Propositional Logic

Declaration
This note was prepared based on the book *Logic for Computer Science* by A. Singha.

1 Introduction

Our goal is to model reasoning as it is used in mathematics and computer science, taking cues from that found in day-to-day communication. We start with the simplest kind of reasoning, called reasoning with propositions and connectives. Here are some propositions:

- No bachelor is married.
- Some unmarried men get married.
- Five men cannot have eleven eyes.
- Alexander the Great did not set foot in India.
- The title of a book on logic could be misspelt.
- The woman who committed the crime did not have three legs.

Propositions are declarative sentences which may be asserted to be true or false. It is quite possible that you may not be able to say for certain whether a given proposition is true or false, without going to its meanings or external factors. For example, the conjectures or open problems below are propositions whose truth we do not know yet:

- **Goldbach's Conjecture:** Every even number bigger than 2 is a sum of two prime numbers.
- **$\mathbf{P} \neq \mathbf{NP}$:** There is at least one non-deterministic polynomial-time solvable problem which is not deterministic polynomial-time solvable.

As of now, we do not have any way of showing the truth or falsity of these propositions. However, each of them is either true or false.

We are not defining here what a proposition is. We are only getting familiarized with the kind of objects in question. A safer way to describe a proposition is to see whether the question, "Is it true that X?" is meaningful or not. If it is, then X is a proposition; otherwise, X is not a proposition.

The sentences which are not propositions include questions, orders, exclamations, etc., for which we may not like to associate a truth value. For example, we do not know how to say whether "Is the night sky beautiful?" is true or false. Similarly, we may not assert that "How beautiful is the morning sky!" is true or false.

Our building blocks here are propositions; we will not try to go beyond them. It is not our concern to determine whether "Each bachelor is married," for we pretend not to know the meanings of the words uttered in the proposition. Our units here are propositions, nothing less and nothing more. However, we seem to know that two propositions such as "I know logic" and "You know logic" can be composed to get another proposition such as "I and you know logic."

We are only interested in propositions and how they are composed to yield other propositions. This is what we mean when we say that propositions are our building blocks. Thus, we are interested in the forms rather than the meanings of propositions. Since propositions can be true or false, we must know how to assign truth values to compound propositions.

If indeed I like logic and you like logic, then we must agree that the proposition "I and you like logic" is true. But what about the proposition:

I like logic and you like logic or you do not like logic?

This is problematic, for we do not know exactly how this compound proposition has been composed or formed. Which of the following ways must we parse it?

(I like logic and you like logic) or (you do not like logic),

(I like logic) and (you like logic or you do not like logic).

We will use parentheses for disambiguating compound propositions. Moreover, we will start with some commonly used connectives; and if the need arises, we will enrich our formalization by adding new ones. Of course, we will explain what "follows from" means.

In the sequel, we will shorten the phrase "if and only if" to "iff," and denote the set of natural numbers $\{0, 1, 2, 3, \ldots\}$ by $\mathbb{N}$.

Exercises

1. Do the following pairs of sentences mean the same thing? Explain.

   (a) Healthy diet is expensive. Expensive diet is healthy.

(b) Children and senior citizens get concession. Children or senior citizens get concession.

2. In Smullyan's island, there are two types of people: knights, who always tell the truth, and knaves, who always lie. A person there asserts: "This is not the first time I have said what I am now saying." Is the person a knight or a knave?

3. In Smullyan's island, a person says A and B where A and B are two separate sentences. (For instance, A is "I have a brother" and B is "I have a sister.") The same person later asserts A, and then after a minute, asserts B. Did they convey the same as earlier?

4. Is the sentence "This sentence is true" a proposition?

5. Is the sentence "This sentence is false" a proposition?

2 Syntax of PL

For any simple proposition, called a propositional variable, we will use any of the symbols $p_0, p_1, \ldots$. For connectives 'not', 'and', 'or', 'if ... then ...', '... if and only if ...', we use the symbols $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$, respectively; their names are negation, conjunction, disjunction, conditional, biconditional. We use the parentheses ')' and '(' as punctuation marks. We also have the special propositions $\top$ and $\bot$, called **propositional constants**; they stand for propositions which are 'true' and 'false', respectively. Read $\top$ as top, and $\bot$ as bottom or falsum. Both propositional variables and propositional constants are commonly called **atomic propositions** or **atoms**. So, the alphabet of Propositional Logic, PL, is the set:

$$\{), (, \neg, \wedge, \vee, \rightarrow, \leftrightarrow, \top, \bot, p_0, p_1, p_2, \ldots\}.$$

Any expression over this alphabet is a string of symbols such as

$$(\neg p_0 \rightarrow () \wedge p_1 \vee, \neg p_{100})(\rightarrow \vee, (\neg p_0 \rightarrow p_1).$$

(Do not read the commas and dots.) Only the last one of these is a propositional formula or a proposition. In fact, we are interested only in such expressions. The propositions (in PL) are defined by the following grammar:

$$w ::= \top \mid \bot \mid p \mid \neg w \mid (w \wedge w) \mid (w \vee w) \mid (w \rightarrow w) \mid (w \leftrightarrow w)$$

Here p stands for any generic propositional variable, and w stands for any generic proposition. The symbol ::= is read as 'can be'; and the vertical bar '|' describes alternate possibilities (read it as 'or'). The same symbol w may be replaced by different propositions; that is what the word "generic" means. This way of writing the grammatical rules is called the Bacus-Naur form or BNF, for short. The grammar can be written in terms of the following formation rules of propositions:

1. $\top$ and $\bot$ are propositions.
2. Each p_i is a proposition, where $i \in \mathbb{N}$.
3. If x is a proposition, then $\neg x$ is a proposition.
4. If x, y are propositions, then $(x \wedge y), (x \vee y), (x \to y), (x \leftrightarrow y)$ are propositions.
5. Nothing is a proposition unless it satisfies some or all of the rules (1)-(4).

The set of all propositions is written as PROP. The formation rule (5) is called the closure rule. It states that PROP is the smallest set that satisfies (1)-(4). The 'smallest' is in the sense that A is smaller than B iff $A \subseteq B$.

Propositions are also called PL-formulas and well-formed formulas, wff for short. The non-atomic propositions are also called compound propositions.

Example 1.1. The following are propositions:

$$p_0, \quad (p_5 \to \top), \quad ((p_{100} \leftrightarrow \bot) \wedge \neg p_7), \quad (p_8 \to ((\neg p_4 \vee p_9) \leftrightarrow (p_2 \wedge (\top \to \bot)))),$$

whereas the following are not propositions:

$$p_0 \wedge p_1, \quad p_0 \to \bot), \quad (\neg\neg(p_0 \vee p_1)), \quad (p_8 \to ((\neg p_4 \vee p_9) \leftrightarrow (p_2 \wedge (\top \to \bot))).$$

In $p_0 \wedge p_1$, a surrounding pair of parentheses is required; in $(\neg\neg(p_0 \vee p_1))$ there is an extra pair of parentheses; etc.

