Contents

1 Propositional logic

1.1 Introduction

1.1.1 True and False

We start off with two statements:

- True - T
- False - F

1.1.2 Propositional variables

We can represent T or F using a symbol:

θ

1.2 Operators

1.2.1 Unary operators

A unary operator takes one input and returns another.

Only negation, $\neg$ is of interest.

The following statements are equivalent:

- T
- $\neg F$

1.2.2 Binary operators

A binary operator takes an additional input.

- If then - $\theta \to \gamma$
- Then if - $\theta \leftarrow \gamma$
- Iff - $\theta \leftrightarrow \gamma$
- And / Conjunction - $\theta \land \gamma$
- Or / Disjunction - $\theta \lor \gamma$

1.2.3 Brackets

Operators can be shown together, with brackets. For example:

$(\alpha \vee \beta) \wedge \gamma$

Is not the same as:

$\alpha \vee (\beta \wedge \gamma)$

1.2.4 Clauses and horn clauses

A clause is a disjunction of atomic formulae.

$A \vee \neg B \vee C$

This can be written in implicative form.

$(A \vee \neg B) \vee C$

$\neg(A \vee \neg B) \to C$

$(\neg A \wedge B) \to C$

A horn clause is a clause where there is at most one positive literal. This means the implicative takes the form.

$(A \wedge B \wedge C) \to X$.

1.2.5 Atomic formulae

Atomic formulae are those without operators taking more than one input.

Literals, and negative literals, are types of atomic formula.

A literal is a formula with no operators.

θ

These are also known as positive literals.

Negative literals are the negation of a literal.

$\neg\theta$

1.2.6 Well-formed formulae

A well-formed formula is one which can be given a truth value.

The following is not a well-formed formula:

$\theta\wedge$

1.2.7 Interpretations

An interpretation assigns meaning to propositional variables in a formula.

For example an interpretation of the formula $\theta \lor \gamma$ assigns values to each of θ and γ.

1.2.8 Satifisable

A formula is satisfisable if there is some interpretation where it is true.

For example θ is satisfisable but $\theta \land \neg\theta$ is not.

1.2.9 Tautology

A formula is a tautology if it is true in all interpretations.

Examples of tautologies include:

- $\theta \lor \neg\theta$

1.3 Semantic consequence

1.3.1 Semantic consequence

A formula, A, semantically implies another, B, if for every interpretation of A, B is true.

We show this with:

$A \vDash B$

Formula B is satisfisable if there is some A where this is true.

For example: $A \land B \vDash A$

Formula B is a tautology if this is true for any A. We can also write this as $\vDash B$.

1.3.2 Logical equivalence

If $A \vDash B$ and $B \vDash A$ we say that A and B are logically equivalent.

This is shown as $A \Leftrightarrow B$.

1.3.3 How many unique operators are there?

An arbitrary operator takes n inputs are returns T or F.

With 0 inputs there is one posible permutation. For every additional input the number of possible permutations doubles. Therefore there are 2^n possible permutations.

For the operator with one permutation there are two operators. For every additional permutation the number of operator doubles. Therefore there are $2^{(2^n)}$ possible operations.

With 0 inputs, we need 2 different operators to cover all outputs. For 1 input we need 4 and for 2 inputs we need 16.

1.3.4 We don't need 0-ary operators

There are two unique 0-ary operators. One always returns T and the other always returns F. These are already described.

1.3.5 We need one unary operator

For the operators with 1 input we have:

- one which always returns T
- one which always returns F
- one which always returns the same as the input
- one which returns the opposite of the input

It is this last one, negation, shown as $\neg$ and is of most interest.

1.3.6 We can use a subset of binary operators

The full list of binary operators are included below.

Of these, the first two are 0-ary operators, and so are not needed. The next four are unary operators, and so are not needed.

The non-implications can be rewritten using negation.

1.3.7 Brackets replace the need for n-ary operators

N-ary operators contain 3 or more inputs.

N-ary operators can be defined in terms of binary operators.

As an example if we want an operator to return positive if all inputs are true, we can use:

$(\theta \land \gamma) \land \beta$

1.3.8 De Morgan's Laws

- $\neg(A \lor B) \Leftrightarrow (\neg A \land \neg B)$
- $\neg(A \land B) \Leftrightarrow (\neg A \lor \neg B)$

This expresses the duality of normal form.

Duality is the principle that binary operators have inverses, and when they are swapped with their inverse, the truth value of the statement is unaffected.

1.3.9 Normal form

This is where a formula is shown using only:

- And / Conjunction- $\land$
- Or / Disjunction - $\lor$
- Negation - $\neg$

The conjunctive normal form (CNF) is where a formula is converted to a normal form with the following layout:

$a \land b \land c \land d$

These letters can represent complex sub-formulae, in normal form.

Statements in this form are easier to evaluate, as each subformula can be evaluated separately. The statement is true only if all formulaes within are also true.

The disjunctive normal form (DNF) is similar for $\lor$.

