Application Unification

clockssugars.blog/appliuni

12/19/2025

Contents

Chapter 1

Preliminaries

philofmath Nascent's Philosophy of Mathematics

Before we discuss mathematics properly, since I want this text to be approachable to people with only cursory mathematical experience, I think we should first specify what doing mathematics actually is. There are a few necessary parts of this: why mathematics is possible from its foundations; the structure of mathematics, i.e. what a mathematical discovery would look like if we found one; and finally the language of mathematics, i.e. the way mathematicians articulate our discoveries in order to make them useful. This chapter will be my attempt to discuss these philosophical foundations and motivations, which are usually and unfortunately left implicit by mathematicians; it is generally expected that being able to see these foundations is merely a prerequisite for a practical interest in mathematics. Instead of following this pattern, I will venture to make these prerequisites and expectations explicit, at the risk of putting on the for clown makeup a single chapter.

If you consider yourself sufficiently familiar with the mathematical format, you may wish to skip this chapter, however I think it may hold some value in its articulation of the mathematical pursuit regardless. If you are unfamiliar with the mathematical format, then you should know now that i.e. means "id est", latin for "that is" and generally used to mean "this can also be stated as", and we will be using it a lot.

philofmath.1 The Possibility of Mathematics

First we must discuss "what is mathematics is at its core"? This is a fairly open question, but I think there are two parts of an answer that stand out as insightful: one about the nature of logic that mathematics is built on top of, and one of the mathematical tradition itself. As this project progresses, I hope you will come to see this pattern I will describe which is: that logic, almost alone, has the peculiar power to describe and characterize concepts on an intuitive level, through the network of their consequences and incompatibilities.

The classic example of this is of rain making the ground wet. If it has rained then you know that the ground is wet. If the ground is wet, that does not necessarily imply that it has rained. Instead, there could have been some intense humidity, or someone washed their car, or someone simply covered

the ground with as much water as they could. If the ground has been wet for a while, then one might deduce that it could not be true that it is a hot dry day, or else the ground would have quickly dried. These statements are incompatible. If I had not told you that the property we were discussing was that "the ground is wet" then from the various things I told you could not be true, or things that would imply this unnamed property, you may even deduce without instruction that we are indeed discussing "is the ground is wet". In the course of 'doing' mathematics, one of the things we do is invent concepts that are unnamed at first, and by their implications, discover what they truly mean or might describe.

One might then say that the realm of the mathematician begins in the full scope of cognitive tools we employ to characterize these propositions. Consider, if we study a two dimensional plane where points have an x and y coordinate so that $(2, 3)$ describes a point, I could pose the proposition to you that $y = 5x + 2$, read as "the y coordinate is equal to five times the x coordinate plus two". Of course this proposition would be wrong in the vast majority of cases, such as $(2, 3)$ (where $y = 5x + 2 = 12$ by setting $x = 2$, contradicting $y = 3$), and neither does it have a 'solution', a point we can single out and speak of alone where it *is true*. However just as we spoke of a space of points in a two dimensional plane, we may perhaps speak of a sub-space where the proposition *is* true, and restrict our attention to this space. If we are to do that, we do not need to limit ourselves to discussing this with a mere proposition $y = 5x+2$. We can define a set of points where the proposition is true, which we would write as $\{(x, y) \in \mathbb{R}^2 \mid y = 5x+2\}$, and then we could use the language of set theory to discuss which other sets it intersected with. We could also draw these points on to a two dimensional plane, obtaining a straight line that meets the x-axis at $(-0.4, 0)$ and passes through at an angle of about 78.69 degrees. On the level of a set, when we show that the intersection of two such sets corresponding to propositions has an empty intersection, we can say that the propositions are mutually exclusive, i.e. if one then not the other. When we draw these graphs, we can check this merely by looking to see if the lines touch; this is a substantially easier way to make deductions about a different class of reasoning, graphs implying statements about set theory or equational propositions.

Mathematics is very much the study of logical abstractions. We study them by restricting them to special cases, seeing which other statements they cannot exist with or must exist with, just as in our example of the ground being wet. However we break from the discipline of logic, vitally, by actively concerning ourselves with the relations between these different settings of reasoning so that we can make deductions using multiple forms of analysis. For example, an equation is a proposition ("this is equal to that"), but as we saw above it is also a set and thus vulnerable to set theoretic attack, and also a graph which we may inspect visually. Just as we have these three lenses from which to study a logical statement, as mathematicians we are often concerned with inventing some fourth intuition which is equally different in its reasoning than those three are from one another. We ask, 'what new things could we learn if we were able to invent a new perspective, or think about this problem in an entirely new way?' and then we discover and formalize the rules of this new way of thinking and apply it.

One might blame this for the sense that mathematicians quickly devolve into speaking about alien concepts, but this is merely the other side of that coin: we are willing to adapt our thinking so drastically to find new ways of understanding. In this way and many others, mathematicians sit necessarily between physicists and logicians on the axis of 'rigor'. We are distinct from the logicians as we (usually imperceptibly) weaken this rigor to understand things better, and distinct from the physicists as we crave to resolve contradictions in our descriptions that we might finally 'know what

we are talking about'.

philofmath.2 The Structure and Utility of a (Mathematical) Theory

To describe what one hopes to achieve from mathematics, it's perhaps easier to first describe what is achieved from a theory in abstract. When one selects an object of study, say for example, mammals, one may first put significant effort into defining the object, and then giving structure to subcategories in order to discover meaning from those substructures.

I am no biologist but naively and for the sake of argument, one may say that it is clear a mouse is a mammal, and clear that a lizard is not. One may then decide that a bird is not a mammal, and thus that mammals give live birth and do not have beaks. However many echidnas have beaks, and platypus have beaks as well as laying eggs. Despite this exception, they both still have only one jaw bone, do not have feathers, and lactate to feed their newborns, so we may broaden or adjust our conditions slightly and still call them mammals. With the boundaries around our study fixed, we may then notice that bears and dogs share a resemblance, and that dogs and cats share a similar body plan, constructing the order Carnivora and the two suborders Feliformia and Caniformia, containing cats and dogs together with bears respectively. Such a structure is necessary for one to argue for Speciation, that just as we have artificially bred kinds of dogs, that perhaps nature has over a long time bred some four legged animal into what became a proto-dog and proto-cat, and the former was later separated into dogs and bears. Now if you were responsible for medically treating one of these animals, you would have the information that the biology of a dog is more similar to a bear than to a cat. And you can conceive of a broader family-tree of life for which you may extend this utility or draw on further examples to better inform it.

This mode of thinking is actually not out of place to describe something such as Group Theory.

For the layman with great interest in mathematics, a naive attempt to learn about what group theory is may yield what a group is axiomatically. That is, a formal definition will tell you (if for no other reason than to illustrate how dense and unapproachable such a description is) that a group is a set G closed under a multiplication rule such that for all $g, h \in G$, $(gh) \in G$, associative such that $(gh)k = g(hk)$ and with identity and inverses so there is some $e \in G$ with $eg = g$ doing nothing (i.e. like multiplying by one) for all $g \in G$ and an inverse $g^{-1} \in G$ for each $g \in G$ such that $gg^{-1} = e$. Or instead you might encounter an explanation grounded in intuition such as the ways in which a shape can be rotated or moved so that it looks the same, or algebraic examples such as the set of matrices with non-zero determinants. To use our previous example, this would be like noticing the existence of the category of mammals and stopping there. It is not clear what deeper understanding we have achieved by constructing a mere name, title, or category, and we have not discovered any particular similarity between groups nor do we have a notion of a family tree with which to discover other valuable patterns.

So let me describe how that works.

Once we have two groups, G and H, we may construct new groups, such as the product group $G \times H$, whose members are pairs (g, h) for each g in G and h in H. If we have a multiplication operation between groups, can we reverse this? Is there a notion of group division? In fact there is. I must emphasize not to worry if you do not fully understand this example, it will be elaborated properly later:

First we find a subgroup H to divide, that is, a subset of G inheriting the multiplication operator, which cannot be escaped into G by multiplying its elements (i.e. the subgroup is *closed*), and also contains all of its own inverses. A group is not always commutative, that is, we cannot say for certain that two elements g and h in G satisfy $gh = hg$, i.e. multiplication cannot be reordered arbitrarily. However there may be subgroups whose elements do reorder with the rest of the group, i.e. a subgroup Z in G for which all elements z satisfy $gz = zg$ for g in G even if this is not satisfied by other elements h in G (we would call Z the 'center' of G). More generally, a subgroup N may not necessarily have elements that each reorder with each element of G, but may reorder as a whole subgroup, so for example, for each g in G, we have $gh = kg$ where $h \neq k$ most of the time but h and k are both in N. So long as this is true, we write $gN = Ng$ to mean 'some element of N' in place of the element itself, i.e. for any h in N, there will be some k in N such that $gh = kg$, and when this is true, we call N a 'normal' subgroup. If we have $gN = Ng$, then elements of N may be in a sense reordered with the rest of the elements in G, and so we may define a new group G/N of elements gN for each g in G. This 'quotient group' factors out the dynamics of the normal subgroup, since we have $(gN)(hN) = g(Nh)N = g(hN)N = (gh)N$ for elements g and h in G, and if h should also be a member of the subgroup N, then we have $g(hN) = gN$, meaning that any elements of N are factored out.

Again, it is not important to understand this example in detail, in fact later we will discuss it properly, however the example does illustrate the point. We take a definition of a group and construct a group product, and a subgroup, then the conditions necessary for a factor group and thus a quotient group, using our intuitions about multiplication and division of numbers as a guide. If we restrict our attention to the finite groups, then we may ask whether or not this notion of divisibility admits a version of 'prime numbers' for groups, and indeed this is true. In fact, mathematicians have completed a de facto periodic table of finite groups, the great accomplishment known as the classification of finite simple groups. In fact it has been proved from underlying axioms that all possible finite groups are accounted for, and that it is impossible for a group to exist which does not classify into our periodic table.

From here, the intuition that groups describe "symmetries" of objects now has strong rules: the symmetries of objects decompose into now well known groups and we may say with certainty what the atomic components of these symmetries are as well as studying those atoms. Our theory has yielded an unfalsifiable taxonomy and practical implications for all systems with discrete symmetries.

Many fields in mathematics follow a similar pattern as laid out in the example about mammals which we then applied to group theory. The protocol, if you can even call it that, is thus: strictly define a core object, preferably abstractly so that later on you can apply the structure to anything that fits the bill (e.g. symmetries described by group theory); create sub-labels (e.g. subgroups, some subgroups are 'normal' or 'central') and characterize their properties to build an understanding of what these labels are or what they mean, possibly relating the structure to something more familiar (e.g. product groups and quotient groups as analogous to multiplication and division of numbers); and finally, substantiate the utility of this theory if you have not already, which we did by saying that 'groups describe symmetries', which is formalized in the study of 'group actions'.

With this pattern in mind, we are able to collect 'theories' or academic sub-disciplines centered around a school of thought, each with a certain descriptive capability. With a large collection of

theories each studying some construction that 'appears everywhere', we assemble an increasingly comprehensive framework that is able to provide already well studied models for new phenomena that we discover. For instance, once the differential equation is well studied, the experimental evidence that not a magnetic field but a *change* in magnetic fields induces a current, allows the articulation of Faraday's law. Or we could apply our models retroactively to draw new conclusions: man has always known that fluid flows and perhaps at times even understood many things about its flow, but only with an understanding of vector calculus and continuum mechanics did it become reasonable to describe Navier-Stokes equations, and only much later did we discover a framework by which to use this model computationally (e.g. finite volume methods) when we could not solve the equations analytically. So to widen this arsenal of theories is to widen our foreknowledge; the physicist need not try every experiment he can imagine, only a choice few and the rest can be tested at the blackboard since the proper mathematical framework is already explored.

philofmath.3 What is Doing Mathematics vs Using Mathematics

With this in mind, I must add an essential disclaimer that will also characterize our motivations and the rules we follow in line with those motivations. We must consider the axis of mathematical 'purity'.

In fact we do call it Pure Mathematics to be concerned primarily with abstractions rather than their consequences, and Applied Mathematics when we try to take our grand collection of theories and see if they make valuable predictions about, or tools for, manipulating the world. I would posit that this is a microcosm of the axis I spoke of earlier, placing mathematicians between logicians and physicists, with the pure mathematicians closer to the logicians and the applied mathematicians closer to the physicists, but with a particular asterisk. One must not think that the applied mathematicians, or the physicists for that matter, are being in some way illogical. Rather, the task of translating a setting we are concerned with studying into a mathematical framework requires reasoning outside of the pure mathematical constructions, using an understanding of the setting itself.

My favorite example is this: say we are studying the way that thermal energy disperses within a solid. That is, we have for example a long metal cylinder laying on a heating pad that keeps the bottom of the cylinder very hot while the top is cooler. This can be studied using the heat equation, which when solved, will provide for each point in a coordinate system, let us say (x, y, z), with a temperature at a given time t. However, what happens if we lift up the cylinder and bend it? Suddenly we have a problem in our mode of analysis: is our coordinate system fixed in space, or fixed relative to the material? If it is fixed in space, our description will surely become inaccurate, because by bending one end of the cylinder down, we have moved material that was further from the heating pad toward the bottom, where hotter material would be if the cylinder had not been bent. If we remain in fixed spatial coordinates, we would go to that location and instead find material colder than it should be.

Although it seems obvious to us, as creatures that live in this world and have become familiar with its rules, this constitutes a very serious alteration to our mathematical model. We know that temperature is not a property of space, it is a property of material, and a hot piece of material retains its heat when moved. Our mathematical model does not know this unless it is told, and to do that, we must construct an alternative coordinate system that can track all

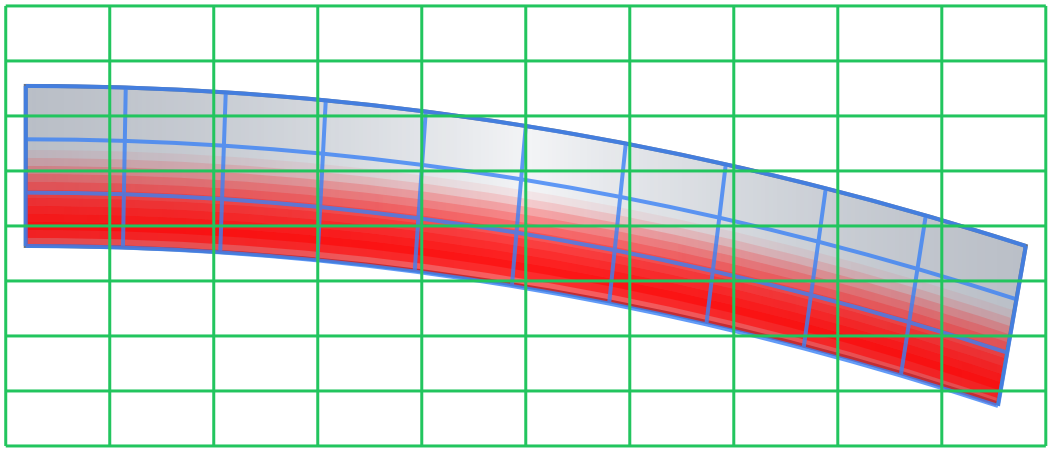

Figure 1.1: The spatial coordinates
(green) do not bend with the material, so these coordinates will predict the wrong temperature. The material coordinates (blue) will continue to provide a good model.

pieces of material based on their locations at some fixed time 'before' they were moved (in fact this is called Lagrangian coordinates). Then the (x, y, z) we speak of would not refer to a position at a given time, but only to the position that a *piece of material* has at time zero, prior to any deformations. It must be emphasized that we decide to use a different coordinate system in this way on the basis of fundamentally qualitative reasoning; there is no formal deduction we can make that tells us 'if material property then material coordinates' other than the knowledge we have that this is the mathematical description of such a system. When we do this, we are necessarily engaging in Natural Philosophy, and this is not something we can escape as formal reasoning *about the world* must be founded upon *knowledge of the world*.

One may argue then that pure mathematicians in fact have it easier than applied mathematicians: everything relevant to a pure mathematician's problem is written down for him clearly. He needs only be a puzzle solver. By contrast, a good applied mathematician must be a Natural Philosopher of his object of study and make qualitative judgements. And this makes him vulnerable, in a way that the pure mathematician is not, to having reasoned about his study incorrectly or formalizing a statement that is in fact not true. (The pure mathematician, when told the model he is studying does not reflect reality, will laugh and say something to the effect of 'I'm sure it models something else then' or 'I'm still curious about this system' and resume their study without regard.)

It is under this mode of operation that we say that every mathematical theory of a phenomena is a formalization of a philosophy of that phenomena. This distinction is an important one, as it is clear that not every philosophy of phenomena is 'true', but it may well be convincing in a way where it is at least logically self consistent. This creates a problem that mathematicians themselves prefer not to be responsible for, and perhaps rightly so, which is that it is entirely possible to create a self consistent mathematical description of phenomena which has little relation to the actual behaviour of the real phenomena. This is what we call 'a bad model'.

This is the double edged sword of the 'large collection of theories each studying a construction which appears everywhere' that pure mathematicians produce for us. Once the space of implications of different sets of presuppositions have been explored, we obtain a relationship between presuppositions

and implications. This means that to select the things we suppose are true about a phenomena is also to select the implications; we are no longer choosing merely the 'starting axioms' but in fact discounting entire branches of thought since we know which axioms lead to those branches. We see problems such as this in cosmology and theoretical physics: it is entirely possible to restrict our attention to models of reality that aren't clearly wrong without selecting a model that is particularly right. If you have a little bit of mathematical training, you may have even seen this yourself when people propose linear models and draw overly simple conclusions.

A different version of this accusation could also be levied at category theory (of which I have reasons to be fond) which finds ways to express things we knew in other theories all within one theory, and then is sometimes granted some unifying ownership over all other theories when all it did was provide a language with no implications of its own. Such a pattern may seem impressive if you are unfamiliar with the subfields such a theory claims to unify, but with awareness it becomes clear that the 'unifying theory' predicts nothing (although I should emphasize category theory is much less guilty of this than other theories which I will not name here).

The broader argument I am making is also true in reverse. The failure of a school of thought to make their descriptions quantitative/algebraic is not a signal that their study cannot be understood or is not sensible, just in the same way that the mathematical self consistency of a model does not make it true. In fact it is a failure mode we are well aware of that we might discover we are "using the wrong model" when a new phenomena arises. The mathematical lens of analysis is one that requires unfalsiable rules as prerequisite for any meaningful deductive implication. Consequently, as the complexity of a system increases, it becomes increasingly inappropriate to apply such a lens; the tools of probability and statistics can be found fit to analyze such systems regardless, however these tools carry their own intense and deep epistemological complexities that are rarely understood without particular mathematical expertise, and famously used to mislead.

What this amounts to is that one must know there are two distinct jobs of mathematicians each with separate responsibilities and different failure modes: the puzzle solvers who *do* mathematics and the natural philosophers who *use* mathematics. For the former, we have a game that, once understood, offers a rich and fascinating landscape of closed-ended problems to solve. It is an unfortunate but necessary (and in fact extremely gratifying) responsibility of the latter, the applied mathematicians, to know the object of their study deeply, to have the knowledge-that-is-power over nature *first* before expecting the symbols of math to yield anything of value.

philofmath.4 The Language of Mathematics

Mathematics remains remarkably successful at cultivating an image of mechanical logic, regardless of the disclaimers I make above, and this is not without reason. There is a strong argument that mathematics is, before everything else, a literary tradition, and it has its cultural and linguistic features of this literary tradition to thank for its achievements. Indeed, in a shortly following chapter, we will discuss how the language of mathematics mostly maps on to a structure robust enough that it can be understood and checked by a computer; that the structure of this tradition lends itself to such a standard, despite the appropriate asterisks, is no small feat.

In order to participate in this literary tradition, we must spend some time (at the risk of becoming a bit drier) discussing the literary structure and features of a mathematical text.

In mathematical literature, there exist much more formal rules for certain features of writing

structure. That is, where an english teacher might tell you an essay comes in discrete units of introductory and distinct argumentative paragraphs, mathematics comes in units of text that are distinct much moreso than in other forms of writing, even to the detriment of typical notions of readability. The reason for this is straight forward once you are familiar with the true goals of mathematical literature: you might say that mathematical writing is in fact intended as almost computer code, (although I would argue the genealogy is reversed given the comparative ages of the two disciplines). When, as both a programmer and one with mathematical literacy, one inspects a terse math book, one immediately sees the telltale pattern of blocks of code, functions, class or typeclass instantiations, each interspersed with comments and documentation. This relationship is made somewhat obvious in the school of literate programming.

This is the error in trying to 'read' a math textbook in the way that one reads any other piece of literature. In fact, many math books are published with the explicit intention of being formal references much in the same way that one writes a package providing high level tools for other developers; e.g. one does not rewrite the Windows API each time one writes an application in Windows that opens a page on the screen, there are sophisticated tools written to do this already that are reused and maintained separately. Similarly one does not always strictly define every term and theorem before publishing a math paper, there is a corpus of written constructions that one can refer to even by name as necessary. The flip side of this is that one does not naively read a math book unless it is explicitly written to be forgiving. Some authors move away from this tradition in certain books which are intended explicitly to introduce a subject, however the consequence of this is that the book then makes a terrible reference after having educated the reader; a problem made worse when you consider that the book may have slight differences in implementation of ideas, and so is not in perfect agreement with other books which *are* intended as a reference.

My intention with this text is to take the best of both literary styles, one that is intensely descriptive and which denominates strict statements side by side.

Some of the description of mathematical literature must be postponed until our discussion of propositional logic, as mathematics inherits a lot from logic. More still must be kept in a separate glossary of mathematical grammar patterns, since there are more than can fit nicely in a narrativized exposition.

There are three core types of formal blocks in mathematical writing, and a few other less formal blocks which are nonetheless made distinct. It is common to either name or number these blocks since they must also be referred to formally.

- Definitions: These define new mathematical constructions, such as labels for functions, or conditions under which we classify certain constructions. These often come in the form of an object or collection of objects, often along with a proposition about that object, and acts as a rule for *constructing* an object in a formal sense which will be discussed more in its own chapter. We may also include in definitions the rules for how we write these constructions and what notation exists around them here or in other literature you may find on them. To set up a definition, it is common for us to set up a circumstance in which the definition may apply, saying "Let $f\colon A \to B$ be a X and ... then we call f and Y together a Z if ..." and Z is the new label we are creating which has no strict meaning other than what we give it here. The use of 'let' should usually be taken to mean something like "should we find ourselves in a circumstance where..." for the purposes of definitions. Definitions (and theorems for that matter) often employ a grammatical rule in which we may use parentheses following

certain terms to duplicate a statement into another statement that says something near identical, ideally with the word 'respectively' included or abbreviated to indicate what we are doing, but this is not always the case. An example of this would be "when a number satisfies $x > 0$ ($x < 0$) we say that it is positive (negative)".

It is informally important to give defined concepts good names, so that people can may develop appropriate associations and intuitions; we may encounter names that seem unintuitive but in these cases it is important to respect and understand historical associations built up by the literary tradition.

- Theorems: These define rules under which one proposition might imply another, or imply each other thereby becoming equivalent statements. They are, in a sense, the functions of mathematical propositions, and they also adopt the language of setting up circumstances and conditions for their application as in definitions. That is, while the exemplary theorem may be written "if X then Y", in practice they are often written "Let X be true. Then Y is true".

  There are a few logical operations or quantifiers such as "for all" and "there exists" which have particular ways or sets of ways to refer to them which may seem at first ambiguous. We will discuss their formal meanings in our chapter on propositional logic, but it is important to know that the formal meanings referred to there are equally referred to by "for any" or "there is a". "For all X, we satisfy Y" can also be written in shorthand as $\forall X, Y$ or referred to as "choose some X. Then Y" implying that we could choose any other. "There exists X such that Y" is written in shorthand as $\exists$, as well as $\exists!$ when we mean "there exists uniquely", i.e. if there exist X and Z such that Y for both X and Z then $X = Z$. Logical implication is often written by a double-lined arrow $\Rightarrow$, and when this implication goes both ways, both $X \Rightarrow Y$ and $Y \Rightarrow X$, we say "X *if and only if* Y" or write $X \Leftrightarrow Y$.

  Since theorems may also show that two definitions are equivalent, they sometimes serve as the follow-through of a set of definitions, carrying the notation we will use when writing about the described concepts. Lastly, theorems are referred to by many names, each with connotations about the theorem being described. These are:

  - Theorem: a particularly important discovery either for its utility to us in the text we find it, or historically.
  - Lemma: a less significant insight which will be very useful for proving other theorems.
  - Proposition: a theorem of moderate importance, between that of a lemma and a theorem. Although "proposition" should refer to a statement that could be true in principle, the name is very often used simply to mean theorem when it punctuates a relevant insight. On rare occasions, one may write a proposition formally as a conjecture, provided without proof, and then disprove it later to make a point, however this practice is not common.
  - Corollary: a theorem that is obvious or especially easy to prove as a consequence of a theorem or proposition. The statement of a corollary is exemplified in the format "it follows simply from X that Y".

- Proof: These usually follow directly after theorems and show the process of deducing the implication of a theorem from its premises. It is common to describe these premises as the 'hypothesis' rather than the theorem itself, and so a proof will usually start with setting up the

context under which the theorem would apply, e.g. "assume/presume by hypothesis that ...". Depending on the style of proof, we may first manipulate the proposition we want to prove into something that is equivalent on purely logical grounds, such as identifying multiple goals for proof (in the case of "if and only if"), producing a contrapositive statement or proof by contradiction. In the case of the latter, a proof might start with "Assume for the sake of contradiction that..." and then later conclude with "...violating our hypothesis". When the statement we are proving includes "there exists X such that Y" we must provide an X that satisfies Y, and in doing so we may say "Propose X =..." and then go on to prove that it satisfies Y; another name for such an X we propose or more generally a form X could take once some details are resolved, is an *ansatz*.

Moreover, in the process of a proof we will often have to apply prior theorems e.g. "by theorem 4.13, we have..." or inspect the meaning of our hypotheses and expose the inner statements of their propositional form, thereby exposing variables. This can be messy, since we may have a theorem for example that is "for all $\epsilon > 0$, there exists $\delta > 0$ such that ..." and then need to apply this *twice* with different ϵ. In such cases, we try to either resolve whatever we were using the theorem for so that it is clear its inner variables have no baring on what comes after, or we label our variables with subscripts when we invoke them i.e. "from our hypothesis, we have that for all $\epsilon_1 > 0$ ..." so that our next invocation can then be labeled ϵ_2. The naming and renaming of such variables that are usually hidden away in the statement of a theorem is a highly contextual thing. This contextual awareness is also often expected for certain 'obvious' deductions, so it is sometimes expected that when we say "applying theorem 4.13..." we have recently used this theorem often enough or defined it in ways that we think it is obvious how to satisfy the hypotheses of a theorem in order to apply it. Additionally, when the proof we are doing has multiple goals but these goals are similar enough, we may only solve one goal and use the expression "without loss of generality". This means that the structure of the proof we have done for the case we cover is near identical to the proof for the case we have not covered, barring some extremely minor alterations.

Finally, it is common to end the proof with some signifier. This is usually a square on the right hand side, indicating the proof is over, but it is also common to write Q.E.D., referring to the latin "quod erat demonstrandum", literally meaning "what was to be shown" i.e. we say this after we have deduced the thing we wanted to show. A professor of mine once joked that it is perfectly valid in your own proof to simply draw a picture of a cat at the end of your proofs as long as you are relatively consistent about it.

These blocks are particularly valuable when using a math text as a reference, since it is clear that anything not highlighted in such a block is merely of pedagogical value and can be ignored when the content is no longer unfamiliar. There are a few other named blocks of information we should list for completeness, as we will also use them.

- Example: an example intended to make the practical usage/instantiation of a theorem or definition concrete, often with an instance you might already be familiar with. Sometimes examples will come in groups, starting with relatively trivial instantiations that don't show much of the consequences of the thing being demonstrated, then gradually moving towards more sophisticated examples. This is a good opportunity to talk about what is meant by 'trivially' in mathematics: you could for example construct a system of numbers that has

a multiplication rule trivially by saying the system of numbers is just the number 1 and the only multiplication you can do is $1 \times 1 = 1$. There is also the notion of a 'pathological case' or an example that is pathological: often times when we build up mathematical theories, we intend for them to describe some intuition formally, and it is sometimes possible within the framework of such a theory to construct an example which violates all of these intuitions but also isn't particularly useful for describing anything. These cases are useful for understanding what structure is *not* implied by the rules we have set, and when we need to add additional rules to avoid such cases.

There is a pattern I find quite irritating in mathematics texts of using 'Examples' not only to give intuition but also to indicate examples which are vitally important for later discussion. It is my opinion that one should be able to skim or skip past examples on repeated readings, and as such, they should only be there only to serve the first-time reader when they feel uncertain about a concept. For that reason, many of the 'examples' of this text will in fact be writte in definitions or theorems of their own, signified as part of the continuous exposition in a way that the boxed 'Examples' will not.

- Remark: As an extension of the blocked-information format, these can sometimes be useful as an extended footnote about something we have recently mentioned, either to give intuition or speak on the limitations of a tool or concept.

- Notation: These will be useful occasionally to define new shorthands for things that we have already defined, or ways of writing things which does not fit neatly under the umbrella of a construction. When defining a symbol in a purely notational manner, it is common to use := to mean 'defined as equal to'.

  We may also sometimes speak of 'abuse of notation' in which we prescribe the notation of one concept onto a different concept when we wish to think of them similarly, even if they are quite different; in practice however, what counts as 'abuse' and what is considered a broadening of what the notation should be used for is a matter of opinion.

You may have also noticed that we primarily use first person plural pronouns; this is almost universal in mathematical writing. I have heard conflicting stories about this but I am told this is called *Royal We* or *Author's We*. Certainly in the way I use it and have come to understand it, is in two interchangable modes: the use of *we* is to mean I, the author, together with you, the reader, as though we step through mathematical procedures together, as if by guidance; and on occasion when I speak of *we* as in mathematicians as a class, or a subclass of mathematicians who have developed some tradition, i.e. "when X *we* write Y as Z". With this in mind, I make a certain distinction about when to use 'we' and when to use 'I' based on whether I am speaking about something of my own personal understanding.

The above, and indeed many of the instruction I have tried to preempt, may seem abstract for now. The following two chapters should serve to specify the firm symbolic meanings described above and contextualise the softer intuitions I have mentioned. Proceeding to them, you may want to keep this section handy as a reference for the time being.

proptypes Propositional Logic: A Programming Inspired Approach

Highschool is usually the first place people encounter any form of mathematics, and along with this setting comes some inappropriate expectations about what lies ahead in mathematics. I might go as far as to say that highschool mathematics and even first year undergraduate mathematics approaches the topic primarily from the perspective of an engineer or experimental physicist; that is mainly to say that you wouldn't call highschool mathematics even particularly 'mathematical' in the ways that a mathematician is actually concerned with the topics studied.

I myself only discovered the seriousness of this discrepency in my first real analysis class. I had lamented at the time that it felt like I was being 'expected to do math as a lawyer who writes computer compileable legalese'. However, as time has gone on, not only have I grown to embrace that description, I've also come to understand that the other classes I was taking at the time (with which I did not have this grievance) were more about applying a physicist's reasoning to geometry and calculus than true mathematics.

This 'true mathematics' is the programmatic legalese. In mathematics, we care very deeply about strict conditions and labels that tell us with certainty under what circumstances our tools will *always work*. A peer once even said to me that the mathematics was not in symbols, but in all of the words around them.

To that end, we will not spend this chapter discussing mathematics either; recalling the notion that mathematics sits in between logic and physics on an axis of rigor, we will swing all the way to the left and discuss a description of formal logic. In particular we are going to discuss formal logic in the context of type theory, which will allow us draw on programming metaphors which our society's culture has become much more familiar with in recent years.

However I must confess that the programming metaphor has a special value to me, and I hope also soon to you. As I was first coming to understand that mathematics was inseparable from its legalese, the pattern of theorems and proofs in particular (and also due in part to my double major in physics), I had a notion that one does not truly understand a proof unless one sees clearly how the steps of the proof intuitively mean that the theorem must be obvious. I now see this as wrong, or at least far too physicist-brained. There was a period in my education when I was very ill, and although feeling like I had lost half my brain, I realised that as long as you are doing pure mathematics, you only need half a brain.

The job in solving the puzzle is to connect all of the hypotheses, like pipes or belts in a factory, to the relevant machines (other theorems) that transform them into whatever they need to be to satisfy the proof obligations. Proving a theorem is then no different than *playing Factorio* or *programming*. For most jobs, the high level understanding of the whole system is completely subserviant to the low level understanding where you merely know how to correctly plug pipes into ports. If you can do that, then with sufficient effort and an open mind to understand your objects/items/constructions and use your tools/machines/theorems, you can *do mathematics*. It is this, among other experiences, that makes me so confident that anyone can follow this text, provided I have explained myself well enough.

proptypes.1 Types, Functions, and Arguments

First I must belabor about some of the basic constructions of programming. You may of course see fit to skip this part, and that would be reasonable if you are familiar with the languages Rust or Haskell, or a similar language with a robust type system. Otherwise, or even in spite of this, it will be valuable to examine what a type is or what we might want it to be, along with what we might expect of functions or arguments.

First, what is a function? There are so many things we could say about this, but first, let us say that a function is a black box that takes an *input* (or an *argument*) and gives us back an *output*, in such a way that we will always get the same output if we use the same input. This could be a function like $x \mapsto x + 1$ which takes a number x and adds to it one before returning it; this notation with the $\mapsto$ symbol (read as 'maps to', and distinct from $\to$) defines a function in terms of what an output looks like assuming a given input.

