5 Propositional logic

Using 'Systems construction and analysis: a mathematical and logical framework: Norman Fenton and Gillian Hill', selected material from sections 3.2, 3.3 and 3.4.

This point of this section is to set up a framework for reasoning by manipulating statements, in particular special kinds of statements which are called propositions.

NB we Can use the IFAD tool here too! (check Fitzgerald for notation)

Definition 5.1 *A statement is a sentence which is not a question or a command. A proposition is a statement which is either true or false, but not both. If a proposition is true we say it has* truth-value $True$. *Otherwise it has truth-value* $False$.

In programming, we usually use 1 to represent $True$ and 0 to represent $False$.

Examples 5.2

1. *Some true propositions:*
   (a) $2 + 3 = 5$.
   (b) *Bombay is in India.*
   (c) *Spurs won the FA cup in 1991.*
2. *Some false propositions.*
   (a) $10 > 87$.
   (b) *Newcastle is south of Leeds.*
3. *Propositions whose 'truth-value' depends on one or more variables - open statements.*
   (a) *n is an even integer. (depends on n).*
   (b) *She is taller than me. (Depends on 'she' and 'me').*
   (c) *$x + y > 7$. (Depends on x and y).*
4. *The following are NOT propositions or statements.*
   (a) *Well done!*
   (b) *How did you manage that?*
5. *The following sentence*

   This sentence is false.

   is a statement, but not a proposition. It cannot be true, and it cannot be false. It is in fact a paradox.

Definition 5.3 *We make compound propositions by glueing together simple propositions using the* logical connectives *'and', 'or' and 'not'.*

So from the two simple propositions

Roses are always red.
Violets are always blue.

we can make various compound propositions

Roses are always red and violets are always blue.
Roses are always red or violets are always blue.
Roses are not always red.

The truth-value of a compound proposition is determined both by the truth-value of the constituent propositions, and by the way in which they are connected together. We'll learn how to manipulate compound propositions; the language and laws of this manipulation are known as the propositional calculus.

In order that our explanation of this doesn't get too wordy, let's start using symbols rather than the English language.

Definition 5.4 *We'll use letters like p, q, r to stand for propositions (we call them propositional variables), $\wedge$ to mean 'and', $\vee$ to mean 'or', and $\neg$ to mean 'not'. We also call $\wedge$ 'conjunction' and $\vee$ 'disjunction'. We use the symbol T to represent a proposition which is true, and F to represent a proposition which is false. We call an expression representing a compound proposition which we build out of these symbols a propositional formula*

The following rules define the way in which 'and', 'or' and 'not' should be interpretated (not necessarily as you might expect in English where the meanings are not always consistent).

5.5 *Interpretation of the logical connectives $\wedge$, $\vee$ and $\neg$.*

Given propositional formulae f, g, $f \wedge g$ is defined to be true if both f is true and g is true, and false otherwise.

$f \vee g$ is defined to be true if f is true or g is true, including when both are true, and false otherwise.

$\neg f$ is defined to be true when f is false, and false when f is true.

Note the meaning of $\vee$ as 'or' is inclusive, as in 'Would you like milk or sugar with your tea?' rather than exclusive as in 'Would you like tea or coffee?'. In the first case it is assumed that you might like both milk and sugar, but in the second not that you would take both drinks. Only one drink is on offer.

Definition 5.6 (Logical equivalence) *Two propositional formulae, f, g are said to be logically equivalent if the formula f is true whenever the formula g is true, and vice versa. We write $f \equiv g$.*

Notice for instance that, for any formula f,

$$f \vee \neg f \equiv T$$

Definition 5.7 (Truth tables.) *We use truth tables to test for logical equivalence between propositional formulae in almost exactly the same way that we test for equality between sets using Venn diagrams. The truth table is simply a table of rows and columns, with one column for each of the variables involved in the formula, usually columns for some of the subformulae if the formula is complicated, and the rightmost column for the formula itself. One row of the table then corresponds to each of the possible combinations of values for the variables involved, and from these, the truth-value of the formula itself is calculated, and inserted into the final column. Two formulae are logically equivalent if and only if their truth tables are the same.*

Truth tables for $p \wedge q$, $p \vee q$, $\neg p$ given variables p, q.

