1
Introduction

In this chapter we introduce logical reasoning and the idea of mechanizing it, touching briefly on important historical developments. We lay the groundwork for what follows by discussing some of the most fundamental ideas in logic as well as illustrating how symbolic methods can be implemented on a computer.

1.1 What is logical reasoning?

There are many reasons for believing that something is true. It may seem obvious or at least immediately plausible, we may have been told it by our parents, or it may be strikingly consistent with the outcome of relevant scientific experiments. Though often reliable, such methods of judgement are not infallible, having been used, respectively, to persuade people that the Earth is flat, that Santa Claus exists, and that atoms cannot be subdivided into smaller particles.

What distinguishes *logical* reasoning is that it attempts to avoid any unjustified assumptions and confine itself to inferences that are infallible and beyond reasonable dispute. To avoid making any unwarranted assumptions, logical reasoning cannot rely on any special properties of the objects or concepts being reasoned about. This means that logical reasoning must abstract away from all such special features and be equally valid when applied in other domains. Arguments are accepted as logical based on their conformance to a general *form* rather than because of the specific *content* they treat. For instance, compare this traditional example:

All men are mortal
Socrates is a man
Therefore Socrates is mortal

with the following reasoning drawn from mathematics:

All positive integers are the sum of four integer squares
15 is a positive integer
Therefore 15 is the sum of four integer squares

These two arguments are both correct, and both share a common pattern:

All X are Y
a is X
Therefore a is Y

This pattern of inference is logically valid, since its validity does not depend on the content: the meanings of 'positive integer', 'mortal' etc. are irrelevant. We can substitute anything we like for these X, Y and a, provided we respect grammatical categories, and the statement is still valid. By contrast, consider the following reasoning:

All Athenians are Greek
Socrates is an Athenian
Therefore Socrates is mortal

Even though the conclusion is perfectly true, this is not logically valid, because it does depend on the content of the terms involved. Other arguments with the same superficial form may well be false, e.g.

All Athenians are Greek
Socrates is an Athenian
Therefore Socrates is beardless

The first argument can, however, be turned into a logically valid one by making explicit a hidden assumption 'all Greeks are mortal'. Now the argument is an instance of the general logically valid form:

All G are M
All A are G
s is A
Therefore s is M

At first sight, this forensic analysis of reasoning may not seem very impressive. Logically valid reasoning never tells us anything fundamentally new about the world – as Wittgenstein (1922) says, 'I know nothing about the weather when I know that it is either raining or not raining'. In other words, if we *do* learn something new about the world from a chain of reasoning, it must contain a step that is *not* purely logical. Russell, quoted in Schilpp (1944) says:

Hegel, who deduced from pure logic the whole nature of the world, including the non-existence of asteroids, was only enabled to do so by his logical incompetence.†

But logical analysis can bring out clearly the necessary relationships *between* facts about the real world and show just where possibly unwarranted assumptions enter into them. For example, from 'if it has just rained, the ground is wet' it follows logically that 'if the ground is not wet, it has not just rained'. This is an instance of a general principle called *contraposition*: from 'if P then Q' it follows that 'if not Q then not P'. However, passing from 'if P then Q' to 'if Q then P' is *not* valid in general, and we see in this case that we cannot deduce 'if the ground is wet, it has just rained', because it might have become wet through a burst pipe or device for irrigation.

Such examples may be, as Locke (1689) put it, 'trifling', but elementary logical fallacies of this kind are often encountered. More substantially, deductions in mathematics are very far from trifling, but have preoccupied and often defeated some of the greatest intellects in human history. Enormously lengthy and complex chains of logical deduction can lead from simple and apparently indubitable assumptions to sophisticated and unintuitive theorems, as Hobbes memorably discovered (Aubrey 1898):

> Being in a Gentleman's Library, Euclid's Elements lay open, and 'twas the 47 *El. libri* 1 [Pythagoras's Theorem]. He read the proposition. *By G—*, sayd he (he would now and then sweare an emphaticall Oath by way of emphasis) *this is impossible!* So he reads the Demonstration of it, which referred him back to such a Proposition; which proposition he read. That referred him back to another, which he also read. *Et sic deinceps* [and so on] that at last he was demonstratively convinced of that trueth. This made him in love with Geometry.