But a function need not be defined by the *procedure* we take to calculate its output; if you draw a circle of radius one, draw a straight line through its center, and then for some angle off of that straight line draw a ray, the distance perpendicular to our line to where the ray intersects the circle would be the *sine* of the angle. While methods to calculate this exist, what the *function* is here is the relationship between an input to an output, not necessarily a sequence of arithmetic operations. A function could be a pair of lists, $[1, 2, 3]$ and $[5, 2, 3]$ such that for an input, we find its position in the first list and take the number at the matching position in the second list, $1 \mapsto 5$, $2 \mapsto 2$ and $3 \mapsto 3$; we just as soon make the lists uncountably infinite, the outputs seemingly random, and we still have a function, and for this reason we also call a function a *map*. The *function* is defined by its repeatability and no more. This ambiguity about *how* we take a function is not a failure of the abstraction but a *feature*: much later on it will allow us to study functions for which we only have partial information, and perhaps even discover from that information how to calculate them.

But in each of these cases, even the most arbitrary, we have declared either implicitly or explicitly, a prerequisite condition for an input. For instance, with possible exception to the final example, we could not give any of these functions as an input, say, a letter of the alphabet. The procedure $x \mapsto x + 1$ does not admit addition of a letter, the relationship between an angle and its sine does not admit non-angles, and yet we can just as well define a function that takes a letter and sends it to its opposite letter in the alphabet $a \mapsto z$ or a word to itself interspersed with some symbol "*hello*" $\mapsto$ "*h.e.l.l.o*", so the fault is not in the concept of a letter or a word. In fact, one might even conceive of a function which we define to take both numbers and letters, which maps a number $x \mapsto x + 1$ or a letter to its next letter $a \mapsto b$, $b \mapsto c$, selecting whether to be a function of letters or numbers be based on its input. In this case one might even say that we have *extended* the function to letters by treating them as numbers, using $+1$ to move from one number to the next. But what of $+27$? What is the sine of a letter? Our naive extensions break down.

It is a simple pragmatism that we decide (perhaps only as an arbitrary choice we make now then cling to dearly) that operations such as addition, or relationships that define things like the sine or cosine, fall into classes or *types* of what they take as inputs. We can have the type of counting numbers $\mathbb{N}$, the type of letters, etc. or the type of functions such as $f\colon n \mapsto n + 1$ which we might write as $f\colon \mathbb{N} \to \mathbb{N}$ to symbolize that it is a *function-type* with input type $\mathbb{N}$ and output type $\mathbb{N}$, and that this type $\mathbb{N} \to \mathbb{N}$ has as a member f. We would equally write $5\colon \mathbb{N}$ in this scheme, to say that 5, as a counting number, has type $\mathbb{N}$, of the counting number.

But we have a problem that must be examined closely if we want to make this concept robust. What is it exactly that populates our types? Where does a counting number come from, and if we have two counting numbers, how should we know that they are distinct or the same? Are complex numbers really numbers? If them then why not every clifford algebra? This may seem pedantic but in fact it is the result of carelessness; we cannot say that a *type* is *of numbers* without being stuck in the problem of 'what is a number' with its own risk of sophistry. There is a simple solution at the cost of a much more arbitrary choice than to declare that functions' inputs have *types*. We make a firm rule about *where types come from* which is the following: a type is defined as having *constructors*, which are *function-like* objects which take an object of the correct type and return as their output a *new* object that does not exist except as the output of this constructor when given that input. This means, among other things, that an object has *only one type*. We now present the typical example.

Definition proptypes.1 — (Inductive Natural Numbers)

We define the **natural numbers**, denoted $\mathbb{N}$, to be a type with the following constructors:

$$\mathbf{Z}\colon \mathbb{N},$$
$$\mathbf{S}\colon \mathbb{N} \to \mathbb{N}$$

which we call *zero* and the *successor* respectively. That is, $\mathbf{Z}$ can be thought of as a function that takes nothing and always produces the same result as a constructor, thus defining one single element of the type; $\mathbf{S}$ can also be thought of a function that takes any other member $n\colon \mathbb{N}$ and uniquely defines a new member $\mathbf{S}(n)\colon \mathbb{N}$. By unique, we mean that if $\mathbf{S}(a) = \mathbf{S}(b)$ then $a = b$ and vice versa.
Outside of this section, we consider the natural numbers to start *from one* not zero.

It is reasonable to see this and wonder how this definition of the natural numbers, defined ostensibly from only two rules, populates the entire set of counting numbers we know. The simple answer to this is that you would read $\mathbf{S}$ as a function $n \mapsto n+1$, however instead of being a function, it is in this instance *the definition* of what it means to count forward one number. What we have done, really, is to define the counting numbers distinctly as 'the first number' and 'the number we count after that'. What that means is that a number like 3 would be written in its full form as $\mathbf{S}(\mathbf{S}(\mathbf{S}(\mathbf{Z})))$, and we would 'read' the number based on counting the successor functions, i.e. counting how many times we counted up one.

You could reasonably ask now 'but what is $\mathbf{Z}$?' or 'but what does $\mathbf{S}$ do?' in a deeper sense of what they *are* rather than merely what we intend these symbols to mean. The answer to that, as in a similar fashion we will keep encountering, is *whatever works*. You could define a notion of natural numbers such as these in python using the notion of a list, or an array in other languages, defining $\mathbf{Z} = []$ the empty list, and $\mathbf{S}(n) = [\mathtt{n}]$ as the numeral contained in a list, such that $3 = [[[]]]$, an empty list contained in a list, repeated three times. You could do a similar thing with nested pairs so that $2 = ([], ([], []))$. The instantiation of these constructors does not matter, and they can correspond 'really' (insofar as one rationalizes that there is something *real* about the constructs of programming languages or stories about what a CPU does with those constructs) to anything you like as long as you are able and willing to *read* that function as symbolizing what we intend it to mean.

In fact, this is almost literally the idea of Alonzo Church, the famous *Church Numerals* as they have come to be known. In his construction, constructors were literally functions, but merely functions which we commit to never defining a way to compute. In such a way, you would read $f \circ f \circ f = x \mapsto f(f(f(x)))$ (we read $\circ$ as 'of' or 'compose' meaning to take the output of the second function as the input of the first, f of f) as merely $f \circ f \circ f$ and no more, since there is nothing to be *done* with the expression, and thus choose to read $f \circ f \circ f$ as 3. However the idea to define natural numbers as a starting number followed by an operation to count forward is attributed to Giuseppe Peano, and typically considered a presentation of the *Peano Axioms*, although in his day the language of type theory was in its cradle with Bertrand Russel, the commonly referred to originator of type theory of *Russel's Paradox* fame. In truth, its origins stretch further back than even Russel's predecessor on the topic, Gottlob Frege, and has been a gradual progression in the development of logic. The school of thought we draw on in this section is downstream of that of the *Intuitionists*, attributed to Martin-Löf and his various colleagues and successors.

This framework has certain powers which we will appreciate soon. One often calls the kind of natural numbers we have defined an *inductive type*, and it enables a suite of logical tools for which the famous 'proof by induction' is only one. However there is a general mental rule here that sums up its powers up here quite well: once a type has been defined as nothing but a collection of constructors, anything we want to do with that type need only be defined in terms of those few constructors. This introduces powers such as proof by induction, and indeed a more general form of it which apply to *all* other inductive types, as well as some unfamiliar responsibilities.

That is, okay, we have our natural numbers and we have confined ourselves to a relatively rigid system, that of *types*, to do so. Can we *do* anything with these natural numbers, or do we have to drop this rather laborious system just after we have defined it and return to intuitive, poorly defined notions of arithmetic? The answer is of course we can do things with these natural numbers, however to do so, almost every operation we do will appear as a recursive function. Let me give you an example.

Definition proptypes.2 — (Addition)

We define the function $\texttt{add}\colon \mathbb{N} \to \mathbb{N} \to \mathbb{N}$, which we also write as $m + n$ for two natural numbers $m, n\colon \mathbb{N}$, by the following procedure: if $n = \mathbf{Z}$ then output $m+n = m$ immediately, but if $n = \mathbf{S}(p)$, where $p\colon \mathbb{N}$ is some other number, then we output the result of $\mathbf{S}(m) + p$. We write these rules by **pattern matching** on the different possible forms of n which we are checking for, and so the same definition is written

$$\texttt{add}(m, \mathbf{Z}) = m,$$
$$\texttt{add}(m, \mathbf{S}(p)) = \texttt{add}(\mathbf{S}(m), p).$$

What we have done then is to define addition in terms of the notion of counting upwards, just as we defined natural numbers in terms of counting upwards. Where a natural number can be thought of a tally of each successor function applied to a zero, we define addition to move each of these tally marks from the right hand side to the left hand side, until we are left with zero on the right and the sum on the left. Only then does our recursion terminate, with all 'plus one's moved to one number. Cumbersome as it may seem to define addition in such a way, the upshot is that now everything we could hope to do with a counting number is now almost exactly that

hard, including some much more complicated operations.

I have slipped something by you however: we have not yet discussed formally what it means for a function to take more than one input, and instead I have merely presented one to you with the nebulous function type $\mathbb{N} \to \mathbb{N} \to \mathbb{N}$ with no explanation. This is not a notation that we will see in the rest of mathematics (in fact, in spite of all the sense it could make, mathematicians generally invent new notations just to not need to use this one), and is usually reserved for the purer functional programming languages such as Haskell, however it exposes certain questions and concepts that are valuable to us here. In a programming language, it is common to write $f(x, y, z)$ to use a function, and in many languages (and indeed to us now) it is reasonable to speak of f as a function alone, but then what is (x, y, z) exactly?

Most programming languages have a notion of a *tuple* related to this, a generalization of an *ordered pair* to as many terms as necessary, allowing one to combine objects into one without forcing them to interact or relate to one another; x and y could be different types entirely and this is fine. We also have a variety of similar notions in mathematics, chiefly the *Cartesian Product* which we invoke when we say $(2, 3)$ is a coordinate on a 2D plane combined of two mere numbers. It will also be necessary for us to define the notion of a *Product type*, the type theory equivalent to the tuple which solves this problem by defining functions such as `add` as of type $(\mathbb{N}, \mathbb{N}) \to \mathbb{N}$, however I pose to you that this is not, at this stage, necessary.

But how then will we pass a second input into a function, if not to force it to ride in a pair with the first input? The answer lies in how one should read $\mathbb{N} \to \mathbb{N} \to \mathbb{N}$. That is, what does the $\to$ operation do exactly? Let us for a moment compare it to our addition function, which also is written in the form $m + n$; then one might say that $\to$ is itself a function with two inputs, except that these inputs are not *elements of a type*, they *are types*, and it produces a function type. But if the function type it produces is to be thought of as *new*, then by the rules we have established, $\to$ is not a function but a constructor!

Definition proptypes.3 — (Function Type)

We define the symbol $\to$ to be a right-associative constructor of form `Type` $\to$ `Type` $\to$ `Type`. It takes two types, A, B: `Type` and constructs the type of functions between them $A \to B$.

This does not resolve our problem of how to read types of form $A \to B \to C$ however; instead that is done by our insistence that the operation is defined *right-associative*. Consider, if $\to$ takes only *two* types and produces another type, then absent all other reasoning, you could immediately place brackets to attempt to make this form sensible, considering either $(A \to B) \to C$ or $A \to (B \to C)$. However, consider the form of the former: we know that $A \to B$ must be a function type, taking inputs of type A and producing outputs of type B, which means a type $(A \to B) \to C$ takes *functions* of type $A \to B$ and somehow turns these functions into elements of C. A function is however, itself, a single value, and so we have not constructed a function of multiple inputs as desired. The latter, $A \to (B \to C)$ considers a function that takes an element of type A and outputs a function $B \to C$; consider, this output which is a function could now have an input passed to it! This is the meaning of a *right-associative* operation, one where there are implicit brackets making the right-most operation be performed first.

$$A \to B \to C \to D \to E$$
$$=$$
$$A \to (B \to (C \to (D \to E)))$$

Right-associativity is perhaps poorly named, as it is distinctly *non-associative*. Usually in mathematics we only care about whether an operation is equivalent no matter where you put the brackets (associative, or equivalently right-associative *and* left-associative) or not (non-associative), and certainly in common mathematics almost every binary operation you have encountered is associative (addition, multiplication) at its core (since subtraction and division are inversions of addition and multiplication, they are fundamentally associative, but as written they are not, hence the common memes about disagreeing calcuators) with exception to the cross product. Here however, right-associativity of function types enables us to do something special.

It is of course possible merely to replace functions of form $A \to B \to C$ with those of form $(A, B) \to C$, but consider the latter, it requires that the inputs come *as a pair* prepared and packaged in one single input. Do inputs need to come prepared as a collection? If we write $5+$, is this object meaningless, or have we invented a function? The right-associative notation $\to$ would imply the latter, since $\mathbb{N} \to (\mathbb{N} \to \mathbb{N})$ is still a function with one input and one output, and with one input (five), that output will be of type $\mathbb{N} \to \mathbb{N}$, an operation that adds 5 to its input. That is, a side effect of this notation is that it allows us to postpone this process of providing inputs; in the same way that a function exists without any inputs, and exists with all of its inputs as its final output, now it exists with only some of its inputs. This is called *Currying* for Haskell B. Curry, and is common in languages such as Haskell (sharing a namesake) although it existed in this form as early as Church's lambda calculus. Such a concept is exceedingly common in algebra under a different presentation (and different historical lineage), although mathematicians have relatively clumsy notation to do this at best and use it only very conservatively, an error on both counts in my opinion.

Since currying is so natural to the functional mode of operation, they dispense with the $f(x, y, z)$ notation of function application entirely and often write $f\,x\,y\,z$, since then it is equally natural to write $f\,x\,y$ meaning the curried form defined $z \mapsto f\,x\,y\,z$. In this frame, one provides an input x to f, at which point we say that 'x has become an *argument* of f', and then when we later provide y, we have $(f\,x)\colon y \mapsto (f\,x\,y)$ where x has become a *parameter*. The distinction between these terms is thus: an argument is what is provided in the moment of a function's *application*, and a parameter is some value that exists fixed in an expression that defines a function's *output*. In our example, x appears as a parameter at the time that y is an argument, due to our currying; if instead we had $(x, y, z) \mapsto f(x, y, z)$ then we might call each of x, y, z arguments at the same time. Equally, you would say that a *function* $x \mapsto x + 1$ has as a parameter 1, since one could imagine this is also the function $(+1)$ curried. This distinction might seem moot now, but later we will see that while the notion of an argument is a formalism, the notion of a parameter is deeply tied up with scientific notions of independent and dependent variables, chiefly as those held *fixed* in an experiment.

proptypes.2 Proposition Types and Basic Propositional Logic

We are now ready to use type theory for its principle purpose in this chapter, bringing home multiple threads we have left open in the previous subsection. First, let us discuss basic logic so we can in some sense define our conceptual targets, what our theory of types must model in order to describe logic.

We have notions of true and false, which we denote $\top$ and $\bot$ respectively, and we have, famously three atomic logical operations we can do with them: `AND`, `OR`, and `NOT`. In many programming languages, these are written &&, || and !, however in formal logic they are written $\wedge$, $\vee$ and $\neg$. Then if we have two propositions P, Q (you can think of these as representing something like "the ground is wet" or "I played basketball") then famously, their rules are:

- $P \wedge Q = \top$ only if both $P = \top$ *and* $Q = \top$, otherwise $P \wedge Q = \bot$.
- $P \vee Q = \top$ if either $P = \top$ *or* $Q = \top$ otherwise $P \vee Q = \bot$ (or equivalently $P \vee Q = \bot$ if $P = \bot$ *and* $Q = \bot$ otherwise $P \vee Q = \top$)
- $\neg P = \bot$ if $P = \top$ and $\neg P = \top$ if $P = \bot$

In order to define propositions as types, we will need to consider, just like the counting numbers type $\mathbb{N}$ is populated with the counting numbers themselves, what is it that populates a proposition? We call these elements of the proposition type 'witnesses' or 'evidence'; you might think of the members of the proposition type "is the ground wet" as testimony that "I went outside and touched the grass and it made my hand wet", however a proposition's witness is never considered in doubt unlike real witnesses.

If we think of propositions as types which are populated (or indeed may not be if they are *false* and thus have no witnesses) then we know some things about what our logical operations have to be. `AND` must create a new type which has witnesses only when both of its argument types have witnesses, `OR` must have the witnesses of both types so that it is true if either is true, and `NOT` must be a type that somehow has witnesses when its argument type has no witnesses. In fact, we have actually already discussed one of these!

Definition proptypes.4 — (Product Type)

For two types $A : \texttt{Type}$ and $B : \texttt{Type}$, define their **product type** denoted $A \times B$, to be a new type with the right-associative constructor $, : A \to B \to A \times B$, the comma. In this way, elements $a : A$ and $b : B$ form an element $(a, b) : A \times B$. We also define the *projection operators* $\texttt{pr}_1 : A \times B \to A$ and $\texttt{pr}_2 : A \times B \to B$ which are $\texttt{pr}_1(a, b) = a$ and $\texttt{pr}_2(a, b) = b$.

This notation introduces an ambiguity, since we do not literally mean every comma we use in a formal context to be an element of the product type, but it has a value. That is, as a right-associative constructor, we have not only defined pair types but also triples, etc. since any (a, b, c, d) is to be read as nested pairs $(a, (b, (c, d)))$. To avoid confusion however, we use brackets where possible.

The product type also allows us to now formalize what we mean when we say curried and uncurried functions are equivalent. That is, we can always turn a function $f : A \times B \to C$ into a function $g : A \to B \to C$ and vice versa, because we can set $f\,(a, b) = g\,a\,b$ or $g\,a\,b = f\,(x, y)$ respectively.

Although it may not be obvious, the product type is exactly our `AND` operation, since to form an element of $P \times Q$ we require some $p\colon P$ and $q\colon Q$ to form $(p, q)\colon P \times Q$. For example, let P be the proposition that the ground is wet with witnesses which are days that the ground was wet, and Q be "I played basketball" with witnesses which are days I played basketball. Then $P \times Q$ is the type of pairs of a day the ground was wet together with a day I played basketball. If there was a day the ground was wet but not a day where I played basketball, such a pair cannot be formed. This is what gives it its logical 'and' structure; if either fails to have a witness, we will be left stuck in producing such a pair. We do not have an element of the product type $A \times B$ with merely $(a,)$, even now that we can curry. Although, knowing what we do about currying, it will also sometimes be appropriate to read a function $P \to Q \to R$ as not just 'if P then if Q then R' but 'if P and Q then R'.

Using this as inspiration, we can immediately figure out that the analogous `OR` operation will also have a more general use, that is, injecting elements of two types into the one.

Definition proptypes.5 — (Sum Type)

For two types $A, B\colon$ `Type`, define their **sum type** denoted $A + B$, to be the new type with constructors $l\colon A \to A + B$ and $r\colon B \to A + B$ (often called 'in left' and 'in right').

These constructors, like all constructors, have no additional structure other than that stated. l will take *any* $a\colon A$ and merely dump it in $A + B$, just as r will, just as **S** would take any natural number and count it up. This is the general case of the example we discussed earlier in which a function has an input type defined of 'either numbers or letters'. In this way, if we have even one of $p\colon P$ *or* $q\colon Q$, we will populate $P + Q$; we could just as well have Q be a type that is empty of witnesses and we will still have $l\,p\colon P + Q$ as witness. One could imagine this type as a hat; we would write down the days the ground was wet on pieces of paper, likewise with the days we played basketball, and throw them into the hat, forgetting which was which. Only on noticing that the hat is indeed not empty (i.e. it was wet or we played basketball on at least one day) that we may pull out the piece of paper and distinguish which type it originated from (via. was it l or r?). This is our logical `OR`.

Before we proceed to `NOT`, we must notice two things. First, that there is another kind of logical operation, one that does not appear in programming in quite so obvious a fashion, that is of great relevance to us. That is, we have been ignoring *logical implication* the entire time. This is perhaps because it has been obvious to us the whole time as well. Our logical implication operator, 'if P then Q', is the function type $\to$. When we say $P \to Q$ we mean to say that witness of one is witness of the other. For instance, if we use our ongoing example about wet ground and playing basketball, we could say that a witness $f\colon P \to Q$ is a witness that says "we play basketball whenever the ground is wet". Thus, on providing a day that the ground was wet, we could clearly say "ah, I remember now, we played basketball on that day when it was wet".

Second, we must think more about what it means for a proposition to be false. The primary thing we are trying to manouver around, given that a proposition type's members are witnesses, is the case where a type has no witnesses and is thus false. This prompts us to codify something we had not said earlier, which is the following:

Definition proptypes.6

Define the type $\top$ to have the constructor $*\colon \top$ which takes no arguments, and the type $\bot$ to be the type with no constructors. We call these the **unit type** and the **empty type**. We may also at times call these the *true type* and *false type*.

Our archetypal false type is explicitly the type that is empty, identified with all other types when they are empty, and thus we can say things like $P \times Q = \bot$ when $Q = \bot$. It is not necessarily so simple when a proposition is true however, since there are often multiple witnesses to a true proposition, and the unit proposition only has one constructor. Regardless, with this codified, we may now ask a question: what does $\to$ do to an empty type, i.e. what does it mean to imply false or derive from it? Let us examine this as removed from analogies for a moment.

Consider our responsibilities when defining a function: a function is a *map* which assigns to every valid input a valid output. So for a type A, the function type $\bot \to A$ is populated with what exactly? There are no valid inputs, and so we say that there exists immediately a *trivial function* $f\colon \bot \to A$. It is common to be confused when seeing this, but consider that from what we know about functions, we have no grounds with which to argue f is not a function; it *does its job* as a function, since in this case *it has no job to do*. There simply are no inputs to which we must find outputs. This has another consequence, which is that for all types P, the trivial function $\bot \to P$ will exist even if P is a false proposition with no witnesses; again, the space of outputs does not matter since there are no inputs. In fact, what we have described in a trivial function is the type theory version of the 'principle of explosion', often stated as 'if *false* then *anything*', with the trivial function as its witness.

The reverse is equally interesting: for a type A, just as $\bot \to A$ is *always* populated with the trivial function, $A \to \bot$ is *never* populated so long as A is populated. In this case, we have inputs for which we must find valid outputs, but no such outputs exist. The only way we can short circuit this is if indeed we are discussing $\bot \to \bot$, in which case there is still the trivial function. In fact, this function-type is our `NOT` operator.

Definition proptypes.7 — (Logical Negation)

For a type $A\colon$ `Type`, define the *function* $\neg\colon$ `Type` $\to$ `Type` by

$$\neg A \mapsto (A \to \bot).$$

Unlike our other logical operations, strictly speaking we have not defined a new type here; that was done back when we formally discussed $\to$. This is merely a function on types, in fact it is a shorthand for the curried $(\to \bot)$, rather than a constructor. It works as logical negation, as discussed earlier, since $\neg P$ is populated with the trivial function only when $P = \bot$ since if we have a witness $p\colon P$, we will have no outputs to assign for a function $f\colon P \to \bot$, thus falsifiying the function and leaving $P \to \bot$ empty. Returning to our example about days the ground was wet and days we played basketball, let Q be days we played basketball, but say that the proposition type is empty, i.e. we never played basketball, and thus $Q = \bot$. We can still construct $f\colon P \to Q$, or $f\colon P \to \bot$ rather, still saying "we played basketball on every day the ground was wet" just so long as there were no days the ground was wet.

This now makes it possible to discuss certain basic deductions of logic, the most basic *theorems*, and see that these will take the form of functions. It is my intention that above all else, what you take

away from this section is the ability to read a mathematical theorem as a kind of function, and so we have a preliminary example.

Theorem proptypes.8

Let P, Q: `Type` be proposition types. Then "if P or Q, then if not P, then Q", or

$$P + Q \to \neg P \to Q$$

i.e. "if the ground was wet or we played basketball, then if the ground was *not* wet, then we played basketball".

Proof.

Let f denote the witness of type $P + Q \to \neg P \to Q$ that we construct. It has as its input a member of $P + Q$ and outputs a function of type $\neg P \to Q$, or $(P \to \bot) \to Q$. Since $P + Q$ has two constructors, let us define f for both: if the constructor passed to f is $r : Q \to P + Q$ then it must have been passed $q : Q$ as $r\, q$. In that case, we have our witness of Q and may pass it directly, defining

$$f\,(r\,q) = (g \mapsto q)$$

where $g : \neg P$ is the evidence that $P = \bot$.
Perhaps counter-intuitively, we must now consider the case where f is given as its input $l\, p$ for some $p : P$ and then some $g : P \to \bot$ as well, the self contradicting case where both P and $\neg P$ are true, which we call the *counterfactual* (meaning roughly 'the case that was not true'). That is, we are above all defining a *function*, and so no matter our intuitions about logic, we must put them aside and say *what the function does*. This is not a problem for us however. The $\bot$ type is empty, however since we are in a case that should never happen, we now have evidence of *false* and can prove anything we like, including our proof goal. We write this as

$$f\,(l\,p)\,g = h\,(g\,p)$$

where $h : \bot \to Q$ is the trivial function. This can be interpretted as "If P or Q via P, and P implies *false*, then since we know P, imply *false*. Since *false* implies anything, imply Q". In the process of this we have momentarily allowed there to be a fictitious element of the *empty* type, as is natural in the counterfactual case.
Then for any $P + Q$ we have constructed $f : P + Q \to \neg P \to Q$. □

Part of the power of using type theory is that, as discussed before but more obvious now, a function (and thus a theorem) is defined so long as one can account only for every constructor the function may receive. This case is simple, asking what constructors $P + Q$ could have and then taking any member of $\neg P$ absent a question about which constructor it is exactly, however as we will see going forward, theorems of arbitrary complexity will yield to this kind of case analysis since we are merely fulfilling our obligation to provide an output for every input. Thus the question 'which witnesses did we receive?' is merely to divide our sets of inputs/witnesses/circumstances into classes which may individually have easier proofs.

Before moving on, note that we have introduced a split in the notation of constructive mathematics and classical mathematics: in type theory, propositions and their operations exist at the same level as objects, and use the same tools. As our intention here is to proceed, after this section, to classical mathematics, it would be better practice to give our operations appropriate aliases so that we are not confused in future.

Notation proptypes.9

When they act on propositions, define the following symbols as aliases for their non-propositional counterparts:

- $\wedge := \times$ read as 'and',
- $\vee := +$ read as 'or',
- $\Rightarrow := \to$ read as 'implies', e.g. $P \Rightarrow Q$ is the type of functions taking witnesses $p\colon P$ and producing some witness $q\colon Q$, along with $(P \Leftarrow Q) = (Q \to P)$,
- $(P \Leftrightarrow Q) := (P \to Q) \times (Q \to P)$, that is, we say P 'if and only if' Q when we can find as witness a function $P \to Q$ as well as a function $Q \to P$ so that both propositions imply one another.

proptypes.3 Dependent Types and Example Proposition Types

We have of course not yet given an example of a real proposition type with its own constructors yet, and the reason for that is this: while we have the tools to discuss a proposition type now, we don't have the tools to discuss *whole classes* of propositions, and thus to deduce anything that could actually have generality or be useful. The unit type and empty type are technically examples of propositions, and yet they do not *say* anything. What we need, more than the ability to discuss witnesses or implications about the ground being wet or having played basketball, is a reliable way to discuss witnesses which specify *which day* the ground was wet or *which day* we played basketball. It is not on its own useful to deduce that that because the ground was once wet *ever* that we did play basketball, we need tools that allow us to discuss propositions like "we play basketball the day after it rains".

Since we are preparing to speak about mathematics, most often of all, the things we will have to *say* with a proposition is to speak of equality, this equals that. And yet equality is not useful if we must manually define a new type for every pair of items we want to claim are equal. We can of course make functions between types, as seen by $\to$, but, consider for equality of natural numbers, we need something of the form $\mathbb{N} \to \mathbb{N} \to$ `Type`. And yet even in that case, we will certainly need constructors that are somehow dependent on the arguments, lest we populate a proposition type that says any random pair of numbers is equal.

Let us think then about what properties equality has that could found its type. In mathematics, we say that a 'relation' in general describes an equivalence when it has three properties: every thing is equal to itself (reflexivity), an equality is the same backwards as it is forwards (symmetry), and equalities can be chained together to make new equalities (transitivity). A broader discussion of equivalence relations will appear in the following section, however for now, one of these does

indeed look like a constructor.

Definition proptypes.10 — (Equivalence Type)

If $A\colon \texttt{Type}$ and $a, b\colon A$, then define by $a \equiv b$, read as 'a equals b', a new type with constructor

$$\texttt{refl}\colon (a\colon A) \to (a \equiv a).$$

There are a number of peculiar things about this type, least of all how to interpret what it means. First, to avoid confusion, we have used the symbol $\equiv$ rather than $=$ in order to separate the rules of $\equiv$ from out existing notions of equivalence; instead we will deduce them. Second, its constructor doesn't look like a function in the way we are familiar, since now the type of output depends on the individual input rather than merely its type, i.e. $(\texttt{refl}\ a)\colon (a \equiv a)$ and $(\texttt{refl}\ b)\colon (b \equiv b)$ are members of separate, mutually incomparable types. The constructor we've given this represents the expectation that equality relations are *reflexive*, so what we've really said is 'in the case where b is both equal to a and literally the symbol a, $a \equiv b$ has constructor $\texttt{refl}\ a$'.

But we cannot use this yet; the same issue that constructors for this type have, a bizarre dependency of their output *type* on their input *element*, is also necessary for any theorem that speaks about equality. Otherwise, more than just having to make each equality type one at a time, we'd have to define a new version of each theorem for each equality type used. We must generalize this concept of dependent function types.

Definition proptypes.11 — (Π-Types, or Dependent Functions)

Let $A\colon \texttt{Type}$ and let B be some family of types, a collection of types each with roughly the same constructors which each use some $a\colon A$ as a parameter. In that case, we write $B\colon A \to \texttt{Type}$ and define the Π-type as the *dependent function* type, written

$$\prod_{x:A} B(x) := (x\colon A) \to B(x)$$

where Π is the capital greek letter 'pi'. In practice, let us also write what is on the right hand side of this definition to denote the dependent function type.

This construction allows us to do something we really have not been able to before. Before, for instance, we could speak of a function which takes letters of the alphabet to words that start with that letter, like $a \mapsto$ Apple or $b \mapsto$ Banana, and in that way, functions have always *depended* on their inputs. But now, we can speak of a function that maps a to a *particular* apple in the type `Apple` or b to a particular banana in the type `Banana`. It is called a dependent function precisely because the *type* of the output now changes depending on the particular input given. To map that onto the formalization above, we might say that A is the type of alphabetic letters, and B is a family of types of foods which each start with a different letter of the alphabet.

This may seem abstract, but in fact the reason we are discussing it in the first place is because it concretizes a much more natural concept; when $P\colon \mathbb{N} \to \texttt{Type}$ is a family of propositions depending on a natural number, we can speak of $n\colon \mathbb{N}$ where $P(n)$ is true; this is the type theoretic form of a *condition* on n. This allows us to do two things very interesting things with respect to functions. We can speak of a function of type $(n\colon \mathbb{N}) \to P(n) \to Q(n)$ which, since it takes a witness of $P(n)$ as

an argument, is a statement of the form 'when P is true for n then Q is true for n'. If instead we have a function of form $(n\colon \mathbb{N}) \to P(n)$ then we have promised that we can reliably produce a witness that condition P is true for any n we give; we have said 'for all $n\colon \mathbb{N}$, $P(n)$ is true'.

In effect, we have gained the ability to speak about propositions in generalities. To say that 'in an entire class of circumstances, it remains true that...'. This concept is so ubiquitous in mathematics that there will be very few sections of this text which do not refer to it literally or implicitly.

Notation proptypes.12 — (For All)

When $A\colon \texttt{Type}$ and $P\colon A \to \texttt{Type}$ is a family of proposition types, define the notation

$$\forall x : A, P(x) := \prod_{x:A} P(x)$$

as the propositional logic version of the Π-type, reading the symbol $\forall$ as "for all". It is also called in some settings the "universal quantifier".

Here, the deeper value of the type theory metaphor for propositional logic begins to show itself. In mathematics, we tend to use Π in a similar way as above to denote large repetative multiplications, which we sometimes call 'product notation'. This is in fact entirely apt as a choice of notation against the backdrop of our other names. Consider what it means to have a *function* with a dependent output type: we require of a function that for every input there is an output, and so, what it means to have found a witness for a dependent function *proposition* type is that we have a member of *every* type in the type family $P\colon A \to \texttt{Type}$.

Said another way, imagine a type family dependent on the unit type, $Q\colon \top \to \texttt{Type}$. Since $\top$ only has one element, the $*$ element, there is only one type $Q(*)$ in the type family, and so $\prod_{*:\top} Q(*)$ is still a type of functions, but with no real *dependency* left. Nonetheless, an element $f\colon (* : \top) \to Q(*)$ still selects an element of $Q(*)$ to be its output, $q = f(*)$. Then the *data*, the information that could be said to be contained in f, the choices required to define it, lie only in the choice of $f(*)\colon Q(*)$. What we have done in finding a member $f\colon \prod_{*:\top} Q(*)$ is merely to find a member $q\colon Q(*)$. Now consider, for a type, let's call it $\texttt{Bool}$ with only two members $\texttt{on}\colon \texttt{Bool}$ and $\texttt{off}\colon \texttt{Bool}$; then the dependent type $\prod_{b:\texttt{Bool}} P(b)$ contains functions that still assign an output to every input, and so it finds a witness for both $P(\texttt{on})$ and $P(\texttt{off})$. The data required to define a function $f\colon \prod_{b:\texttt{Bool}} P(b)$ is then contained in each element of $P(\texttt{on}) \times P(\texttt{off})$, since each $(p_{\texttt{on}}, p_{\texttt{off}})\colon P(\texttt{on}) \times P(\texttt{off})$ corresponds to a possible f which is defined $f(\texttt{on}) = p_{\texttt{on}}$ and $f(\texttt{off}) = p_{\texttt{off}}$. If extended, the conclusion one reaches is that $\forall(n\colon \mathbb{N}), P(n)$ is to be read as $P(0) \times P(1) \times P(2) \times ...$, i.e. "for all n, we have $P(n)$" is an identical statement to "$P(0)$ and $P(1)$ and $P(2)$ and ...". In fact, this interpretation is ostensibly correct, and the intuitions garnered from it generalize.

