

# 1 Logic

**Definition:** A *proposition* is a statement that has a truth value. It is either true or false but not both.

*Which of the following statements are propositions?*

1. This summer was on the cold side.
2. When will this class end?
3.  $3^2 + 4^2 = 6^2$ .
4.  $3x + 5 = 50$ .
5. If  $x$  is an odd number then so is  $3x$ .

*Note:* Mathematicians have adopted the convention that a statement is true provided it is *absolutely true without exception*. Otherwise, it is false. Notice that this may not be the case in casual conversation. Sometimes people will say that a statement is true *and* false. Such ambiguities do not exist in mathematical statements. In the subsequent discussion, we will define more words. Likewise, the definitions are very precise and sometimes different from that of standard usage.

## 1.1 Logical operators

Compound propositions are built from simpler propositions using logical operators. To understand how these logical operators work, we use *truth tables* to specify the truth value of a compound proposition given the truth values of the simpler propositions.

Let  $p$  and  $q$  be propositions. We define six very common logical operators.

1. *negation* of  $p$ ,  $\neg p$ .

Let  $p =$  "It is cold outside." What is  $\neg p$ ?

2. *conjunction* of  $p$  and  $q$ ,  $p \wedge q$ , "*p and q*"

Let  $q =$  "I'm wearing boots." What is  $p \wedge q$ ?

3. *disjunction* of  $p$  and  $q$ ,  $p \vee q$ , “ $p$  or  $q$ ”  
What is  $p \vee q$ ?

4. *exclusive or* of  $p$  and  $q$ ,  $p \oplus q$ , “ $p$  xor  $q$ ”  
What is  $p \oplus q$ ?

5. *implication*  $p \rightarrow q$ , “if  $p$  then  $q$ ”

Statement  $p$  is called the *hypothesis* while  $q$  is called the *conclusion* of the implication.

Consider the following if-then statements:

“If you mow the lawn, I will pay you \$20.”

“If you don’t finish your fish tempura, you can’t have green tea ice cream.”

What’s the difference?

Common *mathspeak* for  $p \rightarrow q$  include:

- $p$  implies  $q$
- $q$  whenever  $p$
- $p$  is sufficient for  $q$
- $q$  is necessary for  $p$
- $p$  only if  $q$

The *contrapositive* of  $p \rightarrow q$  is  $\neg q \rightarrow \neg p$ . The *converse* of  $p \rightarrow q$  is  $q \rightarrow p$ . What are the contrapositive and converse of these statements?

- If  $x$  is an even number then  $3x$  is an even number.
- If you don't leave by 8 AM, you will be late for class.

6. *biconditional*  $p \leftrightarrow q$ , “ $p$  if and only if  $q$ ”  
 $p \leftrightarrow q$  also means  $(p \rightarrow q) \wedge (q \rightarrow p)$ .

Common *mathspeak* for  $p \leftrightarrow q$  include:

- $p$  is necessary and sufficient for  $q$
- if  $p$  then  $q$  and conversely

## Truth Tables of Compound Propositions

To better understand compound propositions, it is helpful (if it is not too time-consuming) to build their truth tables. One important question to ask is *when do these compound propositions evaluate to true?*

1.  $(p \vee \neg q) \rightarrow (p \wedge q)$

2.  $(p \rightarrow q) \rightarrow r$

## Translating English Sentences to Logical Expressions

To *apply* logic, we need to be able to take English sentences and translate them to logical propositions. Here are the basic steps:

- Parse the compound sentence into propositions without logical connectives.
- Represent each proposition by a letter.
- Combine the propositions with logical connectives used in the sentence.

1. The user has paid the subscription fee but did not enter a valid password.

2. You cannot ride the roller coaster if you are under four feet tall unless you are older than 16 years old.

3. To become a professor at Harvard, it is sufficient to be world-famous.

## 1.2 Propositional Equivalences

In algebra, you learned that

$$(3x + 2y)(x - y) = 3x^2 - xy - 2y^2.$$

What does it mean for the left-hand side (LHS) of the equation to be equal to the right-hand side (RHS)?

Similar notions exist for propositions.

**Definition:** Two compound propositions  $P$  and  $Q$  constructed from the same simple propositions are *logically equivalent* whenever their truth values are the same for all the combinations of the truth values of the simple propositions. Notation:  $P \equiv Q$  or  $P \Leftrightarrow Q$ .

*Example 1:* Show that  $\neg(p \wedge q) \equiv \neg p \vee \neg q$

*Example 2:* Show that  $(p \rightarrow q) \equiv \neg p \vee q$

*Note:* Constructing the truth tables for  $P$  and  $Q$  is a straightforward way of verifying that  $P$  and  $Q$  are equivalent. The problem, however, is that the table be very large. *If  $P$  is made up of  $n$  simple propositions, how many rows does its truth table have?*

### Logical Identities

In algebra, the correctness of  $(x + 1)^2 = x^2 + 2x + 1$  is obtained not by making sure that the equation holds for every value of  $x$  but by using various laws such as the distributive law of multiplication over addition, and associative law of addition. We have similar laws for propositions.

Here are the identities:

Let us now use these identities to prove the equivalence of propositions.

1. Show that  $p \vee (\neg p \wedge q) \equiv p \vee q$ .

2. Show that  $p \rightarrow q \equiv \neg q \rightarrow \neg p$ . (That is, an implication is logically equivalent to its contrapositive.)

3. Show that  $p \leftrightarrow q \equiv (p \wedge q) \vee (\neg p \wedge \neg q)$ .

## Tautologies and Contradictions

**Definition:** A compound proposition  $P$  is a *tautology* if no matter what the truth values of the simple propositions that make it up are, it is always true (e.g.,  $p \vee \neg p$ ). Another way of saying this is  $P \equiv T$ , where  $T$  stands for true.

Similarly, A compound proposition is a *contradiction* if no matter what the truth values of the simple propositions that make it up are, it is always false (e.g.,  $p \wedge \neg p$ ). That is,  $P \equiv F$ .

*Note:* If two compound propositions  $P$  and  $Q$  are equivalent, then the statement  $P \leftrightarrow Q$  is a tautology.

*Example:* Show that  $p \rightarrow (p \vee q)$  is a tautology.