of a particular variable within a certain part of its structure, which we call its *scope*. Here,  $\forall$  *binds* the variable  $n$  in its scope.
- We call variables that are in the scope of a matching binder *bound variables*.

More examples:

(20)  $\exists y . [\text{Green}(y)]$

(21)  $\exists x . [\text{Like}(x, \text{Taro})]$

(22)  $\{x : \text{Like}(x, \text{Taro})\}$

Just as we saw with  $\{\dots : \dots\}$  notation before, we can write some weird meanings with  $\exists$  and  $\forall$ :

(23)  $\forall y . [\text{Dog}(\text{Fido})]$  *vacuous quantification*

Note that bound variables are interchangeable, if you carefully substitute the binder and bound variables together:

(24)  $\exists y . [\text{Green}(y) \wedge \text{Dog}(y)] = \exists z . [\text{Green}(z) \wedge \text{Dog}(z)]$

(25)  $\{x : \text{Like}(x, y)\} \neq \{y : \text{Like}(x, y)\}$