Holding that thought for a moment, we must first complete our suite of tools. If we are to think of the Π-type as generalizing the notion of a product type to entire families of types, then can we generalize the sum type to an entire family of types? Let us think for a moment: if a dependent function generalizes the product type since there must exist an output in each type of the type family to correspond to an input, then to generalize the sum type means to say there exists at least one type with a witness amongst an entire family of types. The tool we are looking

for is a *dependent pair* type.

Definition proptypes.13 — (Σ-Type, or Dependent Pairs)

Let $A\colon \texttt{Type}$ and $B\colon A \to \texttt{Type}$ be a family of types, each which take a member $a\colon A$ as a parameter. We define the Σ-type as the *dependent pair* type, written

$$\sum_{x:A} B(x),$$

where Σ is the capital greek 'sigma', to be the type of all pairs (a, b) where $a\colon A$ and $b\colon B(a)$, with the type of the second term depending on the individual choice of element of the first in A. Analogously to the product type, we have the comma constructor

$$,\, : (a\colon A) \to B(a) \to \sum_{x:A} B(x).$$

It is of course possible to set $B\colon A \to \texttt{Type}$ to be some constant family, say, a type $C\colon \texttt{Type}$, in which case the Σ-type trivializes back down to the standard pair, $A \times C$. This is important however; as we discussed earlier about currying, a pair argument (a, b) for a function $A \times B \to C$ is the same as a curried function $A \to B \to C$. But then we defined dependent functions that could be $(a\colon A) \to P(a) \to Q(a)$; can we uncurry those? With the dependent pair, we can try writing $\sum_{a:A} P(a) \to Q(a)$ but this would be very bad practice. The natural currying of dependent functions allows us to speak of $(a\colon A) \to P(a) \to Q(a)$ as a dependent function where indeed the family of types $A \to \texttt{Type}$ is $P(a) \to Q(a)$, or $\prod_{a:A} \big(P(a) \to Q(a)\big)$ with the entire function type as dependent on a; if we hide this a in a dependent pair, it is not reasonable to say that it suddenly starts behaving as a dependent function argument later.

The principle is however, sound in a sense. It is reasonable to desire that in the way $P \to Q \to R$ can be read as 'if P and Q then R' via currying, that we might use dependent pairs to speak of $\sum_{n:\mathbb{N}} P(n)$ as 'we have a number n and we know that $P(n)$'. In fact, if we have $(a\colon A) \to P(a) \to Q$ where Q is merely a proposition rather than a family of propositions, we can in fact uncurry this to 'if we have a number n and $P(n)$ is true then Q'. This should be peculiar in a sense; if Q does not depend on a choice of n, then we are saying that Q is implied true merely because there exists a $n\colon \mathbb{N}$ which satisfies the condition P, without care of which number n is. This an extremely useful concept.

In propositional logic, the Σ-type $\sum_{x:A} B(x)$ describes the quantifier 'exists', the same one as in statements like "for all n, there exists a prime number p which is greater than n". In fact, pending a definition of $>$ (greater than) and letting $P(n)$ be the proposition that n is prime, we could now describe that proposition literally by the type $\prod_{n:\mathbb{N}} \sum_{p:\mathbb{N}} P(p) \wedge (p > n)$. Such a pattern of pairing 'for all' with 'there exists' is a common one, however, it is usually written with the following notation.

Notation proptypes.14 — (There Exists)

When $A\colon \texttt{Type}$ and $P\colon A \to \texttt{Type}$ is a family of proposition types, define the notation

$$\exists x : A, P(x) := \sum_{x:A} P(x)$$

as the propositional logic version of the Π-type, reading the symbol $\exists$ as "there exists" and the comma as "such that". When there exists exactly one $a\colon A$ with a populated $P(a)$ type, we say that $\exists!x\colon A, P(x)$, reading $\exists!$ as "there exists uniquely". $\exists$ is also called in some settings the "existential quantifier".

It should also be mentioned at least briefly that although in type theory, all objects must belong only to one type, dependent pairs give us the closest thing we have to a subtype. Consider for example, the type

$$\sum_{n:\mathbb{N}} \exists(m\colon \mathbb{N})\ (n \equiv m + m)$$

The use of Σ here together with $\exists$ is a purely notational choice (indeed, the symbols are intended to be equivalent as just defined) but we write it this way because we are in fact interested in the particular $n\colon \mathbb{N}$. Then in fact all $n\colon \mathbb{N}$ for which there exists a pair of type $\exists(m\colon \mathbb{N})(n \equiv m+m)$ must be the *even numbers*, since we will be unable to form such pairs if n is odd, and thus no m with $2m = n$ exists. That is, our closest notion of a subtype of all $a\colon A$ that satisfy a proposition family $P\colon A \to \texttt{Type}$ is exactly the dependent pairs of $a\colon A$ together with the evidence that they satisfy P.

Together, the operations we have described above show how this dependent type theory, one that exists in computer programming languages with operations and objects $\to, \times, +, \top, \bot, \Pi, \Sigma$, has a one-to-one mapping with the logical operations used in mathematics, $\Rightarrow, \wedge, \vee$, true, false, $\forall, \exists$. This is a consequence of the Curry-Howard correspondance, often stated simply as "proofs are programs".

proptypes.4 Formal Theorems

The whole reason we are discussing type theory is to recontextualise what it means to prove a theorem; what we do in pure mathematics often has less in common with the deductions of a detective eliminating possible stories and more in common with a programming problem. That is, although we are certainly making deductions, the manner in which one does this in every day life can sometimes lead one to think logical statements can be weakened or strengthened with synonyms or by persuading oneself with some argumentative sleight of hand. Although for much of this text we will write proofs in the form of worded deductions, those deductions are written each as the application of some function that transforms a logical context.

To that end, it helps to see propositions as not merely statements to be read as true or false, and theorems as statements which are conditionally true, but that propositions have witnesses, and those theorems *act* on those witnesses and are incomplete if every single hypothesis is not proven. Moreover, we do not accept a statement merely because it seems like it should be true, but because we have a transformation of existing witnesses to propositions that produces a new witness that the statement is true. With addition to some core axioms, we hold ourselves to this high a standard in mathematics.

This also means that everything you think you know to be true about the typical mathematical objects, rules like $a+(b+c) = (a+b)+c$ are, if not assumed, propositions that have explicit witnesses. It is no longer enough to expect that things *should* be true, or *ought* to be true of mathematical objects or objects which we intend to describe things in the real world. If they are true, then they have witnesses. This will expose certain things about how the mathematical mode of thought differs from the logical one, or rather, that many things one believes to be true about mathematics are not necessarily true but are in fact distinct choices we make about the system of logic we work in.

Let us begin with some proofs of familiar concepts.

Theorem proptypes.15 — (Transitivity of Equality)

Let $A\colon \texttt{Type}$ and $a, b, c\colon A$. If $a \equiv b$ and $b \equiv c$ then $a \equiv c$, or

$$\forall(a, b, c)\colon A \times A \times A,\ (a \equiv b) \Rightarrow (b \equiv c) \Rightarrow (a \equiv c)$$

has a witness.

Let us digress briefly to discuss how the statements of theorems correspond to their propositional counter parts. In mathematics we will often see statements of this form, in particular 'Let a, b, c be of type A. If $a \equiv b$ and $b \equiv c$ then $a \equiv c$'. Obviously the proposition 'if $a \equiv b$ and $b \equiv c$ then $a \equiv c$' doesn't make a lot of sense on its own since we must specify what a, b, c we are talking about, and this is done by 'let a, b, c' and interpretted as $\forall(a, b, c)\colon A \times A \times A$.

But since most theorems do not have their propositional forms written symbolically, we must be able to read which parts of the theorems are contextualised and which are not. For instance, when we say $a, b, c\colon A$, we do this only after we have said 'let $A\colon \texttt{Type}$'. Perhaps then it would be more appropriate to first write $\forall(A\colon \texttt{Type})$ but this would be superfluous since we can't even state the theorem without a choice of type in the first place. Similarly, we may then only write the theorem as $(a \equiv b) \Rightarrow (b \equiv c) \Rightarrow (a \equiv c)$ since we have already contextualised that a, b, c must be of the same type.

It is common in mathematics to drop conditions from statements of the form 'if ... then ...' and instead propose them first as statements 'let ...'; since our theorem can only be applied in a relevant context anyway, our *let* statements turn arguments into contextual requirements for purely notational clarity. While there exists a broader study of logical context, its rules and manipulations (and if you are interested in this, look into the *turnstile*), this study exists to split more hairs than even dependent type theory. Contextual manipulations are, as far as we are concerned, equivalent to additional 'for all' arguments.

Proof.

We are to define a dependent function f which takes a triple $(a, b, c)\colon A \times A \times A$ and then a witness of $a \equiv b$ and of $b \equiv c$. Presuming we are supplied an appropriate triple, we then need to define

$$f(a, b, c)\colon (a \equiv b) \to (b \equiv c) \to (a \equiv c).$$

Presume also then that we are given a witness $p\colon (a \equiv b)$. Now we will *pattern match* on the remaining argument, that is, since there is only one constructor for the equality type,

we know that an argument of type $b \equiv c$ will be supplied as the witness $\texttt{refl}\, b \colon (b \equiv b)$; this is only as appropriate, since equality types only have constructors, and thus are populated, when the left hand side of the equality *is equal* to the right side. And if it were not so, then no witness would be supplied at all (this is indeed a power of the inductive type). In this way, by merely asking what constructor of $b \equiv c$ would be supplied, we have immediately shown that what we are proving could only be the following

$$f(a,b,b) \colon (a \equiv b) \to (b \equiv b) \to (a \equiv b).$$

This is substantially easier job, since we may define f as

$$f\ (a,b,c)\ p\ (\texttt{refl}\, b) = p$$

since, by requiring a witness of $b \equiv c$ and thus ensuring c can only literally be b, our final output must be a witness $a \equiv b$ just as our first witness $p \colon (a \equiv b)$. □

The mode of reasoning in this proof also applies immediately to certain things about numbers. For instance, the idea that $m \equiv n$ implies $m + 1 \equiv n + 1$ is immediate in this manner: a function

$$\forall (m,n) \colon \mathbb{N} \times \mathbb{N}, (m \equiv n) \Rightarrow (\mathbf{S}(m) \equiv \mathbf{S}(n))$$

has its witness easily since in the moment that we pattern match on $m \equiv n$, we see that the only constructor could have been $\texttt{refl}\ n$, and thus the case we are proving is exactly

$$(n \equiv n) \Rightarrow \big(\mathbf{S}(n) \equiv \mathbf{S}(n)\big)$$

with witness given trivially by $\texttt{refl}\ (\mathbf{S}(n))$. This is the extended utility of the *inductive type*. The mere fact of saying that a function is defined if we know what output to assign to every case of constructor inputs is equivalent also to eliminating the cases where our prerequisite propositions were untrue. In this instance, the only way we could say the relatively trivial solution $\texttt{refl}\ (\mathbf{S}(n))$ fails to be a solution is if our final type were not $\mathbf{S}(m) \equiv \mathbf{S}(n)$. And yet it must be that type since evidence of $n \equiv m$ was supplied, and the only form that evidence could have been was $\texttt{refl}\, n$, meaning $n \equiv m$ must have been $n \equiv n$.

But our system can do much more than just make things we knew intuitively to be true much more burdensome to prove, it can make things which are often difficult for students to comprehend become equally as concrete as those we knew to be obvious already. The famous 'proof by induction' is brought into crystal clarity by our technique, and in fact *deduced*.

Theorem proptypes.16 — (Proof by Induction for $\mathbb{N}$)

Let $P \colon \mathbb{N} \to \texttt{Type}$ be a type family. If $P(\mathbf{Z})$ has witness and there is a function $\forall (n : \mathbb{N}), P(n) \to P(\mathbf{S}(n))$, then we have proven $\forall (n : \mathbb{N}), P(n)$. That is, there is a witness for

$$\forall (P \colon \mathbb{N} \to \texttt{Type}), P(\mathbf{Z}) \Rightarrow \big(\forall (n : \mathbb{N}), P(n) \Rightarrow P(\mathbf{S}(n))\big) \Rightarrow \forall (n \colon \mathbb{N}), P(n).$$

Proof.

o complete this proof, we will need to speak of two functions: f, the witness of the theorem (type omitted for length) and $g \colon \forall (n : \mathbb{N}), P(n) \Rightarrow P(\mathbf{S}(n))$, the function we require that

can propogate a proof of $P(n)$ to its successor $P(\mathbf{S}(n))$. Although it may not be obvious by the statement of the theorem, by the right-associativity of function types (yes, even dependent ones), we will also be able to provide $n\colon \mathbb{N}$, defining our function f by the final output of type $P(n)$ it produces.
Let us pattern match on n: it must either be $\mathbf{Z}$ or $\mathbf{S}(m)$ for some $m\colon \mathbb{N}$. In the first case, we are trying to produce a member of $P(n) = P(\mathbf{Z})$, for which an evidence is already provided as prerequisite. So we may define

$$f\ P\ p\ g\ \mathbf{Z} = p$$

where $p\colon P(\mathbf{Z})$ and the other arguments are defined above. In the case of $\mathbf{S}(m)$, we write

$$f\ P\ p\ g\ \mathbf{S}(m) = g\ m\ (f\ P\ p\ g\ m)$$

defining a recursive function just as we had done for addition. What we have done on the right hand side is this: since $g\colon \forall(n : \mathbb{N}), P(n) \Rightarrow P(\mathbf{S}(n))$, we provide it with $m\colon \mathbb{N}$ so that we are discussing $g(m)\colon P(m) \Rightarrow P(\mathbf{S}(m))$. To complete the proof, we need only hand $g(m)$ some proof of $P(m)$, which we hand off recursively to f itself in its specialized form, $f\ P\ p\ g\ m\colon P(m)$ as we described above. In this way, we hand the obligation of actually proving something off through recursion until finally the tally marks are taken off and we are in fact proving $f\ P\ p\ g\ \mathbf{Z}\colon P(\mathbf{Z})$, which has witness p as was provided. □

Proposition proptypes.17

We can use proof by induction to extend our earlier observation that $a \equiv b$ implies $\mathbf{S}(a) \equiv \mathbf{S}(b)$ to a general form that $b \equiv c$ implies $b + a \equiv c + a$ for all $a\colon \mathbb{N}$. That is, we find a witness of

$$\forall a, \forall b, \forall c, (b \equiv c) \Rightarrow (b + a \equiv c + a)$$

by setting it as our $P\colon \mathbb{N} \to \texttt{Type}$ and performing induction on it. The case $a = \mathbf{Z}$ is covered immediately, recalling the inductive definition $\texttt{add}(n, \mathbf{Z}) = n$, i.e. $P(\mathbf{Z})$ is in fact the type $\forall b, \forall c, (b \equiv c) \to (b \equiv c)$, proven trivially by passing the witness along. Next we need $\forall(n : \mathbb{N}), P(n) \to P(\mathbf{S}(n))$, specialized to our case as

$$\begin{aligned} \forall(n\colon \mathbb{N}), \quad & \forall b, \forall c, (b \equiv c) \Rightarrow (b + n \equiv c + n) \\ \Rightarrow \quad & \forall b, \forall c, (b \equiv c) \Rightarrow \big(b + \mathbf{S}(n) \equiv c + \mathbf{S}(n)\big). \end{aligned}$$

But again we may apply our inductive definition of addition $\texttt{add}(m, \mathbf{S}(n)) = \texttt{add}(\mathbf{S}(m), n)$, and so the output proposition for this is in fact

$$\forall b, \forall c, (b \equiv c) \Rightarrow (\mathbf{S}(b) + n \equiv \mathbf{S}(c) + n).$$

Now recall our observation that $(m \equiv n) \Rightarrow (\mathbf{S}(m) \equiv \mathbf{S}(n))$, which we write in our case with witness function $g\colon \forall b, \forall c, (b \equiv c) \Rightarrow (\mathbf{S}(b) \equiv \mathbf{S}(c))$. Since our inductive function gives us as hypothesis, the witness $h\colon \forall b, \forall c, (b \equiv c) \Rightarrow (b + n \equiv c + n)$ (i.e. this is the $P(n)$ from which we must show $P(\mathbf{S}(n))$, we can now specialize that as $\big(h\ (\mathbf{S}(b))\ (\mathbf{S}(c))\big)\colon \big(\mathbf{S}(b) \equiv \mathbf{S}(c)\big) \to \big(\mathbf{S}(b) + n \equiv \mathbf{S}(c) + n\big)$. We can now chain these together

$$\Big(h\,(\mathbf{S}(b))\,(\mathbf{S}(c))\,(g\,b\,c)\Big) : (b \equiv c) \Rightarrow \big(\mathbf{S}(b) + n \equiv \mathbf{S}(c) + n\big)$$

forming our *inductive step*. Then (and you should check this by looking at the proof by induction's form) we have satisfied the requirements for applying the theorem form of proof by induction, and we deduce $\forall a, \forall b, \forall c, (b \equiv c) \Rightarrow (b + a \equiv c + a)$.
If we use f to denote the inductive rule then the full proof can be read monolithically as

$$\begin{aligned} &f\,\big(\forall a, \forall b, \forall c, (b \equiv c) \Rightarrow (b + a \equiv c + a)\big) \\ &\quad (b \mapsto c \mapsto p \mapsto p) \\ &\quad \big(b \mapsto c \mapsto h\,(\mathbf{S}(b))\,(\mathbf{S}(c))\,(g\,b\,c)\big). \end{aligned}$$

In practice, we will rarely reason about mathematical objects themselves at the type constructor level. While it is a useful tool for being absolutely certain no logical error has been made, we typically try to adjust our intuitions so that we can reason such steps automatically. That is not to say, in this moment at least, that the rigidity we expect of pure mathematics is any weaker, but that statements such as $a = b \Rightarrow a + c = b + c$ is one we train our intuitions on so that we know automatically how to make *algebraic* deductions.

When we lack such intuitions, or when we know our intuitions are at risk of failing us, this system or modes of reasoning like it are there to catch us. This is often the case when we speak about the various refactorings we can do to statements to make them easier to prove related to negation. It is a common mistake for students to set out to prove by contradiction, only to structure the negated form of the proposition incorrectly and make the proof trivial but wrong. With the benefit of this system, we will not be making such mistakes.

Theorem proptypes.18 — (Some of De Morgan's Laws)

Let P, Q: `Type` be propositions, and let A: `Type` with R: $A \to$ `Type` a type family. Then we have the following:

- $\neg P \vee \neg Q \Rightarrow \neg(P \wedge Q)$ i.e. 'if P or Q is false then they are not both true'
- $\neg P \wedge \neg Q \Rightarrow \neg(P \vee Q)$ i.e. 'if both are false then not one of them are true'
- $\neg(P \vee Q) \Rightarrow \neg P \wedge \neg Q$ i.e. 'if not either then both false'
- $\neg(\exists(a\colon A), R(a)) \Rightarrow \forall(a\colon A), \neg R(a)$ i.e. 'if not one example, then for all false'
- $\forall(a\colon A), \neg R(a) \Rightarrow \neg(\exists(a\colon A), R(a))$ i.e. 'if for all false, then there does not exist an example'
- $\exists(a : A), \neg R(a) \Rightarrow \neg(\forall(a : A) \to R(a))$ i.e. 'if counterexample then not for all'

Many of the proofs for this theorem will depend on applying right-associativity, since if the final type we want to show is of form $\to \neg P$, then this is the same as $\to P \to \bot$. Just like in our previous proofs involving negation, we can indulge a counterfactual case, however proofs ending in negation are special in a certain respect. That is, when we engage the counterfactual, we do not need to apply the principle of explosion (the trivial function) in order to obtain any proposition we want, because our final proposition type is $\bot$. Said another way, we can reinterpret $\to \neg P$ as $\to P \to \bot$ and supply a witness p: P, and this is equivalent to saying "P is false and we will show it's

false by showing absurdity is derived from it being true". This is a very specific form of proof by contradiction.

This proof will be the longest we've ever seen before, a record to be almost immediately broken again. For that reason, it is on the website displayed as a minimized box by default. I do recommend looking over this proof, so please click it.

Proof.

- Let us proceed as if P, Q were normal types. Then we read $\neg P \vee \neg Q = (P \to \bot) + (Q \to \bot)$ and $\neg(P \wedge Q) = P \times Q \to \bot$. Together then, we must find a witness function of the type

$$(P \to \bot) + (Q \to \bot) \to (P \times Q \to \bot).$$

  Let us call this witness f. Then to pattern match f for its argument in $(P \to \bot) + (Q \to \bot)$, we must define its procedure in when it receives either $l\, g$ where $g \colon P \to \bot$ or $r\, h$ where $h \colon Q \to \bot$. In either case, recalling the right-associativity of $\to$, we may then define its behaviour as a function with output in $\bot$ if it receives an input $(p, q) \colon P \times Q$ thus solving in the counterfactual. Then, for example, we may extract the desired value p and apply it to our function $g \colon P \to \bot$. Thus f is defined

$$f\ (l\, g)\ (p, q) = g\, p$$
$$f\ (r\, h)\ (p, q) = h\, q$$

- Once again, note that the desired statement is written fully as

$$(P \to \bot) \times (Q \to \bot) \to (P + Q \to \bot).$$

  with the benefit of familiarity with the previous proof, we must define the behaviour assuming we are given an argument $(g, h) \colon (P \to \bot) \times (Q \to \bot)$ and, using right-associativity, let us say we are also given an argument $P + Q$ which must be pattern matched as either $l\, p$ or $r\, q$ for $p \colon P$ or $q \colon Q$ respectively. Then we can define f as

$$f\ (g, h)\ (l\, p) = g\, p$$
$$f\ (g, h)\ (r\, h) = h\, q$$

  a near re-ordering of the previous proof.

- This statement is written as

$$((P + Q) \to \bot) \to (P \to \bot) \times (Q \to \bot).$$

  Its witness, which we once again call f, will take as an argument $g \colon P + Q \to \bot$. Our output must be a pair of functions that take some $p \colon P$ to $\bot$ and some $q \colon Q$ to $\bot$. Since we can inject either into $P + Q$ with l or r, this is simple.

$$f\ g = \Big(\big(p \mapsto g\ (l\ p)\big), \big(q \mapsto g\ (r\ q)\big)\Big)$$

- Despite the different symbols, this is remarkably similar to the previous proofs. First, observe what we want to prove is really

$$\left(\sum_{a:A} R(a) \to \bot\right) \to (a : A) \to (R(a) \to \bot)$$

with $\sum_{a:A} R(a)$ the type of dependent pairs (a, r) with $r\colon R(a)$. Then, once again providing our witness f with arguments $g\colon \sum_{a:A} R(a) \to \bot$, $a\colon A$ and $r\colon R(a)$ just as in previous proofs, we define f as

$$f\ g\ a\ r = g\ (a, r)$$

since the pair (a, r) is exactly the argument g needs to produce an element of $\bot$, completing the counterfactual.

- Repeating our procedure, what we aim to prove is really

$$\big((a\colon A) \to (R(a) \to \bot)\big) \to \left(\sum_{a:A} R(a) \to \bot\right).$$

So we will provide our proof f with $g\colon (a\colon A) \to R(a) \to \bot$ and a pair $(a, r)\colon \sum_{a:A} R(a)$. We then have immediately

$$f\ g\ (a, r) = g\ a\ r$$

thus providing our fictitious element of $\bot$ and completing the theorem.

- The non-propositional form is

$$f\colon \left(\sum_{a:A} R(a) \to \bot\right) \to \big((a : A) \to R(a)\big) \to \bot$$

so presume the arguments are supplied (a, g) with $a\colon A$ and $g\colon R(a) \to \bot$ and $h\colon (a : A) \to R(a)$. We need simply specialise h on a to obtain an element of $R(a)$, which we can then apply to g to obtain a member of $\bot$.

$$f\ (a, g)\ h = g\ (h\ a).$$

□

Notice however that these four statements we prove are clearly not all the logical statements we tend to consider about negation. For instance, we cannot prove 'if not both then one is false' $(\neg(P \wedge Q) \Rightarrow \neg P \vee \neg Q)$ because our system has hamstrung us. Or more interestingly, it has in fact revealed to us that there are things we believe to be true about logic that are not true *a priori*.

Consider for example, the concept of proof by contradiction. Since $\neg\neg P$ is read as 'not not P' you would be forgiven for thinking that it is true only when P is true, but that is not the

case. In fact, to believe $\neg\neg P \Rightarrow P$ is a strong statement, implying in particular that you believe all propositions are either true or false ($\forall(P\colon \texttt{Type}), P \vee \neg P$), a property called 'being *decidable*'. In our every day life, this seems to reflect our experiences, and yet there are rare cases when a question must be undecidable, as is famously the case in Alan Turing's solution to the Halting Problem.

In fact, to consider dependent type theory as the basis of logic would force one to accept a bizarre truth: that there is more than just true or false, but a secret third thing, the *not false*. We can prove (and this is a fun exercise you are now well equipped to do) that $\neg\neg\neg P \to \neg P$, that is, 'not not not P implies not P', but we cannot do anything about $\neg\neg P$. So if we (circularly) think of $\bot = \neg\top$, then we must accept that there is in fact a third outcome, that of $\neg\bot = \neg\neg\top$ in a distinct and meaningful way where there is no fourth (since $\neg\neg\bot = \neg\neg\neg\top$ would be $\neg\top$ as above). We must ask ourselves, is this the system of logic we want to work within? Does this reflect our goals in mathematics?

proptypes.5 The Problem with Constructive Mathematics

To believe that all propositions are either true or false is in fact a distinct **axiom**, called the *Law of Excluded Middle* (LEM) in this case, a statement which we consider to be true and do not question precisely because it is part of the foundation of our system of reasoning. To claim an axiom in one's reasoning is a profoundly dangerous step that must be exercised with caution; presume the wrong thing, and you will end up in a counterfactual, but axiomitise the wrong thing, and your entire logical system is vulnerable to the principle of explosion.

And yet if it is dangerous to believe this axiom since it has clear contradictions, why is it that proof by contradiction is commonly accepted in mathematics?

Here we meet the difference between mathematics and logic; there is a sense in which mathematicians are more interested in the metaphysical behaviour of the objects they study than the logical soundness of them. This is not to say that mathematicians are willing to accept contradictions (in fact they are repulsed by them!) but it is just not their primary concern. To a mathematician, the fact that almost every object they encounter is decidable is enough to consider these rare cases the exception to the rule, i.e. statements are decidable until proven or cautioned otherwise.

Moreover, this more metaphysical interest means that there are circumstances when even mathematicians, looking at a statement that agrees with their metaphysical understanding of an object, will be tempted to say 'this seems true enough'. Such is famously the case with the *axiom of choice*.

This axiom is famously difficult to comprehend, and many on encountering it are unclear on why exactly it is an axiom which must be asserted rather than a straight forward logical operation. To us, with the benefit of type theory, we see the axiom of choice as a clear assertion of type

$$\Big(\forall(a : A), \exists(b : B), P(a, b)\Big) \;\to\; \exists(f : A \to B), \forall(a : A), P(a, f(a))$$

given a family of types $P\colon A \to B \to \texttt{Type}$. We read this statement as "if for all a there exists a b so that $P(a, b)$ is true, then there exists a *choice* function f which for all a will choose the $b = f(a)$ satisfying $P(a, b)$". Yet even here you may begin to see why such a statement makes one question why it is an axiom: if we know the b exists, then surely it is right there, and we can just *choose it*, right? But in fact, it can be shown that the axiom of choice implies the law of excluded middle. It is the nature of the mathematician (more particularly the naive one) to look at such a statement and be unconcerned with it; their attention is instead focused on the

nature of the type or *set* that they are choosing from. One might remark even in the other direction, that the great success of mathematics is not in spite of a blindness to logic but *because* of their singleminded metaphysical curiosity. For much of the remainder of this text, we will embrace that curiosity, and the mindset of the mathematician.

We may even point out a distinct failure of the system we have been working within. That is, while we can prove quite sophisticated mathematical theorems in dependent type theory, we do so with a stipulation that is not the case in mathematics. A witness to the type $\exists(a : A), P(a)$ is not just evidence that it is true that there exists a satisfying $P(a)$, it is an example of it being true (a, p). The functional nature of our deductions means that we cannot merely prove an example exists, we must *construct* it, hence the name **constructive mathematics**.

This burden of proof is distinctly higher than necessary. There are valuable things we can say when we are able to prove examples or counter-examples exist where the example itself does not particularly matter. We can even reverse the accusatory finger; since constructive proofs are computational (indeed there are many languages that will run them), to expect all of mathematics to be fully constructive corresponds in some sense to an axiom itself, that the world we live in is a computable one. Does every question that has an answer have a procedure to find that answer? This is an assumption mathematicians do not make.

proptypes.6 De Morgan's Laws and Proof Techniques

As I said many times before, the rules of dependent type theory remain valuable to us despite our disinterest in its strict adherence. Instead, we know now what it means exactly to presume something is true: treating our theorems as types and our proofs as functions, we can use our intuition to guide our steps and know what type-form a presumption might have, noting immediately if it is reasonable or in fact a much more dangerous statement.

Accordingly, we can now give a more full set of rules for how to manipulate negations, paying off one of the uses of type theory that we will continue to return to throughout this text.

Theorem proptypes.19 — (De Morgan's Laws, Proof by Contradiction, etc.)

Let P, Q : `Type` be *decidable* types, that is, we have witnesses of $P \vee \neg P$ and $Q \vee \neg Q$. Then we have the following:

- **Proof by Contradiction** or *Reductio ad absurdum*, i.e. 'if P is true or false, then if we can show that P being false is logically inconsistent, then P must be true since it is either true or false and can not be false'

$$\texttt{RAA} : \forall(P : \texttt{Type}), P \vee \neg P \Rightarrow (\neg\neg P \Rightarrow P)$$

- $\neg(P \wedge Q) \Leftrightarrow \neg P \vee \neg Q$, i.e. 'not both, if and only if either is false'
- $\neg(P \vee Q) \Leftrightarrow \neg P \wedge \neg Q$ i.e. 'neither, if and only if both are false'
- **Proof by Contrapositive** i.e. 'the rain must wet the ground if and only if knowing the ground is not wet tells us that it did not rain'

$$\texttt{PBC}\colon (P \Rightarrow Q) \Leftrightarrow (\neg Q \Rightarrow \neg P)$$

- **Negation of implication** i.e. 'P does not cause Q if and only if there are examples of P happening but not Q'

$$\neg(P \Rightarrow Q) \Leftrightarrow P \wedge \neg Q$$

Additionally, let $R\colon A \to \texttt{Type}$ be a family of types with $A\colon \texttt{Type}$ a non-empty type. Moreover, allow that $R(a)$ and its derivative types are decidable as necessary, such as $\forall(a : A), R(a)$, or $\exists(a : A), \neg R(a)$, etc. Then we also have:

- $\neg\big(\forall(a : A), R(a)\big) \Leftrightarrow \exists(a : A), \neg R(a)$ i.e. 'not for all if and only if there exists a counter example '
- $\neg\big(\exists(a : A), R(a)\big) \Leftrightarrow \forall(a : A), \neg R(a)$ i.e. 'there does not exist an example if and only if for all it is false'

This is the promised even-longer proof, so we have minimized it on the web. I do recommend taking a look however, particularly in the last thing we prove.

Proof.

We have already shown some parts of these in theorem proptypes.18, so we only need to show the remaining parts.

- We have already assumed by hypothesis $P\colon \texttt{Type}$ and some witness $P \vee \neg P$, so let us pattern match on this witness. It is either $l\,p$ with $p\colon P$ or $r\,f$ with $f\colon P \to \bot$. In the former case, by right-associativity we are done since we may assign the function $\neg\neg P \Rightarrow P$ to be merely the constant function p. In the latter case, we must also be given an element of $\neg\neg P$, that is, some $g\colon (P \to \bot) \to \bot$. In this case, $g\,f\colon \bot$ produces the fictitious element of the empty type, allowing us to use the principle of explosion with trivial function $h\colon \bot \to P$.

$$\begin{aligned}\texttt{RAA}\,(l\,p)\,g &= p\\ \texttt{RAA}\,(r\,f)\,g &= h\,(g\,f)\end{aligned}$$

- Recall that $\Leftrightarrow$ merely means a pair of implications going in either direction. We have already shown $\neg P \vee \neg Q \Rightarrow \neg(P \wedge Q)$ so we need now to show in our context that $\neg(P \wedge Q) \Rightarrow \neg P \vee \neg Q$, or rather

$$\big((P \times Q) \to \bot\big) \to \big((P \to \bot) + (Q \to \bot)\big).$$

  To show this, we will need to actively use our information that P and Q are decidable, so in fact, the above type as written may not have witnesses, nor are we trying to prove it in a general sense. Our full theorem, accounting for the desired context, is

$$\begin{aligned}f\colon P \vee \neg P \to Q \vee \neg Q \to \big((P \times Q) \to \bot\big)\\ \to \big((P \to \bot) + (Q \to \bot)\big)\end{aligned}$$

We must now pattern match on $P \vee \neg P$ and $Q \vee \neg Q$ with $p\colon P$ or $g_p\colon P \to \bot$ and $q\colon Q$ or $g_q\colon Q \to \bot$ respectively. Then let $h\colon (P \times Q) \to \bot$ be the next argument and $k\colon \bot \to \neg P \vee \neg Q$ be a trivial function. Our four cases become

$$\begin{aligned} f\ (l\ p)\ (l\ q)\ h &= k\ \big(h\ (p, q)\big) \\ f\ (l\ p)\ (r\ g_q)\ h &= r\ g_q \\ f\ (r\ g_p)\ (l\ q)\ h &= l\ g_p \\ f\ (r\ g_p)\ (r\ g_q)\ h &= l\ g_p \end{aligned}$$

Notice that whenever either decidable proposition is decided false, we simply use that to form our witness of "not P or not Q".

- Both $\neg P \vee \neg Q \Rightarrow \neg(P \wedge Q)$ and $\neg(P \wedge Q) \Rightarrow \neg P \vee \neg Q$ have been proven in theorem proptypes.18, so their pair forms $\neg(P \vee Q) \Leftrightarrow \neg P \wedge \neg Q$.