The key fact is that any object that has been formed (generated) by this grammar can also be parsed. That is, you can always find out the last rule that has been applied to form a proposition and then proceed backward. Such an unfolding of the formation of a proposition can be depicted as a tree, called a parse tree.

Example 1.2. Construct a parse tree for the proposition

$$\neg((p_0 \wedge \neg p_1) \to (p_2 \vee (p_3 \leftrightarrow \neg p_4))).$$

For the given proposition, the last rule applied was $w ::= \neg w$. This means that it is a proposition if the expression $((p_0 \wedge \neg p_1) \to (p_2 \vee (p_3 \leftrightarrow \neg p_4)))$ is also a proposition. Look at the root and its child in the left tree of Figure 1.1.

Further, $((p_0 \wedge \neg p_1) \to (p_2 \vee (p_3 \leftrightarrow \neg p_4)))$ is a proposition if both the expressions $(p_0 \wedge \neg p_1)$ and $(p_2 \vee (p_3 \leftrightarrow \neg p_4))$ are propositions (the rule $w ::= (w \to w)$). If you proceed further, you would arrive at the left parse tree in Figure 1.1. The corresponding abbreviated tree is on the right.

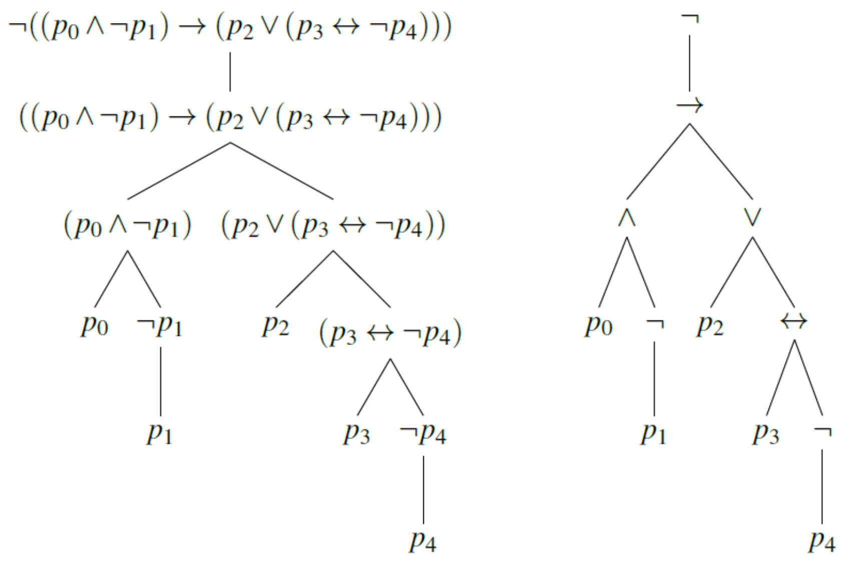

Figure 1: Parse tree for Example 1.2 and its abbreviation

Example 1.3. Consider the string $(\vee(p_1 \wedge p_2) \to (\neg p_1 \leftrightarrow p_2))$. We cannot apply the rule for $\vee$, since to its left is just a parenthesis. But we find that by taking x as $\vee(p_1 \wedge p_2)$ and y as $(\neg p_1 \leftrightarrow p_2)$, the string appears as $(x \to y)$, which can be parsed. Look at the left tree in Figure 1.2. We cannot parse $\vee(p_1 \wedge p_2)$ any further.

Similarly, the string $(\vee \to \neg p_1 \leftrightarrow p_2)$ can be parsed in two ways; first with $\to$, and next with $\leftrightarrow$. Look at the middle and the right trees in Figure 1.2. Neither can we parse $\vee, \neg p_1 \leftrightarrow p_2$ nor $\vee \to \neg p_1$.

Notice that the leaves of the trees of Figure 1.1 are atomic propositions, while the leaves of the trees in Figure 1.2 are not. The corresponding expressions in the latter cases are not propositions.

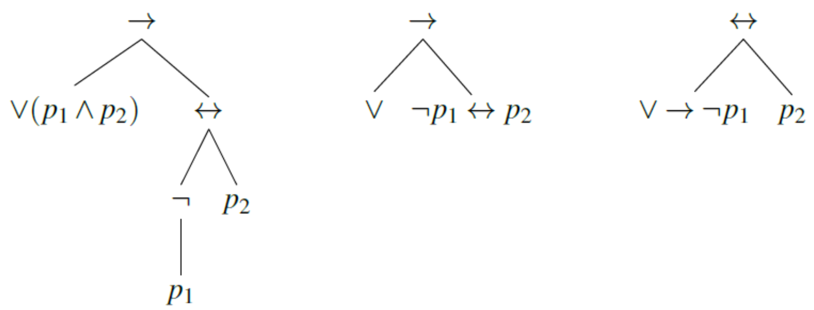

Figure 2: Parse trees for Example 1.3

Exercises

1. Which of the following strings are in PROP and which are not? Why?

(a) $p_0 \vee (p_1 \to \neg p_2)$
(b) $((p_3 \leftrightarrow p_4) \wedge \neg p_1)$
(c) $((p_5) \to (p_2 \leftrightarrow p_3))$
(d) $((p_3 \leftrightarrow p_4) \wedge \neg p_1)$
(e) $(((p_0 \wedge \neg(p_1 \vee p_2)) \to (p_3 \leftrightarrow \neg p_4)) \vee (\neg(p_5 \to p_4) \to \neg p_1) \wedge p_2)$
(f) $((p_1 \wedge \neg p_1) \vee (p_0 \to p_1)) \wedge (\neg(p_0 \wedge \neg\neg p_1) \to ((\neg p_3 \vee \neg p_1) \leftrightarrow p_2))$

2. Construct parse trees for the following propositions:

(a) $(\neg((p_0 \wedge \neg p_1) \to (p_0 \to p_1)) \leftrightarrow \neg(p_1 \vee p_2))$
(b) $((p_3 \vee (p_4 \leftrightarrow \neg(p_3 \wedge p_1))) \wedge \neg(p_2 \to p_5))$
(c) $(\neg(\neg(p_1 \to p_2) \wedge \neg(p_3 \vee p_4)) \leftrightarrow (p_5 \leftrightarrow \neg p_6))$

3. Construct parse trees for the following strings, and then determine which of them are propositions.

(a) $(\neg((p_0 \wedge \neg p_1) \to (p_0 \to p_1))) \leftrightarrow \neg(p_1 \vee p_2)$
(b) $(\neg(((p_0 \wedge \neg p_1) \to (p_0 \to p_1)) \leftrightarrow \neg p_1 \vee p_2))$
(c) $(\neg((p_0 \wedge \neg p_1) \to p_0) \to p_1) \leftrightarrow (\neg(p_1 \vee p_2))$