Indeed, Euclid's seminal work *Elements of Geometry* established a particular style of reasoning that, further refined, forms the backbone of present-day mathematics. This style consists in asserting a small number of *axioms*, presumably with mathematical content, and deducing consequences from them using *purely logical reasoning*.‡ Euclid himself didn't quite achieve a complete separation of logical and non-logical, but his work was finally perfected by Hilbert (1899) and Tarski (1959), who made explicit some assumptions such as 'Pasch's axiom'.

† To be fair to Hegel, the word *logic* was often used in a broader sense until quite recently, and what we consider logic would have been called specifically *deductive logic*, as distinct from *inductive logic*, the drawing of conclusions from observed data as in the physical sciences.

‡ Arguably this approach is foreshadowed in the Socratic method, as reported by Plato. Socrates would win arguments by leading his hapless interlocutors from their views through chains of apparently inevitable consequences. When absurd consequences were derived, the initial position was rendered untenable. For this method to have its uncanny force, there must be no doubt at all over the steps, and no hidden assumptions must be sneaked in.

1.2 Calculemus!

'Reasoning is reckoning'. In the epigraph of this book we quoted Hobbes on the similarity between logical reasoning and numerical calculation. While Hobbes deserves credit for making this better known, the idea wasn't new even in 1651.† Indeed the Greek word *logos*, used by Plato and Aristotle to mean reason or logical thought, can also in other contexts mean computation or reckoning. When the works of the ancient Greek philosophers became well known in medieval Europe, *logos* was usually translated into *ratio*, the Latin word for reckoning (hence the English words rational, ratiocination, etc.). Even in current English, one sometimes hears 'I reckon that ... ', where 'reckon' refers to some kind of reasoning rather than literally to computation.

However, the connection between reasoning and reckoning remained little more than a suggestive slogan until the work of Gottfried Wilhelm von Leibniz (1646–1716). Leibniz believed that a system for reasoning by calculation must contain two essential components:

- a universal language (*characteristica universalis*) in which anything can be expressed;
- a calculus of reasoning (*calculus ratiocinator*) for deciding the truth of assertions expressed in the *characteristica*.

Leibniz dreamed of a time when disputants unable to agree would not waste much time in futile argument, but would instead translate their disagreement into the *characteristica* and say to each other '*calculemus*' (let us calculate). He may even have entertained the idea of having a machine do the calculations. By this time various mechanical calculating devices had been designed and constructed, and Leibniz himself in 1671 designed a machine capable of multiplying, remarking:

> It is unworthy of excellent men to lose hours like slaves in the labour of calculations which could safely be relegated to anyone else if machines were used.

So Leibniz foresaw the essential components that make automated reasoning possible: a language for expressing ideas precisely, rules of calculation for manipulating ideas in the language, and the mechanization of such calculation. Leibniz's concrete accomplishments in bringing these ideas to fruition were limited, and remained little-known until recently. But though his work had limited direct influence on technical developments, his dream still resonates today.

† The Epicurian philosopher Philodemus, writing in the first century B.C., introduced the term *logisticos* (λογιστικός) to describe logic as the science of calculation.

1.3 Symbolism

Leibniz was right to draw attention to the essential first step of developing an appropriate language. But he was far too ambitious in wanting to express all aspects of human thought. Eventual progress came rather by extending the scope of the symbolic notations already used in mathematics. As an example of this notation, we would nowadays write '$x^2 \leq y + z$' rather than 'x multiplied by itself is less than or equal to the sum of y and z'. Over time, more and more of mathematics has come to be expressed in formal symbolic notation, replacing natural language renderings. Several sound reasons can be identified.

First, a well-chosen symbolic form is usually shorter, less cluttered with irrelevancies, and helps to express ideas more briefly and intuitively (at least to cognoscenti). For example Leibniz's own notation for differentiation, $\mathrm{d}y/\mathrm{d}x$, nicely captures the idea of a ratio of small differences, and makes theorems like the chain rule $\mathrm{d}y/\mathrm{d}x = \mathrm{d}y/\mathrm{d}u \cdot \mathrm{d}u/\mathrm{d}x$ look plausible based on the analogy with ordinary algebra.