- The implication of the contrapositive form of a proposition $(P \Rightarrow Q) \Rightarrow (\neg Q \Rightarrow \neg P)$ can be proven without assuming decidability. In fact, reading the type as

  $$(P \to Q) \to (Q \to \bot) \to P \to \bot$$

  we can supply our proof f_1 with three arguments $g\colon P \to Q$, $h\colon Q \to \bot$ and $p\colon P$, immediately solving this form as

  $$f\ g\ h\ p = h\ (g\ p)$$

  by using g to turn p into a member of Q and then using h to turn that member of Q into a member of $\bot$.

  In the other direction, we require decidability of Q, so let us extend the type as we did in the previous proof which we used decidability in.

  $$f_2\colon Q \vee \neg Q \to (\neg Q \to \neg P) \to (P \to Q)$$

  Similarly to the last proof which used decidability, our proof in fact looks the same so long as Q is true, since by right-associativity, we may simply ignore the rest of the arguments and say that $P \Rightarrow Q$ since no matter P, we have $q\colon Q$ as a constant function. This means, for other cases, we assume Q to be false. In that case, with argument $h\colon \neg Q \to \neg P$, we may read $\neg P = P \to \bot$ and supply the final $p\colon P$ argument from the end of the type $P \to Q$ (i.e. we are applying right-associativity) to obtain an element of $\bot$, then apply the trivial function $k\colon \bot \to Q$.

  $$\begin{aligned} f_2\ (l\ q)\ h\ p &= q \\ f_2\ (r\ g)\ h\ p &= k\ (h\ g\ p) \end{aligned}$$

  Our complete proof is then the pair $\texttt{PBC} = (f_1, f_2)$.

- To show the form of negation of implications, once again we prove two forms

$$f_1 \colon P \vee \neg P \to Q \vee \neg Q \to \neg(P \to Q) \Rightarrow P \wedge \neg Q$$
$$f_2 \colon P \vee \neg P \to Q \vee \neg Q \to (P \wedge \neg Q) \Rightarrow \neg(P \to Q)$$

where f_2 turns out to be substantially easier to prove, and in fact can be done without decidability. For that reason, let us provide it the dummy arguments $x \colon P \vee \neg P$ and $y \colon Q \vee \neg Q$ and the pair $(p, g) \colon P \times \neg Q$ where $p \colon P$ and $g \colon Q \to \bot$. Finally, we once again apply right-associativity to provide the argument $h \colon P \to Q$ changing our goal type to $\bot$. In this case our proof is merely

$$f_2\, x\, y\, (p, g)\, h = g\, (h\, p).$$

The reverse will prove to be the most complicated non-dependent proof we've done so far, and will involve reusing our proof of contrapositive which we named so we could use it now, `PBC`. In particular, we will need to extract its second form

$$(\mathtt{pr}_2\, \mathtt{PBC}) \colon Q \vee \neg Q \to (\neg Q \to \neg P) \to (P \to Q)$$

with the projection operator $\mathtt{pr}_2$ which we use for extracting the second item in a pair. We will also need two different trivial functions k_1, k_2, and so we will name their types shortly; the more important thing to recognise is that whenever we see a trivial function used, it is because we are in a false case.

Now consider, since P and Q are decidable by hypothesis, they are either true or false. First let us pattern match on the decidability of Q, and indeed the truth of P will not matter if Q is true, so we'll need a dummy variable symbol x, and a few others y, z. Recalling the form we want to show

$$f_1 \colon P \vee \neg P \to Q \vee \neg Q \to \neg(P \to Q) \Rightarrow P \wedge \neg Q,$$

we also take as an argument $g \colon \neg(P \to Q)$. In the case where Q is true, our evidence $q \colon Q$ allows us to define a constant function $(y \mapsto q) \colon P \to Q$, and this can be fed to g to produce $\bot$. If P is true and Q is false, we are in exactly the situation desired and may provide $p \colon P$ and $h \colon \neg Q$ as output $(p, h) \colon P \wedge \neg Q$.

If however both P and Q are false, let us say that we have $h_p \colon P \to \bot$ and $h_q \colon Q \to \bot$. Since $h_q \colon \neg Q$, we can pass it to $(\mathtt{pr}_2\, \mathtt{PBC})$ as our proof that Q is decidable and thus witness that contrapositive applies. This gets us

$$(\mathtt{pr}_2\, \mathtt{PBC})\, h_q \colon (\neg Q \to \neg P) \to (P \to Q).$$

To use this, we'll need a function of type $\neg Q \to \neg P$, however we have $h_p \colon \neg P$, which is enough to define a constant function $(z \mapsto h_p) \colon \neg Q \to \neg P$. Once fed to $(\mathtt{pr}_2\, \mathtt{PBC})\, h_q$, we obtain now a function of type $P \to Q$. Recall now that we have the argument $g \colon (P \to Q) \to \bot$, which can eat $(\mathtt{pr}_2\, \mathtt{PBC})\, h_q\, (z \mapsto h_p)$ to give us our element of $\bot$. Finally we can apply the trivial function $k_2 \colon \bot \to P \wedge \neg Q$.

All together, these three cases are written

$$f_1\, x\, (l\, q)\, g \;= k_1\big(g\, (y \mapsto q)\big)$$
$$f_1\, (l\, p)\, (r\, h)\, g = (p, h)$$
$$f_1\, (r\, h_p)\, (r\, h_q)\, g = k_2\Big(g\, \big((\mathtt{pr}_2\, \mathtt{PBC})\, h_q\, (z \mapsto h_p)\big)\Big)$$

with $k_1 \colon \bot \to P \wedge \neg Q$.

We proceed now to the dependent statements. In theorem proptypes.18 we showed both directions of $\neg\big(\exists(a : A), R(a)\big) \Leftrightarrow \forall(a : A), \neg R(a)$. So we set our attention on $\neg\big(\forall(a : A), R(a)\big) \Leftrightarrow \exists(a : A), \neg R(a)$, for which we have also proven the direction $\exists(a : A), \neg R(a) \Rightarrow \neg(\forall(a : A) \to R(a))$ without decidability.
The other direction will prove substantially more difficult, requiring helper functions as well as decidability of all $R(a)$ and of the dependent pair type $\sum_{a:A} \neg R(a)$. We also need an example of a so we are sure the family R is non-empty. Then we will need dependent trivial functions $\phi_a \colon \bot \to R(a)$ (spoken as 'phi', consider this if you like to be $(a : A) \to \bot \to R(a)$ with the argument a given in the subscript) and $\psi_a \colon \bot \to \neg R(a)$.
Our goal is then

$$\begin{aligned} f \colon A \to &\big((a : A) \to R(a) \vee \neg R(a)\big) \to \\ &\left(\sum_{a:A} \neg R(a)\right) \vee \neg \left(\sum_{a:A} \neg R(a)\right) \to \\ &\Big(\big((a : A) \to R(a)\big) \to \bot\Big) \to \sum_{a:A} \neg R(a) \end{aligned}$$

consisting of four arguments,

$$\begin{aligned} b &\colon A, \\ g &\colon (a : A) \to R(a) \vee \neg R(a), \\ \text{either } &(l\, e_1) \text{ or } (r\, e_2) \text{ pattern matched} \\ \text{with } &e_1 \colon \exists(a : A), \neg R(a) \text{ or } e_2 \colon \big(\exists(a : A), \neg R(a)\big) \to \bot \\ \text{and finally } &h \colon \big((a : A) \to R(a)\big) \to \bot \end{aligned}$$

We will also need helper functions alpha and beta

$$\begin{aligned} \alpha &\colon (a : A) \to R(a) \vee \neg R(a) \to R(a) \\ \beta &\colon (a : A) \to R(a) \end{aligned}$$

which will exist only in this context in the case with e_2 and as a consequence of it. You may think of them as effectively a notational sugar-coat for readability.
Proceed in the case with e_1. This is the case where we imagine that decidability of $\exists(a : A), \neg R(a)$ is true; in this case our proof is exactly this example.

$$f\, b\, g\, (l\, e_1)\, h = e_1$$

In the other case, we can and must first define α and β. The statement of α is effectively 'for all $a \colon A$ with $R(a)$ true or false, we have $R(a)$ true'. This obviously does not make sense in general, but in this specific case we have e_2 which one can consider to be the statement 'there are no counter-examples to $\forall(a : A), R(a)$' (read the type of e_2 carefully to get this). The other reason we define α is so that we can pattern-match on $(g\, b) \colon R(b) \vee \neg R(b)$. Thus our α takes two arguments with two cases, one where it takes $c \colon A$ and $(l\, p)$ with $p \colon R(c)$ and one where it takes instead $(r\, q)$ with $q \colon R(c) \to \bot$. Recalling we defined the trivial function $\phi_a \colon \bot \to R(a)$, we can now write

$$\alpha\, c\, (l\, p) = p$$
$$\alpha\, c\, (r\, q) = \phi_c\, \big(e_2\, (c, q)\big)$$

i.e. evidence of decidability of $R(c)$ must always give us evidence that it is true since we have e_2 to tell us there exists no counter examples. If we try to say there exists $q\colon R(c) \to \bot$ then we get a contradiction.
This is nearly the form we need, that of $\forall(a : A), R(a)$, which will allow us to apply $h\colon \big((a : A) \to R(a)\big) \to \bot$ to say that e_2 itself causes a contradiction. To do that though, we need to find some way to get rid of the dependence on proof of decidability as an argument. Since, in our context, we already have g which takes some $a : A$ and tells us if its $R(a)$ is decidable, we don't need to take that as an argument; in this context it can be taken as a constant. Thus we define

$$\beta\, c\, = \alpha\, c\, (g\, c)$$

which is merely α with decidability of $R(c)$ provided externally by $(g\, c)$.
Finally we provide β to h, obtaining a contradiction, and then applying our trivial function $\psi_a\colon \bot \to R(a)$ to eliminate the counterfactual.

$$f\, b\, g\, (l\, e_1)\, h = \big(b, \psi_b\, (h\, \beta)\big)$$

We are finally done.

□

It is worth noting that this also pays off our earlier mention of universal quantifiers generalizing *and* statements and existential quantifiers generalizing *or* statements. The generalized De Morgan's laws show that the way a *for all* statement negates with respect to a *there exists* statement is indeed the same relationship we see between *and* and *or*. For instance, we have both 'not both, if and only if either is false' and 'not for all if and only if there exists a counter example'.

proptypes.7 Into *Mathematics*

This system of deductions will remain close with us. It is not strictly that the system of dependent type theory is a bad one for logic, but merely that it does not automatically provide a great model for our interests in mathematics. This is largely solved by augmenting our system with the axioms we insist to be true.

In the context of dependent type theory, an axiom can be thought of as insisting a witness of some proposition or theorem exists with no way to define it or find it. Once we add even one axiom, we are in the non-constructive regime, since if theorems are functions, our axioms are functions with no procedure to execute them. Computers can still check such theorems however, and they do this at the level that a computer program merely checks to see that there are no type errors; the lack of a procedure does not change the fact that if such a procedure existed, the theorem would then run *without error*.

But as we move forward, we will be doing more than merely making our theorems non-constructive with our axioms. We, as mathematicians, are *not* logicians. There will be times when we strike off one of our axioms and insist that it does not apply in the case we are in. We will even do this without strict rules, but by reasoning about our system, or simply by a known counter-

example. That is, as mathematicians, we can know an inconsistency exists somewhere in our system, and simply choose to avoid it as a false branch of conclusions. We do not, for instance, neurotically emphasize solutions to Russel's paradox to avoid even the slightest risk of paradox, we simply avoid the mechanism that causes Russel's paradox to occur as implicitly solving the paradox.

That is neither to say that we merely allow the risk of paradoxes due to our axioms. Due to over a century of knit-picking out of concern for this exact problems, we know what our axioms are in mathematics, and just as in our proof of De Morgan's laws we wrote axioms not as functions that were merely available in context, but as specific conditions, we can simply fail to provide those conditions in the situations where they are not true.

Dependent type theory, with the appropriate asterisks, will form a scaffold by which we continue to reason about logic, and by which we intuit what behaviours we might desire from other structures. For instance, moving forward, we largely put the idea of constructive numbers behind us, but we know that we *can* construct inductive natural numbers, and thus we can do proof by induction on them. Moreover, seemingly qualitative statements can now be made quantitative by interpretting our conditions for when they are true as *constructors*, by our so called *universal* and *existential* quantifiers.

mathhink Rewrites and Sets: The Cognitive Weapons of Math

Despite the problems with constructive mathematics that we discussed previously, I think it is necessary to first see mathematics as a much more rigid system. The two topics of this section, set reasoning and relational reasoning, while both formal studies, are at their best when one simply develops an intuition for them. Going from no mathematical training to having these intuitions is perhaps the hardest part, the barrier that makes people think that math is hard, and it is made significantly worse by the lack of structure at the bottom of the reasoning.

That is, set theory is not a constructive theory in the slightest. It cannot exist in a vacuum and provides a very poor pedagogical lens since it was invented for existing mathematicians to better understand their craft. It does not say anything about where an object comes from or what it fundamentally is, but rather assumes an object could be *anything* and then whittles that *anything* down to something which can be reasoned about. It appears as a top down theory rather than a bottom up theory, like type theory is.

And while (pure) mathematics has evolved into an especially symbolic field, it did not begin that way; the symbols are merely a way to help us formalize our thoughts but the thoughts themselves have historical roots in logical *statements*, reasoned about just as one reasons about in philosophy. Mathematics has moved less from those roots than people realize, thus the great friction between computer readable mathematics and human readable mathematics; human readable mathematics involves *context* and it skips steps where they are *obvious*. Nonetheless, we have (quite nervously) formalized into rules whatever we can to ensure that our *reasoning* is not merely elaborate sophistry, and to move the foundation of our thought further down.

It is because mathematics is merely a formalized discipline of human reasoning that it is significantly broader than any one system we have invented within it. This is what it means when we say something like that type theory is a system of reasoning built up within set theory (or rather that it can be). Not because sets are themselves the perfect abstraction, but because as an abstraction,

they were made to model the broadest reasoning that consistently made sense to us rather than narrowing reasoning to something a computer could follow.

So in this section we aim to explore some of the cognitive primitives of mathematics, namely our relationship to relations themselves, the mathematician's notion of a set, and what it means for an object or theory to be founded on axioms. To do this, we will touch on the notion of a rewriting system and the Zermelo-Fraenkel-Choice axioms, discussing how an axiomatic system works. We will see in this section that, although in broader mathematics we can often say very little about what a thing *is* (in the same way that we have constructors in type theory), we trade this knowledge for an increased focus on what a thing *does*.

mathhink.1 Symbolic Reasoning

Consider what is meant when we write the abstract symbol x. If you were comfortable with the previous section then you'd expect it could be an element of a type, or even breaking convention, perhaps a type itself. As we enter into non-constructive mathematics, the rule will be that a symbol is essentially any possible thing until otherwise constrained (perhaps by context) to some set of rules. As we will discuss more when we get into set theory, classical mathematics does not have types or constructors, it has only rules, and sets of things that obey those rules. We have the set of natural numbers, the set of integers, etc. Not because those sets are themselves types, or because the members they contain are *necessarily* fundamentally different from one another, but merely because it is a collection of *abstract things* which happen to obey conditions.

This is a clean break from our notion of types; just as we had constructed a type of natural numbers, we could also construct a type of integers (counting numbers with negatives) but this type would have its own constructors. It would be reasonable to create a function that sends a natural number to its corresponding integer in the integer type, but this is not the same as saying that 'all natural numbers are also integers'. In our type theory, a natural number can *become* an integer but in classical mathematics one may say that members of what should be a type, *are* also members of another type.

This creates a problem in some modes of thinking; if we can say that some integers are natural numbers (the positive ones), and some real numbers are integers (the whole numbers), etc. etc., can we say that there is some *true thing* that each of these systems of numbers *are*? Not only is the answer no, the question would probably illicit some laughter from a mathematician. The lack of knowing what a number *is* is in fact a feature which allows our tools to be general. When we say that 'integers can be added, subtracted, multiplied, and in some cases divided', we do not mean that we are referring to some fundamental object that *exists*. We mean that we are referring to a collection of objects, the extent of which we do not know and do not care of, which we can describe *in general*. We are studying the nature of *anything* which is counted whole with positives and negatives when we study integers. The apples in bags, the candies divided amongst children, anything that we think of as whole.

This creates a different problem too, for if we are discussing *all things* that our rules describe, then how are we to intuit what we should do with a problem? One might imagine this like an actor being told to display a certain emotion in front of a green screen, only to watch the movie themselves and discover that an entirely different performance would be more appropriate in that scene. Or like being handed a simple puzzle interface and being told to solve puzzles, only to discover that you were

in fact designing the electrical grid for a nation. Surely it is appropriate to know the material nature of a problem so that you can better approach it, right? To a (pure) mathematician, the answer is no. In fact, it is of great value to the pure mathematician that someone else can make formal the rules of the game we are playing so that we may merely play a game.

And this is entirely appropriate for the mathematician. While it is certainly true that knowing more about a material setting allows us extra information to deduce more things, there is a value in reasoning out everything there is to be known about the general case. Moreover, the way that we investigate this general case is by rejecting all other information about the system other than the symbolic rules we have been given and merely monkeying around with them.

An example is in order, if for no other reason than to explain what it means to reject information and settle for deduction by rules. Consider the equation $x^2-4=0$, although do not consider it so much as to be afraid; we will step through this example very slowly. To a mathematician, a problem such as this is examined as an algebraic one, and is thus attacked with algebraic tools. If we want to find the real number values of x that solve the equation, we consider some things we know about algebraic numbers. We know about addition and multiplication. We know that for every addition there exists a number which, when added, reverses the addition (the negative). We know that for every multiplication (except zero) there exists a number which reverses the multiplication (the inverse). We would write these rules as follows:

$$\forall x \in \mathbb{R}, \exists y \in \mathbb{R}, (x+y=0)$$
$$\forall x \in \mathbb{R}, \forall y \in \mathbb{R}, (x+y=0) \Leftrightarrow (x=-y)$$
$$\forall x \in \mathbb{R} \setminus \{0\}, \exists y \in \mathbb{R}, (xy=1)$$
$$\forall x \in \mathbb{R} \setminus \{0\}, \forall y \in \mathbb{R} \setminus \{0\},$$

Now, it is common when reasoning about equalities to think of alterations as "surely this implies that" or "this is equal to that", "if we add 2 to both sides, then the sides remain equal". While this mode of reasoning is helpful, on this occasion we reject it entirely.

Instead, we will think in terms of *rewrites*. Which is to say, when we see terms of a certain form, we do not need to ask what they are other than the relevant *rules*, and we *replace symbols*. In this case, we see the second rule we wrote above, that $x+y=0$ implies $x=-y$ and vice versa for all x, y, and we *perform a rewrite*. In this case, the only symbol in the expression which is fixed is *zero*, and the equality itself. So can we see a zero and an equality in $x^2-4=0$? Yes we can, and identify x in our general expression as x^2 in our specific expression, likewise with y and -4, and we rewrite $x^2-4=0$ to $x^2=4$. We do this without any particular reasoning about this equals that, but rather on the basis of a *rule* that is known about real number algebra. It is only at the point that we are *stuck* (i.e. no rewrites apply) that we try to reason, not about the equation itself exactly, but about what other rules may apply. Then perhaps we would reason about when a square root is appropriate.

This mindset radically simplifies the mental overhead of doing mathematics. We can of course, when necessary, make more reasoned deductions about why a thing *is true*, but if we did this all the time then it would be exhausting as well as slow. Mathematicians do not do this, their theorems (insofar as their theorems describe relations such as equals or less than/greater than) provide for them rewrite rules which they remember and apply as their fingers reaching into the world of symbols.

But perhaps that example was too unfamiliar. We have in fact already studied at least one rewrite, which was the addition operation for inductive natural numbers. First, consider this in abstract: if we say we have two numbers a and b whose sum is c, let us write d to mean the sum $a + b$ not as their sum *quantity* but as a representative of the symbols $a + b$. Then what is the difference between c and d? The former is the *result* of the calculation whereas d is the calculation itself, in a state before any rewrites had been applied that might produce its result. We can think of c and d as 'equal' in the sense of the numbers we might discuss, but still representing different objects. d, representing $a+b$ must be *rewritten* into c through whatever rules we have to perform this addition. Our notion of equality is then a choice of granularity: do we speak of symbolic equality, i.e. equality before reductions, or equality after reductions? What is it exactly that we want equality to mean?

Recall that the function we defined for addition in the previous section was based on two cases, that of the inductive zero and the inductive succession operator, $\mathbf{Z} \colon \mathbb{N}$ and $\mathbf{S} \colon \mathbb{N} \to \mathbb{N}$. The rules were

$$\begin{aligned} \texttt{add}(m, \mathbf{Z}) &= m \\ \texttt{add}(m, \mathbf{S}(p)) &= \texttt{add}(\mathbf{S}(m), p) \end{aligned}$$

in which we either shifted a 'tally mark' from the second argument to the first and then repeated, or we were in the case where the right hand side had no tally marks and we simply returned the left number. This was in fact the first proper rewrite rule we discussed, in which every instance of the left hand sides of these equalities, the $\texttt{add}(m, \mathbf{Z})$ and the $\texttt{add}(m, \mathbf{S}(p))$ are automatically replaced with the right hand sides, m and $\texttt{add}(\mathbf{S}(m), p)$. In the second case, the case of the succession operation $\mathbf{S}$, we would then perform this rewrite as many times as we need to until we found ourself in the first case. But consider that knowing that these rewrite rules are *equal*, we could have also rewritten them backwards. In fact it is often common to name such a theorem $\texttt{add}(m, \mathbf{Z}) \equiv m$ a theorem of *right-identity*.

Even in the type theory case however, one must noticing something. In the previous section we showed transitivity of our $\equiv$ relation, and so if we had $m \equiv n$ we could show $\texttt{add}(m, \mathbf{Z}) \equiv n$, however we could not show that $m+a \equiv n+a$ implied $\texttt{add}(m, \mathbf{Z})+a \equiv n+a$ directly. While we had other rules to do that, the idea that we could insert $\texttt{add}(m, \mathbf{Z}) \equiv m$ into the existing expression $m + a$ simply does not exist a priori. What we are reaching for there is a rewrite rule, the idea that equivalence implies that a symbol can be replaced with what it is equal to. But the only rewrite rules in the type theory we discussed, without significant extra effort or axioms, is that of function definitions.

The version of type theory we layed out in the previous section is one that can be computer interpretted, like a programming language. In the proper study of rewriting systems, we require that programming languages are *normalizable*, meaning that either there is a proper order of rewrites, or no matter which rewrites one does, there is a promised *final state* that all valid rewrites will arrive at. A correct *normal form* which we arrive at as the simplification of whatever we had written. Our addition operator is an example of this, as we have a rewrite that can almost always be applied $\texttt{add}(m, \mathbf{S}(p))$ to $\texttt{add}(\mathbf{S}(m), p)$ and a rewrite that could be applied in the one case where that rewrite did not, the termination case arriving finally to normal form which is the sum.

Mathematics is not like this. In mathematics it is expected that we have a rewrite rule for every equality we discover. Whenever it is that we write $A = B$, then we would swap A for B no matter how complex the expression we would find either symbol in. Moreover, rules like addition,

subtraction, any operation, are constantly reversable even in contexts where they did not originally occur. We write $a + b = c$ and then we write $c = c + 0 = c + d - d$ and thus $a + b = c + d - d$ as necessary, inventing new terms if it serves our deductive goals. Sometimes I have tried to explain to programmers that mathematics is like programming but with a time machine; if one thinks of the reductions of a program in the process of being executed in one direction towards its final simplified form as the flow of time, mathematical deductions reverse this process freely, swap timelines, etc. Our goal is not a final form but some observation about the operation we are studying, and to do this we must actively reject a final form.

In fact, one might say that mathematics as a practice (a *procedural skill* distinct from its discipline or its theory) revolves almost entirely around rewrite rules. We are able to study concepts far more abstract than just numbers precisely because we put what a thing *is* to the side and focus only on how its symbols behave. We can then define objects which have entirely different symbolic behaviours from numbers and try to understand what they must represent in our world to behave in such a way. The beginning of the following chapter will be almost entirely devoted to elaborating how our notions about numbers are deduced from rewrites, however our first true example of this will be the set.

maththink.2 Sets and ZFC from Axioms

Set theory is famously described as being developed in a period of panic about the foundations of mathematics. We did not have then, as we do now, a notion of how one builds up mathematics from a small and strictly limited set of rules so that we are certain that what we are doing in mathematics makes sense. One could at the time (very anxiously) worry that perhaps all of the mathematics we had developed was somehow fraudulant. This is important context because it tells us about what set theory was meant to do and what it was not meant to do. It was developed *for* existing mathematicians so that they could understand what it was that they were doing. This means that it does not mention the wider expectations about what one would be doing *with* its rules. Consequently, set theory is a woefully incomplete manual for 'how to do mathematics'; it does not even try to be that. Set theory exists *within* what we have been calling the set theoretic lens, and the set theoretic lens is the lens that existed *for all mathematics* before we had invented notions of starting from scratch in a constructive setting.

So let us first discuss some mental primitives to keep in mind. A *set* is supposed to be a collection of objects without ordering or multiplicity (i.e. a set either contains an item or it does not, with no notion of before or after and never containing it twice). A set containing three objects a, b, c can be written $\{a, b, c\}$. In fact, with certain asterisks which we discuss shortly, one can think of a set also as a proposition, much as in the way that we discussed propositions in the previous section. That is, one writes $\{x \mid P(x)\}$ to denote the set of all objects which we might label x, for whom the condition $P(x)$ is true. This notation is called a **set comprehension**. One may naively narrate sets as things like 'the set of all people' for which you can select a random member and get a person, or restrict this to the set of people you know as a subset, etc. As discussed earlier, we have shed type theory now, so a symbol means only the rules that apply to it; a set $\{x \mid P(x)\}$ that we draw members x from could now be anything, and all we know about x is that it obeys whatever kind of condition P is. In this practice of mathematics, we reject a fixation on what a symbol *is* or *means* in favor of what it *does*, as P will encode information necessary to make rewrites valid. If we set P to be a condition that "x is either one or some counting number with one added to it" then we have defined

the *set* of natural numbers, not by construction but by restriction from the set of all things, and this enables rewrites involving sums for example. Thus forth, these kinds of restrictions, restrictions that *add rules* to our objects of study *will be* what we consider construction.

This more naive description of what a set is is unreasonable however, as famously shown by Bertrand Russel. Russel's paradox showed that we cannot allow sets to naively contain 'anything that satisfies some condition', as this would allow sets to contain themselves. Moreover, one can start asking questions such as, "does the set of all sets which do not contain themselves contain itself?" and as Russel showed, deduce that it both contains itself and does not contain itself.

This and a plurality of other matters that needed clearing up or formalization culminated in what is now called the Zermelo-Fraenkel set axioms, usually supplemented with the axiom of choice to be called ZFC.

As this will be our first exploration of a mathematical concept from its axioms, we should discuss a bit about how this will work. Our notion of propositional logic will carry over quite strongly from the previous section, and our axioms will define a collection of propositions asserted true about sets. These will be the requirements, the rules, that restrict down the could-be-anythings of unspecified symbols down to a collection of certainly-are-somethings from which we can make deductions. Importantly, a set of axioms is often far from a literate series of statements about what a thing is or how it should be interpretted; that information is certainly *encoded*, but it is decoded as a deduction from what is said or not said in the axiomatic statement. (It is often the case that axioms will try to be relatively minimal, making as *weak* a statement as possible so that properties that are desireable can be encountered or deduced rather than enforced).

As such, our reading of the Zermelo-Fraenkel axioms will start from the axioms themselves, how one can read or interpret each axiom, and some basic consequences, in that order. I must emphasize that you should acknowledge this order and not take any part of it to do the job of a part that follows; the list of axioms themselves are, once again, intended to make strict statements that may not be immediately fully understood. One should not try to draw some philosophical notion of what axioms describe immediately from their list, or else you are likely to halt unnecessarily when the answer, the clearly stated description, lies just a paragraph or two below as it does in this case. Instead, axioms are to be read with a note taken of anything that does not immediately make sense to you (until such a time that you are an experienced mathematician) so that you may await the text to answer all such questions you may have. This is the process we must eventually go through habitualy when encountering a new set of axioms. As we progress, we will develop an intuition for the kinds of concepts mathematical descriptions map on to, and our corresponding intuition prescriptions will get necessarily shorter.

Definition maththink.1 — (Zermelo-Fraenkel Set Theory Axioms)

Inherit the notion of a proposition or a condition (i.e. loosely in the sense of a family of types) from the previous section along with logical operations. Consider a set to be a collection of objects with a notion of equivalence between those objects and a propositional relation written $\in$ which is true when an object x is *in* a set A (and thus read $\in$ as 'in' or 'is an element of').
Then sets are defined to obey the following.

- (Axiom of Extensionality) Two sets A and B are equal if they have the same elements.

$$\big(\forall x, (x \in A) \Leftrightarrow (x \in B)\big) \Rightarrow A = B$$

- (Axiom of Pairing) For all objects a and b there exists a set a, b that contains exactly and only a and b.

$$\forall a, \forall b, \exists A, \big(\forall x, (x \in A) \Leftrightarrow (x = a \vee x = b)\big)$$

- (Axiom of Comprehension) Let P be a condition and A be a set. There exists a set B such that $x \in B$ if and only if $x \in A$ and $P(x)$.

$$\forall P, \forall A, \exists B, \big(\forall x, (x \in B) \Leftrightarrow (x \in A) \wedge P(x)\big)$$

  Then we write B using the **set comprehension** notation

$$B = \{x \in A | P(x)\}.$$

- (Axiom of Power Set) For every set A there exists a set $\mathcal{P}(A)$ of subsets of A.

$$\forall A, \exists B, \forall C, \big((\forall x, x \in C \Rightarrow x \in A) \Leftrightarrow C \in B\big)$$

  When $B \in \mathcal{P}(A)$ then we write $B \subseteq A$.

- (Axiom of Union) Let $\mathcal{A}$ be a set which contains sets. There exists A such that for all sets X and all objects x, if $x \in X$ and $X \in \mathcal{A}$ then $x \in A$.

$$\forall \mathcal{A}, \exists A, \forall X, \forall x, \big((x \in X) \wedge (X \in \mathcal{A})\big) \Leftrightarrow (x \in A)$$

  Then we write A using the **union notation**, which is either $A = B \cup C$ when $\mathcal{A} = \{B, C\}$ or more generally

$$A = \bigcup_{X \in \mathcal{A}} X.$$

- (Axiom of Replacement) Let A be a set and f be a function for which $f(x)$ is meaningful when $x \in A$. Then there exists a set $B = f(A) := \{f(x) | x \in A\}$, extending the *set comprehension* notation. To state this formally, let P be the relation such that $P(x, y)$ is true when we say $y = f(x)$, and in this sense let P define f.

$$\forall P, \big(\forall x \forall y \forall z (P(x, y) \wedge P(x, z) \Rightarrow y = z)$$
$$\Rightarrow \exists B (\forall y, y \in B \Leftrightarrow \exists x, P(x, y)\big)$$

  We then call B the **image** of f or its **range**.

- (Axiom of Foundation) Every set A which is non-empty contains an element B with which it is disjoint.

$$\forall A, \big((\exists a, a \in A) \Rightarrow \exists B, (B \in A) \wedge \neg \exists b, (b \in B \wedge b \in A)\big)$$

- (Axiom of Infinity) There exists a set containing infinitely many objects

$$\forall A \exists S, \big(\{x \in A | \bot\} \in S \wedge (\forall x, x \in S \Rightarrow x \cup \{x\} \in S)\big).$$

Once again, we must emphasize a change in the way that we write propositions. That is, none of our objects (and thus none of our *conditions* or families of propositions) are typed anymore. Set theory does not make claim to 'where objects come from'. It instead assumes mostly through other mechanisms or as prescribed later by the mathematician, that there *are objects*. One might even say that set theory specifically exists in a perspective where all objects that could exist do exist already, which we then draw on. It is then our responsibility to pick and choose from these objects by appropriately setting conditions, propositions, etc. This, on its own, obviously creates problems such as the one we mentioned earlier, of Russel's paradox, but these axioms carefully skirt around those.

Our notions of propositions as families of types, involving a function from a type to the type of types, must then be modified. Our notion of a function in general must also be modified, as we will see when we discuss the axiom of replacement. It will still mostly make sense to think of our propositions as types though, and in particular the way we think about $\wedge, \vee, \neg, \forall, \exists$ as being $\times, +, (\to \bot), \Pi, \Sigma$, but our notions of the relationship to a type must change. For instance, we say $x \in A$ is a proposition. This means that we can also speak of the proposition $\neg(x \in A)$, which we write $x \notin A$. The analogous statement does not make sense in the context of types; if $a \colon A$, it is *of type* A, and it doesn't make sense to say otherwise because a is *constructed* in accordance of the rules of the type A. Objects have no such loyalty in set theory.

Let us now take these axioms as they are and pay attention to what they do say rather than what they don't. While the axioms we have above are intended to describe roughly the intuition we mentioned about what a set is, we cannot take any of that intuition for granted; the whole point of axioms is to give a rigid foundation for those intuitions, smoothing them where we may intuit contradictory ideas. This also means that we must be on the lookout for which axioms (or deductions from them) tell us the basic things we expect to be true about sets.

First, the axiom of extensionality. Since the place where objects come from is not particularly discussed, it is unclear how or when we might speak about those objects being equal to one another, or when any notion of mathematical structure might arise. We will discuss that problem in general later, but this axiom solves that problem in the specific case of sets by saying that two sets are equal when they share members. Specifically, the reading of the formal form is 'if for all objects x, to say x is a member of A is the same as saying it is a member of B, then we are speaking of the same set'. We are to take from this that a set contains no information other than what objects it contains, disregarding notions of an order or a multiplicity (i.e. sets are not lists and they never contain objects twice) since they are irrelevant to equivalence/identity.