Example 1.4. Construct parse trees for the following strings and determine which of them are propositions:

1. $(\neg((p_0 \wedge \neg p_1) \to (p_0 \to p_1))) \leftrightarrow \neg(p_1 \vee p_2)$
2. $(\neg(((p_0 \wedge \neg p_1) \to (p_0 \to p_1)) \leftrightarrow \neg p_1 \vee p_2))$
3. $(\neg((p_0 \wedge \neg p_1) \to p_0) \to p_1) \leftrightarrow (\neg(p_1 \vee p_2))$
4. $((p_0 \to p_1) \vee (p_2 \to (\neg p_1 \wedge (p_0 \leftrightarrow \neg p_2))))$
5. $((p_0 \to p_1) \vee (p_2 \to (\neg p_1 \wedge (p_0 \leftrightarrow \neg p_2)))) \leftrightarrow (p_3 \to p_5)$
6. $(((p_5 \to (p_6 \vee p_8)) \leftrightarrow (p_3 \wedge \neg p_2)) \vee (\neg(p_1 \leftrightarrow p_3) \to p_{10}))$
7. $(((p_5 \to p_6 \vee p_8) \leftrightarrow (p_3 \wedge \neg p_2)) \vee (\neg(p_1 \leftrightarrow p_3 \to p_{10})))$

2.1 Parsing Propositions: Unique Parsing and Examples

Given a proposition, we aim to determine in which way it has been formed. If the proposition is of the form $\neg x$, then the x here is uniquely determined from the proposition. If it is in the form $(x \vee y)$ for propositions x and y, can it also be in the form $(z \wedge u)$ for some (other) propositions z and u? Consider the proposition:

$$w = ((p_1 \to p_2) \wedge (p_1 \vee (p_0 \leftrightarrow \neg p_2))).$$

We see that $w = (x \wedge y)$, where $x = (p_1 \to p_2)$ and $y = (p_1 \vee (p_0 \leftrightarrow \neg p_2))$. If we write it in the form $(z \vee u)$, then $z = (p_1 \to p_2) \wedge (p_1$ and $u = (p_0 \leftrightarrow \neg p_2)$. Here, z and u are not propositions.

Recall that the prefix of a string means reading the string symbol by symbol from left to right and stopping somewhere. The prefixes of w above are:

$$(, ((, ((p_1, ((p_1 \to, ((p_1 \to p_2, ((p_1 \to p_2), ((p_1 \to p_2)\wedge, \ldots$$

None of these is a proposition except w itself, as the condition of matching parentheses is not met. Similar situations occur when a proposition starts with one or more occurrences of $\neg$. That is, a proper prefix of a proposition is never a proposition. We express this fact using the following observation:

Observation 1.1. Let u and v be propositions. If u is a prefix of v, then $u = v$.

Theorem 2.1. *(Unique Parsing). Let w be a proposition. Then exactly one of the following happens:*

1. $w \in \{\top, \bot, p_0, p_1, \ldots\}$.
2. $w = \neg x$ *for a unique* $x \in PROP$.
3. $w = (x \circ y)$ *for a unique* $\circ \in \{\wedge, \vee, \to, \leftrightarrow\}$ *and unique* $x, y \in PROP$.

The theorem is so named because it asserts that each proposition has a unique parse tree, which demonstrates how it has been formed by applying the rules of the grammar. Of course, unique parsing does not give any clue as to how to determine whether a given string over the alphabet of PROP is a proposition or not.

For example, $p_1 \to p_2$ is not a proposition; you need a pair of parentheses. Similarly, $(p_0 \wedge p_1 \to p_2)$ is also not a proposition. But that is not the question. The question is: Can you give a procedure to do this for all possible expressions?

Observe that the unique parse tree for a proposition has only atomic propositions on its leaves. Whereas if an expression is not a proposition, in any parse tree for the expression some leaf will contain an expression other than an atomic proposition.

Algorithm 1 Procedure PropDet

```
Input: Any string x over the alphabet of PL.
Output: 'yes', if x is a proposition, else, 'no'.

if x is a (single) symbol and x ∉ {), (, ¬, ∧, ∨, →, ↔} then
    Report 'yes'
else
    Report 'no'; and stop
end if

Otherwise, scan x from left to right to get a substring w in one of the forms
¬p, (p ∧ q), (p ∨ q), (p → q), (p ↔ q)
where p, q are symbols not in the set {), (, ¬, ∧, ∨, →, ↔}.

if no such substring is found then
    Report 'no'; and stop
end if

if a valid substring w is found then
    Replace w by p0
    Go to Step 1
end if
```

Example 1.4. We apply PropDet on the strings $(\vee(p_1 \wedge p_2) \to (\neg p_1 \leftrightarrow p_2))$ and $(\vee \to \neg p_1 \leftrightarrow p_2)$.

String 1: $(\vee(p_1 \wedge p_2) \to (\neg p_1 \leftrightarrow p_2))$

Step 1: The string is not a single symbol, so we proceed to Step 2.

Step 2: We scan from left to right to find a substring of the form $\neg p$, $(p \wedge q)$, $(p \vee q)$, $(p \to q)$, or $(p \leftrightarrow q)$. The substring $(p_1 \wedge p_2)$ matches the form $(p \wedge q)$.

Step 3: We replace $(p_1 \wedge p_2)$ with p_0, so the string becomes $(\vee p_0 \to (\neg p_1 \leftrightarrow p_2))$.

Step 4: We repeat Step 2. The substring $(\neg p_1 \leftrightarrow p_2)$ matches the form $(p \leftrightarrow q)$.

Step 5: We replace $(\neg p_1 \leftrightarrow p_2)$ with p_0, so the string becomes $(\vee p_0 \to p_0)$.

Step 6: The string now consists of atomic propositions and the logical operator $\to$, which confirms that it is a valid proposition. Therefore, the output is 'yes'.

String 2: $(\vee \to \neg p_1 \leftrightarrow p_2)$

Step 1: The string is not a single symbol, so we proceed to Step 2.

Step 2: We scan from left to right to find a substring of the form $\neg p$, $(p \wedge q)$, $(p \vee q)$, $(p \to q)$, or $(p \leftrightarrow q)$. However, no valid substring is found.

Step 3: Since no valid substring is found, we report 'no'. Therefore, the output is 'no'.

Correctness of PropDet:

PropDet works by identifying a substring of the given string as a proposition, and then replacing the substring by a propositional variable, p_0. Thus, if it starts from a proposition, it must end with one. The invariance of the loop executed by PropDet is the proposition-hood of the string. Moreover, by any such replacement, PropDet reduces the length of the string. Thus, the given string is a proposition if and only if the final string of length 1 is p_0, which is also a proposition. This criterion is checked in Step 1. This completes the correctness proof of PropDet.

