Logic

Readings: C&C, Ch. 4.1–4.3.1

1. Propositional logic

- *Propositional logic* is concerned with complex propositions built from simple propositions with the help of logical connectives.

1.1. Introducing logical connectives

- Assuming p and q are propositions:
  - conjunction: $p \wedge q$, equivalent to *p and q* in natural language;
  - disjunction: $p \vee q$, equivalent to *p or q* in natural language;
  - material implication: $p \rightarrow q$, roughly equivalent to *if p, then q* in natural language;
  - negation: $\neg p$, equivalent to *not p* in natural language.

1.2. Semantics of logical connectives via truth tables

1.2.1. *Conjunction*

(1) All the figures have the same shape and all the figures have the same color.

- For (1) to be true, it has to be the case that both conjuncts are true. That is the case for any sentence of the form $p \wedge q$. We can thus represent the semantics of conjunction using a *truth table*:

(2)

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

1.2.2. *Disjunction*

(3) All the figures have the same shape or all the figures have the same color.

- For (3) to be true, one or both disjuncts have to be true. That is the case for any sentence of the form $p \vee q$.
- You might have an intuition that a sentence of the form *p or q* is true only if one of the disjuncts is true, but not both. This reading corresponds to *exclusive* (as opposed to *inclusive*) *disjunction*. In propositional logic we sometimes write *p* XOR *q* to represent exclusive disjunction.

In-class Exercise 1

- Draw the truth tables for $p \vee q$ and p XOR q.

- Exclusive disjunction seems to be the default reading of English sentences involving *or*. It is commonly assumed, however, that this reading arises as a conversational implicature. That is, *or* is taken to denote inclusive disjunction and this meaning is strengthened via Gricean reasoning. Go through the steps of the Gricean reasoning process (i.e., which maxims are involved etc.).

- Show that the exclusive reading is indeed a conversational implicature.

1.2.3. Material implication

(4) *Context: You are playing a slot machine that displays three figures when you pull the lever. You win if and only if all the figures have the same shape.*

If all the figures have the same shape, you win.

- Given the context, the sentence in (4) should be true if the slot machine is working well. In order to check whether it really does work well, one could go through the following scenarios:
  - All the figures have the same shape and you win — the machine is working well.
  - The figures don't all have the same shape and you win — the machine is working well.
  - All the figures have the same shape and you don't win — problem!
  - The figures don't all have the same shape and you don't win — the machine is working well.
- In other words, (4) is only false when its *antecedent* is true and its *consequent* is false. We will assume that the same holds for any sentence of the form $p \to q$.

In-class Exercise 2

- Draw the truth table for $p \rightarrow q$.

- To capture the meaning of natural language conditionals (*if p*, *(then) q*), we often need something more sophisticated then material implication. E.g., treating (5) via material implication would predict it to be true, but is (5) intuitively true? We will not talk about natural language conditionals in this course, however.

(5) If the moon is made of green cheese, then I had yogurt for breakfast this morning.

1.2.4. Negation

- The connectives above are all *binary connectives*. Negation is a *unary connective*.

(6) It's not the case that all the figures have the same shape.

- (6) is true if and only if the sentence *All the figures have the same shape* is false. More generally: $\neg p$ is true iff p is false.

In-class Exercise 3

- Draw the truth table for $\neg p$.

1.3. Reasoning with truth tables

- Reminder: φ ([faɪ]) entails ψ ([saɪ]) iff whenever φ is true, ψ is also true.
- We can thus see if φ entails ψ by looking at all the rows of the truth table where φ is true and checking if ψ is also true in those rows.

In-class Exercise 4

- Does p entail $p \vee q$? Explain using the relevant truth table from In-class Exercise 1.
- Does $[p \rightarrow q] \wedge \neg p$ entail $\neg q$? Explain using a truth table.

- φ contradicts ψ iff whenever φ is true, ψ is false (and vice versa).
- We can thus see if φ contradicts ψ by looking at the truth table and checking if in all rows the values of φ and ψ are different.

In-class Exercise 5

- Show that p contradicts $\neg p$ using the truth table from In-class Exercise 3.
- Does $p \rightarrow q$ contradict $p \wedge \neg q$? Explain using a truth table.