A convenient way of displaying truth tables for complicated formulae:-

When we have a complicated formula like $(p \wedge q) \vee (p \wedge r)$ the most simple-minded way to draw the truth table is with columns headed, $p, q, r, p \wedge q, p \wedge r$ and then finally $(p \wedge q) \vee (p \wedge r)$, but in fact it can be a lot clearer (from many points of view) if we simply have 3 narrow columns for the variables p, q, r, then simply write out the formula $(p \wedge q) \vee (p \wedge r)$ and divide the space below it into 3. The space below the first $\wedge$ then refers to the subformula $p \wedge q$, the space below the second wedge to the subformula $p \wedge r$, and the space below the $\vee$ to the entire formula $(p \wedge q) \vee (p \wedge r)$.

We shall use this representation.

Example. $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$.

This may look familiar. It should do. We've had almost the identical formulae for sets, with intersection and union. In fact

5.8 *we can verify all the following, using truth tables:-*

For propositional formulae f, g, h

1. Commutative laws.

$$f \wedge g \equiv g \wedge f \quad f \vee g \equiv g \vee f$$

2. Associative laws.

$$f \wedge (g \wedge h) \equiv (f \wedge g) \wedge h \quad f \vee (g \vee h) \equiv (f \vee g) \vee h$$

3. Distributive laws.

$$f \wedge (g \vee h) \equiv (f \wedge g) \vee (f \wedge h) \quad f \vee (g \wedge h) \equiv (f \vee g) \wedge (f \vee h)$$

4.

$$f \wedge T \equiv f \quad f \vee F \equiv f$$

5.

$$f \wedge \neg f \equiv F \quad f \vee \neg f \equiv T$$

6. De Morgan's laws.

$$\neg(f \wedge g) \equiv (\neg f) \vee (\neg g) \quad \neg(f \vee g) \equiv (\neg f) \wedge (\neg g)$$

7. Idempotent laws.

$$f \wedge f \equiv f \quad f \vee f \equiv f$$

We've seen rules very much like these before, in the section on set theory. It's clear that there's a connection between set theory and propositional calculus. It's like two different languages. Sets translate to propositional formulae, $\cap$ to $\wedge$, $\cup$ to $\vee$, $\mathcal{U}$ to T and $\emptyset$ to F.

There's a reason for this, which may become clearer later in the course.

Of course we see the duality again, that we saw with sets, with $\wedge$ being paired with $\vee$ and T with F.

More examples.

Definition 5.9 (Tautology, contradiction) *A propositional formula which is always true is called a tautology. A propositional formula which is always false is called a contradiction.*

Example:

Show that $\neg(\neg p \cap q) \vee (\neg p)$ is a tautology.

Construct the truth table. All values of p, q and r give a T in the right hand column.

5.10 *Translation of English sentences into language of propositional logic*

This isn't always easy, or even possible to do, but we need to get used to the idea of breaking down a proposition into its consituent parts. We have to think very hard about what the sentence really means.

Examples:

'Sue is a beautiful woman' can be broken down as 'Sue is beautiful and Sue is a woman'.

'Mike is a wonderful guitarist' is already as simple as it can get, because noone said that Mike was wonderful more generally.

'The work is easy to do but not satisfying' can be broken down as 'The work is easy to do' and not 'The work is satisfying'.

More examples. This section is important.

We can build further propositional formulae with the introduction of additional connectives.

Definition 5.11 *The connectives $\Rightarrow$, $\Leftarrow$ and $\iff$, by specifying their truth tables.*

We read $f \Rightarrow g$ as 'f implies g', or 'if f then g '. Similarly we read $f \Leftarrow g$ as 'f is implied by g' and $f \iff g$ as 'f if and only if g', and the truth tables fit with the meaning of the English, in that $f \Rightarrow g$ is true provided that whenever f is true then g is also true. Maybe the English meaning doesn't really suggest that $f \Rightarrow g$ is true whenever f is false, irrespective of the value of g. But this is true in logic. Hence the proposition 'If pigs can fly then the pope is a protestant' is true.

5.12 *More translation of English sentences into language of propositional logic*

More examples.

We see (by studying the truth tables) that

$$
\begin{aligned}
f \Rightarrow g &\equiv (\neg f \vee g) \\
f \Leftarrow g &\equiv g \Rightarrow f \\
f \iff g &\equiv (f \Rightarrow g) \wedge (g \Rightarrow f)
\end{aligned}
$$

Lots more examples with truth tables.