Convention 1.1. Instead of writing $p_0, p_1, p_2, \ldots$, we write propositional variables as $p, q, r, s, t, \ldots$. We write $u, v, w, x, \ldots$ for propositions. Sometimes we may write $p, q, r, s, t, \ldots$ for propositions as well, provided no confusion arises.

To write less, we put down some precedence rules and omit the outer parentheses. Recall the precedence rule that multiplication has higher precedence than addition. This means that the expression $x \times y + z$ is rewritten as $((x \times y) + z)$, and not as $(x \times (y + z))$.

Convention 1.2. Our precedence rules are the following:

- $\neg$ has the highest precedence.
- $\wedge, \vee$ have the next precedence, their precedence being equal.
- $\rightarrow, \leftrightarrow$ have the lowest precedence, their precedence being equal.

Using the precedence rules, the proposition

$$((p_1 \vee (p_3 \wedge p_6)) \rightarrow (p_{100} \leftrightarrow \neg p_1))$$

can be abbreviated to

$$p_1 \vee (p_3 \wedge p_6) \rightarrow (p_{100} \leftrightarrow \neg p_1).$$

Using abbreviations p, q, r, s for p_1, p_3, p_6, p_{100}, respectively, the abbreviated proposition becomes:

$$p \vee (q \wedge r) \rightarrow (s \leftrightarrow \neg p).$$

However, you must not stretch too much the notion of abbreviated propositions. For instance, $p_0 \rightarrow p_1$ is not a sub-proposition of $(p_0 \rightarrow (p_1 \wedge p_2))$. The reason is that the abbreviated proposition in full form looks like $(p_0 \rightarrow (p_1 \wedge p_2))$, and the propositions $p_0 \rightarrow p_1$ and $(p_0 \rightarrow p_1)$ are not even substrings of $(p_0 \rightarrow (p_1 \wedge p_2))$.

Exercises

1. Apply PropDet on the strings

(i) $((p \rightarrow (\neg q \vee r)) \leftrightarrow (\neg p \vee r))$

(ii) $((\neg p \leftrightarrow (\neg q \wedge r)) \to (\neg p \to (r \to s)))$

(iii) $(((\neg p \leftrightarrow (\neg q \wedge r))) \to (\neg p \to ((r \to s) \leftrightarrow \neg q)))$

2. Insert parentheses at appropriate places using the precedence rules so that the following strings become propositions.

(a) $(p \to q) \wedge \neg(r \vee q \leftrightarrow p) \leftrightarrow (\neg p \vee q \to r)$

(b) $(p \to q) \leftrightarrow (r \to t \vee p) \wedge (p \vee q \to \neg p \wedge t)$

(c) $p \vee (\neg q \leftrightarrow r \wedge p) \leftrightarrow (p \vee p \to \neg q)$

(d) $\neg(p \vee (q \to r \vee s) \leftrightarrow (q \leftrightarrow (p \wedge r \wedge \neg q) \vee (r \vee p)))$

3 Semantics of PL

3.1 Interpretation

The meaning associated with any proposition is of two kinds, called true and false, for convenience. In what follows we write 0 for false, and 1 for true. These two tokens, true and false, or 0 and 1, are called the truth values. Propositions are built from the atomic propositions with the help of connectives. The propositional constants are special; $\top$ always receives the truth value true, and $\bot$ always receives the truth value false. Depending on situations, the propositional variables may receive either of the truth values. We must then prescribe how to deal with connectives.

The common sense meaning of the connectives $\neg$, $\wedge$, $\vee$, $\to$, and $\leftrightarrow$ are respectively, not, and, or, implies, and if and only if. It means, $\neg$ reverses the truth values. That is, if x is true, then $\neg x$ is false; and if x is false, then $\neg x$ is true. When both x and y are true, $x \wedge y$ is true; and when at least one of x or y is false, $x \wedge y$ is false. If at least one of x or y is true, then $x \vee y$ is true; and if both x and y are false, $x \vee y$ is false. Similarly, $x \leftrightarrow y$ is true when x and y are true together, or when x and y are false together; $x \leftrightarrow y$ is false if one of x, y is true and the other is false. The problematic case is $x \to y$. We will consider some examples to see what do we mean by this phrase implies, or as is commonly written if . . ., then . . ..

The sentence if x then y is called a conditional sentence with antecedent as x and consequent as y. In main stream mathematics, the meaning of a conditional sentence is fixed by accepting it as false only when its antecedent is true but its consequent is false. It is problematic in the sense that normally people think of a conditional statement expressing causal connections; but this view does not quite go with that. However, this meaning of the conditional statement is not so counter intuitive as is projected by the philosophers. Let us take some examples.

Your friend asks you whether you have got an umbrella, and you answer, ”If I have got an umbrella, then I would not have been wet”. Suppose, you do not have an umbrella.

Then is your statement true or false? Certainly it is not false if you had been really wet. It is also not false even if you had not been wet since it did not rain at all. That is, the statement is not false whenever its antecedent "I have got an umbrella" is false.

Since it is tricky, we consider one more example. At the bank, you asked me for a pen to fill up a form. Before searching I just replied "If I have a pen, I will oblige you." I searched my pockets and bag, but could not find a pen. Looking around I spotted a friend, and borrowed a pen from him for you. Did I contradict my own statement? Certainly not. I would have done so had I got a pen and I did not lend it to you. Even if I did not have a pen, I obliged you; and that did not make my statement false. That is, the sentence "if x then y" is true, if its antecedent x is false.

Thus we accept this meaning of the conditional statement. The proposition $x \to y$ is false only when x is true and y is false; in all other cases, $x \to y$ is true. The true cases of $x \to y$ are when x is false, or when y is true. The former case, that is, when x is false, the conditional statement $x \to y$ looks vacuous; consequently, philosophers call the connective $\to$ as material implication. A rose is a rose, with whatever name you call it!

Formally, the assumed association of truth values to the propositional variables is called a truth assignment. That is, a **truth assignment** is any function from $\{p_0, p_1, \ldots\}$ to $\{0, 1\}$. An extension of a truth assignment to the set of all propositions that evaluates the connectives in the above manner is called an **interpretation**. That is, an interpretation is any function $i : \text{PROP} \to \{0, 1\}$ satisfying the following properties for all $x, y \in \text{PROP}$:

1. $i(\top) = 1$.
2. $i(\bot) = 0$.
3. $i(\neg x) = 1$ if $i(x) = 0$, else, $i(\neg x) = 0$.
4. $i(x \wedge y) = 1$ if $i(x) = i(y) = 1$, else, $i(x \wedge y) = 0$.
5. $i(x \vee y) = 0$ if $i(x) = i(y) = 0$, else, $i(x \vee y) = 1$.
6. $i(x \to y) = 0$ if $i(x) = 1$, $i(y) = 0$, else, $i(x \to y) = 1$.
7. $i(x \leftrightarrow y) = 1$ if $i(x) = i(y)$, else, $i(x \leftrightarrow y) = 0$.