The axiom of pairing, in some sense, states that it is possible to form small sets of specifically chosen objects regardless of the nature of those objects. The reading of its formal form is 'for all objects a and all objects b, there exists a set A for which, for any object x that we find to be an element of A, it must either be $x = a$ or $x = b$', thus implying the set contains only a and b alone. This also means we can, in one set, put a number and a letter, or a set and a set containing sets, etc. A set is not itself an object with type, and it does not discriminate on any basis whatsoever when accepting members.

The axiom of comprehension, in one framing, can be stated as 'sets are propositions' as mentioned earlier. That is, a set can be defined specifically to be containing the objects that satisfy a proposition or condition. This comes with a specific asterisk however, relating to Russel's paradox. For instance, we cannot speak of a set containing all sets, or more specifically a set containing all sets which do not contain themself, lest we ask whether this set contains itself. Such a concept is in

contradiction with the axiom of foundation anyways. This problem is solved by defining a set comprehension as requiring a *base set* from which to draw its elements. We then state the axiom as 'for all conditions P and all sets A, there exists a set B which contains the elements of A which satisfy P'. That set B must then be a *subset* of A in accordance with the axiom of comprehension, a set that is a restriction on a larger set.

The axiom of power set extends this notion. We say next that it is not merely that for each set there exists sets containing fewer members where each satisfies a proposition, but in fact that every *subset* that could exist does exist, and populates a new set which we call the power set (denoted as a function $A \mapsto \mathcal{P}(A)$ where $\mathcal{P}(A)$ denotes the set of all subsets of A). A more literal example is given if we define the set $F = \{1, 2, 3, 4\}$, where we can see its powerset is

$$\mathcal{P}(F) = \left\{ \begin{array}{c} \{1\}, \{2\}, \{3\}, \{4\}, \\ \{1,2\}, \{2,3\}, \{3,4\}, \\ \{1,3\}, \{2,4\}, \{1,4\}, \\ \{1,2,3\}, \{2,3,4\}, \\ \{1,3,4\}, \{1,2,4\}, \\ \{\}, \{1,2,3,4\} \end{array} \right\}$$

including both F itself and every set that is F without some of its elements. Consequently, every set comprehension, since it is a subset by condition of some P (the members of the set that satisfy P), is also a member of the power set. This axiom is profoundly important for moving from the countable regime to the uncountable regime since it says that one can pack *every* subset into a single set. Although it is a bit of a stretch, one could in a sense blame this axiom, together with the axiom of infinity, for the continuity of real numbers. Going forward, we use $B \subseteq A$ to denote that B is either a subset or equal to the set A. This is expressed formally within the statement of the powerset, in particular $\forall C$ (that is, for any set we propose as a subset) $\forall x$ we have that $x \in C$ implies $x \in A$, so all elements of C are elements of A. We also use the symbol $\subset$ to speak of a *proper subset* where there exists some x for which $x \in A$ but $x \notin B$, i.e. $B \subset A$ if $B \subseteq A$ *and* $A \neq B$.

The axiom of union can be thought of as a flattening in the way it is seen above. This, as with the previous axiom, the axiom of power set, are the first times we have used a caligraphic script to denote something higher. That is, in type theory it was common to write $\mathbb{N}$ to denote the set of natural numbers, looking like a capital N, and a lower case $n \colon \mathbb{N}$ to denote a single number in the natural numbers. We have continued to use capital letters to denote *sets* and lower case letters to denote the things inside the sets, as this connotation (much in the same way as in natural language) allows us to reduce certain confusions. But if we think of capital letters as containing things, then how should we denote the containers of containers? This is one use for calligraphic capital letters, as we discuss more closer to the end of the section. In this instance, we describe a set of sets $\mathcal{A}$ and how it can be flattened into just a set A, where our choice of letters of the alphabet connotes that they are similar objects. The more literal reading of the axiom is that for all sets $\mathcal{A}$ there exists a set A with the property that for all objects X and all objects x, you can say that $x \in X$ (implying that X is a set) and that $X \in \mathcal{A}$ if and only if x is a member of the set A. In effect, there exists a set which is $\mathcal{A}$ but flattened down a level of 'container'. We establish a notation for this (elaborated later in the chapter), the *union notation* as a binary operation $B \cup C$ and a big-operator notation $\bigcup_{X \in \mathcal{A}} X$. This union is also the same one that one thinks of in a Venn diagram as including the contents of both circles.

The axiom of replacement forces us to redefine what it is we meant by a function. In the previous section we defined functions by their relationship to programming and types, where they take an input and produce an output. This intuition holds, but since we do not have types, the strict manner by which we define a function is different. Now, a function is a condition P on two objects x, y, which is true if y is the output corresponding to the input x. As we see in the formal definition of the axiom of replacement, we also require of this condition that the property of *being an output* is unique. In particular, if $P(x, y)$ and $P(x, z)$ then we expect that $y = z$, which we expect since for each input, a function should only have one output; we stated this previously when we said that functions are defined as a concept almost exclusively by their *consistency*. This is now brought to primacy; since the concept of types are gone for the moment, it seems as though functions need not even assign an output to every input, since the inputs we consider are now unindexed and undescribable (in fact the set A we mention is *nowhere to be found* in the formal statement!); all that is left to say about a function is that it is consistent. The axiom of replacement then says, fundamentally, that if we can define a function f, then we have a set of all outputs which *do* exist. This makes it meaningful to speak of *transformations of sets* and allows us to extend set comprehension notation so long as we can define a function and input set.

And still it will be valuable to give some structure to functions. In fact, the structure that is appropriate for functions in mathematics general does not necessarily line up with what formal set theory prepares for us here. So we will have to *overrule* these axioms in some sense, and define functions in the following way.

Definition maththink.2 — (Set Functions or *Set Morphisms*)

A function f amongst sets is a condition P on two objects together with two sets $\mathrm{Dom}\, f$ and $\mathrm{Cod}\, f$ called the **domain** and **codomain**. These must satisfy the following:

$$\forall x \forall y \forall z, P(x, y) \wedge P(x, z) \Rightarrow y = z$$
$$\forall x, x \in \mathrm{Dom}\, f \Leftrightarrow \exists y, P(x, y)$$
$$f\big(\mathrm{Dom}\, f\big) \subseteq \mathrm{Cod}\, f$$

read as "functions are consistent", "f acts on x if and only if it is in the domain" and "the image of a function is a subset of its codomain". When all these are true, we write $f(x) = y$ when $P(x, y)$ and

$$f \colon \mathrm{Dom}\, f \to \mathrm{Cod}\, f.$$

Importantly a codomain is very strictly *not the whole image* of the function (i.e. there may exist items y in the codomain for which no input x can produce $f(x) = y$), codomains are the same as if not larger than the image set. The ideas of domains and codomains allow us to, in many contexts, recover a semblance of our notion of typed inputs and outputs; this is because the sets we speak of will usually be identified not by items but by what rules their elements obey as discussed earlier. We will return to this thread later.

The axiom of foundation seems largely exists to iron out potential problems. It implies that sets do not contain themselves, that there is no infinite tower of sets containing sets, and that sets cannot mutually contain each other. The prerequisite $\exists a, a \in A$ is merely to say that A

is a non-empty set (i.e. there exists an object contained in A); we can also say this by defining the notation $\emptyset = \{x|\bot\}$ which represents a set with no members and saying $A \neq \emptyset$. Once established, we can also reformulate other parts of the statement; using a set comprehension, the statement $b \in B \wedge b \in A$ discusses the members shared by both sets, and this is called the *intersection* and denoted $A \cap B = \{x \in A | x \in B\}$. The statement then becomes the following

$$\forall A, A \neq \emptyset \Rightarrow \exists B, B \in A \wedge A \cap B = \emptyset$$

or "every set which is not the empty set contains an object it is disjoint with", disjointness referring to sharing no members. Personally this is not an axiom I make the slightest attempt to remember.

Finally, the axiom of infinity can be read as an embedding of the Peano axioms (the natural numbers we spoke of in the previous chapter) into the context of sets. That is, it defines a zero $\mathbf{Z} = \emptyset$ and a succession $\mathbf{S}(n) = n \cup \{n\}$ so that numbers become $2 = \{\{\}, \{\{\}\}\}$ and consequently a set of all natural numbers made in this way. I personally find this to be rather contrived (indeed the nesting of empty sets is simply to give each one some uniqueness since they are no longer equal by the axiom of extensionality). But this axiom is valuable nontheless, since by the axiom of replacement, we can construct functions from this set of natural numbers, thus establishing that even infinite sets of objects may exist in a single set. As you may be much more familiar with the consequences of this axiom than the axiom itself, this is one you can forget as we move forward.

Together, these axioms, the Zermelo-Fraenkel axioms, establish formally what it is that we expect sets to do and what we do not expect them to do. They can be infinite, they can be empty, they can be finite, they can be flattened or combined, they can be propositions (in a rough sense as by a subset), they can be the outputs of functions, they contain no data other than their memberships (such as ordering or type), and they can not contain themselves.

The fact that they contain no data other than their memberships and are at times representative of propositions however is somewhat telling to me personally however; the combination of set comprehension notation and the notation used (commonly) in the axiom of replacement, appearing in the format of an unrestricted set comprehension indicates why this might have been tempting to mathematicians. And containing no other data than membership is the other side of this coin; if a set is a condition, then discussing $A = \{x \mid P\}$ is to translate $P(x)$ is true into $x \in A$ is true. One may ask then, why involve sets at all? Especially when restrictions must be placed on them to avoid contradictions, when we could simply speak in terms of propositions? I would posit, very much as an opinion which I hope will give your thoughts structure on the matter, that a set is a device for simultaneously for mentally disregarding all x which are not $x \in A$ and for indexing propositions. That is, to say $P(x)$ is true is still to consider the counterfactual that it might be false. But when P is a strong enough set of conditions providing rewrite rules such as those describing the real numbers in the following chapter, our mental focus is shifted from the possibility of $P(x)$ being false to treating A as an almost type-like object, with its elements obeying rewrite rules which place it in an appropriate mental category. Moreover, it can be intimidating and frustrating to think about 'all possible conditional statements on a set of objects' and yet in the framing of sets, we have the much more intuitive 'set of all subsets' notion in the power set, an object which we be much more important to us later.

However using sets as the basis of much of mathematical reasoning imparts on us to job of reconstructing certain familiar notions within its rules. We can of course still do this, as seen in the example of the analogy to the product type, the cartesian product set.

Theorem mathtthink.3 — (Cartesian Product is a ZF Set)

Let an ordered pair (a, b) be a set $\{\{a\}, \{a, b\}\}$ in order to identify the first item as the one that applies in both member sets and the second item as the one that applies only once. Then there exists an operation $\times$ called the **Cartesian product** that takes two sets A, B and uniquely defines the set of their pairs

$$A \times B = \{(a, b) \mid a \in A \wedge b \in B\}.$$

Proof.

Note first that the set comprehension notation we use in the theorem statement is not a true set comprehension, and can not be since it does not define a set which it draws elements from. The goal of this proof will be to show that there is such a set to draw elements from that makes a Cartesian product valid under the Zermelo-Fraenkel axioms.
Without loss of generality, proceed under the assumption that A and B are non-empty; if they are, this discussion is simply moot in such a case since the cartesian product is the empty set. We are ensured that an ordered pair (a, b) with $a \in A$ and $b \in B$ in the way we have defined it exists, since we may construct $\{a, b\}$ by the axiom of pairing and then restrict it down to $\{a\}$ via axiom of comprehension $\{x \in \{a, b\} \mid x = a\}$. From there, we may apply the axiom of pairing again to obtain the ordered pair $(a, b) = \{\{a\}, \{a, b\}\}$.
Now consider the following. The set $\{a, b\} \subseteq A \cup B$ where $A \cup B$ exists by the axiom of union (implicit invocation of axiom of pairing by setting $\mathcal{A} = \{A, B\}$ before the axiom of union $\cup\mathcal{A}$), and by the axiom of powerset, we have $\{a, b\} \in \mathcal{P}(A \cup B)$ and thus also $\{a\} \in \mathcal{P}(A \cup B)$. The ordered pair $\{\{a\}, \{a, b\}\}$ produced by the axiom of pairing containing two such sets in $\mathcal{P}(A \cup B)$ is then

$$\{\{a\}, \{a, b\}\} \subseteq \mathcal{P}(A \cup B)$$
and thus
$$\{\{a\}, \{a, b\}\} \in \mathcal{P}\big(\mathcal{P}(A \cup B)\big)$$

Then this is our set which we draw from for our comprehension. Define the Cartesian product to be the set comprehension

$$A \times B = \big\{x \in \mathcal{P}\big(\mathcal{P}(A \cup B)\big) \mid \exists a \exists b, (a \in A) \wedge (b \in B) \wedge (x = (a, b))\big\}$$

Since it contains all such pairs by construction, it is unique since any other set containing all such pairs will, by the axiom of extensionality, be the same set.

This example is far from archetypal with respect to rewrites (see the beginning of the following chapter where we discuss real numbers for a much more obvious one) but it is indeed an example. Our rewrites here are existential, since our axioms exist mostly to tell us what kinds of sets exist; the procedure of this proof is thus. First to case-split on A and B being empty sets or not (in a similar way that we pattern matched in the previous section), and if they are then we are done immediately. With the case of empty sets handled, we are ensured that A and B are not empty sets and thus *there exists* some a and some b such that $a \in A$ and $b \in B$; the rest of the proof serves to put these a and b into a pair by set comprehension. We have a context in which there is some $a \in A$ and $b \in B$, so we rewrite by axiom to existence of

$\{a, b\}$, and we have a different axiom that rewrites to existence of $\{a\}$. Another axiom uses existence of these to rewrite to the existence of $\{\{a\}, \{a, b\}\}$, etc. etc. More important than the particular lack of symbolic clarity of how a rewrite is occurring here, is that during the proof we do not think too hard about whether or not a step of the proof aligns with our intuitions or why it might be true. In fact, we do not even use our intuitions when they tell us something is simply true, we *rewrite* using the axioms provided exclusively.

It will of course often be the case that we have developed intuitions for some field and may make some statements more informally, but this is only with the confidence that we have mastered the use of axioms behind those intuitions. When this occurs in a textbook or in lecture notes, it is often jokingly referred to as "proof by intimidation", in the sense that the lecturer/author is insisting 'I/we know this to be true or obvious, and if it is not obvious to you then go verify this fact as you see fit'. This will eventually become impossible to avoid, as our tower of abstractions will grow to a point where complete statement of why every single reasoning step (and indeed its sub-steps) would blow proofs up to enormous size, making their reasoning much harder to follow. This is one case where the mathematician's metaphysical fascination is so valuable; we are not logicians, and while logic is of near supreme value to us, understanding the nature of an object (through its defining axioms) will prove a much greater substitute for totalizing reasoning.

Returning to our foundational discussion, it is typical to append to the Zermelo-Fraenkel axioms the axiom of choice, forming the famous 'ZFC' foundations of mathematics. This is in some sense quite a meaningful alteration to what it is that a set does; in a sense, it may be stated as 'existence within a set is the same as accessability'. The axiom of comprehension for instance allows us to restrict a set by a condition, and from there we might even show that the set is non-empty, i.e. that *something* satisfies that condition. It is the axiom of choice however that says that when we can show that *something* satisfies that condition, we may take it and use it without any knowledge of the set's structure. We do not need to *define* a choice function, we will simply always have one.

Definition maththink.4 — (Axiom of Choice)

For every family of non-empty sets there exists a choice function (in the sense that we defined a function in maththink.2) that selects an element of each set.

$$\forall \mathcal{A}, \left(\forall X, X \in A \Rightarrow X \neq \emptyset\right) \Rightarrow \\ \exists (f : \mathcal{A} \to \cup \mathcal{A}), \forall X, X \in \mathcal{A} \Rightarrow f(X) \in X$$

or 'for all sets $\mathcal{A}$ which has all members which are not the empty-set, then there must exist a function from $\mathcal{A}$ to its union which outputs a choice of element from each set X'.

This axiom of choice is formulated differently from the one we saw for type theory due to the disagreements between the two theories of how exactly functions work, and using sets instead of types to contain members. Nonetheless, it performs a similar role to the law of excluded middle in the previous chapter. In constructive mathematics we had the problem that existential statements, rendered to us as dependent pairs, could only say that a thing existed by giving an example of it; consequently, we had difficulty deducing that a thing exists from surrounding information that it must surely exist (e.g. 'if not for all then a counter example exists' from De Morgan's laws). We have a similar problem here, that one may say $A \neq \emptyset$ to say that the set is not empty, and yet, knowing that it is not empty does not really tell you how you grab an element out

of it. The axiom of choice says that there must be a function which does just that.

maththink.3 Object/Symbol Identity and the Concept of Equality

We are now in a position to extend our discussion on symbolic reasoning. That is, earlier, we discussed the notion of equality as related to which rewrites are valid, and considered the corseness of equality as a decision about whether an operation is truly equal to its result. In our description of the ZFC axioms, we also mentioned equality a number of times, but never specified what that equality meant exactly except when it meant equality of sets. This is complicated more, as mentioned earlier, by set theory being a theory built on top of a mathematician's intuitions; a theory where sets are able to contain any *thing* that a mathematician might want to talk about, without specifying where that thing came from, what it is, or how its equality works.

Consider the following example. If we interpret the construction we see in the axiom of infinity to be the natural numbers, then we can apply the axiom of extensionality and say that our zero has $\mathbf{Z} = \mathbf{Z}$ since $\mathbf{Z} = \emptyset$ and the axiom of extensionality says $\emptyset = \emptyset$ since they share all members (none). We can subsequently do this for every nested-set integer implied by the axiom of infinity since, while we may not have a notion of equality from elsewhere, we do have one for sets and objects which are merely sets in disguise as in the set-numbers implied by the axiom of infinity. But the moment we declare that the natural numbers we use are some other object which is not itself a set but are contained in a set via the axiom of replacement (e.g. say we define a function which takes axiom-of-infinity set-numbers to inductive natural numbers from type theory and use the axiom of replacement to define this as a 'set of natural numbers') we have to import our own notion of equality.

This is actually common. It is natural for definitions of many constructions to simply *declare* that a set of objects with the desired rewrite rules/equivalences exist and thus bring with it the necessary conditions for equality. Even in the previous section, this was why it was important for us to declare that inductive natural numbers are equal only when they are either both $\mathbf{Z}$ or both $\mathbf{S}$ acting on the same n; irrespective of our later equivalence defined by $\equiv$, we specified that objects such as constructors have such a quality as equivalence under same constructor and same arguments.

There will be times when our constructions provide unsatisfactory notions of equivalence however, where the notion of equivalence does not draw directly to any nice intuition of what an object *is*. In fact, it will often be the case that we actively abstain from referencing a broader system of constructions in order to make a theory more general, or a construction that itself defines its equivalencce rule based on another one which is not specified. This is the case in our set extensionality example; the moment our sets contain something other than sets themselves, we say two sets are equal when they have equal elements, but we cannot compare those elements without them having their own equality. This places us in something of a pickle which will require us to study equality more closely.

Let us study an example from type theory to ground ourselves. In most systems of constructive mathematics it is considered necessary to either derive from axioms or suppose explicitly what is called *function extensionality*. This axiom or rule says effectively that two functions are equal if and only if they both assign the same outputs to the same inputs.

$$\forall (f, g \colon A \to B), f = g \Leftrightarrow \big(\forall (a \colon A), f(a) = g(a)\big)$$

This is similar to our set theory extensionality, that said that sets also hold no data other than their

membership. Consider though, what it would mean if we did not suppose function extensionality. Then it would be true that functions contain some data in addition to their map between inputs and outputs. Since functions are defined with no additional information in type theory, the only thing left to say that they are different when they share inputs and outputs is their *symbol*. To say that $f \neq g$ specifically because f is written f and not g.

This is a valuable proposition: since we explicitly claim no knowledge unless stated otherwise about members of sets, in effect, they have untold amounts of identifying data and in effect are defined exclusively by their symbols. This is precisely what we meant at the beginning of the section when we said that an abstract symbol x could be anything and everything until *restricted down*. When we say 'let A be a set and $a, b \in A$', there is potentially infinite information held behind the symbols a and b; that potential for infinite information in fact does not go away when we say something like 'let $\mathbb{N}$ be a set containing a number 1 and an operation $\mathbf{S}\colon n \mapsto n+1$'. It only goes away at the point that we add a *natural number extensionality* and say '$1_a = 1_b$ for every $1_x \in \mathbb{N}$ (i.e. there is only one *one*) and $\mathbf{S}(n) = \mathbf{S}(n)$ for every $n \in \mathbb{N}$ (i.e. there is only one succession)' that we have successfully papered over that additional information, rendering it irrelevant.

Rather than a fault, this is a power of the set theoretic lens which otherwise might seem needlessly ontologically cautious. In set theory we have the power to define or construct a set and give it a rule for what is equal or what is not equal a priori, a power that does not exist quite so simply in type theory. In fact it can be valuable at times to informally think of a set as coming with its own equivalence rule. We can, just as we papered over what could have been untold infinite information held in the objects that we write as 1 or 2, paper over information that we *did* know about. This gives us a brand new way to construct things.

Definition maththink.5 — (Integers)

Define the set $\mathbb{Z}$ to be the set $\mathbb{N} \times \mathbb{N}$ with the following rule: when a two pairs $(a, b), (x, y) \in \mathbb{N} \times \mathbb{N}$ satisfy

$$a + y = x + b$$

we say that they are equal. Then we call $\mathbb{Z}$ the set of **integers**. A natural number can become an integer via the inclusion $n \mapsto (n + 1, 1)$.

This is the standard construction of integers in mathematics, the various conversations we will shorly have about formally stating the equivalence rule put aside momentarily. If by counting numbers we mean numbers that are strictly greater than or equal to one (this is the standard in mathematics as well, not starting from zero) then we invent negative numbers in this way. From the perspective of already having negative numbers, one easily manipulates this equation to see that it is read $a - b = x - y$, but this is invalid on natural numbers since there is no result for $x - y$ when $y > x$. But if we speak only of positive numbers, then we avoid these problems and instead speak of larger numbers rather than smaller ones, moving y and b to their opposite sides.

The technique we have utilized to do this does not exist without significant effort in type theory; over there, constructors with some set of parameters are *unique*, and one cannot weaken this. So what they do is to define integers similarly to $\mathbb{N} + \mathbb{N}$ with a l in-left constructor for negatives, a r in-right constructor for positives, and a constructor $\mathbf{Z}$ for zero. To construct rational numbers

which are much more obviously pairs, fractions such as $1/1$ and $2/2$ are equal, and so they must put in substantial work to reduce create a sigma type which is a pair of $\mathbb{Z} \times \mathbb{Z}$ along with a witness that it is in simplified form.

It may be valid to protest however, just as type theoretic constructions are too restrictive about equality, the above notion of equality is perhaps dangerously forgiving. Surely we are not saying that all pairs of natural numbers (m, n) are literally always integers, right? Integers are supposed to be at the same standing as natural numbers, or real numbers. Enabling different operations of course, but still distinct kinds of objects themselves.

This is in part our expectations about types creeping in, but these are not exactly bad expectations to have. The set theorists, prior to a similar popularity of type theory as we enjoy today, saw fit to try to formalize what we have done above. Thus we have the formal notion of a partition, equivalence relation, and equivalence class. In such a way, we can say that what we did above was not true equality (and thus not overwriting the properties of pairs nor natural numbers) but a different notion of equivalence, thus indicating the existence of a different set of which objects obey that equivalence natively.

To explore this, we will need to formally investigate some terms that were mentioned briefly in the previous section when we spoke about equivalence.

Definition maththink.6 — (Equivalence Relation)

An equivalence relation can be considered to be a set of pairs A. When a pair $(a, b) \in A$ then we write $a \sim b$ and either say that a is *similar* to b or that a is *equivalent* to b *under* $\sim$. Let B be the set of all items that $\sim$ relates to one another (we might write $B = \cup\cup A$ if using the notion of pairs defined in the cartesian product definition). Then in order to be an equivalence relation, $\sim$ must satisfy the following:

- (Reflexivity) For all $a \in B$, we have $a \sim a$ since a thing must be equal to itself
- (Symmetry) For all $a, b \in B$, if we have $a \sim b$ then we have $b \sim a$ since equality (or similarity) is not an ordered property; i.e. $(a, b) \in A \iff (b, a) \in A$
- (Transitivity) For all $a, b, c \in B$, if we have $a \sim b$ and we have $b \sim c$ then we have $a \sim c$, since if b is the same as both a and c then they must be the same as one another as well.

These three properties, capturing what we expect of the notion of 'sameness', formalize our notion of what it means for something to describe a kind of equivalence.

It is also possible to define relations in which one of these assumptions fail, the most important example being ordering. That is, an ordering relation $>$ is not symmetric, since $b > a$ does not imply $a > b$ and quite frankly it is important that it does not for it to be an ordering. However it does obey transitivity, since $c > b$ and $b > a$ imply $c > a$. There are some cases even where it is desireable for reflexivity to fail, such as in computer's floating point numbers: when you divide by zero in floating point numbers you get NaN, and it is not necessarily meaningful to say that two 'not-a-number's are equal to each other just because they are both not numbers, however the rest of the time reflexivity may work perfectly fine. If one speaks of the set of options one may play in rock-paper-scissors then there is a relation of 'who wins' which is not reflexive, symmetric

or even transitive, since rock beats scissors and scissors beats paper but rock *loses* to paper, so it fails to appear like an ordering because the ordering forms a cycle.

This notion of similarity translates the condition of set membership into a different condition; we can now describe integers by speaking of a set

$$[-3] = \{(a,b) \in \mathbb{N} \times \mathbb{N} \mid (a,b) \sim (1,4)\}$$

which is an *equivalence class* of pairs which would be treated as equal to -3 under our desired equivalence relation. Instead of resorting to simply saying that pairs *are equal* in order to form the integers, we have instead done this: we have defined a similarity relation, $(a,b) \sim (x,y)$ if $a+y = x+b$ in the sense of natural numbers, and then created a set of all pairs in $\mathbb{N} \times \mathbb{N}$ which also obey the condition that they are similar to $(1,4)$, one of many representatives of -3, via set comprehension. Since we restrict on the set of *all pairs* in $\mathbb{N} \times \mathbb{N}$, we have then captured *all pairs* which our similarity relation says represent -3, and thus we have the set of all -3 representatives. What we may choose to do, mostly as an exercise in semantics and making ourselves comfortable with our own tools, is to say that this set *is* the integer -3.

If we say this, then any other object that wants to *be* -3 will also need to have all representatives of -3 under the relation $\sim$ as members, and then will be equal via set extensionality. In such a way, we can also create a set for each other class which is unique under $\sim$, the -4, the $+5$, etc. each with their own set $[-4] = \{(a,b) \in \mathbb{N} \times \mathbb{N} \mid (a,b) \sim (1,5)\}$ or $[+5] = \{(a,b) \in \mathbb{N} \times \mathbb{N} \mid (a,b) \sim (6,1)\}$ and populate an entire set of integers in such a way.

$$\mathbb{Z} = \left\{ \left\{ (a,b) \in \mathbb{N} \times \mathbb{N} \,\middle|\, (a,b) \sim (c,d) \right\} \in \mathbb{N} \times \mathbb{N} \,\middle|\, (c,d) \in \mathbb{N} \times \mathbb{N} \right\}$$

This structure is called a *partition* on $\mathbb{N} \times \mathbb{N}$, and is defined by the property that it is a set of sets where every member of $\mathbb{N} \times \mathbb{N}$ goes into precisely one set; so $(1,1)$ goes into the set we designate $[0]$, which contains $(1,1)$ but also $(5,5)$ and $(23,23)$, and yet no other $[n]$ will contain $(1,1)$.

This makes precise our notion of papering over additional information, and is generally referred to as a *set quotient*. That is, in the definition we have given above, we would write $\mathbb{Z} = (\mathbb{N} \times \mathbb{N})/\sim$ as though we had divided by our similarity relation.

The reason I have chosen not to describe these partitions and equivalence classes in a more precise manner however (breaking out the green boxes as it were) is because, at least for the objects we have described so far, how much of these constructions matter when discussing the integers is a matter of perspective. For instance, we have convinced ourselves with the above constructions that it is certainly valid to speak of integers, since we may construct a set with their desired properties and transplant any desired computations we want integers to do as required (e.g. $[-3] - [+2]$ meaning some $(a, a+3)$ and $(b+2, b)$ and we take the result to be the equivalence class containing $(a+b, a+b+5)$). And yet, it is usually the case that we simply take the integers to exist, having been generally convinced that they can exist, in which case -3 and $+2$ are themselves distinct objects with no internal structure, least of all sets of pairs.

These concepts, of similarity relations, partitions, equivalence classes, find their formal use not in constructing objects we are already familiar with but in constructing new objects for which we do not have existing intuitions. Indeed, we will next see them when we discuss point-set topology; when applied to patches of the real number line, these quotients can take any notions of closeness defined for some space and say that 'we say that these two points are similar' thus creating a sort of portal or *gluing* the two points together in space, creating loops.

mathtthink.4 Remaining Prerequisites: Some Concepts and Syntax

By now, we have discussed most of the concepts which mathematical reasoning implicitly expects awareness of but does not discuss formally. More examples of how axioms construct concepts will readily become apparent in the following chapter which will provide ample examples. There are however a few loose ends to cover.

First, we must return to our notion of functions, or set morphisms as we described them. Amongst all the discussion of how sets replace our notion of types, and how our objects are now made via restriction rather than construction, it is perhaps difficult to imagine that in many circumstances sets will not be so bad a replacement for types. The notion of a domain and a codomain for functions, a set from which they draw their inputs and a set from which they assign outputs respectively, will usually and desirably be sets defined by structure that makes them nearly tantamount to types themselves. From there, the following concepts will become relevant to us.

Definition mathtthink.7 — (Properties of Set Functions)

Let A, B and C be sets, and $f\colon A \to B$ be a function f with domain A and codomain B, likewise $g\colon B \to C$. Then:

- we call f **injective** or **one-to-one** if for all $x_1, x_2 \in A$ we have $f(x_1) = f(x_2)$ if and only if $x_1 = x_2$. That is, f is injective if every unique input has a unique output. An injective function is invertible but only on the subset of its codomain which is its image.
- we call f **surjective** or **onto** if for all $y \in B$ there exists some $x \in A$ such that $f(x) = y$. That is, f is surjective if we can find an input for every possible output, or that the set of f's outputs *fills* the codomain. A surjective function can find an inverse for every output but there may be more than one inverse, in which case we describe the function as *many-to-one*.
- as mentioned in the previous section, the $\circ$ operation takes two functions and **composes** them so that the output of one is the input of the other. When two functions f and g have the property that the domain of one is the codomain of the other (B in this case), then we can compose them and write $g \circ f\colon A \to C$. For any $x \in A$ it is defined $g \circ f(x) = g(f(x))$. We can also do this when the codomain of f is a subset of the domain of g.
- we call f **isomorphic** if it is both injective and surjective. In this case, since we can find an input for every output, and that input is unique, we can establish a mapping *from* the output *to* the input, an **inverse** map which we call $f^{-1}\colon B \to A$ with domain and codomain reversed. It has the properties that $f \circ f^{-1}\colon B \to B$ and $f^{-1} \circ f\colon A \to A$ are **identity functions**, meaning they send every input to itself and change nothing. When an isomorphism exists between two sets A and B then we may write $A \cong B$ to mean that they are **isomorphic**.

Additionally, we write $f(A)$ to mean the image of a set, the set of its outputs; if f is surjective then $f(A) = B$. Even when f is not isomorphic and thus not invertible, we may at times speak of $f^{-1}(D)$ if D is a subset of the image of f, to mean

$$f^{-1}(D) = \{x \in A \mid f(x) \in D\}$$

which we call the **preimage** of D.

Sometimes when there exists an isomorphism between two sets, we may say that the sets themselves are isomorphic, and sometimes we may even then say that they represent the same underlying set and treat them as equal. This will be something to discuss later but treating sets as equal when they are isomorphic is a strong claim that will at times be appropriate and most of the time not be (I expect a full discussion of this to appear in chapter 3).

Now that we have formally discussed relation properties, we should also collect into one place the various rules that a binary operation can have, such as $\circ$, $+$ or $\div$.

Definition maththink.8 — (Properties of Binary Relations)

Let $+\colon A \times A \to A$ represent a binary operation, taking two members of A to another member of A. Let $a, b, c \in A$. Then there are three rules that it may or may not have that are important to remember.

- (**Associativity**) the operation is associative if $(a+b)+c = a+(b+c)$, so the order in which we calculate individual pairs does not matter
- (**Commutativity**) the operation is commutative if $a+b = b+a$ so that individual pairs may be flipped without changing their value
- (**Identity**) the operation has an identity if there is some $e \in A$ for which $e + a = a$ and $a + e = a$, that is, an object which does nothing to the other object when the operation acts. If it satisfies only $e + a = a$ then we call it a *left-identity* and if it satisfies only $a + e = a$ then we call it a *right-identity*.

Many binary operations we have seen so far obey these, but notably operations such as subtraction or division which are simply operations composed with an inverse are not. Here some are listed:

- Addition ($+$) is associative and commutative with identity zero.
- Multiplication ($\times$ or $\cdot$) is associative and commutative with identity one.
- Logical-And ($\wedge$) is associative and commutative with identity $\top$ the true proposition.
- Logical-Or ($\vee$) is associative and commutative with identity $\bot$ the false proposition.
- Set Union ($\cup$) is associative and commutative with identity $\emptyset$ the empty set.
- Set Intersection ($\cap$) is associative and commutative, and when it exists in the context of some total-set X (as is often the case when discussing subsets of a *space*) then it has identity X.
- Function composition ($\circ$) is associative but not commutative, but has identity which is the identity function, the function that takes an input and returns it as output unchanged.
- Cartesian Product ($\times$) is in practice associative since we are generally unconcerned with finer structure beyond the creation of tuples, so $(A \times B) \times C$ is treated as equal to $A \times (B \times C)$ since we only care that their members are triples (a, b, c) with $a \in A$, $b \in B$ and $c \in C$. If we

enforce that cartesian products only make pairs, i.e. $A \times (B \times C)$ has elements $(a, (b, c))$ then it is not associative. It is also not commutative since $A \times B$ has members (a, b) but not members (b, a). It does in a sense have an identity however, since any set $\{z\}$ with only one element will produce a set of pairs $(a, z) \in A \times \{z\}$ which is isomorphic to A via $(a, z) \mapsto a$ and $a \mapsto (a, z)$, however this is the sort of thing that we care about contextually.