Second, using a more stylized form of expression can avoid some of the ambiguities of everyday language, and hence communicate meaning with more precision. Doubts over the exact meanings of words are common in many areas, particularly law.† Mathematics is not immune from similar basic disagreements over exactly what a theorem says or what its conditions of validity are, and the consensus on such points can change over time (Lakatos 1976; Lakatos 1980).

Finally, and perhaps most importantly, a well-chosen symbolic notation can contribute to making mathematical reasoning itself easier. A simple but outstanding example is the 'positional' representation of numbers, where a number is represented by a sequence of numerals each implicitly multiplied by a certain power of a 'base'. In decimal the base is 10 and we understand the string of digits '179' to mean:

$$179 = 1 \times 10^2 + 7 \times 10^1 + 9 \times 10^0.$$

In binary (currently used by most digital computers) the base is 2 and the same number is represented by the string 10110011:

$$10110011 = 1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0.$$

† For example 'Since the object of ss 423 and 425 of the Insolvency Act 1986 was to remedy the avoidance of debts, the word 'and' between paragraphs (a) and (b) of s 423(2) must be read conjunctively and not disjunctively.' (Case Summaries, *Independent* newspaper, 27th December 1993.)

These positional systems make it very easy to perform important operations on numbers like comparing, adding and multiplying; by contrast, the system of Roman numerals requires more involved algorithms, though there is evidence that many Romans were adept at such calculations (Maher and Makowski 2001). For example, we are normally taught in school to add decimal numbers digit-by-digit from the right, propagating a carry leftwards by adding one in the next column. Once it becomes second nature to follow the rules, we can, and often do, forget about the underlying meaning of these sequences of numerals. Similarly, we might transform an equation $x-3 = 5-x$ into $x = 3+5-x$ and then to $2x = 5+3$ without pausing each time to think about *why* these rules about moving things from one side of the equation to the other are valid. As Whitehead (1919) says, symbolism and formal rules of manipulation:

> [...] have invariably been introduced to make things easy. [...] by the aid of symbolism, we can make transitions in reasoning almost mechanically by the eye, which otherwise would call into play the higher faculties of the brain. [...] Civilisation advances by extending the number of important operations which can be performed without thinking about them.

Indeed, such formal rules can be followed reliably by people who do *not* understand the underlying justification, or by computers. After all, computers are expressly designed to follow formal rules (programs) quickly and reliably. They do so without regard to the underlying justification, and will faithfully follow even erroneous sets of rules (programs with 'bugs').

1.4 Boole's algebra of logic

The word *algebra* is derived from the Arabic 'al-jabr', and was first used in the ninth century by Mohammed al-Khwarizmi (ca. 780–850), whose name lies at the root of the word 'algorithm'. The term 'al-jabr' literally means 'reunion', but al-Khwarizmi used it to describe in particular his method of solving equations by collecting together ('reuniting') like terms, e.g. passing from $x + 4 = 6 - x$ to $2x = 6 - 4$ and so to the solution $x = 1$.† Over the following centuries, through the European renaissance, algebra continued to mean, essentially, rules of manipulation for solving equations.

During the nineteenth century, algebra in the traditional sense reached its limits. One of the central preoccupations had been the solving of equations of higher and higher degree, but Niels Henrik Abel (1802–1829) proved in

† The first use of the phrase in Europe was nothing to do with mathematics, but rather the appellation 'algebristas' for Spanish barbers, who also set ('reunited') broken bones as a sideline to their main business.

1824 that there is no general way of solving polynomial equations of degree 5 and above using the 'radical' expressions that had worked for lower degrees. Yet at the same time the scope of algebra expanded and it became generalized. Traditionally, variables had stood for real numbers, usually unknown numbers to be determined. However, it soon became standard practice to apply all the usual rules of algebraic manipulation to the 'imaginary' quantity i assuming the formal property $i^2 = -1$. Though this procedure went for a long time without any rigorous justification, it was effective.