$a \lor b \lor c \lor d$

1.3.10 Properties of the normal form

The normal binary operators are commutitive - $A \land B \Leftrightarrow B \land A$ and $A \lor B \Leftrightarrow B \lor A$

Both binary operators are associative - $(A \land B) \land C \Leftrightarrow A \land (B \land C)$ and $(A \lor B) \lor C \Leftrightarrow A \lor (B \lor C)$

Negation is complementary.

$A \land \neg A \Leftrightarrow F$

$A \lor \neg A \Leftrightarrow T$

Normal binary operators are absorbative.

$A \land (A \lor B) \Leftrightarrow A$ $A \lor (A \land B) \Leftrightarrow A$

Identity.

$A \land T \Leftrightarrow A$

$A \lor F \Leftrightarrow A$

Distributivity.

$A \land (B \lor C) \Leftrightarrow (A \land B) \lor (A \land C)$

$A \lor (B \land C) \Leftrightarrow (A \lor B) \land (A \lor C)$

1.4 Inference

1.4.1 Substitution

If we have a tautology, then we can substitute the formula of any propositional variable with any formula to arrive at any other tautology.

For example, we know that $\theta \lor \neg\theta$ is a tautology. This means that an arbitrary formula for θ is also a tautology.

An example is $(\gamma \land \alpha) \lor \neg(\gamma \land \alpha)$, which we know is a tautology, without having to examine each variable.

1.4.2 Syntactic consequence

Let us call the first formula A and the second B. We can then say:

$A \vdash B$

This says that: if A is true, then we can deduce that B is true using steps such as substitution.

1.4.3 Modus Ponens

Modus Ponens is a deduction rule. This allows us to use stpes other than substitution to derive new tautologies.

If A implies B, and A is true, then B is also true.

$(\theta \to \gamma) \land \theta \Rightarrow \gamma$

That is, if we can show that the following are true:

$\theta \to \gamma$

θ

We can infer that the following is also true:

γ

1.4.4 Inference with horn clauses

If the horn clause is true, and so is the normal form part, then X is also true.

As all inference with horn clauses uses Modus Ponens, it is sound.

Inference with horn clauses is also complete.

1.4.5 Theory

Results derived from substitution or induction are called theorems. Theorems often divided into:

- Theorems - important results
- Lemmas - results used for later theorems
- Corollaries - readily deduced from a theorem

We take a set of axioms, as true, and a deduction rule which enables us to derive additional formulae, or theorems. The collection of axioms and theorems is known as the theory.

1.4.6 Principle of explosion

If axioms contradict each other then it is possible to derive anything. That is:

$P \land \neg P \vdash Q$

We can prove this. If P and $\neg P$ are true, then the following is also true:

$P \lor Q$

We can then use $P \lor Q$ and $\neg P$ to imply Q.

This works for any proposition Q, including $\neg Q$.

As we can derive Q and $\neg Q$, our axioms are not consistent.

1.4.7 Resolution rule

1.4.7.1 Proof by resolution

If we have a string of or statements, $A \vee B \vee C$, and another which contains the completement of one element $X \vee \neg B \vee Y$, we can infer:

$A \vee C \vee X \vee Y$

If the second statement has only one formula, then we have:

$A \vee B \vee C$ and $\neg B$ implying $A \vee C$

1.5 Axioms for propositional logic

1.5.1 Motivation for axioms for propositional logic

We discussed in the previous section the ability to derive new tautologies from others using substitution and Modus Ponens.

We now aim to identify a group of axioms from which all tautologies can be derived.

1.5.2 The axioms

The first is known as "Simplification". In words, this is "if it is cloudy, then if it is a Tuesday it is also cloudy."

$\theta \to (\gamma \to \theta)$

The second is called "Frege".

$(\alpha \to (\beta \to \gamma)) \to ((\alpha \to \beta) \to (\alpha \to \gamma))$

The third is "Transposition". Consider the statement "If there are no clouds in the sky, it is not raining." If this is true then it is also true that "If it is raining there are clouds in the sky."

$(\neg\theta \to \neg\gamma) \to (\gamma \to \theta)$

1.5.3 Independence of axioms

These axioms are independent. That is, if you take one away, you cannot derive it from the others.

These axioms are also effective. One could define all true formulae as axioms, however this is not effective.

1.5.4 Soundness of axioms

Soundness implies that all theories are true.

$T \vdash A \Rightarrow T \vDash A$

These axioms and the deduction rule are sound. We know that the axioms are tautologies, and we know that the inference rule is valid.

As the axioms are sound, the theories are consistent. That is, it is not possible for both θ and $\neg\theta$ to be theories.

1.5.5 Completeness of axioms

Completeness implies that all true formulae are theories.

$T \vDash A \Rightarrow T \vdash A$

1.5.6 Axioms and definitions

A definition is a conservative extension of the language. A definition statement, for example that a new symbol Z is always evaluated as false allows us to make additional statements, but it does not allow us to make additional statements in the original language.

An axiom allows us to generate additional statements in the original language, a definition does not.