1.4. Grammar of propositional logic

- Propositional logic is a language, so it has a syntax (specifying which formulas are well-formed) and a semantics (assigning denotations to all well-formed formulas).

1.4.1. *Syntax of propositional logic*

- Syntactic rules of propositional logic:
    1. Atomic formulas: p, q, r, ...
    2. Negation: if φ is a formula, then $\neg\varphi$ is a formula.
    3. Binary connectives: if φ and ψ are both formulas, then so are:
        (a) $[\varphi \wedge \psi]$
        (b) $[\varphi \vee \psi]$
        (c) $[\varphi \rightarrow \psi]$
        (d) $[\varphi \leftrightarrow \psi]$
    4. Nothing else is a formula.
- Brackets are important. E.g., $[\neg p \wedge q]$ and $\neg[p \wedge q]$ are two different formulas. Outermost brackets are usually omitted.

In-class Exercise 6

- For each of the strings below say if it is a formula of propositional logic.

(7) a. $\neg\neg p$
b. $\neg \wedge p$
c. $p, q \vee r$

1.4.2. *Semantics of propositional logic*

- The truth of a formula depends on what the world is like, i.e., on the *model M*.
- In propositional logic a model will specify for each simple proposition whether it's true or false. For example, if our model M_1 specifies that p is true and q is false, we can write:

(8) $M_1(p) = 1$
$M_1(q) = 0$

- *Semantic values* are assigned to linguistic expressions by the *valuation function*, written as $[\![\cdot]\!]$, which will be relativized to a model: $[\![\cdot]\!]^M$.
- Semantics of propositional logic:
    1. Atomic formulas: if φ is an atomic formula, then $[\![\varphi]\!]^M = M(\varphi)$ (i.e., whichever truth value the model M specifies for φ).
    2. Negation: $[\![\neg\varphi]\!]^M = 1$ iff $[\![\varphi]\!]^M = 0$.
    3. Binary connectives:
        (a) $[\![\varphi \wedge \psi]\!]^M = 1$ iff $[\![\varphi]\!]^M = 1$ and $[\![\psi]\!]^M = 1$
        (b) $[\![\varphi \vee \psi]\!]^M = 1$ iff $[\![\varphi]\!]^M = 1$ or $[\![\psi]\!]^M = 1$
        (c) $[\![\varphi \rightarrow \psi]\!]^M = 1$ iff $[\![\varphi]\!]^M = 0$ or $[\![\psi]\!]^M = 1$
        (d) $[\![\varphi \leftrightarrow \psi]\!]^M = 1$ iff $[\![\varphi]\!]^M = [\![\psi]\!]^M$

In-class Exercise 7

- Assuming a model M_1 such that $[\![p]\!]^{M_1} = 1$ and $[\![q]\!]^{M_1} = 0$, compute the semantic value of the formulas below step by step. For each step say which semantic rule of propositional logic you used.

(9) a. $[\![\neg[p \wedge q]]\!]^{M_1} =$

b. $[\![\neg p \vee q]\!]^{M_1} =$

c. $[\![p \rightarrow \neg q]\!]^{M_1} =$

d. $[\![\neg p \leftrightarrow q]\!]^{M_1} =$

2. Predicate logic

- Predicate logic (a.k.a *first-order logic*, *first-order predicate calculus*) adds predication and quantification to propositional logic.

2.1. Predication

- In propositional logic, we'd express the natural language sentences in (10) as atomic sentences p, q and r. But this can't capture what these sentences have in common (namely, (10a) and (10b) have the same subject, and (10b) and (10c) the same predicate).

(10) a. Neil giggles.
b. Neil sings.
c. Marilyn sings.

- In predicate logic, we have *individual constants*, e.g., n (Neil) and m (Marilyn), and we have *predicates*, e.g., Giggles and Sings. This allows us to express the sentences in (10) as in (11):

(11) a. Giggles(n)
b. Sings(n)
c. Sings(m)

- In addition to one-place predicates like Sings, there are also two-place predicates like Sees, and three-place predicates like Gives.

(12) a. Sees(m,n)
b. Gives(n,m,e)

- Just as in propositional logic, atomic sentences can be connected with $\wedge$, $\vee$, $\rightarrow$ and $\leftrightarrow$ to form complex sentences.

(13) Sings(n) $\rightarrow$ Giggles(m)

In-class Exercise 8

- For each of the strings below say if it is a formula of predicate logic.