There also exists a notion of subtraction on sets, which we call *set minus* and denote by the backwards slash $\setminus$. The set $A \setminus B$ is then the set containing all elements which are in A but not in B. Consequently you might write

$$A \setminus B = \big\{a \in A \,\big|\, a \notin B\big\}.$$

Before moving on from this chapter, we should also emphasize some notations for writing functions which were mentioned multiple times but should be clarified together here in case details were missed. That is, in general mathematics, we write $f\colon A \to B$ to say that f is a function with domain A and codomain B, and we write it in this way specifically with the arrow $\to$. This arrow is distinct from the arrow starting with a bar $\mapsto$, which describes not the set of inputs and the set of outputs A and B, but instead a particular pattern for how to evaluate a function. For instance, we might write "a" $\mapsto 1$ to describe that a function on letters specifically maps the letter "a" to the number 1, but we might write $x \mapsto x + 1$ to describe a function that maps *any* number x to the number $x + 1$, and this is primarily contextual.

It is very important that you **do not confuse these arrows** $\to$ and $\mapsto$. We may at times write that instead of $f\colon A \to B$, it is $f\colon a \mapsto b$ where b is some expression involving what to do with a; in this context, we are basically writing that f is defined to be the function that does $f(a) = b$ without explicitly setting the domain and codomain, but we can write $f\colon a \mapsto b$ and $f\colon A \to B$ together as well. A function which does different things in different *cases* may be written with *piecewise notation*,

$$f(x) = \begin{cases} x + 1, & x < 0 \\ x + 2, & x > 0 \\ 0 \end{cases}$$

where in general we have a list of different things f could do to x and conditions beside them that say when to do these things. Sometimes the propositions will not describe every case of x, in which case an option should be provided at the bottom without a proposition, which is the fallback case that the function takes when no other case applies.

We must also discuss notations relating to sequential patterns, starting with *big operator notation*. Once in a while this notation makes rounds on the internet as 'just being a for loop' and half that time that description is quite apt. When we have a binary operator such as $+$ or $\times$ which is associative and commutative, it is meaningful to simply take a list of values and say 'simply apply the binary operator to all of them, in any order' since, as commutativity and associativity describe, ordering really does not matter. In this case, we may write

$$\sum_{i=1}^{n} f(i) = f(1) + f(2) + f(3) + ... + f(n)$$

which we call **summation** notation. The rule with this pattern is that we have an expression on the right hand side of the capital greek sigma Σ involving a variable i which we declare below it (this can be some other variable such as n or k or j) and we add the expression repeatedly, going through each counting number starting with the number we set i equal to, 1 in this case, upwards until we reach n. We can also as necessary start with a negative number, in which case it is understood that we are counting upwards in the integers. A matching notation exists for multiplication, called the **product notation**

$$\prod_{i=1}^{n} f(i) = f(1) \times f(2) \times f(3) \times ... \times f(n)$$

in which we use capital Pi Π with the same system. In situations we discuss later, it may even become appropriate to write infinity in the top instead of n to mean summing or multiplying the expression over all counting numbers.

This notation is a special case of a more general notation in which we do an operation over an entire set. That is, when we write $i = 1$ on the bottom and n on the top, the more general notation would be to write $A = \{1, 2, 3, ..., n\}$ and then

$$\sum_{i \in A} f(i) = f(1) + f(2) + f(3) + ... + f(n)$$

meaning that we perform the operation over all members of the set A. At times also we can, when it is well enough communicated, write a short proposition on the bottom of a big-operator notation to describe the operation applying over values which are true. In such a fashion, we can rewrite the above as

$$\sum_{1 \leq i \leq n} f(i) = f(1) + f(2) + f(3) + ... + f(n)$$

as long as we have some reasonable assumption that it is implied $i \in \mathbb{N}$. We can also stack these big operators together to do many more sums, for instance

$$\sum_{i=1}^{n} \sum_{j=1}^{n} ij = 1 \times 1 + 1 \times 2 + ...1 \times n + 2 \times 1 + 2 \times 2 + ... + n \times n$$

or it may instead be appropriate to put these values together in order to add an extra condition, such as in

$$\sum_{1 \leq i < j \leq n} f(i, j) = f(1, 2) + f(1, 3) + ... + f(1, n) + f(2, 3) + ... + f(n - 1, n)$$

in which the structure of the proposition only adds terms corresponding to pairs (i, j) when i is less than j.

We may circumstancially apply these big operators when the operation they describe is not associative or commutative, so long as we define exactly what we mean by this. For instance, while it will rub many mathematicians the wrong way, for a collection of indexed functions $f_1, f_2, ..., f_n$ we can compose them all if as long as we say first that we mean composition to act in order from left to right or right to left. For instance, we would then write

$$\bigcirc_{i=1}^{n} f_i = f_1 \circ f_2 \circ f_3 \circ ... \circ f_n$$

if we are specifying that it starts leftmost with the first choice of i and counts upwards to the right.

We must also mention the symbol ... itself. Obviously this is meant to describe the full extension of a pattern, but it will sometimes be used so frequently that it almost constitutes a symbol itself. As such I should specify the general practice with ..., the **ellipsis**, that we generally begin the sequence with enough items to establish a pattern and then end it with a final term if indeed there is a final term. See above that in each time we have used it we have used two or three examples of the pattern before the ellipsis and then completed its final element. For illustrative purposes, it may at times be appropriate to write two or three examples at the end of the pattern, especially when we want to do something with one of those examples for the purposes of a mathematical manipulation. When we are pressed for space and the pattern has already been well established, only then will we write $f(1) + ... + f(n)$, always buffered on each side by the appropriate operation.

In a case where for some reason we are adding to infinity (a concept which we will need to strictly define later) then it may become appropriate to not write a final term

$$\sum_{n=1}^{\infty} f(n) = f(1) + f(2) + f(3) + ...$$

These notations also extend to union and intersection, so a sequence of sets may have

$$\bigcup_{i=1}^{n} A_i = A_1 \cup A_2 \cup A_3 \cup ... \cup A_n$$

$$\bigcap_{1 \leq i < j \leq n} A_{i,j} = A_{1,2} \cap A_{1,3} \cap ... \cap A_{1,n} \cap A_{2,3} \cap ... \cap A_{n-1,n}$$

We will often reuse symbols such as $+$ or $\times$ or indeed our big operators Σ or Π to mean different things when we have defined other concepts which we consider analogous to addition or multiplication. For instance, while we have $\int$ to represent integrals in calculus, the same way integration can be thought of as a sum over an area, there is an analogous notion of a product over an area which is generally denoted with product notation for lack of a better symbol (when it is not denoted $\exp(\int \log f(x)dx)$ instead or something of the sort), or a large number of multiplications in the sense of group theory may also use big Π.

Finally, we have discussed context in the labelling of variables before. Mathematicians (especially in particular fields) tend to have specific symbols which they use to denote specific things as part of their language with one another. Accordingly, one tends to know that f is a function or x is a real number without any instruction. We list some of these here for your reference, and so you are not left out of these conventions, but in doing so we mention some things you may encounter elsewhere or later, so be sure to ignore anything you do not understand here until later.

- x can be used to represent a point in a space or an input to a function. Accordingly, we sometimes write a point in space as $\mathbf{x}$ or $\vec{x}$, etc. in which case it may have coordinates x_1, x_2, x_3, but in other circumstances (especially in two or three dimensions) these may be labeled x, y, z.

- y or z are often used as an output for a function or as a spatial coordinate. It may also be used when x is taken as a sort of second-preference x.
- When x, y, z are in use, our next set of preference letters are generally u, v, w. These may also be used to describe coordinates in some transformed space where x, y, z are the inputs or base space, as is the case in UV-maps in 3d model skins; it is precisely because a 3d model's skin is a bizarre map from 2d to 3d that we consider the 3d space version of the skin to be x, y, z and the virtual space in which the skin were flat as the u, v coordinates.
- n is typically used to mean a counting number in $\mathbb{N}$ with m used as a second preference for this
- i, with j and k as second and third preferences are often used to *index* some indexed object. That is to say, if we have three things x_1, x_2, x_3, we often write x_i or x_j to mean 'one of our labels'.
- r is often used to describe a radius, but this may also mean that it describes a radius *vector*, and thus a position in space *relative to a origin*.
- f, second preference g and third preference h are preferred letters to refer to functions
- Capital letters may be used to describe things which we do not think of as numbers but as containing numbers or number-adjacent things. Accordingly, sets are generally written with capital letters such as A, B, S, X, and we often use the lower case version of the letter to denote a member of that set, so $A \times B$ will often be written with elements (a, b), implying $a \in A$ and $b \in B$.
- As we move on, we may encounter collections of objects which are collections of collections, and our first way of saying that something does not merely contain numbers or number-like things but contains things which themselves contain those, is to write it with a special script. This will often be a **caligraphic script**, and so we might write $\mathcal{A}$ to mean a set of sets for example.
- q may be found used in place of x in many texts related to physics to describe position, however physics has many many more conventions and expectations about which letters mean what, such as p for momentum, V for kinetic energy, etc. and frankly too many to list here.
- In general, it is often the preference when speaking about some collection of objects G to call some member in it its lowercase letter, so we might say $g \in G$. When this letter is taken, or indeed we wish to speak about more than variable, the usual practice is to go up in the alphabet, often trying to exclude letters which are known to have strong connotations as a first preference to mean something else. so we would say $g, h, k \in G$, with g and h included since g is the lowercase G and h comes after k, but since i and j are first and second preference letters for indicating index, it is convention to skip them and go to k.
- We usually skip the letter o to be used as a variable name since it looks too much like zero.
- Often when there is not a good choice of letter in latin script, we use greek letters. Many greek letters have strong associations in mathematics for what they mean, for instance, the greek iota ι tends to mean some kind of *inclusion*, a way of sending a value to basically itself but as

contained in a larger set. Our first choices for greek letters are generally ψ, ϕ called psi and phi, however it is often preference to use a variant writing of phi written φ (I am a fan of this one). Mathematicians tend to have conventions for handwriting these greek letters especially when it comes to those which look a lot like latin characters (i.e. κ and k) but you are better off asking one and receiving a jubilant ideosyncratic rant. Some mathematicians, when there is no good greek character left in a context, will resort to using cyrillic letters such as Zhe although my editor will crash if I try to include that here.

- We double up the use of letters sometimes by marking them in special ways. Some common ones are $\bar{x}$ called 'x bar', $\hat{x}$ called 'x hat', x^* or $x^\star$ which are generally referred to as 'x star', and $\tilde{x}$ called 'x tilde' or 'x squiggle'. Some other markings have special meanings; for instance y' is often called 'y prime' and $\dot{y}$ is called 'y dot', however these markings both often refer to derivatives of y. Each of these markings when used on a letter that already means something in a context often means that the value assigned to that marked letter is strongly tied or related to the main letter but different somehow.

- We use a special bold face script to denote certain canonical sets of numbers or number-like things which are important or well studied. For instance we have $\mathbb{N}$ for the natural numbers, $\mathbb{Z}$ for the integers, $\mathbb{Q}$ for the rationals and $\mathbb{R}$ for the real numbers. There are also other sets which may be stranger such as $\mathbb{C}$ the complex numbers, $\mathbb{H}$ the quaternions, or $\mathbb{PR}^n$ the real projective space in dimension n. Some authors will generalize their theorems and say something like "for any field $\mathbb{F}$ ..." to mean that their theorems work on both $\mathbb{R}$ and $\mathbb{C}$ or any other algebraic field satisfying the properties to make it behave like a similar number system; in that case $\mathbb{F}$ is but a variable for some set.

Again, many terms here are mentioned in advance of the time they show up so that this list may also act as something of a reference. In future, our preferences for what symbols we want to mean what will generally become apparent simply because we will use them a lot, so be sure to pay some attention to which letters we preferentially use as we develop our mathematical language.

maththink.5 On to Chapter Two

If you are still reading, congratulations and thank you. You have made it through the hurdle of learning the rules of mathematical writing and most of the unwritten rules of mathematical thought.

In the following chapter, we will start doing some real math in all its symbolic glory, now that we have seen, hopefully, that symbolic manipulation is much less scary than it looks, and moreover mainly a proxy for intuitive although strict reasoning. We will develop that reasoning over the next chapter as we study the properties of real numbers and space as thought of as real numbers.

Thank you again for following along, and I hope this text proves useful and insightful to you.

Chapter 2

Anatomy of $\mathbb{R}^n$: A Brief Introduction to Real Analysis

realnumsax Real Numbers from Axioms

As I have said in the previous chapter, real analysis usually proves a novel difficulty for the newly interested mathematician, since they are generally practiced on the intuitions of physics dressed up for the purposes of analytic geometry. This can make foundational statements about the real numbers appear tautological since it is unclear how one applies 'strict rigor' when we are unsure what rules we have and which we do not, thus returning us to the potholes of a more naive intuition's reasoning.

My sense about this is that it is most easily remedied by simply working within the rules proper from the very beginning. One discovers on meeting real analysis, for the first time, that they did not truly know what a real number really was. Natural numbers, integers, rational numbers, each have strict rules for where new instances come from, either by counting rules, or extending subtraction or division as necessary, but this is not so immediately true for the real numbers, not in any useful way anyway. It could equally be said inversely that real numbers are so easy to create that it is hard to understand how exhaustive (or indeed restrictive, if you have been convinced that dx is a number) the reals are.

So we will continue, in some sense, our discussion from the previous chapter into this one, and describe the real numbers as a set defined with member objects that obey certain symbolic rules. In this way, a real number will be no more and no less than the totality of these rules to us than their consequences and reinterpretations.

A final note that must be said at the beginning of this chapter: it is presumed that you read the previous chapter. If you did not, either because you felt sufficiently familiar with the language of mathematics or because it somehow struck you as unapplicable, then first and foremost, my apologies. Secondly however, you will certainly need to know if you did not read or follow the previous section, that the symbol $\in$ which is read 'in' or 'is a member of'; this means $a, b \in \mathbb{R}$ is to be read as 'a and b are in the set of real numbers' and thus interpretted that a and b are real numbers by loose analogy to a type relation. Moreover I highly recommend that you at least keep an open tab of the latter parts of Nascent's Philosophy of Mathematics and Rewrites

and Sets, both which contain a few lists of definitions which you should check should you encounter a word or notation you are unable to immediately understand.

realnumsax.1 The Real Number Axioms

There are of course multiple axiomitizations of the real numbers, but to avoid confusing you, we will use these fourteen, most of which are borrowed from other constructions. That is, the set of real numbers is a *complete ordered field*, meaning that it is first an algebraic field defined with addition and multiplication, subtraction and division, and other appropriate rules; it has an ordering so every number can be compared to one another, and it is complete in a way we will discuss shortly. Short of completeness, these axioms in fact describe the rational numbers, and in fact each of those terms, completeness, orderedness, or an algebraic field, are important and relevant on their own. For our purposes, by the time we meet many of these concepts formally, we will think of them in many ways as 'real numbers but without' some property.

Definition realnumsax.1 — (Real Number Axioms)

Denote by $\mathbb{R}$ the set which is an ordered field for which all subsets have a least upper bound. That is:

- (Algebraic Field) $\mathbb{R}$ is a set with a binary operation called addition and a binary operation called multiplication. For elements $a, b \in \mathbb{R}$, we denote addition as $a + b$ and multiplication as ab or sometimes $a \cdot b$, rarely $a \times b$. For any $a, b, c \in \mathbb{R}$ these operations satisfy
  - (RNFA1) addition is associative, meaning $(a + b) + c = a + (b + c)$.
  - (RNFA2) addition is commutative, meaning $a + b = b + a$.
  - (RNFA3) addition has an identity z_+, satisfying $a + z_+ = a$ and $z_+ + a = a$.
  - (RNFA4) addition is invertible, so for all $a \in \mathbb{R}$ there exists a unique w such that $a + w = z_+$ leaving only the additive identity.
  - (RNFA5) multiplication is associative, meaning $(ab)c = a(bc)$.
  - (RNFA6) multiplication is commutative, meaning $ab = ba$.
  - (RNFA7) multiplication has an identity $z_\times$, satisfying $az_\times = a$ and $z_\times a = a$
  - (RNFA8) multiplication is almost always invertible, so for any $a \in \mathbb{R} \setminus \{z_+\}$, that is, any number a which is not the additive identity, there exists a unique w such that $aw = z_\times$ and $wa = z_\times$.
  - (RNFA9) multiplication is **distributive**, meaning that $a(b + c) = ab + ac$.
- (Ordered) $\mathbb{R}$ is a set with a binary relation $<$. For all $a, b, c \in \mathbb{R}$ it satisfies
  - (RNOA1) the relation forms a *trichotomy*, so either $a < b$ or $a = b$ or $b < a$, but there is no fourth option and exactly one of the three is always true.
  - (RNOA2) the relation is transitive, so if $a < b$ and $b < c$ then $a < c$.
  - (RNOA3) translations preserve orderings, so if $a < b$ then $a + c < b + c$.

  - (RNOA4) positive dilations preserve orderings, so if $a < b$ and $z_+ < c$ then $ac < bc$.

- (Completeness) For all nonempty sets $A \subset \mathbb{R}$ which have an **upper bound**, that is, some number $b \in \mathbb{R}$ such that $a < b$ or $a = b$ (i.e. $a \leq b$) for all $a \in A$, there must exist some **least upper bound** called the **supremum**, which we write $\sup(A) \in \mathbb{R}$. The least upper bound must simultaneously satisfy that $a \leq \sup(A)$ for all $a \in A$, the property of an upper bound, and that for all upper bounds b of A, it is $\sup(A) \leq b$.

You may immediately deduce the way that your understanding of the real numbers maps onto this description. That is, although wrapped up in symbols, $z_+ = 0$ and $z_\times = 1$, and that the additive inverse for some $a \in \mathbb{R}$ is $-a$ while the multiplicative inverse is $1/a$. However I must emphasize that notions such as these, the idea that the inverse of a number is one divided by it is a notation, and writing a negative sign in front of a number to denote its additive inverse (via $(-1)\cdot a$) is yet to be validated. Indeed, even the statement $z_+ = 0$ and $z_\times = 1$ is a choice that defines how we write the real numbers, fixing its scale by saying that the quantity $z_\times - z_+$ will be written by us as 1.

This collection of fourteen axioms combines three different things that we tend to want from a 'number system' that could enable us to speak on matters of geometry and algebra simultaneously. It is an *algebraic field*, in the sense of a number system with addition, subtraction, multiplication, division, a zero that adds nothing, and a one that multiplies nothing. These properties are also satisfied by the complex numbers, the rational numbers, and indeed the binary field which is composed of only the elements $\{0, 1\}$ where computer science is concerned. It is ordered in the way we need it to be in order to *measure* things and compare those measurements on one strict axis, like the counting numbers, the integers, and the rationals; it enables us to say that one thing is greater than another or smaller or equal but never incomparable.

But then the real numbers have one other strange property which I might argue enables it to be useful in matters of geometry, which is that it is *complete*. The axis created by the order axioms cannot ever be partitioned in such a way that there are two non-overlapping sets with no element between them, because *there are no gaps* since $\mathbb{R}$ is complete. In fact all of the axioms stated, up until the condition of completeness, describes the rational numbers which are not themselves complete. To state a collection of axioms for the reals is significantly easier than it is to construct the real numbers in any type theoretic sense precisely because of the axiom of completeness. The description I just gave, of real numbers as a real ordered set *with no gaps* corresponds to the 'Dedekind cut' construction of the real numbers, where every number is uniquely identified by a point you can split the real number line in two at leaving a point in the middle.

While the nuances of constructing the real numbers is far beyond the scope of this section, it is valuable to understand for later that real numbers are not nearly as easily discussed or pointed at as one might think. Indeed, we will discuss later in this chapter how almost all real numbers are effectively impossible to discuss. But it is our job for the majority of this chapter, so long as we wear the hats of 'real analysts' that we remain uninterested in the various difficulties of that construction; the study of real numbers has many much more interesting things to tell us even with that put aside.

First, since I had promised that this chapter would explore the practical implementation of rewriting reasoning with real numbers, let us write explicitly some rules which are not written above since they are a combined consequence of the real numbers and the properties of equality. This will begin our promised elaboration of math as simply a list of rewrite rules; we are in some sense rejecting a pursuit of *truth* to demonstrate the power of truth-as-process, since all we know about our

real numbers is these fourteen symbolic rules.

Lemma realnumsax.2 — (Real Number Algebra Rewrite Rules)

Let $a, b, c \in \mathbb{R}$ be real numbers and let $f\colon \mathbb{R} \to \mathbb{R}$. Then the following rewrites are conditionally valid.

a. $a = c$ if and only if $a + b = c + b$

b. if $b \neq 0$ then $a = c$ if and only if $ab = cb$

c. if $a = c$ then $f(a) = f(c)$

d. if f is injective then $f(a) = f(c)$ if and only if $a = c$

Proof.

Recall that equality is reflexive, meaning that for all symbols x, we have $x = x$. For part a of the lemma, this means that both $a = a$ and $a + b = a + b$. Since equality implies a rewrite rule, we can rewrite $a + b$ to $c + b$ on the right hand side, obtaining $a + b = c + b$. For the reverse direction, take $d \in \mathbb{R}$ to be the additive inverse of b, so that $b + d = 0$. Apply the rule once more to obtain

$$a + b + d = c + b + d$$

followed by applying associativity of addition (RNFA1) to place brackets

$$a + (b + d) = c + (b + d)$$

and then applying the additive inverse property (RNFA4, i.e. rewrite $b + d$ to 0) followed by the identity property (RNFA3) to eliminate zeroes via $x + 0 = x$.

$$\begin{aligned} a + 0 &= c + 0 \\ a &= c \end{aligned}$$

Thus $a + b = c + b$ also implies $a = c$.
We can repeat this process also for multiplication, saying that $ab = ab$ by reflexivity of equality, and then rewriting ab on the right hand side to cb. Once more, obtain the reverse direction by assigning d the multiplicative inverse to b satisfying $bd = 1$ and applying the rule once more

$$\begin{aligned} abd &= cbd \\ a(bd) &= c(bd) \\ a(1) &= c(1) \\ a &= c \end{aligned}$$

applying this time RNFA5, RNFA8 and then RNFA7 to repeat the steps mentioned before for multiplication instead of addition.

Once again, we do not do this by thinking about what *is true*, we do this on the basis of the rules that we have, those being reflexivity and rewrite under equality.
For part c this is even simpler, taking $f(a) = f(a)$ by reflexivity and rewriting the right hand side to $f(c)$ and obtaining $f(a) = f(c)$. In general, the reverse direction does not hold unless f is injective, in which case we move to part d.
For part d, recall the definition of injectivity. A function f is injective when it satisfies the rules that $f(x_1) = f(x_2)$ if and only if $x_1 = x_2$. In this case, our x_1 is a and our $x_2 = c$, so if $f(a) = f(c)$, we notice that what we are trying to prove is the literal definition of injectivity, so we are done.

□

These rewrite rules will be taken as trivial from now on and exercised simply by saying 'we [do a thing] to both sides'. In fact if one thinks of $f(x) = x + b$ or $f(x) = xb$ then both parts a and b's forward implications were merely special cases of part c; this means as well that the rule extends to subtraction and division. In fact we never particularly used the property that f was a function on real numbers either, so in fact these *algebraic rewrites* remain valid on any system of numbers for which we have a valid function f.

A series of obvious constructions must follow, along with some less obvious ones. Given the ordering $<$, we have the dual ordering $>$ in which $a < b$ if and only if $b > a$, as well as the partial order $a \leq b$ which can be treated as the proposition $a < b \vee a = b$, with its dual $\geq$. We also consider subtraction to be the operation of taking a number's additive inverse and adding it instead, likewise for division multiplying a number's multiplicative inverse. This can be stated as saying, literally, that $a-b = c$ is *defined* as the solution c to $c+b = a$, and $a/b = c$ is defined as the solution c to $bc = a$. It is preferable at this stage however, since these definitions correspond to unique solutions, to simply consider these inverses as functions which we will do shortly.

Let us show some well known facts in order to explore how we arrive at common intuitions from these pure symbolic rules.

Proposition realnumsax.3

Let $a, b, c \in \mathbb{R}$. For the purposes of notation, let $I_+ \colon \mathbb{R} \to \mathbb{R}$ be a function that maps each real number to its unique additive inverse, i.e. $a + I(a) = z_+ = 0$ for all $a \in \mathbb{R}$. Likewise, define $I_\times \colon \mathbb{R} \setminus \{0\} \to \mathbb{R} \setminus \{0\}$ be the multiplicative inverse which sends each number that is not zero to its unique multiplicative inverse, so that $a \cdot I(a) = z_\times = 1$. Given our above description of subtraction and division, this means that we read $x - y$ as $x + I_+(y)$ and x/y as $x \cdot I_\times(y)$ and take these as rewrite rules. Then the following hold.

a. $I_+(I_+(a)) = a$ and $I_\times(I_\times(a)) = a$.

b. If $ac = 1$ then $c = 1/a$ which we also write $c = a^{-1}$. In effect, $I_\times(a) = 1/a$.

c. Any number multiplied by zero is zero, i.e. $z_+ \cdot a = z_+$.

d. If $a + c = 0$ then $c = (-1)a$, or rather, in general $I_+(a) = I_+(z_\times)a$. In effect, $I_+(a) = -a$ where $-a$ is shorthand for $(-1)a$.

Proof.

First let us show part a. This is the statement that for all real numbers, for both addition and multiplication, the inverse of a number's inverse is itself. For addition we can apply the property that $x + I_+(x) = 0$ with $x = I_+(a)$ to obtain

$$I_+(a) + I_+\big(I_+(a)\big) = 0$$

Applying commutativity of addition (RNFA2) we can swap the sides of the addition and obtain

$$I_+\big(I_+(a)\big) + I_+(a) = 0$$

As stated above, this matches the form of subtraction, $x+I_+(y) = x-y$, where $x = I_+\big(I_+(a)\big)$ and $y = a$. This means that we are considering

$$I_+(I_+(a)) - a = 0$$

But we had said earlier that subtraction is defined $x - y = z$ by being the z that solves $z + y = x$, so this may be rewritten as

$$I_+(I_+(a)) = 0 + a.$$

Finally applying RNFA3 to eliminate zeroes, we obtain $I_+(I_+(a)) = a$. The process of the proof is the same for multiplication, except that at the end we apply RNFA7 to eliminate $z_\times$, the one's.

Part b is significantly easier, as it is basically tautological. Recall the definition of division as above, that is, $x/y = w$ has w that solves $wy = x$, presuming that $y \neq 0$. Set $x = 1 = z_\times$ and $y = a$; then $wa = z_\times$. But by RNFA8 this w must be the unique multiplicative inverse of a, $w = I_\times(a)$. Rewriting the original division, we then have $1/a = I_\times(a)$.

For part c, we apply RNFA3 in reverse to turn z_+ into $z_+ + z_+$, adding a zero. Thus z_+a is also

$$\begin{aligned} z_+a &= (z_+ + z_+)a \\ &= z_+a + z_+a \end{aligned}$$

by distributivity (RNFA9). Subtracting $z_+(a)$ from both sides (or rather adding $I_+(z_+a)$) we obtain via RNFA4 and RNFA3

$$\begin{aligned} (z_+a) + I_+\big(z_+(a)\big) &= (z_+a) + (z_+a) + I_+\big(z_+(a)\big) \\ z_+ &= z_+a \end{aligned}$$

showing that zero multiplied by anything is zero.

Part d consists of proving $I_+(a) = I_+(z_\times)a$. Begin from reflexivity of equivalence, writing $I_+(a)/a = I_+(a)/a$, a number's additive inverse divided by itself. Set $w = I_+(a)/a$ and then add one to both sides, obtaining

$$w + 1 = \frac{I_+(a)}{a} + 1$$

Now by RNFA8, we replace $1 = aI_\times(a) = a/a$ and write

$$w + 1 = \frac{I_+(a)}{a} + \frac{a}{a}$$

These divisions by a are of course shorthands for multiplying by $I_\times(a)$, so we may apply distributivity (RNFA9) and rewrite the right hand side to $\frac{1}{a}(I_+(a) + a)$.

$$w + 1 = \frac{I_+(a) + a}{a}$$

But now by RNFA4, the top of the fraction must be zero since we are adding a number with its additive inverse. This also means that the entire right hand side is zero by part c of the lemma, since anything ($I_\times(a)$ in this case) multiplied by zero is zero. So that means

$$w + 1 = z_+$$

implying that w is the unique additive inverse to $1 = z_\times$ by RNFA8. Thus $w = I_+(z_\times)$ is the value we had in $w = I_+(a)/a$. Displaying this division in its alternate form with $w = I_+(z_\times)$ substituted, this is finally

$$I_+(a) = I_+(z_\times)a$$

At this stage, we have effectively demonstrated what the inverse operation is for addition and multiplication, specifically that $I_\times(a) = 1/a$ and $I_+(a) = -a$. We can now, with complete certainty, take these notations as proven to be consistent within our number system. As we prove more such statements as these, statements which we might normally take to be obvious, our list of rewrite rules will grow until they begin to form something like an intuition, or at least a feel for the game we have created for ourselves.

Our next stop in making this game easier for ourselves will be to fill out some of our intuitions about how ordering operations work. As you may have seen above, we take as axioms that orderings obey translations and dilations (i.e. they remain true under additions and multiplications) even though we did not take this as an axiom for equivalence (where we had to derive these rewrite rules as in lemma realnumsax.2). This is in part because we've never seen a formal ordering before, and we are in some sense defining simultaneously what an ordering *does*.

In fact, we have not said anything at all about what it *means* to write $a < b$, or any of these operations so far within a formal context. Of course we know the symbol $<$ to mean **less than** but the point in some sense is that its meaning here will appear not from what we name it but from what properties it has. Only having chosen those properties well, as is the case with these axioms, will we know for sure that this symbol and its properties reflects the intuition we wish it to match with.

Proposition realnumsax.4 — (Ordering Rules for Negatives)

Let $a, b, c, d \in \mathbb{R}$ and let $a < b$.

a. Then we may write $-b < -a$, reversing the order on the number's negatives.

b. If $c < 0$ then we have $bc < ac$ (by comparison to RNOA4 which required $0 < c$ and implied $ac < bc$).

c. If we have $c < d$ then we have $a + c < b + d$.

Proof.

Begin with the hypothesis $a < b$. By RNOA3 we may translate by the value $-a - b$.

$$a + (-a - b) < b + (-a - b)$$

From there, it will be necessary to apply associativity of addition (RNFA1) and commutativity of addition to rearrange terms (RNFA2),

$$\begin{aligned} a + (-a - b) &< b + (-a - b) \\ (a - a) - b &< b + (-b - a) \\ 0 + (-b) &< (b - b) - a \\ &< 0 + (-a) \end{aligned}$$

followed finally by RNFA3 to eliminate zeros.

$$-b < -a.$$

For part b we will need to use part a and proposition realnumsax.3 in the following manner. By part a, we have that $c < 0$ implies $0 < -c$ (that is, when c is a negative number), which means $-c$ satisfies the requirement for RNOA4 to rewrite $a < b$ to

$$a(-c) < b(-c).$$

Applying proposition realnumsax.3, we know that these negatives (or rather additive inverses) are simply numbers multiplied by -1, so applying commutativity of multiplication (RNFA6) we have

$$\begin{aligned} a(-1)c &< b(-1)c \\ (-1)ac &< (-1)bc \\ -ac &< -bc \end{aligned}$$

Applying part a of proposition realnumsax.3, $I_+(I_+(a)) = a$ or in this case $-(-a) = a$, together with part a of this proposition, we use $-ac < -bc$ to imply

$$\begin{aligned} -(-bc) &< -(-ac) \\ bc &< ac \end{aligned}$$

as desired.

Part c follows by applying translations and transitivity (RNOA3 and then RNOA2). That is, add c to both sides of $a < b$ to get $a + c < b + c$. Now add b to both sides of $c < d$ to obtain $b+c < b+d$. Now the right hand side of $a+c < b+c$ is the same as the left hand side of $b + c < b + d$, so by transitivity we have

$$a + c < b + c < b + d$$

implying

$$a + c < b + d$$

and we are done.

From this point on, we will begin taking some of these symbolic properties increasingly more for granted. For instance, we will slowly stop mentioning when we apply commutativity of addition or mulitplication (RNFA2 or RNFA6) or associativity (RNFA1 or RNFA5) when we reorder terms in sums or products. Nonetheless, the fact that we understand that we can do these reorderings is always strictly because they were written here. Because we *have the property* that these rewrites are valid.

realnumsax.2 The Archimedean Property

Now that we have seen some examples of demonstrating things that we already intuit to be true using strict axiomatic rewrites, we are in a position to demonstrate some things we might not know with equal certainty. In particular, we are ready to demonstrate that real numbers do not contain infinitessimals.

The concept of an infinitessimal in popular understanding is a confused one despite its ubiquity. For people in physics or certain less formal branches of applied mathematics such as engineering or financial mathematics, an infinitessimal is the dx in a derivative $\frac{df}{dx}$ or in an integral $\int f dx$. However most people without such a background encounter the concept as the idea of a number 0.9999 with infinite repeating 9s; it is common once such a quantity is proposed to then debate whether that number is equal to 1 or is its own number. Importantly, the number is not 0.999 or 0.999999 but an infinite number of unwritten 9s, and so the whole point of the number is that you are in some sense unable to pin it down as anything but 'closer but not equal to one', since more 9s must be written. In some sense the concept is a modern instantiation of Zeno's paradox, of the distance divided into infinite segments, first 0.9 then 0.09 then 0.009, which can never reach their target of 1.