The same conditions are also exhibited in Table 1.1, where the symbols u,x,y stand for propositions. The conditions are called **boolean conditions**; and such a table is called a **truth table**. You must verify that these conditions and the truth table convey the same thing.

$\top$	$\bot$	u	$\neg u$	x	y	$x \wedge y$	$x \vee y$
1	0	1	0	1	1	1	1
0	1	1	0	0	1	0	1
0	1	0	1	1	0	0	1
0	0	0	1	0	0	0	0

Table 1: Truth table for connectives

Alternatively, the boolean conditions can be specified in the following way:

$$\begin{aligned}
i(\top) &= 1, \quad i(\bot) = 0, \\
i(\neg x) &= 1 - i(x), \\
i(x \wedge y) &= \min\{i(x), i(y)\}, \\
i(x \vee y) &= \max\{i(x), i(y)\}, \\
i(x \to y) &= \max\{1 - i(x), i(y)\}, \\
i(x \leftrightarrow y) &= 1 - |i(x) - i(y)|.
\end{aligned}$$

An interpretation is also called a valuation, a boolean valuation, a state, a situation, a world, or a place. We view an interpretation in the bottom-up way. We start with any function $i : \{p_0, p_1, \ldots\} \to \{0, 1\}$. Then, following the above rules, we extend this i to a function from PROP to $\{0, 1\}$.

Is the bottom-up way of extending a function from propositional variables to all propositions well-defined by the required properties? Can there be two different interpretations that agree on all propositional variables?

Theorem 3.1. *Let $f : \{p_0, p_1, \ldots\} \to \{0, 1\}$ be any function. There exists a unique interpretation g such that $g(p_j) = f(p_j)$ for each $j \in \mathbb{N}$.*

Convention 1.3. Due to Theorem 3.1, we write the interpretation that agrees with a truth assignment i as i itself.

The following result implies that if a propositional variable does not occur in a proposition, then changing the truth value of that propositional variable does not change the truth value of that proposition.

Theorem 3.2 (Relevance Lemma)**.** *Let w be a proposition. Let i and j be two interpretations. If $i(p) = j(p)$ for each propositional variable p occurring in w, then $i(w) = j(w)$.*

The **Relevance Lemma** shows that in order to assign a truth value to a proposition, it is enough to know how an interpretation assigns the truth values to the propositional variables occurring in it. We do not need to assign truth values to the propositional variables which do not occur in the proposition.

If a proposition w is built up from n propositional variables, say, $q_1, \ldots, q_n$, then due to the relevance lemma, an interpretation of w will depend only on the truth values assigned to $q_1, \ldots, q_n$. Since each q_i may be assigned 0 or 1, there are 2^n possible (relevant) interpretations of w in total. A truth table for w will thus have 2^n rows, where each row represents an interpretation.

Example 3.1. *The truth table for $(p \to (\neg p \to p)) \to (p \to (p \to \neg p))$ is shown in Table 1.2, where we write $u = p \to (\neg p \to p)$, $v = p \to (p \to \neg p)$, and the given proposition as $u \to v$.*

p	$\neg p$	$p \to \neg p$	$\neg p \to p$	u	v	$u \to v$
0	1	1	0	1	1	1
1	0	0	1	1	0	0

Table 2: Truth Table for Example 3.1

The first row is the interpretation that extends the truth assignment i with $i(p) = 0$ to the propositions $\neg p$, $p \to \neg p$, $\neg p \to p$, u, v, and $u \to v$. The second row is the interpretation j with $j(p) = 1$. We see that $i(u \to v) = 1$ while $j(u \to v) = 0$.

Example 3.2. *The truth table for $u = \neg(p \wedge q) \to (p \vee (r \leftrightarrow \neg q))$ is given in Table 1.3. The first row is the assignment i with $i(p) = i(q) = i(r) = 0$. This is extended to the interpretation i where:*

$$i(\neg q) = 1, \quad i(p \wedge q) = 0, \quad i(\neg(p \wedge q)) = 1, \quad i(r \leftrightarrow \neg q) = 0, \quad i(p \vee (r \leftrightarrow \neg q)) = 0, \quad i(u) = 0.$$

Similarly, other rows are constructed.

p	q	r	$\neg q$	$p \wedge q$	$\neg(p \wedge q)$	$r \leftrightarrow \neg q$	$p \vee (r \leftrightarrow \neg q)$	u
0	0	0	1	0	1	0	0	0
1	0	0	1	0	1	0	1	1
0	1	0	0	0	1	1	1	1
1	1	0	0	1	0	1	1	1
0	0	1	1	0	1	1	1	1
1	0	1	1	0	1	1	1	1
0	1	1	0	0	1	0	0	0
1	1	1	0	1	0	0	1	1

Table 3: Truth Table for $u = \neg(p \wedge q) \to (p \vee (r \leftrightarrow \neg q))$

Exercises

1. For the interpretation i with $i(p) = 1$, $i(q) = 0$, $i(r) = 0$, $i(s) = 1$, compute:

   (a) $i(r \leftrightarrow p \vee q)$

   (b) $i(p \vee q) \to (r \wedge \neg s)$

   (c) $i(r \to p \vee s)$

   (d) $i(\neg s \vee q) \leftrightarrow (r \vee p)$

   (e) $i(p \vee \neg q \vee s \leftrightarrow s \vee \neg s)$

   (f) $i(r \leftrightarrow (p \to \neg r))$

2. Construct truth tables for the following propositions:

(a) $p \to (q \to (p \to q))$

(b) $\neg p \lor q \to (q \to p)$

(c) $\neg(p \leftrightarrow q)$

(d) $(p \leftrightarrow q) \leftrightarrow (p \to q)$

(e) $p \to (p \to q)$

(f) $(p \to \bot) \leftrightarrow \neg p$

3. For an interpretation i, we know that $i(p) = 1$. Then which of the following can be determined uniquely by i with this information only?

(a) $(p \to q) \leftrightarrow (r \to \neg p)$

(b) $(q \to r) \to (q \to \neg p)$

(c) $p \to (q \leftrightarrow (r \to p))$

(d) $p \leftrightarrow (q \to (r \leftrightarrow p))$

4. Using the vocabulary:

- p: It is normal,
- q: It is cold,
- r: It is hot,
- s: It is small,

translate the following into acceptable English:

(a) $p \lor q \land s$

(b) $p \leftrightarrow q$

(c) $p \to \neg(q \lor r)$

(d) $p \lor (s \to q)$

(e) $p \to \neg(q \land r)$

(f) $p \leftrightarrow ((q \land \neg r) \lor s)$

3.2 Models

The proposition $p \vee \neg p$ is always evaluated to 1 and $p \wedge \neg p$ always to 0. Whereas the proposition u in Table 1.3 is evaluated to 0 in some interpretations and to 1 in some other interpretations. It is one of the aims of semantics to categorize propositions according to their truth values under all or some interpretations.