(14) a. $\neg\neg$Purrs(c)
b. Smart(Student)
c. Student(h,r)
d. Kissed(r)
e. Petted(h,c) $\vee$ Petted(c,h)
f. Likes(h,r) $\leftrightarrow$ Likes(r,h)

2.2. Quantification

- Predicate logic also has *variables* (as opposed to *constants*) and quantifiers. With these tools, we can capture meanings of sentences such as:

(15) a. Some cat purrs.
b. Every cat purrs.

- (15a) is true iff there is an individual *x* such that *x* is a cat and *x* purrs. To represent this meaning, we will use the symbol $\exists$, which stands for the *existential quantifier* (meaning 'there exists' or 'there is'), and variables:

(16) $\exists x.[\text{Cat}(x) \wedge \text{Purrs}(x)]$

- (15b) is true iff for all *x*, if *x* is a cat, it purrs. To represent this meaning, we will use the symbol $\forall$, which stands for the *universal quantifier* (meaning 'for all'), and variables:

(17) $\forall x.[\text{Cat}(x) \rightarrow \text{Purrs}(x)]$

- Brackets are used to indicate the *scope* of the quantifier (but are often omitted when the scope is unambiguous). A quantifier only *binds* the variables that are in its scope. E.g., in (18a) $\exists$ binds both occurrences of *x*, but in (18b) the second occurrence of *x* is *unbound*.

(18) a. $\exists x.[\text{Cat}(x) \wedge \text{Purrs}(x)]$
b. $\exists x.[\text{Cat}(x)] \wedge \text{Purrs}(x)$

In-class Exercise 9

- Read the following formulas aloud using such words as *there exists*, *for all*, *it's not the case that*, etc.:

(19) a. $\forall x.[\exists y.[\text{Likes}(x, y)]]$
b. $\forall x.[\text{Student}(x) \rightarrow \exists y.[\text{Cat}(y) \wedge \text{Petted}(x, y)]]$
c. $\neg\exists x.[\text{Cat}(x) \wedge \text{Purrs}(x)]$
d. $\forall x.\neg[\text{Cat}(x) \rightarrow \text{Purrs}(x)]$

- Express the following sentences in predicate logic (you can use as many constants as you need):

(20) a. Some student is tall.
b. Some man petted some cat.
c. It's not the case that some woman petted Fido.
d. All cats are white or gray.

2.3. Semantics of predicate logic

- The semantics of predicate logic is a bit more involved than that of propositional logic and we will skip the details here (if you are interested, you can have a look at Coppock and Champollion textbook draft, Ch. 3.2–3).
- But on an intuitive level, if we understand what a sentence in predicate logic expresses, we can also determine which semantic value this sentence receives in a given model.

In-class Exercise 10

- Assume a model M_2, where the set of individuals consists of Chloe and Rocky ({c, r}), who are both cats. Only Chloe purrs. Chloe scratches Rocky but Rocky doesn't scratch Chloe. Determine which semantic value the sentences below receive in M_2.

(21) a. $\text{Purrs}(r)$
b. $\forall x.[\text{Cat}(x)]$
c. $\exists x.[\text{Cat}(x) \wedge \text{Purrs}(x)]$
d. $\forall x.\forall y.[\text{Scratches}(x, y) \rightarrow \text{Purrs}(x)]$
e. $\forall x.\forall y.[\text{Scratches}(x, y) \leftrightarrow \text{Scratches}(y, x)]$

What you need to know

Key notions: exclusive vs. inclusive disjunction, antecedent, consequent, binary vs. unary connectives, model, valuation function, individual constant, variable, existential quantifier, universal quantifier, quantifier scope, bound variable, unbound variable

Skills:

- Draw truth tables for complex sentences in propositional logic.
- Determine whether two complex sentences in propositional logic entail or contradict one another using truth tables.
- Say if a given string is a well-formed formula in propositional logic.
- Read formulas written in propositional and/or predicate logic using such words as *there exists*, *for all*, *it's not the case that*, etc.
- Represent meanings of natural language sentences consisting of names, predicates, quantifiers such as *some cat* and *every cat*, and counterparts of logical connectives (*and*, *or*, *if... then*, and negation) using predicate logic.
- Compute semantic values of formulas in propositional logic given a model.