Why infinitessimals might be real numbers at all is largely a confusion born of the tricks used in physics courses and earlier studies of calculus, since looking at derivatives written $\frac{df}{dx}$ and thinking that df and dx are themselves values can sometimes yield correct and valuable results. In fact there is a sense in which it is true that they are *values*, or at least, that there are formalisms within which we can speak of df and dx as being mathematical objects with distinct algebraic rules similar to that of a number. But for the sake of a discussion of real numbers, it is important to emphasize that these objects, whatever they may be, are *not* real numbers. Real numbers obey something called the *Archimedian property* which forbids the notion of infinitessimal numbers.

There are many ways to express the Archimedian property, each focusing on various aspects of it. Here, we will write the Archimidean property specifically as the property that there are no such thing as infinitessimal reals, and *because* there are no infinitessimal reals, any two real numbers are non-equal if and only if have some number between them. Expressing this statement poses a difficulty however; if there *were* infinitessimal real numbers, then they would of course lay on the number line just as any other number and we would have difficulty distinguishing them from other non-infinitessimal real numbers. So instead, in this context, we will think of the idea of an infinitessimal number as one that is smaller than all *positive rational numbers*, that is, all $q \in \mathbb{Q}$ which are $q > 0$, the set of positive rational numbers $\mathbb{Q}^+$.

Theorem realnumsax.5 — (Archimedian Property)

There are no elements $h \in \mathbb{R}$ which are $h > 0$ and also $h < q$ for all positive rational numbers $q \in \mathbb{Q}^+ \subset \mathbb{R}$.

Proof. (Adapted from the proof on Wikipedia)

We proceed with a proof by contradiction. Accordingly, assume for the sake of contradiction that there did exist a real number $h \in \mathbb{R}$ which were both $h > 0$ and also $h < q$ for all $q \in \mathbb{Q}^+$. Then there would exist a set of all infinitessimal numbers

$$Z = \left\{ h \in \mathbb{R} \,\middle|\, \forall q \in \mathbb{Q}^+, (h < q) \wedge (h > 0) \right\}$$

and since we are assuming such an infinitessimal real number exists, we are insisting that $Z \neq \emptyset$, that this set of all infinitessimals is non-empty. By the axiom of completeness for $\mathbb{R}$, since the set has any positive rational number as an upper bound and the set is non-empty by hypothesis, it must have a least upper bound $c = \sup(Z)$.
Consider now that c must dominate all would-be infinitessimals, that is, $z \leq c$ for all $z \in Z$, and that any number larger than c must not be in Z since c *bounds* Z. Then we can say with certainty that $2c$ is *not* an infinitessimal because $c < 2c$ (which we deduce by adding c to both sides of $0 < c$). Since $2c \notin Z$ and $2c > 0$, in order to have avoided being a member of Z there must have existed some $k \in \mathbb{Q}^+$ which is $k \leq 2c$.
On the other side, $c/2$ is small enough that it should be an infinitessimal, since c is the *least* (smallest) upper bound of Z, so whether or not c itself is in Z, anything smaller than c but greater than zero is less than all $q \in Q^+$ and thus must be in Z. Since c is a least upper bound of Z and Z is defined by its elements being greater than zero, we know that any x which satisfies $0 < x < c$ must be $x \in Z$. Moreover this means that any x which satisfies $0 < c/4 < x < c/2 < c$ is an infinitessimal.
Now, there must exist some $k \in \mathbb{Q}^+$ which is $c \leq k \leq 2c$ or else any other number between c and $2c$, say $\frac{3}{2}c$, would be an infinitessimal thus violating c's *least* upper bound property. But since such a rational number exists, we may simply divide the transitive inequality by four, obtaining $c/4 < k/4 < c/2$. As discussed earlier, any number satisfying that property must be infinitessimal, but at the same time, $k/4$ is a rational number divided by another rational number and thus a rational number itself. Since it is smaller than $c/2$, its existence implies that $c/2$ was not an infinitessimal, but that would also mean that $c/2$ bounds the infinitessimals, making it a lesser bound than c which was supposed to be the least upper bound.
This provides the contradiction which implies Z must be empty.

As a corollary, this property may be thought of as saying that numbers such as 0.999 repeating are equal to the number they infinitely approximate. That is, if infinitessimals do not exist in the reals, then there is also no notion of a number being infinitessimally larger or infinitessimally smaller than another; recall the property of trichotomoy for orderings (RNOA1) which says that two numbers a, b are either $a < b$, $a = b$ or $a > b$, so a number such as 0.999 repeating which we suppose is larger than all other numbers we can *specifically point to* less than one (0.998, 0.9999453, etc.) must only be equal to one. It seems obvious to say then that when the Archimedean property is true, as stated earlier, two real numbers are non-equal if and only if there exists a number between them. Superficially, we could already say that this was $a \neq b$ if $a \neq \frac{a+b}{2} \neq b$ with $(a+b)/2$

their average between them when they are not equal; but now we can be certain that no consequence of real numbers mean that there would be a quantity which is *hard to refer to* strange way.

We will later substantiate the meaning of infinitessimals in an informal sense, mostly as a tool for visualization wrapped up in significant algebraic (or, context dependently, analytic) abstractions, but these tools will never be "real numbers". No matter how much abstraction we build up, if we are ever asked to find *a number* which is somehow smaller than any other thing but greater than zero, we will say confidently that it does not exist.

realnumsax.3 The Triangle Inequality

We have one last serious property of the real numbers to prove, known as the *triangle inequality*. This inequality in some sense is our first taste of the flavor of real analysis as a discipline; indeed we will use this inequality heavily despite its seemingly inane symbolic statement.

The triangle inequality is born of the obvious fact that one cannot draw a triangle with any one side greater than the length of the other two. This property is obvious in the setting of euclidean geometry (no doubt, you can demonstrate right now with a piece of paper) but as we discuss more abstract spaces, expecting the triangle inequality to hold will constitute a very particular choice about exactly how abstract a notion of space we wish to describe. For instance, should a *space of functions* with a notion of distance satisfy the triangle inequality with that notion of distance? We will later say that the answer is generally yes. Should a space of possible outcomes, an *event space*, which we measure in probabilities, have a triangle inequality? Generally we say the answer is no in that case. But the kinds of things an abstract definition of *space* can describe are downstream of the properties we expect of that space, and expecting or not expecting the triangle inequality will define what kind of space it is.

In this case, we will be demonstrating the not-so-obvious fact that the triangle inequality, the inability to draw a three sided shape with one side greater than the sum of the other two, holds not just in two or more dimensions, but even on only one dimension. Yes, even on merely the number *line*, we have the triangle inequality, and we can prove it as a natural consequence of the real number axioms.

But to do this, we will first need the *absolute value* operation. You may know this operation as the the one that leaves positive values the same but makes negative values become positive; in fact that will be our definition today, but I must foreshadow that despite its simplicity, the full scope of properties the absolute value has on other spaces will, in time, prove fascinating on its own.

Definition realnumsax.6 — (Absolute Value)

Given $a \in \mathbb{R}$, denote by $|a|$ the **absolute value** of a, that is, the number of equal magnitude to a which is non-negative. Formally, if $0 \leq a$ then $|a| = a$, but if $a < 0$ then $|a| = -a$ in order to make it positive.

$$|x| = \begin{cases} -x, & x < 0 \\ x, & x \geq 0 \end{cases}$$

As a simple consequence, $|-a| = |a|$ and $|k||a| = |ka|$ for all $k \in \mathbb{R}$ and $|ka| = k|a|$ for all $k \geq 0$.

Theorem realnumsax.7 — (Triangle Inequality)

Any three $a, b, c \in \mathbb{R}$ satisfy the property

$$|a - c| \leq |a - b| + |b - c|.$$

called the **triangle inequality** in $\mathbb{R}$.

This symbolic statement can then be read as the property we mentioned earlier; we read each $|a-c|$, $|a-b|$ and $|b-c|$ as the distances between points, since in $\mathbb{R}$ distances between numbers are merely their positive differences. Then it is literally that the distance between any two points is less than or equal to the sum of the other two distances.

Proof.

This proof proceeds by brute force case analysis. There are six ways to order the three values, i.e. each case of ordering is something like $a \leq b \leq c$ or $b \leq c \leq a$ or something like that (this is applying RNOA1, since each pair of numbers must be either less than or equal to each other or greater than or equal to each other). Given that we are only concerned with the magnitude of their *differences* however, the proof where we assume $a \leq b \leq c$ is also actually a proof for the case where $c \leq b \leq a$, since we could change a to $-a$, b to $-b$ and c to $-c$, and each term, each difference $|a - c|$ would become

$$\begin{aligned} |a - c| \mapsto |(-a) - (-c)| &= |-a + c| \\ &= |c - a| \\ &= |a - c| \end{aligned}$$

and so our six cases become only three that we care about where the triangle inequality is concerned. These are cases are labelled (1), (2) and (3) respectively.

$$\begin{aligned} &(1) \quad a \leq b \leq c \\ &(2) \quad b \leq a \leq c \\ &(3) \quad a \leq c \leq b \end{aligned}$$

Let's start with case (1). In this case, $a - b$, $b - c$ and $a - c$ are all negative, so we may apply the appropriate sign changes to compute the absolute values in the triangle inequality we wish to prove as follows.

$$c - a \leq (b - a) + (c - b)$$

We can obtain this easily by starting with reflexivity of equality $c - a = c - a$ stated as $c - a \leq c - a$ and then deduce this statement, but in this case it will be more obvious if we complete the proof by *rewriting on the goal*. That is, if we can show that the thing we aim to prove can be rewritten as a fact which we know to be trivially true, then the proof is completed. We have already begun this process by deducing that when $a \leq b \leq c$, the triangle inequality is written as above. We continue this process by merely simplifying the statement.

$$\begin{aligned} c - a &\leq (b - a) + (c - b) \\ &\leq b - b + c - a \\ &\leq 0 + c - a \\ &\leq c - a \end{aligned}$$

This of course obtains the same statement we noted earlier was true by reflexivity of equality. We could of course do these steps in reverse, proceeding from $c - a \leq c - a$, adding $b - b$ to the right hand side using proposition realnumsax.4.c (with $\leq$ instead of $<$), but the fact that we could do these steps in reverse is exactly the reason we do not have to.
Concluding case (1), we study case (2) by applying a similar method of computing the absolute values when $b \leq a \leq c$.

$$\begin{aligned} c - a &\leq (a - b) + (c - b) \\ &\leq a + b - 2b \end{aligned}$$

Now to both sides, add $b - c$ (i.e. apply RNOA3) to obtain

$$\begin{aligned} c - a(b - c) &\leq a + c - 2b + (b - c) \\ b - a &\leq a - b \end{aligned}$$

This is then the statement we need to prove. Since $b \leq a$ as in $b \leq a \leq c$, we have $b - a \leq 0$ by subtracting a from both sides of $b \leq a$, and at the same time we could subtract b from both sides of $b \leq a$ and obtain $0 \leq a - b$. By transitivity (RNOA2) we have

$$\begin{aligned} b - a \leq 0 \leq a - b \\ b - a \leq a - b \end{aligned}$$

which is what we wanted to show. Thus case (2) is covered.
Case (3) will proceed similarly as in case (2). Our goal of the triangle inequality, when $a \leq c \leq b$, is read

$$\begin{aligned} c - a &\leq (b - a) + (b - c) \\ c - a &\leq 2b - a - c. \end{aligned}$$

Adding $a - c$ to both sides we obtain

$$\begin{aligned} c - a + (a - c) &\leq 2b - a - c + (a - c) \\ 0 &\leq 2b - 2c. \end{aligned}$$

Since $c \leq b$ as in $a \leq c \leq b$ of case (3), we know by RNOA4 that $2c \leq 2b$, and subtracting $2c$ from both sides, that $0 \leq 2b - 2c$. But this is what we wanted to show, the case (3) form of the triangle inequality.
Then we have shown all three relevant cases for the triangle inequality on $\mathbb{R}$ and know it to be true in general.

realnumsax.4 Section Appendix: Some Loose Ends

The above proofs hopefully serve to show that one can prove statements that might seem so simple as to be tautological so long as we have a strong axiomatic basis. However, as you can see, this can lead to quite verbose proofs. From here on, going into the rest of the chapter and chapters beyond it, we will be a little more terse unless we are discussing a brand new concept, with the understanding that we know a bit about how to work with real numbers.

Accordingly, we will largely mostly forget the axiomatic description of the reals except when it is necessary to disentangle a specific property related to numbers themselves. We will however need to bring some things with us foward, such as the triangle inequality, and we must pay attention in future for when a mathematical construction has many or most of the axioms of the real numbers, since they may share properties. It will also serve us well to state some earlier mentioned concepts in a self contained form to establish notational rules and strict definitions.

Here we will begin to establish a formal pattern that most sections will come with a section appendix, a collection of statements, proofs, or intuitive descriptions which are either tangential to the direction of our exposition or do not fit neatly within it. In general, it is recommended that treat section appendices as either references or as almost standalone discussions. In fact it is somewhat optional to read a section appendix at the moment you encounter it, but almost all section appendices will become relevant later, and are written so that you *can* understand them when you encounter them, even if their true importance will only be revealed later. Nonetheless, I hope that you enter the following section trying to follow where the thread left off just before the appendix began.

Definition realnumsax.8 — (Upper and Lower Bounds)

Let $A \subset \mathbb{R}$ be a set of real numbers. We say that b is a **upper bound** of A (respectively a **lower bound**) if for all $a \in A$, b satisfies $a < b$ (respectively $b < a$ for lower bounds).
If A has both an upper bound and a lower bound, we say that A is **bounded**.

Definition realnumsax.9 — (Supremum and Infimum of a Set)

Let $A \subset \mathbb{R}$ be a set of real numbers. Denote by $\sup(A)$ the **supremum** of A, meaning a value $\sup(A) \in \mathbb{R}$ which is an upper bound for A, but is also $\sup(A) \leq b$ for all $b \in \mathbb{R}$ which are upper bounds for A. We also have the **infimum** denoted $\inf(A)$ which is the *greatest lower bound* satisfying $\inf(A) > b$ for all b which are lower bounds of A, or equivalently $\inf(A) = -\sup(\{-x | x \in A\})$.
The supremum and infimum can be thought of as analogous to the **maximum** and **minimum** of a set, but where it may be impossible to specify a largest or smallest value which is itself included in the set (say because a set does not include its endpoints). When it is possible, the supremum is equal to the maximum (and the infimum to the minimum), but when it is not the supremum will select the excluded endpoint, thus sometimes $\sup(A) \notin A$.

Finally, we include one last property of functions with respect to orderings which will be valuable in the next section. That is, we showed under what circumstances $x = y$ can be rewritten to $f(x) = f(y)$ or back (namely when f is injective), but we did not show the corresponding property for orders. That is in part because we must speak about a new property of functions when they go from ordered sets to ordered sets. We must discuss *monotonicity*.

Definition realnumsax.10 — (Monotonic Functions)

Let $f : \mathbb{R} \to \mathbb{R}$. If f has the property that for all $x, y \in \mathbb{R}$ $x < y$ implies $f(x) < f(y)$ then we say that f is a **strictly monotonic increasing** function, that is, it is a function

for which a larger input y will always yield a larger output $f(y)$ than the output for the previous input $f(x)$. There is also the corresponding property of **strictly monotonic decreasing** function for which $x < y$ implies $f(y) < f(x)$.
When instead we have that $x < y$ implies $f(x) \leq f(y)$, we say that f is a **monotonic increasing** function, or a **monotonic non-decreasing** function, signifying that larger inputs may not always lead to larger outputs, but that the outputs are at least as big as all inputs before and no smaller. Correspondingly, a function for which $x < y$ implies $f(y) \leq f(x)$ is a **monotonic decreasing** or **monotonic non-increasing** function.

In the context of this, we can see that addition by a constant $x \mapsto x + a$ is a strictly monotonic increasing function, since $x < y$ has $x + a < x + y$ by RNOA3 and similarly $x \mapsto xa$ is monotonic increasing when $0 < a$ by RNOA4.

Corollary realnumsax.11 — (Composition of Monotonic Functions)

Let $f, g \colon \mathbb{R} \to \mathbb{R}$ be functions.

- if f and g are monotonic increasing, then $g \circ f$ is also monotonic increasing
- if f and g are monotonic decreasing, then $g \circ f$ is monotonic *increasing* since the ordering is reversed twice.

Corollary realnumsax.12 — (Some Basic Monotonic functions)

I encourage you to go to desmos.com/calculator or anywhere else you like to graph functions and graph these functions yourself to see that they are monotonic in the way described. When we speak of conditional monotonicity, we mean that $x < y$ implies $f(x) < f(y)$ or $f(x) > f(y)$ respectively when both x and y satisfy the condition.

- $x \mapsto x$ is strictly montonic increasing
- $x \mapsto -x$ is strictly montonic decreasing
- $x \mapsto x^3$ is strictly monotonic increasing
- $x \mapsto e^x$ is strictly monotonic increasing
- $x \mapsto \sqrt{x}$ is strictly monotonic increasing when valid, i.e. on $x \geq 0$
- $x \mapsto \log x$ is strictly monotonic increasing when valid, i.e. on $x > 0$
- $x \mapsto x^2$ is strictly monotonic increasing when $x > 0$ and strictly monotonic decreasing when $x < 0$
- $x \mapsto 1/x$ is strictly monotonic decreasing on $x < 0$ and on $x > 0$ separately but not together due to the discontinuity of one divided by zero

Finally, some facts should be elaborated about absolute values before we move forward. This is by no means the rich properties they hold which we must discuss much later, but it will save us a great deal of time explaining why these properties are true on a case by case basis later.

Lemma realnumsax.13 — (Absolute Value Properties)

It will prove valuable to understand some basic properties of absolute values. Let $a, k \in \mathbb{R}$. Then the following are true:

a. $|-a| = |a|$

b. $|ka| = |k||a|$

c. if $k \geq 0$ then $|ka| = k|a|$

d. $x \mapsto |x|$ is monotonic increasing when $x \geq 0$ and monotonic decreasing on $x \leq 0$.

e. $|a| \leq k$ is an equivalent statement to $-k \leq a \leq k$

Proof.

a. Consider the two possibilities: if a is positive, then $|a| = a$, and since $-a$ is negative, we get $|-a| = -(-a) = a$. Consequently it is clear that $|-a| = a = |a|$. A similar argument follows if a is negative, since $|a| = -a$, and $-a$ is positive, making $|-a| = -a$. Once again we get $|-a| = -a = |a|$. Although in the first case they are both a and in the second case they are both $-a$, ultimately $|-a| = |a|$ remains true either way.

b. This is a more general case of the latter. There are four relevant cases, $k \geq 0$ and $a \geq 0$, $k \leq 0$ and $a \geq 0$, $k \geq 0$ and $a \leq 0$, and finally $k \leq 0$ and $a \leq 0$. In each of these cases, the first thing to notice is whether they are both positive or both negative since this will make their product positive, or rather if one is positive and one is negative in which case their product is negative. Thus we have

$$\begin{aligned}(k \geq 0) \wedge (a \geq 0) &\implies |ka| = ka \\ (k \leq 0) \wedge (a \geq 0) &\implies |ka| = -ka \\ (k \geq 0) \wedge (a \leq 0) &\implies |ka| = -ka \\ (k \leq 0) \wedge (a \leq 0) &\implies |ka| = ka\end{aligned}$$

and on the individual factors we have

$$\begin{aligned}(k \geq 0) \wedge (a \geq 0) &\implies (|k| = k) \wedge (|a| = a) \\ (k \leq 0) \wedge (a \geq 0) &\implies (|k| = -k) \wedge (|a| = a) \\ (k \geq 0) \wedge (a \leq 0) &\implies (|k| = k) \wedge (|a| = -a) \\ (k \leq 0) \wedge (a \leq 0) &\implies (|k| = -k) \wedge (|a| = -a)\end{aligned}$$

but we see that $(-k)(-a) = ka$ and that $(-k)a = k(-a) = -ka$. So we may write

$$\begin{aligned}(k \geq 0) \wedge (a \geq 0) &\implies |ka| = ka = |k||a| \\ (k \leq 0) \wedge (a \geq 0) &\implies |ka| = -ka = |k||a| \\ (k \geq 0) \wedge (a \leq 0) &\implies |ka| = -ka = |k||a| \\ (k \leq 0) \wedge (a \leq 0) &\implies |ka| = ka = |k||a|\end{aligned}$$

and thus in every case we obtain $|ka| = |k||a|$.

c. This is clearly just one of the cases of the previous proof and is included for emphasis only.

d. Let $x, y \in \mathbb{R}$. If $0 < x < y$ then x and y are positive numbers, meaning $|x| = x$ and $|y| = y$, so it is clear that $x < y$ implies $|x| < |y|$ and $x \mapsto |x|$ is behaving as strictly monotonic increasing. If $x < y < 0$, then x and y are negative numbers, and so $|x| = -x$ and $|y| = -y$. We can take our existing statement in this case $x < y$ and multiply both sides by -1 (this applies proposition realnumsax.4.b with $c = -1$) to obtain $-x > -y$ which is then rewritten $|x| > |y|$, flipping the sign as a strictly monotonic decreasing function does.

e. Consider that $|a|$ is positive regardless of whether a is positive, and since $k \geq |a|$, it must be that k is positive. This also means that if a is positive, $k \geq |a| = a$ but if a is negative, as a negative number smaller than zero, $k \geq a$ anyway. So we have that $k \geq |a|$ implies $k \geq a$.

   On the other side, if $|a| = -a$ then $-a$ is a positive number, and we may write $k \geq |a| = -a$. Multiplying both sides of this by -1, we obtain $-k \leq a$ when a is negative. When a is positive, $-k \leq a$ remains true anyway, since k is less than zero and a is more than zero.

   Together, we have $-k \leq a$ and $a \leq k$ regardless of the positivity/negativity of a, completing the proof in one direction.

   In the other direction, assume it is known that $-k \leq a \leq k$. Once again this implies that k is positive, since by transitivity we have $-k \leq k$. If a is positive then by $a \leq k$, we know that $|a| \leq k$. If a is negative then by $-k \leq a$ we can multiply the inequality by -1 to obtain $k \geq -a$ and since $|a| = -a$ in this case, it is $k \geq |a|$ as well. Once again, in either case we obtain $k \geq |a|$.

seqlimsinR Sequences and Limits in $\mathbb{R}$

Here we can begin to substantiate at least one notion of what happens *at infinity*, in so far as you could ever say that, which, for our formal purposes, I will continue to say we cannot. In fact, one could execute a proof of the Archimedian property from the previous section in the opposite manner, showing not that there exist no infinitessimals in the reals but no infinity objects in $\mathbb{R}$. So infinity believers are in a rough spot right now. We certainly know what we mean informally when we want to discuss infinity, but we cannot actually do so without producing contradictions via the Archimedean property. It is perhaps one of the motivations of real analysis that we are able to construct ways to launder infinity into our work.

Before we are ready to use infinity to speak of things infinitessimally close to a point however, it will be easier to develop our conceptions and mathematical tools if we focus on the simplest kind of infinity first, not an infinity we imagine in $\mathbb{R}$ (or indeed an infinitessimal), but an infinity in $\mathbb{N}$. This is no mere building block either; we will continue to use limits of countable sequences for probably the entirety of this text since they provide an alternate, often simpler, way to prove things or make statements.

Our notion of 'what is at infinity' will be defined not as a literal infinity but as what happens 'on the way there'; consequently it will be easier to speak about 'on the way there' if the process of 'going

there' is a countable one, i.e. we will work with sequences instead of paths.

seqlimsinR.1 Formal Definitions

Definition seqlimsinR.1 — (Sequence on $\mathbb{R}$)

A sequence is an ordered set of real numbers indexed by $\mathbb{N}$, or equivalently, it is a function $a : \mathbb{N} \to \mathbb{R}$ which we write $(a_n)_{n\in\mathbb{N}}$ and denote elements of the ordered set (or equivalently outputs of the function) as a_n for the corresponding counting number n.
Sometimes (and in some math texts) we use a_n with n completely unspecified as a shorthand to refer to the whole sequence $(a_n)_{n\in\mathbb{N}}$.

Definition seqlimsinR.2 — (Limits on Sequences)

We say that a sequence $(a_n)_{n\in\mathbb{N}}$ **converges** to $a \in \mathbb{R}$ or 'has **limit**' $a \in \mathbb{R}$ if the following is true:
for all $\varepsilon > 0$, there exists $N \in \mathbb{N}$ such that for all $n \geq N$, we have $|a_n - a| \leq \varepsilon$.
This condition is often formally written:

$$\forall \varepsilon > 0, \exists N \in \mathbb{N}, (n \geq N) \Rightarrow \big(|a_n - a| \leq \varepsilon\big).$$

When the sequence converges to a, we write

$$\lim_{n\to\infty} a_n = a$$

or say that $a_n \to a$ as $n \to \infty$. (The symbol ε is the greek letter 'epsilon'.)

Personally I found the statement for the requirement of a limit quite difficult to parse the first time I saw it, not having been practiced at proposition heavy mathematics. And especially I was frustrated that I did not see why this seemingly arbitrary requirement had anything to do with infinity. I hope that the following description will mean you share no such grievance.

First, the limit requirement. The way this is to be read is thus. Imagine I gave you a sheet of metal and I told you, "hey, I found this sheet of metal, and it's *perfectly smooth*". You're suspicious, so you run your finger over it and feel that it is smooth, meaning that it appears perfectly smooth at least to the detection standard of your fingers, which are not very precise instruments. So you're still suspicious, and you get a magnifying glass and look it over for imperfections, and you still find none. Next, you get a microscope, and under the microscope, the sheet of metal still appears to be perfectly smooth. It is not that you need to view the atomic structure of the sheet to know that it is perfectly smooth, but in particular that you know it is perfectly smooth once you can *trust* that "no matter how closely I look, my measuring device will always tell me the sheet is smooth". This is our definition of 'perfect' here; that the property remains true no matter how much more precision you use to check if the property holds.

The definition of the limit is exactly the same as above, but instead of perfect smoothness we are saying that the sequence *becomes perfectly close* to a.

I tell you that $(a_n)_{n\in\mathbb{N}}$ converges to a. You're suspicious, so you pick some measuring threshhold

$\varepsilon \in \mathbb{R}$ which is $\varepsilon > 0$, and you ask me if I can find some point in the sequence where the entire sequence after that point is less than ε away from a. That is, you ask me to find some $N \in \mathbb{N}$ such that for all $n \geq N$, we satisfy $|a_n - a| \leq \varepsilon$; the condition $n \geq N$ ensures that a_n and all a_n after it is past the point of sufficient closeness. You're still suspicious, so you pick a smaller ε, and once again I am able to find a larger $N \in \mathbb{N}$ for which all a_n with n after N are less than ε away from a. You pick an even smaller ε, but this time, instead of giving you a *particular* $N \in \mathbb{N}$, I give you a method by which you can find your own $N \in \mathbb{N}$ for any ε. By providing this method, you can trust that no matter how small a ε, there will exist a satisfactory N, and you can *trust* that the sequence converges. This is what it means to prove the limit.

Example seqlimsinR.3 — ($1/n^2$ goes to 0 as n goes to ∞)

Let us show that the sequence $\left(\frac{1}{n^2}\right)_{n\in\mathbb{N}}$ converges to 0. Observe first that $\left|\frac{1}{n^2} - 0\right| = \frac{1}{n^2}$ since $\frac{1}{n^2}$ is always positive. Moreover, the function (and the sequence it defines) $n \mapsto \frac{1}{n^2}$ is *monotonically decreasing* as defined in the previous section, meaning that for all n, $\frac{1}{n^2} > \frac{1}{(n+1)^2}$. (If the reason why in this case is not obvious to you, set $n = 2$ and remember that a quarter of a pizza is larger than a ninth of one. See corollaries at the end of the previous chapter). This means that if we pick $\varepsilon = 1/N^2$, this would satisfy the limit $|1/n^2 - 0| \leq 1/N^2$ for all $n \geq N$ since when n is greater than N, by the monotonically decreasing property, $1/n^2$ must be smaller than $1/N^2$.
But the whole point of the reasoning of the limit is that we, as the ones proving the limit, cannot pick ε. We must imagine someone else picking ε and ourselves making a procedure to find N for which $n \geq N$ implies $|1/n^2 - 0| < \varepsilon$. Still, noticing that the limit is satisfied when $\varepsilon = 1/N^2$ helps us, since we can invert this relation. Naively we can take

$$\begin{aligned} \varepsilon &= \frac{1}{N^2} \\ \frac{1}{\varepsilon} &= N^2 \\ \frac{1}{\sqrt{\varepsilon}} &= N \end{aligned}$$

And if this alone worked, then in fact we would be done. Setting N like this would indeed mean that when $n \geq N$ then we have $|1/n^2 - 0| < \varepsilon$. The only trouble is that $N \in \mathbb{N}$ needs to be a counting number, and we have no guarentee that the ε picked will produce a counting number when we take its inverse squareroot. This is easily solved however by rounding up $1/\sqrt{\varepsilon}$ to the largest whole number, an operation called taking the *ceiling* and written

$$N = \left\lceil \frac{1}{\sqrt{\varepsilon}} \right\rceil \geq \frac{1}{\sqrt{\varepsilon}}.$$

with the property that it is greater than or equal to its counterpart that is not rounded up. Now when we are given some ε we are able to automatically produce some N for which $n \geq N$ will satisfy $|1/n^2 - 0| \leq \varepsilon$.
Let's test this to be certain. Say we are given ε and thus set $N = \lceil 1/\sqrt{\varepsilon} \rceil$. Since N is rounded up, first and foremost we have $N \geq 1/\sqrt{\varepsilon}$ larger than its unrounded version.
Now consider that since $x \mapsto 1/x$ is a monotonic decreasing function on positive numbers (which ε is required to be), we deduce from $N \geq 1/\sqrt{\varepsilon}$ that

$$\frac{1}{N} \leq \sqrt{\varepsilon}$$

flipping the ordering by the monotonic decreasing property in realnumsax.10. We also have that $x \mapsto x^2$ is a monotonic increasing function on positive numbers, so undoing the square root will preserve the ordering.

$$\frac{1}{N^2} \leq \varepsilon.$$

Recall that the property we wanted out of N was that $|1/n^2 - 0| \leq \varepsilon$ was only true when $n \geq N$, so let us operate on that next. Once again, $x \mapsto 1/x$ will flip the sign, and $x \mapsto x^2$ will preserve it.

$$\begin{aligned} n &\geq N \\ \frac{1}{n} &\leq \frac{1}{N} \\ \frac{1}{n^2} &\leq \frac{1}{N^2} \end{aligned}$$

We also had that $|1/n^2 - 0| = 1/n^2$. So together we have a transitivity between equality and addition,

$$\left|\frac{1}{n^2} - 0\right| = \frac{1}{n^2} \leq \frac{1}{N^2} \leq \epsilon$$

implying

$$\left|\frac{1}{n^2} - 0\right| \leq \epsilon$$

as we had wanted. Notice that we proved this, as we needed to, *for any* $\varepsilon > 0$ that *there exists some* $N \in \mathbb{N}$ which for us was $N = \lceil 1/\sqrt{\varepsilon} \rceil$, and if we assumed $n \geq N$ then we could deduce $|1/n^2 - 0| \leq \varepsilon$. This is the pattern of a proof of limit.

In general, the pattern of proving a limit is roughly as above; in particular, we first identify how small a ε a given N can satisfy, which we did when we noticed that $\varepsilon = 1/N^2$ will mean we are trying to prove $n \geq N$ implies $1/n^2 \leq 1/N^2$, which we know to be true immediately from the monotonic decreasing property of $x \mapsto 1/x^2$. Next, we treat the earlier relationship $\varepsilon = 1/N^2$ as a function $\varepsilon \mapsto 1/N^2$ and we see if we can invert it, finding either some way to calculate $1/N^2 \mapsto \varepsilon$ or the next best thing; this ended up being our $N \mapsto 1/\sqrt{\varepsilon}$ as our inverse function, and we had to round it up to make it a whole number. Most proofs will omit these steps, since the crux of the proof is not that we can find some relationship but rather that a relationship exists and that it *works*. So having found our relationship $N = \lceil 1/\sqrt{\varepsilon} \rceil$ we try to in some sense unwrap whatever transformations we've made to the function while taking n along for the journey; that is, we map $n \geq N$ to $1/n \leq 1/N$ together, then square them together. At this point, some minor amount of symbolic massaging will generally get you your limit.

Example seqlimsinR.4 — (2^{-n} goes to 0 as n goes to ∞)

Let us repeat the steps we did prior but more streamlined this time. In this case we notice we are studying a strictly monotonic decreasing function $n \mapsto 2^{-n}$ which is always $2^{-n} > 0$, so immediately a condition $|2^{-n} - 0| \leq \varepsilon$ can be read instead as $2^{-n} \leq \varepsilon$. This may inspire us to take an inverse, and indeed we can notice that if we could choose $\varepsilon = 2^{-N}$ then $n \geq N$ would imply $2^{-n} \leq \varepsilon$ as we desire, so we want to propose something like $N = -\log_2 \varepsilon$. The negative sign here is not to make N negative (indeed we do not want this) but instead to make it positive, since a $\varepsilon < 1$ will have $\log_2 \varepsilon \leq 0$, a property of logarithms. The problem we actually have is the same as in the previous example, that we have no guarentee that $-\log_2 \varepsilon$ is a whole number, so we will want to round it up as $N = \lceil -\log_2 \varepsilon \rceil$.
With this N proposed, we can now prove the limit. First observe that

$$N = \lceil -\log_2 \varepsilon \rceil \geq -\log_2 \varepsilon$$

so we may apply the monotonic decreasing function to the inequality, $x \mapsto -x$, obtaining

$$-N \leq \log_2 \varepsilon$$

with the appropriate sign flip, followed by $x \mapsto 2^x$ to undo the logarithm (which is a monotonic increasing function),

$$2^{-N} \leq \varepsilon.$$

If we repeat these operations on $n \geq N$ we obtain

$$-n \leq -N$$
$$2^{-n} \leq 2^{-N}$$

providing the transitive chain

$$|2^{-n} - 0| = 2^{-n} \leq 2^{-N} \leq \varepsilon$$
$$|2^{-n} - 0| \leq \varepsilon$$

completing the proof of the limit.