Let w be a proposition. A **model** of w is an interpretation i with $i(w) = 1$. The fact that i is a model of w is written as $i \models w$. It is also read as "i verifies w;" and "i satisfies w." The fact that i is not a model of w is written as $i \not\models w$; and is also read as "i does not satisfy w," "i does not verify w," and "i falsifies w."

Example 3.3. *In Table 1.3, let i be the interpretation as given in the first row. That is, $i(p) = i(q) = i(r) = 0$. The table says that $i \models u$. Check that*

$$i \models \neg(p \wedge q) \vee r \rightarrow \neg p \vee (\neg q \vee r), \quad i \models (p \wedge q \rightarrow r) \leftrightarrow p \wedge (q \wedge \neg r).$$

The interpretation j with $j(p) = 1$, $j(q) = j(r) = 0$ is a model of u. Which line in Table 1.3 is the interpretation j?

A proposition w is called:

- **valid**, written as $\models w$, if and only if each interpretation of w is its model;
- **invalid**, written as $\not\models w$, if and only if it is not valid;
- **satisfiable**, if and only if it has a model;
- **unsatisfiable**, if and only if it is not satisfiable;
- **contingent**, if and only if it is both invalid and satisfiable;
- the proposition $\top$ (true) is defined to be satisfiable;
- the proposition $\bot$ (false) is defined to be invalid.

Valid propositions are also called **tautologies**, and unsatisfiable propositions are called **contradictions**.

Notice that:

- $\top$ is both valid and satisfiable, whereas $\bot$ is both unsatisfiable and invalid;
- each propositional variable is contingent.

Example 3.4. *The proposition $p \vee \neg p$ is valid, i.e., $\models p \vee \neg p$, since each interpretation evaluates it to 1. Each of its interpretations is its model.*

The proposition $p \wedge \neg p$ is unsatisfiable since each interpretation evaluates it to 0. No interpretation is its model.

Let $u = \neg(p \wedge q) \to (p \vee (r \leftrightarrow \neg q))$*. Look at Table 1.3. Let* i*,* j *be interpretations of* u *with* $i(p) = 1$*,* $i(q) = i(r) = 0$ *and* $j(p) = j(q) = j(r) = 0$*. It is clear that* $i \models u$ *whereas* $j \not\models u$*. Therefore,* u *is contingent.*

Example 3.5. *Categorize the following propositions into valid, invalid, satisfiable, and unsatisfiable:*

(a) $(p \to (q \to r)) \to ((p \to q) \to (p \to r))$

(b) $((p \to q) \to (p \to r)) \wedge \neg(p \to (q \to r))$

You can construct a truth table to answer both (a) and (b). Here is an approach that avoids the construction of a truth table.

(a) Suppose $i((p \to (q \to r)) \to ((p \to q) \to (p \to r))) = 0$, for an interpretation i. Then, $i((p \to (q \to r)) = 1$ and $i((p \to q) \to (p \to r)) = 0$. The last one says that $i(p \to q) = 1$ and $i(p \to r) = 0$. Again, $i(p \to r) = 0$ gives $i(p) = 1$ and $i(r) = 0$. Now, $i(p) = 1 = i(p \to q)$ implies that $i(q) = 1$. Then $i(p) = 1$, $i(q \to r) = 0$ gives $i((p \to (q \to r)) = 0$. It contradicts $i((p \to (q \to r)) = 1$.

Hence, $i((p \to (q \to r)) \to ((p \to q) \to (p \to r))) = 1$, whatever be the interpretation i. Thus, $\models (p \to (q \to r)) \to ((p \to q) \to (p \to r))$. It is also satisfiable, since it has a model, e.g., the interpretation that assigns each of p, q, and r to 0.

(b) Suppose $i(((p \to q) \to (p \to r)) \wedge \neg(p \to (q \to r))) = 1$, for an interpretation i. Then, $i((p \to q) \to (p \to r)) = 1$ and $i(\neg(p \to (q \to r))) = 1$. The last one says that $i(p) = 1$, $i(q) = 1$, and $i(r) = 0$. Then, $i(p \to q) = 1$, $i(p \to r) = 0$; consequently, $i((p \to q) \to (p \to r)) = 0$, which is not possible.

Therefore, $i(((p \to q) \to (p \to r)) \wedge \neg(p \to (q \to r))) = 0$ for each interpretation i. That is, the proposition is unsatisfiable. It is also invalid, e.g., the interpretation that assigns each of p, q, and r to 0 falsifies the proposition.

In general, each valid proposition is satisfiable and each unsatisfiable proposition is invalid. Validity and unsatisfiability are dual concepts. Further, if i is an interpretation, then $i(w) = 1$ iff $i(\neg w) = 0$. This proves the following statement.

Theorem 3.3. *A proposition is valid if and only if its negation is unsatisfiable. A proposition is unsatisfiable if and only if its negation is valid.*

Exercises

1. Categorize the following propositions into valid, invalid, satisfiable, and unsatisfiable:

   1. $p \to (q \to p)$
   2. $(p \to (q \to r)) \to ((p \to q) \to (p \to r))$
   3. $p \wedge (p \to q) \to q$

4. $(\neg p \to \neg q) \to ((\neg p \to q) \to p)$
5. $p \vee q \leftrightarrow ((p \to q) \to q)$
6. $p \wedge q \leftrightarrow ((q \to p) \to q)$
7. $(\neg p \vee q) \to ((p \vee r) \leftrightarrow r)$
8. $(p \wedge q \leftrightarrow p) \to (p \vee q \leftrightarrow q)$
9. $(p \vee q \to p) \to (q \to p \wedge q)$
10. $(\neg p \to \neg q) \leftrightarrow (\neg p \vee q \to q)$
11. $(q \to p) \to p$
12. $((p \leftrightarrow q) \leftrightarrow r) \leftrightarrow ((p \leftrightarrow q) \wedge (q \leftrightarrow r))$
13. $((p \wedge q) \leftrightarrow p) \to q$
14. $\neg((\neg(p \wedge q) \wedge (p \leftrightarrow \bot)) \leftrightarrow (q \leftrightarrow \bot))$

3.3 Equivalences and Consequences

In the context of reasoning, it is important to determine whether saying this is the same as saying that. It is the notion of equivalence. Along with this, we must also specify the meaning of "follows from."

Propositions u and v are called **equivalent**, written as $u \equiv v$, iff each model of u is a model of v, and each model of v is also a model of u. We write $u \not\equiv v$ when u is not equivalent to v.