Algebraic methods were even applied to objects that were not numbers in the usual sense, such as matrices and Hamilton's 'quaternions', even at the cost of abandoning the usual 'commutative law' of multiplication $xy = yx$. Gradually, it was understood that the underlying interpretation of the symbols could be ignored, provided it was established once and for all that the rules of manipulation used are all valid under that interpretation. The state of affairs was described clear-sightedly by George Boole (1815–1864).

> They who are acquainted with the present state of the theory of Symbolic Algebra, are aware, that the validity of the processes of analysis does not depend upon the interpretation of the symbols which are employed, but solely on their laws of combination. Every system of interpretation which does not affect the truth of the relations supposed, is equally admissible, and it is true that the same process may, under one scheme of interpretation, represent the solution of a question on the properties of numbers, under another, that of a geometrical problem, and under a third, that of a problem of dynamics or optics. (Boole 1847)

Boole went on to observe that nevertheless, by historical or cultural accident, all algebra at the time involved objects that were in some sense quantitative. He introduced instead an algebra whose objects were to be interpreted as 'truth-values' of true or false, and where variables represent *propositions*.[†] By a proposition, we mean an assertion that makes a declaration of fact and so may meaningfully be considered either true or false. For example, '$1 < 2$', 'all men are mortal', 'the moon is made of cheese' and 'there are infinitely many prime numbers p such that $p + 2$ is also prime' are all propositions, and according to our present state of knowledge, the first two are true, the third false and the truth-value of the fourth is unknown (this is the 'twin primes conjecture', a famous open problem in mathematics).

We are familiar with applying to numbers various arithmetic operations like unary 'minus' (negation) and binary 'times' (multiplication) and 'plus' (addition). In an exactly analogous way, we can combine truth-values using

† Actually Boole gave two different but related interpretations: an 'algebra of classes' and an 'algebra of propositions'; we'll focus on the latter.

so-called *logical connectives*, such as unary 'not' (logical negation or complement) and binary 'and' (conjunction) and 'or' (disjunction).† And we can use letters to stand for arbitrary *propositions* instead of *numbers* when we write down expressions. Boole emphasized the connection with ordinary arithmetic in the precise formulation of his system and in the use of the familiar algebraic notation for many logical constants and connectives:

0	false
1	true
pq	p and q
$p+q$	p or q

On this interpretation, many of the familiar algebraic laws still hold. For example, 'p and q' always has the same truth-value as 'q and p', so we can assume the commutative law $pq = qp$. Similarly, since 0 is false, '0 and p' is false whatever p may be, i.e. $0p = 0$. But the Boolean algebra of propositions satisfies additional laws that have no counterpart in arithmetic, notably the law $p^2 = p$, where p^2 abbreviates pp.

In everyday English, the word 'or' is ambiguous. The complex proposition 'p or q' may be interpreted either inclusively (p or q or both) or exclusively (p or q but not both).‡ In everyday usage it is often implicit that the two cases are mutually exclusive (e.g. 'I'll do it tomorrow or the day after'). Boole's original system restricted the algebra so that $p + q$ only made sense if $pq = 0$, rather as in ordinary algebra x/y only makes sense if $y \neq 0$. However, following Boole's successor William Stanley Jevons (1835–1882), it became customary to allow use of 'or' without restriction, and interpret it in the inclusive sense. We will always understand 'or' in this now-standard sense, 'p or q' meaning 'p or q *or both*'.

Mechanization

Even before Boole, machines for logical deduction had been developed, notably the 'Stanhope demonstrator' invented by Charles, third Earl of Stanhope (1753–1816). Inspired by this, Jevons (1870) subsequently designed and built his 'logic machine', a piano-like device that could perform certain calculations in Boole's algebra of classes. However, the limits of mechanical

† Arguably *disjunction* is something of a misnomer, since the two truth-values need not be disjoint, so some like Quine (1950) prefer *alternation*. And the word 'connective' is a misnomer in the case of unary operations like 'not', since it does not connect two propositions, but merely negates a single one. However, both usages are well-established.