Example 3.6. *To determine whether $p \vee q \equiv (p \to q) \to q$, let i be an interpretation.*

If $i(p \vee q) = 0$, then $i(p) = i(q) = 0$; thus $i((p \to q) \to q) = 0$.

Conversely, if $i((p \to q) \to q) = 0$, then $i(p \to q) = 1$ and $i(q) = 0$. So, $i(p) = i(q) = 0$. Hence, $i(p \vee q) = 0$.

That is, $i(p \vee q) = 0$ iff $i((p \to q) \to q) = 0$. So, $i \models p \vee q$ iff $i \models (p \to q) \to q$. Therefore, $p \vee q \equiv (p \to q) \to q$.

To show that $p \to (q \to r) \not\equiv (p \to q) \to r$, consider the interpretation i with $i(p) = 0$, $i(q) = 1$, and $i(r) = 0$. Now, $i(q \to r) = 0$ and $i(p \to q) = 1$. Consequently, $i(p \to (q \to r)) = 1$ and $i((p \to q) \to r) = 0$. Therefore, $p \to (q \to r) \not\equiv (p \to q) \to r$.

A **consequence** is a formalization of an argument found in ordinary discourse. A typical argument goes as follows:

$$w_1, w_2, \ldots, w_n \quad \text{Therefore,} \quad w.$$

The propositions w_i may not be valid. The argument compels us to imagine a world where all of $w_1, w_2, \ldots, w_n$ become true. In any such world, it is to be checked whether w is also true. In order for the argument to be declared correct, all those interpretations which are simultaneously satisfying all the propositions $w_1, w_2, \ldots, w_n$ must also satisfy w.

Let Σ be a set of propositions, and w a proposition. An interpretation i is called a **model** of Σ, written $i \models \Sigma$, iff i is a model of each proposition in Σ. Every interpretation is taken as a model of the empty set $\emptyset$, as a convention.

The set Σ is called **satisfiable** iff Σ has a model. Σ **semantically entails** w, written $\Sigma \models w$, iff each model of Σ is a model of w. $\Sigma \models w$ is also read as "w follows from Σ" and also as "the consequence $\Sigma ! \vdash w$ is valid." For a consequence $\Sigma ! \vdash w$, the propositions in Σ are called the **premises** or **hypotheses**, and w is called the **conclusion**.

The abstract notion of a consequence $\Sigma ! \vdash w$ refers to an argument which may or may not be valid. Once $\Sigma ! \vdash w$ is valid, we write $\Sigma \models w$. For a finite set of premises Let $\Sigma = \{w_1, w_2, \ldots, w_n\}$. We write the consequence $\Sigma ! \vdash w$ as $w_1, w_2, \ldots, w_n ! \vdash w$ and $\Sigma \models w$ as $w_1, w_2, \ldots, w_n \models w$, without the braces.

Thus, $\Sigma \models w$ if and only if for each interpretation i, if i falsifies w, then i falsifies some proposition from Σ. Moreover, $\{w_1, w_2, \ldots, w_n\} \models w$ if and only if $w_1 \wedge w_2 \wedge \cdots \wedge w_n \models w$. It also follows that $x \equiv y$ if and only if $x \models y$ and $y \models x$.

Example 3.7. *Is the consequence $p \to q, \neg q \models \neg p$ valid?*

We try out each model of the set of premises and check whether it is also a model of the conclusion. So, let $i \models p \to q$ and $i \models \neg q$. We see that $i(q) = 0$. Since $i(p \to q) = 1$, we have $i(p) = 0$. The only model of all premises is the interpretation i with $i(p) = i(q) = 0$. Now, this i is also a model of $\neg p$. Therefore, the consequence is valid.

Thus, we write $p \to q, \neg q \models \neg p$.

Example 3.8. *Show that the following argument is correct (Stoll (1963)):*

> *If the band performs, then the hall will be full provided that the tickets are not too costly. However, if the band performs, the tickets will not be too costly. Therefore, if the band performs, then the hall will be full.*

We identify the simple declarative sentences in the above argument and build a vocabulary for translation:

p : *the band performs,* q : *the hall is (will be) full,* r : *tickets are not too costly.*

Then the hypotheses are the propositions $p \to (r \to q)$, $p \to r$, and the conclusion is $p \to q$. We check the following consequence for validity:

$$p \to (r \to q), \quad p \to r \quad ! \vdash \quad p \to q.$$

Since there are only three propositional variables, by the Relevance Lemma, there are $2^3 = 8$ *interpretations. These are given in the second, third, and fourth columns of the table below.*

Row No.	$\boldsymbol{p}$	$\boldsymbol{q}$	$\boldsymbol{r}$	$p \to r$	$r \to q$	$p \to (r \to q)$	$p \to q$
1	*0*	*0*	*0*	*1*	*1*	*1*	*1*
2	*1*	*0*	*0*	*0*	*1*	*1*	*0*
3	*0*	*1*	*0*	*1*	*1*	*1*	*1*
4	*1*	*1*	*0*	*0*	*1*	*1*	*1*
5	*0*	*0*	*1*	*1*	*0*	*1*	*1*
6	*1*	*0*	*1*	*1*	*0*	*0*	*0*
7	*0*	*1*	*1*	*1*	*1*	*1*	*1*
8	*1*	*1*	*1*	*1*	*1*	*1*	*1*

For the time being, do not read the column for $p \to q$ *in the above table. You must find out all (common) models of both* $p \to (r \to q)$ *and* $p \to r$*. They are in rows 1, 3, 5, 7, and 8. In order for the argument to be correct, you must check whether* $p \to q$ *is true (evaluated to 1) in all these rows. This is the case.*

Therefore, $p \to (r \to q), p \to r \models p \to q$*; the argument is correct.*

Example 3.9. *Let* Σ *be a set of propositions,* $x \in \Sigma$*, and* y *a proposition. Assume that* $\Sigma \models y$*. Show that* $\Sigma \setminus \{x\} \models y$*.*

Let i *be a model of* $\Sigma \setminus \{x\}$*. As* $\Sigma \models x$*, we have* $i \models x$*. So,* $i \models \Sigma$*. Since* $\Sigma \models y$*, we also have* $i \models y$*. Therefore,* $\Sigma \setminus \{x\} \models y$*.*

Example 3.10. *We show that for all propositions* x *and* y*,* $\{x, x \to y\} \models y$*.*

Of course, we should use the definition of $\models$*. So, let* i *be a model of* x *and also of* $x \to y$*. If* i *is not a model of* y*, then* $i(x) = 1$ *and* $i(y) = 0$*. Consequently,* $i(x \to y) = 0$*, which is not possible. Therefore,* i *is a model of* y*. Hence,* $\{x, x \to y\} \models y$*.*

3.4 Five results about PL

Theorem 3.4. *Let* u *and* v *be propositions. Then the following are true:*

1. $u \equiv v$ *iff* $\models u \leftrightarrow v$ *iff* ($u \models v$ *and* $v \models u$).
2. $\models u$ *iff* $u \equiv \top$ *iff* $\top \models u$ *iff* $\emptyset \models u$ *iff* $\neg u \equiv \bot$.
3. u *is unsatisfiable iff* $u \equiv \bot$ *iff* $u \models \bot$ *iff* $\neg u \equiv \top$.

Theorem 3.5 (Paradox of Material Implication). *A set of propositions* Σ *is unsatisfiable if and only if* $\Sigma \models w$ *for each proposition* w*.*

Proof. Suppose Σ has no model. Then it is not the case that there exists an interpretation which satisfies the propositions in Σ and falsifies any given proposition w. Thus, $\Sigma \models w$ never holds, which means $\Sigma \models w$.

Conversely, if $\Sigma \models w$ for each proposition w, then in particular, $\Sigma \models \perp$. Then Σ is unsatisfiable. $\square$

Theorem 3.6 (**M: Monotonicity**). *Let Σ and Γ be sets of propositions, $\Sigma \subseteq \Gamma$, and let w be a proposition.*

1. *If Σ is unsatisfiable, then Γ is unsatisfiable.*
2. *If $\Sigma \models w$, then $\Gamma \models w$.*

Proof. Let $i \models \Gamma$. This means i satisfies each proposition in Γ. If $y \in \Sigma$, then $y \in \Gamma$; so $i \models y$. Hence, $i \models \Sigma$. Thus, each model of Γ is also a model of Σ.

1. If Σ has no model, then Γ has no model.
2. Suppose $\Sigma \models w$. Let $i \models \Gamma$. Then $i \models \Sigma$. Since $\Sigma \models w$, $i \models w$. So, $\Gamma \models w$.

$\square$

Theorem 3.7 (**RA: Reductio ad Absurdum**). *Let Σ be a set of propositions, and let w be a proposition.*

1. *$\Sigma \models w$ iff $\Sigma \cup \{\neg w\}$ is unsatisfiable.*
2. *$\Sigma \models \neg w$ iff $\Sigma \cup \{w\}$ is unsatisfiable.*

Proof. **(1)** Let $\Sigma \models w$. Let i be any interpretation. If $i \models \Sigma$, then $i \models w$. So, $i \models \neg w$. Thus, $i \models \Sigma \cup \{\neg w\}$. If $i \models \Sigma$, then $i \models x$ for some $x \in \Sigma$; hence $i \models \Sigma \cup \{\neg w\}$. In any case, $i \models \Sigma \cup \{\neg w\}$. That is, $\Sigma \cup \{\neg w\}$ is unsatisfiable.

Conversely, let $\Sigma \cup \{\neg w\}$ be unsatisfiable. Let $i \models \Sigma$. Then $i \models \neg w$; that is, $i \models w$. Therefore, $\Sigma \models w$.

(2) Take w instead of $\neg w$, and $\neg w$ instead of w in the proof of (1). $\square$

Theorem 3.8 (**DT: Deduction Theorem**). *Let Σ be a set of propositions, and let x, y be propositions. Then,*

$$\Sigma \models x \to y \quad \textit{iff} \quad \Sigma \cup \{x\} \models y.$$

Proof. Suppose $\Sigma \models x \to y$. Let $i \models \Sigma \cup \{x\}$. Then $i \models \Sigma$, and $i \models x$. If $i \models y$, then $i(x) = 1$ and $i(y) = 0$. That is, $i(x \to y) = 0$; so $i \models x \to y$. This contradicts $\Sigma \models x \to y$. Thus, $i \models y$. Therefore, $\Sigma \cup \{x\} \models y$.

Conversely, suppose $\Sigma \cup \{x\} \models y$. Let $i \models \Sigma$. If $i \models x \to y$, then $i(x) = 1$ and $i(y) = 0$. That is, $i \models \Sigma \cup \{x\}$ and $i \models y$; this contradicts $\Sigma \cup \{x\} \models y$. So, $i \models x \to y$. Therefore, $\Sigma \models x \to y$. $\square$

Example 3.11. *We use Monotonicity, Reductio ad Absurdum and/or Deduction Theorem to justify*

$$p \to (r \to q), \quad p \to r \quad ! \vdash \quad p \to q.$$

Solution 1. *Due to the Deduction Theorem (DT),*

$$\{p \to (r \to q), p \to r\} \models p \to q \quad \textit{iff} \quad \{p \to r, p \to (r \to q), p\} \models q.$$

We show the latter. Let i be an interpretation such that $i(p \to r) = 1$, $i(p \to (r \to q)) = 1$, and $i(p) = 1$. If $i(r) = 0$, then it contradicts $i(p \to r) = 1$. So, $i(r) = 1$. Similarly, from the second and third premises, we have $i(r \to q) = 1$, and hence $i(q) = 1$. Therefore, the consequence $p \to q$ is valid.

Solution 2. *Due to Reductio ad Absurdum (RA), we show that $\{p \to r, p \to (r \to q), p, \neg q\}$ is unsatisfiable. If the set is satisfiable, then there exists an interpretation i such that $i(p \to r) = i(p \to (r \to q)) = i(p) = i(\neg q) = 1$. Now, $i(p) = 1$ and $i(p \to r) = 1$ force $i(r) = 1$. Then, $i(q) = 0$ gives $i(r \to q) = 0$, and it follows that $i(p \to (r \to q)) = 0$, which contradicts $i(p \to (r \to q)) = 1$. Therefore, there exists no such i, and the set $\{p \to r, p \to (r \to q), p, \neg q\}$ is unsatisfiable.*

Solution 3. *We know that $\{p \to r, p\} \models r$ and $\{p \to (r \to q), p\} \models r \to q$. Again, $\{r, r \to q\} \models q$. Using monotonicity, we have $\{p \to r, p \to (r \to q), p\} \models q$.*

Exercises

1. Determine whether the following consequences are valid:

   (a) $\neg(r \wedge \neg\neg q) \models (\neg q \vee \neg r)$

   (b) $p \vee \neg q, p \to \neg r \models q \to \neg r$

   (c) $p \vee q \to r \wedge s, t \wedge s \to u \models p \to u$

   (d) $p \vee q \to r \wedge s, s \vee t \to u, p \vee \neg u \models p \to (q \to r)$

   (e) $p \to q \wedge r, q \to s, d \to t \wedge u, q \to p \wedge \neg t \models q \to t$

   (f) $p, \neg r \to \neg p, (p \to q) \wedge (r \to s), (s \to u) \wedge (q \to t), s \to \neg t \models \perp$

2. Translate to PL and then decide the validity of the following argument:

The Indian economy will decline unless cashless transaction is accepted by the public. If the Indian economy declines, then Nepal will be more dependent on China. The public will not accept cashless transaction. So, Nepal will be more dependent on China.