‡ Latin, on the other hand, has separate phrases 'p vel q' and 'aut p aut q' for the inclusive and exclusive readings, respectively.

engineering and the slow development of logic itself meant that the mechanization of reasoning really started to develop somewhat later, at the start of the modern computer age. We will cover more of the history later in the book in parallel with technical developments. Jevons's original machine can be seen in the Oxford Museum for the History of Science.[†]

Logical form

In Section 1.1 we talked about arguments 'having the same form', but did not define this precisely. Indeed, it's hard to do so for arguments expressed in English and other natural languages, which often fail to make the logical structure of sentences apparent: superficial similarities can disguise fundamental structural differences, and vice versa. For example, the English word 'is' can mean 'has the property of being' ('4 is even'), or it can mean 'is the same as' ('$2 + 2$ is 4'). This example and others like it have often generated philosophical confusion.

Once we have a precise symbolism for logical concepts (such as Boole's algebra of logic) we can simply say that two arguments have the same form if they are both instances of the same formal expression, consistently replacing variables by other propositions. And we can use the formal language to make a mathematically precise definition of logically valid arguments. This is not to imply that the definition of logical form and of purely logical argument is a philosophically trivial question; quite the contrary. But we are content not to solve this problem but to finesse it by adopting a precise mathematical definition, rather as Hertz (1894) evaded the question of what 'force' means in mechanics. After enough concrete experience we will briefly consider (Section 7.8) how our demarcation of the logical arguments corresponds to some traditional philosophical distinctions.

1.5 Syntax and semantics

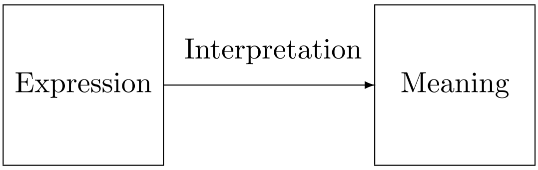
An unusual feature of logic is the careful separation of symbolic expressions and what they stand for. This point bears emphasizing, because in everyday mathematics we often pass unconsciously to the mathematical objects denoted by the symbols. For example when we read and write '12' we think of it as a number, a member of the set $\mathbb{N}$, not as a sequence of two numeral symbols used to represent that number. However, when we want to make precise our formal manipulations, whether these be adding decimal numbers

[†] See `www.mhs.ox.ac.uk/database/index.htm?fname=brief&invno=18230` for some small pictures.

Cambridge University Press
978-0-521-89957-4 - Handbook of Practical Logic and Automated Reasoning
John Harrison
Excerpt
More information

digit-by-digit or using algebraic laws to rearrange symbolic expressions, we need to maintain the distinction. After all, when deriving equations like $x + y = y + x$, the whole point is that the mathematical objects denoted are the same; we cannot directly talk about such manipulations if we only consider the underlying meaning.

Typically then, we are concerned with (i) some particular set of allowable formal expressions, and (ii) their corresponding meanings. The two are sharply distinguished, but are connected by an *interpretation*, which maps expressions to their meanings:



The distinction between formal expressions and their meanings is also important in linguistics, and we'll take over some of the jargon from that subject. Two traditional subfields of linguistics are *syntax*, which is concerned with the grammatical formation of sentences, and *semantics*, which is concerned with their meanings. Similarly in logic we often refer to methods as 'syntactic' if 'like algebraic manipulations' they are considered in isolation from meanings, and 'semantic' or 'semantical' if meanings play an important role. The words 'syntax' and 'semantics' are also used in linguistics with more concrete meanings, and these too are adopted in logic.

- The *syntax* of a language is a system of grammar laying out rules about how to produce or recognize grammatical phrases and sentences. For example, we might consider 'I went to the shop' grammatical English but not 'I shop to the went' because the noun and verb are swapped. In logical systems too, we will often have rules telling us how to generate or recognize well-formed expressions, perhaps for example allowing '$x + 1$' but not '$+1\times$'.
- The *semantics* of a particular word, symbol, sign or phrase is simply its meaning. More broadly, the semantics of a language is a systematic way of ascribing such meanings to all the (grammatical) expressions in the language. Translated into linguistic jargon, choosing an interpretation amounts exactly to giving a semantics to the language.