We could do more examples (and indeed I may have to return here and add more) but the majority of the limits we will prove over this chapter are limits related to other limits. So instead of the particular technique described above, what we will frequently be doing is saying that some sequence $(a_n)_{n\in\mathbb{N}}$ converges to a, and that since it does, we can take the existence of the N which it implies and use it to imply that some other sequence $(b_n)_{n\in\mathbb{N}}$ also converges to some b.

At the beginning of chapter 1 we discussed the idea of the development of a theory; this involved defining core concepts, sub-concepts, relationships between those concepts, and finally substantiation of the implication of the theory onto other areas of study. We have now defined our core concepts for our theory of limits, and so in line with stage two, we must define sub-concepts and properties.

Definition seqlimsinR.5 — (Subsequences on $\mathbb{R}$)

Let $(a_n)_{n\in\mathbb{N}}$ and $(b_n)_{n\in\mathbb{N}}$ be sequences, and let there exist a monotonic increasing map $\varphi\colon \mathbb{N}\to\mathbb{N}$ (so for all $n > m$ we have $\varphi(n) > \varphi(m)$). If $b_n = a_{\varphi(n)}$ for all $n \in \mathbb{N}$ then we call $(b_n)_{n\in\mathbb{N}}$ a **subsequence** of $(a_n)_{n\in\mathbb{N}}$. That is, a subsequence is a sequence that skips elements, or that is missing elements.

A subsequence can do a few things for us. We can define a subsequence with $\varphi\colon n \mapsto n + m$ where $m \in \mathbb{N}$ is some number that skips the sequence forwards by a constant. We could have $\varphi\colon n \mapsto 2n$ which would skip every second entry; this is useful in cases like the sequence $\left((-1)^n\right)_{n\in\mathbb{N}}$ which alternates between 1 and -1 and does not converge, unless of course we were to skip all the -1 entries as the subsequence $\left((-1)^{\varphi(n)}\right)_{n\in\mathbb{N}}$ allows us to do, making it a constant sequence.

We are now at the point where we can make a firm statement about what is and is not possible given the definitions we have constructed. If a sequence converges, no subsequence of that sequence can converge to a different value. This should make sense to us, since if we think of a converging sequence as one that becomes as close as desired to a point it converges at, then of course skipping some of the points in the sequence should simply make the subsequence converge faster.

Proposition seqlimsinR.6 — (Subsequences Preserve Limits)

Let $(a_n)_{n\in\mathbb{N}}$ be a sequence which converges at $a \in \mathbb{R}$. Then all subsequences $(b_n)_{n\in\mathbb{N}}$ converge to a as well.

Proof.

Since we know $(a_n)_{n\in\mathbb{N}}$ converges to a, we have that any $\varepsilon_a > 0$ we choose, there exists N_a such that for all $n \geq N_a$, $|a_n - a| \leq \varepsilon_a$. From this, we want to prove that for all ε_b, there exists N_b such that for all $n \geq N_b$ we have $|b_n - a| \leq \varepsilon_b$. We also know that $(b_n)_{n\in\mathbb{N}}$ is a subsequence, and thus that there exists a monotonic increasing map $\varphi\colon \mathbb{N}\to\mathbb{N}$ for which $b_n = a_{\varphi(n)}$, and so we can rewrite the goal of our proof instead as $\varphi(n) \geq N_b$ implies $|a_{\varphi(n)} - a| \leq \varepsilon_b$.
Then when we are given some $\varepsilon_b > 0$, set our ε_a from earlier as $\varepsilon_a = \varepsilon_b$ so that it tells us there exists a N_a for which $n \geq N_a$ implies $|a_n - a| \leq \varepsilon_a = \varepsilon_b$. Because φ is an increasing map, it has $n \leq \varphi(n)$, which is the same as either $n = \varphi(n)$ or $n < \varphi(n)$. If it is the former, then the statement $\varphi(n) \geq N_b$ implies $|a_{\varphi(n)} - a| \leq \varepsilon_b$ is literally $n \geq N_b$ implies $|a_n - a| \leq \varepsilon_b = \varepsilon_a$, and so we can set $N_b = N_a$ and *inherit* the limit directly from $(a_n)_{n\in\mathbb{N}}$.
In the other case where $\varphi(n) > n$, notice that if we knew $n \geq N_a$ then we would have by transitivity that

$$\varphi(n) > n \geq N_a$$

meaning that we have $\varphi(n) \geq N_a$. But since we know $(a_m)_{m\in\mathbb{N}}$ converges, we have that $m \geq N_a$ implies $|a_m - a| \leq \varepsilon_a$, and this holds for all m (which we write instead of n to avoid confusion about which implications we are using) including if we write $\varphi(n)$ in place of m. So, since we know $\varphi(n) \geq N_a$, we know $|a_{\varphi(n)} - a| \leq \varepsilon_a$. Earlier we said though that $b_n = a_{\varphi(n)}$, and we also set $\varepsilon_a = \varepsilon_b$. This means that we have shown $n \geq N_a$, together with other facts we know to be true, implies

$$|b_n - a| = |a_{\varphi(n)} - a| \leq \varepsilon_a \leq \varepsilon_b$$
$$|b_n - a| \leq \varepsilon_b$$

So we need a way to know that $n \geq N_a$, and remember that our main goal is to show $n \geq N_b$ implies $|b_n - a| \leq \varepsilon_b$, and we have not yet decided what N_b should be. We can solve this conundrum by setting $N_b = N_a$, the same N_a we were told exists since $(a_n)_{n\in\mathbb{N}}$ converges when we set $\varepsilon_a = \varepsilon_b$. This means that assuming $n \geq N_b$ implies $n \geq N_a$, which we showed earlier implies that $|b_n - a| \leq \varepsilon_b$. So we are done.

seqlimsinR.2 Sketching an Intuition On Sequences

Here we take pause to do something we will need to do many times moving forward. While it is certainly true that all one needs to solve many problems in pure mathematics is the symbolic awareness to throw rocks together and see which ones stick, our time will be much better spent of we develop an internal philosophy of our objects of study. The internal philosophy on its own is not of formal value, since formally it says nothing, but if our philosophy is a good one then it will suggest to us things which should be true which we can aim at proving.

We can forecast some intuitions if we think about the above statement; recall earlier that we are studying sequences in place of paths, but for the sake of intuition, imagine something like a path, a route through $\mathbb{R}$ punctuated by pitstops a_n or a frog on a journey through $\mathbb{R}$ and tracing a path by each lilypad it stops at. A converging sequence can be thought of as the dot moving around in $\mathbb{R}$ until it eventually settles into a region around the point it will eventually converge to; indeed it must settle in such a fashion, or else it would be impossible to say that eventually the rest of the sequence past some N will be less than ε distance away. Then of course, we reason that skipping the start of this journey or speeding this process along should only make it settle faster. In this way, the above proposition is obvious.

However we can draw more insights from this picture, each which will translate into theorems.

Now imagine the path it takes; that region it settles into must be bounded, because just as eventually (for some $N \in \mathbb{N}$) it will be less than ε away from its destination, its limit, it is bounded on either side of the limit by a distance of ε. Prior to this (i.e. for $n < N$), it is not bounded, except that on any *particular* sequence it takes *particular* values a_n for each $n < N$, and let us not forget that this moving dot moves not continuously but from each a_n to the next a_{n+1}. As such, although it will have an infinite amount of time to move *closer* to its limit, it has only a finite amount of time (since $0 < n < N$) to get within ε of its limit, and only a finite number of stops $a_1, a_2, ...$ before it does so. And as a finite number of particular points, we may simply draw a circle around them and bound them as well. So we conclude that any sequence that converges will have the entire path it takes bounded as well.

Conversely, if a sequence is bounded but not necessarily converging, then it has an infinite number of points a_n to stop at and only a finite amount of space to place them. Those points have to go somewhere, and by skipping the points that do not move closer to some infinite collection of points, we can find a subsequence which converges even if the main sequence itself does not.

By similar reasoning, we can also say that a sequence which has an upper bound but is monotonic increasing must keep going up yet only has a finite amount of up it is allowed to go; it may not necessarily stop somewhere, but it must either stop or make its movements upwards slow down until the manner of its movements look like a convergence, and so the upper-bounded monotonic increasing sequence must too converge.

There is one more extremely valuable insight we can glean from this kind of pictoral thinking. There

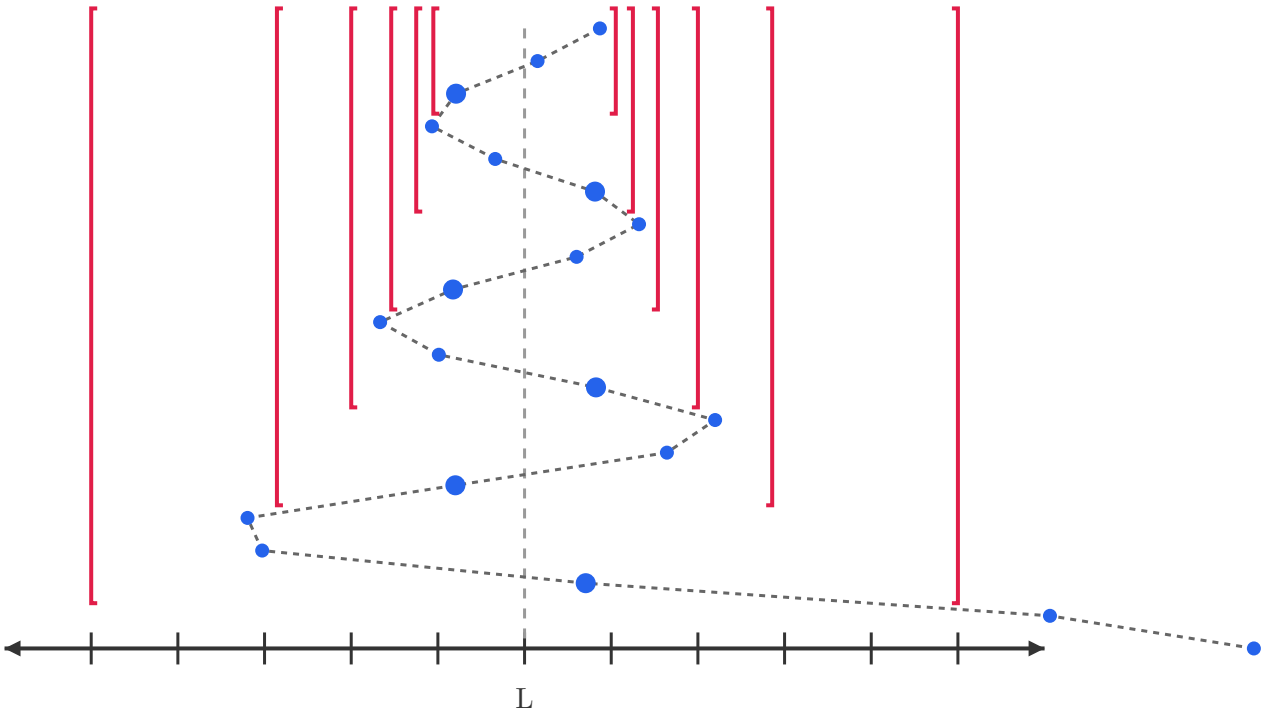

Figure 2.1: A visualisation of a converging sequence beside the real number line, with the sequence points represented by dots converging on the limit L. We represent the region of convergence corresponding to a shrinking ε as the sequence continues with red braces. Eventually we are able to say that 'the rest of the sequence' is contained in smaller and smaller regions.

is a sort of parallel criterion to the limit, another thing we could prove which is ultimately the same as proving convergence which may be easier to prove sometimes. That is, the same way we picture a sequence jumping from a_n to a_{n+1} until it is bounded in a region around its convergence, it must also be bounded around its own path. If a sequence were to converge, one should hope that the steps it takes should get smaller and smaller over time, and so we can imagine a converging sequence as not just being ε away from its limit, but perhaps ε away from its next point and all points after that.

This notion of a 'region of convergence' is not merely imaginary, but to substantiate it, we will need to learn more strange things about the real numbers, which we will do in the next section. Once we do, we will find that it is a concept of utmost importance.

Now let us translate these intuitive deductions into formal ones.

Corollary seqlimsinR.7 — (Subsequence Limit Theorem)

Let (a_n) be a sequence and $(b_n)_{n\in\mathbb{N}}$ be a subsequence which converges to $b \in \mathbb{R}$. Then if $(a_n)_{n\in\mathbb{N}}$ to $a \in \mathbb{R}$, we have $a = b$.

This follows directly from the previous proposition, and is in some sense a direct restatement of subsequences preserving limits. In fact it is somewhat difficult to prove this explicitly since in order to propose a fault from which to proceed with proof by contradiction, one must begin with something that is so blatantly false on its face that the rest of the proof by contradiction is

not particularly necessary.

Proposition seqlimsinR.8 — (Convergent Sequences Are Bounded)

If a sequence converges, then it is bounded. That is, if $(a_n)_{n\in\mathbb{N}}$ is a sequence which converges to a, then there exists some $b_- \in \mathbb{R}$ and some $b_+ \in \mathbb{R}$ which satisfy

$$b_- \leq a_n \leq b_+$$

for all $n \in \mathbb{N}$.

Proof.

We formalize almost exactly the story we told above. For some $\varepsilon > 0$ we have that there exists $N \in \mathbb{N}$ such that when $n \geq N$, $|a_n - a| \leq \varepsilon$, since by hypothesis $(a_n)_{n\in\mathbb{N}}$ converges. Then fix some choice of $\varepsilon > 0$ (although the particular choice does not matter) so that it implies the existence of an appropriate $N \in \mathbb{N}$ with the properties we expect for convergence. We'll set a variable $\hat{b}_+ = \sup\{|a_n - a| \mid n \geq N\}$ to be the largest distance, $|a_n - a|$, the sequence takes from its limit when $n \geq N$; this means $a + \hat{b}$ will be greater than all a_n after $n \geq N$.
Then $a+\hat{b}$ would be an upper bound for $(a_n)_{n\in\mathbb{N}}$ if not for the fact that some a_n with $n < N$ might be greater than this upper bound; to fix this, we will define an upper bound for all a_n before $n < N$, and we do this first by defining the set of points before $n < N$, $A_N = \{a_n \mid n < N\}$ (strictly this is not a set comprehension but an image of $\{n \in \mathbb{N} \mid n < N\}$ through the sequence acting as a function $a_\square \colon \mathbb{N} \to \mathbb{R}$). With this set, we can take its maximum $\max(A_n)$ and either $\max(A_N)$ or $a + \hat{b}$ will be the upper bound for the entire sequence, so let us define

$$b_+ = \max\left\{ \max(A_N), a + \hat{b}_+ \right\}$$

Since A_N is literally contains $N - 1$ or less numbers, this quantity can be computed so long as we have a method to find N from ε, which is guaranteed by hypothesis of convergence. The corresponding proof to calculate b_-, the lower bound, is exactly the same but with minimums instead of maximums and $a - \hat{b}$.

$$b_- = \min\left\{ \min(A_N), a - \hat{b} \right\}$$

Theorem seqlimsinR.9 — (Monotone Convergence Theorem)

Let $(a_n)_{n\in\mathbb{N}}$ be a sequence which is bounded above and monotone increasing, i.e. $a_n \leq a_m$ for all $m, n \in \mathbb{N}$ satisfying $m \geq n$, and there exists $b \in \mathbb{R}$ for which $a_n \leq b$ for all $n \in \mathbb{N}$. Then $(a_n)_{n\in\mathbb{N}}$ must converge.

Proof.

Once again, we take the story we told and formalize it. Let $A = \{a_n \mid n \in \mathbb{N}\}$ be the unordered set of all a_n in the sequence. Since the set is bounded and non-empty, it has a supremum by the axiom of completeness, and we call this $s = \sup(A)$.
Now we aim to show that s is the limit of $(a_n)_{n\in\mathbb{N}}$. For any $\varepsilon > 0$ we are given, there must exist some $a^* \in A$ which satisfies $s - \varepsilon \leq a^* \leq s$, since if no such $a^* \in A$ existed in this range then the least upper bound would be $s - \varepsilon$ not s. Now, since A is the set of all a_n in the sequence, $a^* \in A$ means there must exist some $N \in \mathbb{N}$ for which $a^* = a_N$, where N is the n which indexes the particular a^* we pulled out of the sequence. Then since $(a_n)_{n\in\mathbb{N}}$ is monotonic increasing, we have for all $n \geq N$ that

$$s - \varepsilon \leq a_N \leq a_n$$
$$s - \varepsilon \leq a_n$$

by transitivity of ordering. Now, if we add $\varepsilon - a_n$ to both sides of this inequality, we obtain

$$s - \varepsilon + (\varepsilon - a_n) \leq a_n + (\varepsilon - a_n)$$
$$s - a_n \leq \varepsilon.$$

This is neat, because $s - a_n$ is ensured to be positive (s is the *supremum* of the set of all points in the sequence, and thus is larger than all a_n) meaning $s - a_n = |s - a_n|$. Moreover, $|s - a_n| = |a_n - s|$ because the absolute value will undo any multiplication by -1 inside its argument. This means we get to write

$$|a_n - s| \leq \varepsilon$$

and notice that this followed as an implication from $n \geq N$ (since we could not have written $s - \varepsilon \leq a_n$ otherwise). Then we have proven the limit.

If you're interested, you should note something about the above proof. That is, unlike in previous examples, we did not actually find a method to produce N for a given $\varepsilon > 0$. What we did instead was show that *some* N exists, and indeed if you had a picture of a number line with all the points a_n marked and numbered, you would certainly be able to look between $s-\varepsilon$ and s and select some a_n to become our a_N. But instead, we selected our a_N ultimately using the axiom of choice.

Definition seqlimsinR.10 — (Cauchy Criterion)

A sequence satisfies the **Cauchy criterion** if it satisfies the following property:
for all $\varepsilon > 0$, there exists $N \in \mathbb{N}$ such that for all $m, n \geq N$, we have $|a_m - a_n| \leq \varepsilon$.
When this is true of a sequence, we call it a **Cauchy sequence**.

It is worth noting something about the Cauchy criterion, which is that it is not exactly a formal statement that 'the steps the sequence takes get smaller over time' as I said earlier. Instead it is that *any* two points in the sequence have shorter distances over time, since indeed *all* $m, n \geq N$ will satisfy $|a_m - a_n| \leq \varepsilon$. Or one could read this as the statement that the sequence gets closer to the rest of itself over time.

This criterion is in some sense parallel to our existing definition of limit convergence; at this stage we have not proven that the Cauchy criterion has anything formally to do with limits,

and yet it has a very obviously similar propositional form. Given the stories we told earlier, we can intuitively conclude some more things which we should formalize. For instance, if the sequence converges 'to itself', then for similar reasons as discussed earlier for convergent sequences, it should also be bounded. And as we discussed above, a sequence which does converge should then be Cauchy.

Proposition seqlimsinR.11

a. Let $(a_n)_{n\in\mathbb{N}}$ be a Cauchy sequence. Then the sequence is bounded.

b. Let $(a_n)_{n\in\mathbb{N}}$ be a sequence which converges to $a \in \mathbb{R}$. Then the sequence is Cauchy.

Proof.

a. Since the sequence is assumed Cauchy, we may pick any $\varepsilon > 0$ (the choice does not matter) and the Cauchy assumption says that there must exist some $N \in \mathbb{N}$ for which all $m, n \geq N$ has $|a_m - a_n| \leq \varepsilon$. Now fix a choice of m (again the particular choice does not matter). The proof will now proceed very similarly to the proof of proposition seqlimsinR.8.

Let

$$\hat{b} = \sup\{|a_m - a_n| \mid n \geq N\}$$

be the supremum difference a_m from all other elements of the sequence a_n with $n \geq N$ (i.e. the value which bounds all other differences, the largest and then some perhaps). Then $a_m + \hat{b}$ forms an upper bound for the sequence of elements $n \geq N$.

Since n is a counting number in $\mathbb{N}$, it is non-negative and thus any $n < N$ is also $0 < n < N$, meaning once again that the set of elements a_n which come before N are finite of length $N - 1$ or less. Let's call this set $B = \{a_n \mid n < N\}$ (once again, this is not technically a set comprehension but an image of one), so either $\max(B)$ or $a_m + \hat{b}$ will now bound the sequence. Our upper bound is then

$$b_+ = \max\left\{\max(B), a_m + \hat{b}\right\}$$

and by a similar argument, our lower bound is

$$b_+ = \min\left\{\min(B), a_m - \hat{b}\right\}.$$

b. Since the sequence is assumed to converge, let us say we are given some $\varepsilon > 0$ not in the sense of an epsilon that proves convergence but in the sense of an epsilon that proves it is Cauchy. Since we have the limit $a_n \to a$ as $n \to \infty$, we may also set two epsilons $\varepsilon_m, \varepsilon_n = \varepsilon/2$, each half of the epsilon we are given. For each of these epsilon, since $(a_n)_{n\in\mathbb{N}}$ converges, they each induce some N which we call $N_m, N_n \in \mathbb{N}$, both such that when $m \geq N_m$ or $n \geq N_n$ we have

$$|a_m - a| \leq \varepsilon_m$$
$$|a_n - a| \leq \varepsilon_n$$

respectively. Note also that $N_m = N_n$ since $\varepsilon_m = \varepsilon_n = \varepsilon/2$. Now we can apply the triangle inequality on the values $a_m, a_n, a \in \mathbb{R}$. We obtain

$$|a_m - a_n| \leq |a_m - a| + |a_n - a|.$$

Now consider that since $(a_n)_{n\in\mathbb{N}}$ converges, both terms $|a_m - a|$ and $|a_n - a|$ obey the relations above (each being less than $\varepsilon_m = \varepsilon_n = \varepsilon/2$) when $n \geq N_m = N_n$. For each of these, we may add a missing term to both sides by translation of orderings ($|a_n - a|$ to the first and ε_m to the other)

$$|a_m - a| + |a_n - a| \leq \varepsilon_m + |a_n - a|$$
$$\varepsilon_m + |a_n - a| \leq \varepsilon_m + \varepsilon_n$$

to construct the transitive sequence

$$|a_m - a_n| \leq |a_m - a| + |a_n - a| \leq \varepsilon_m + |a_n - a| \leq \varepsilon_m + \varepsilon_n$$

which is

$$|a_m - a_n| \leq \varepsilon_m + \varepsilon_n = \frac{\varepsilon}{2} + \frac{\varepsilon}{2} = \varepsilon$$

meaning that

$$|a_m - a_n| \leq \varepsilon$$

as desired. Thus the $N \in \mathbb{N}$ we wished to prove existed for a given $\varepsilon > 0$ was in fact the same $N \in \mathbb{N}$ induced by our limit when we used $\varepsilon/2$ for convergence. When $n \geq N = N_m = N_n$, we have $|a_m - a_n| \leq \varepsilon$, satisfying the Cauchy criterion.

The above proofs are minimized by default since many of them appear at once; I highly recommend that you take a look at them but if you read them in order, you may begin to notice the proofs repeating themselves a little here. What are beginning to appear are *proof techniques*, recurring patterns that are useful to us when we need to prove something. If indeed you are trying to follow along with exercises or trying to reproduce these proofs yourself (an exercise of its own that I highly recommend) then these techniques will be indispensible to you. Before long, the particular techniques we use will find their applications few and far between, as we package up each of their implications into theorems which we apply to greatly simplify the process.

We will shortly be able to prove that Cauchy sequences always converge in $\mathbb{R}$, but to do so (and indeed for the nature of the proof to be meaningful to us) we will need some additional structures which we are yet to discuss. Accordingly, we postpone both the proof of this and the proof that bounded sequences have convergent subsequences until the following section, when we are able to discuss the formal reality of a *region of convergence*.

We close this section instead with some properties of limits that bare a resemblance to the real numbers. That is, in much the same way that real numbesr are defined with addition, subtraction, multiplication, ordering, and an Archimedean property, we will see that we can do most of these things for limits.

All these will be discussed and proven in the section appendix.

seqlimsinR.3 Section Appendix: The Limit is a Homomorphism

In the following chapter we will spend much of our focus on algebraic properties. These can be difficult to understand as the algebraic frame of mind is far removed from that of analysis, much less the frame of analytic geometry that most people encounter as highschool mathematics. However we have in fact already met these concerns when we discussed real numbers and the limit of a sequence.

That is, one first encounters the term algebra to refer to the ways in which one manipulates symbols standing for numbers using rewrites related to addition, subtraction, multiplication, and division. But this new structure we have introduced, the sequence, can also have these notions since they are related to real numbers. That is, for any sequences $(a_n)_{n\in\mathbb{N}}$ and $(b_n)_{n\in\mathbb{N}}$, it is clear certainly that we can define a new sequence $(c_n)_{n\in\mathbb{N}}$ which is either $c_n = a_n + b_n$, $c_n = a_n - b_n$, $c_n = a_n b_n$, or when $b_n \neq 0$ for all $n \in \mathbb{N}$, even $c_n = a_n/b_n$.

The question then appears; we have an operation, the limit, which can turn some sequences (those that converge) into real numbers, but how much structure does it preserve? Which manipulations can we do to sequences before will take the limit will remain after we take the limit? Can we ensure that $\lim_{n\to\infty} c = (\lim_{n\to\infty} a_n) + (\lim_{n\to\infty} b_n)$?

The question we are ultimately asking is "is the limit homomorphic", the formal name for asking if the function we are discussing preserves the features and structure of its input when it becomes an output. Since both sequences and real numbers can be added, subtracted, multiplied, and divided, there is indeed a lot of structure that the limit could potentially preserve, and if it does, this would grant us a great many tools in proofs going forward.

In fact it is true that the limit is homomorphic in a sense; our notion of division must surely be weakened since in order to preserve the structure of a division, the divisor must *never* be zero. But this is still enough to satisfy the requirements for a *partially ordered ring homomorphism*, a map that preserves the structures of addition, subtraction, multiplication and ordering, but not division. Strictly speaking it preserves even more structure than that, but that is certainly the neatest existing class of homomorphism we can put the limit in. In practice, one rarely thinks of the limit in this way with relation to algebraic rings, but rather focus on exactly the structure preservation we are about to discuss with each preserved feature in mind individually.

The implication of this however, if you wish to dwell on it, is that there is something *number-like* about convergent sequences of real numbers, much in the way we expect real numbers to function as numbers

Let us begin to show this.

Theorem seqlimsinR.12 — (Algebraic Limit Theorem for Sequences)

Let $(a_n)_{n\in\mathbb{N}}$ and $(b_n)_{n\in\mathbb{N}}$ be two sequences which converge to a and b respectively. Then we have the following.

a. The sequence defined $c_n = ka_n$ for any constant $k \in \mathbb{R}$ converges to $\lim_{n\to\infty} c_n = ka$. If we write this without c_n, this is

$$\lim_{n\to\infty} (ka_n) = k\left(\lim_{n\to\infty} a_n\right)$$

b. The sequence defined $c_n = a_n + b_n$ converges to $\lim_{n\to\infty} c_n = a + b$. If we write

this without c_n, this is

$$\lim_{n\to\infty}(a_n + b_n) = \left(\lim_{n\to\infty} a_n\right) + \left(\lim_{n\to\infty} b_n\right)$$

c. The sequence defined $c_n = a_n b_n$ converges to $\lim_{n\to\infty} c_n = ab$. If we write this without c_n, this is

$$\lim_{n\to\infty}(a_n b_n) = \left(\lim_{n\to\infty} a_n\right)\left(\lim_{n\to\infty} b_n\right)$$

d. When $b_n \neq 0$ for all $n \in \mathbb{N}$ and $b \neq 0$, the sequence defined $c_n = a_n/b_n$ converges to $\lim_{n\to\infty} c_n = a/b$. If we write this without c_n and the requirements are met, this is

$$\lim_{n\to\infty}\frac{a_n}{b_n} = \frac{\lim_{n\to\infty} a_n}{\lim_{n\to\infty} b_n}$$

Note here that a specific carve out is made for division, since it has additional requiremets, but not for subtraction; in place of subtraction we have *scalar multiplication*, since it is then possible to set $k = -1$ for part a and use it on conjunction with part b to define $c_n = a_n + (-1)b_n = a_n - b_n$.

Proof.

Let us first notice what we know and what we need to prove. We know $(a_n)_{n\in\mathbb{N}}$ and $(b_n)_{n\in\mathbb{N}}$ converge to a and b respectively, so for each of them we have a way to find N_a and N_b for any given ε_a or ε_b we choose, satisfying the rest of the limit properties. We are proving convergence of $(c_n)_{n\in\mathbb{N}}$ so we presume we are given some $\varepsilon_c > 0$ and need to find $N_c \in \mathbb{N}$ such that when $n \geq N_c$, we have the appropriate absolute value difference smaller than ε.

a. For $\varepsilon_c > 0$, we must find $N_c \in \mathbb{N}$ such that when $n \geq N_c$, we have $|ka_n - ka| \leq \varepsilon_c$. So set $\varepsilon_a = \varepsilon_c/k$; by convergence of $(a_n)_{n\in\mathbb{N}}$, we have that when $n \geq N_a$, we have

$$|a_n - a| \leq \varepsilon_a = \frac{\varepsilon_c}{k}$$

so we may merely multiply both sides by k to obtain

$$k|a_n - a| \leq k\left(\frac{\varepsilon_c}{k}\right)$$
$$|ka_n - ka| \leq \varepsilon_c$$

by distributivity and the property $k|a| = |ka|$ of absolute values. This means that $c_n = ka_n \to ka$ as $n \to \infty$ as we desired to show.

b. For $\varepsilon_c > 0$, we must find $N_c \in \mathbb{N}$ such that when $n \geq N_c$, we have

$$|(a_n + b_n) - (a + b)| \leq \varepsilon_c.$$

So set $\varepsilon_a, \varepsilon_b = \varepsilon_c/2$ so that they induce $N_a, N_b \in \mathbb{N}$ each with

$$n \geq N_a \implies |a_n - a| \leq \varepsilon_a = \frac{\varepsilon_c}{2},$$
$$n \geq N_b \implies |b_n - b| \leq \varepsilon_b = \frac{\varepsilon_c}{2}.$$

Set $N_c = \max\{N_a, N_b\}$ so that both $n \geq N_a$ and $n \geq N_b$ are satisfied when $n \geq N_c$. Now we apply the triangle inequality on the values $(a_n - a), (b_n - b), 0 \in \mathbb{R}$ and obtain

$$|(a_n - a) - (b_n - b)| \leq |(a_n - a) - 0| + |(b_n - b) - 0|$$

which we may easily rearrange some terms to turn it into

$$|(a_n + b_n) - (a + b)| \leq |a_n - a| + |b_n - b|.$$

Now we may translate our existing relations which are true when $n \geq N_c$ by $|b_n - b|$ and ε_a respectively

$$|a_n - a| + |b_n - b| \leq \varepsilon_a + |b_n - b|$$
$$\varepsilon_a + |b_n - b| \leq \varepsilon_a + \varepsilon_b$$

and this forms the transitive sequence

$$\begin{aligned} |(a_n + b_n) - (a + b)| &\leq |a_n - a| + |b_n - b| \\ &\leq \varepsilon_a + |b_n - b| \\ &\leq \varepsilon_a + \varepsilon_b \\ &= \frac{\varepsilon_c}{2} + \frac{\varepsilon_c}{2} \\ &= \varepsilon_c \end{aligned}$$

which is exactly our desired

$$|(a_n + b_n) - (a + b)| \leq \varepsilon_c$$

when $n \geq N_c$. We are done.

c. For any $\varepsilon_c > 0$ that we are given, we want some $N_c \in \mathbb{N}$ such that $|(a_n b_n) - ab| \leq \varepsilon_c$. Let us say that we choose some $\varepsilon_a, \varepsilon_b > 0$ related to ε_c, but for now it is better to postpone explaining what we will set them too since an excellent choice will become obviously shortly. If ε_a and ε_b are chosen then they induce $N_a, N_b \in \mathbb{N}$ such that for each $n \geq N_a$ and $n \geq N_b$, we have $|a_n - a| \leq \varepsilon_a$ and $|b_n - b| \leq \varepsilon_b$. Set $N_c = \max\{N_a, N_b\}$ once again so that both $n \geq N_a$ and $n \geq N_b$ are satisfied when $n \geq N_c$.

Apply the triangle inequality on the elements $b_n(a_n - a)$, $a(b - b_n)$ and $0 \in \mathbb{R}$. We obtain

$$|b(a_n - a) - a(b - b_n)| \leq |b_n(a_n - a) - 0| + |a(b - b_n) - 0|$$

rearranging terms and simplying to

$$|a_n b - ab_n + ab_n - ab| \leq |b_n||a_n - a| + |a||b_n - b|,$$
$$|a_n b - ab| \leq |b_n||a_n - a| + |a||b_n - b|.$$

A mess of symbols though this might seem, we have a trick up our sleeves in proposition seqlimsinR.8. Since $(b_n)_{n \in \mathbb{N}}$ converges we know it has a bound $B \in \mathbb{R}$ such that $|b_n| \leq B$ for all $n \in \mathbb{N}$. We can focus on individual terms of the above inequality to see where B applies

$$|b_n||a_n - a| \leq B|a_n - a|$$

which we know by taking $|b_n| \leq B$ and dilating it by a factor $|a_n - a|$, and we can then translate this by $|a||b_n - b|$ to obtain

$$|b_n||a_n - a| + |a||b_n - b| \leq B|a_n - a| + |a||b_n - b|.$$

We can apply this transitively on our old relation then to obtain

$$|a_n b - ab| \leq B|a_n - a| + |a||b_n - b|$$

We are now ready to define $\varepsilon_a = \varepsilon_c/2B$ and $\varepsilon_b = \varepsilon_c/2|a|$, since neither B nor a rely on a choice of n or N or anything downstream of our pretending we had already chosen ε_a and ε_b. There is one problem with this though, which is that $|a|$ could be zero (B can not, think about why) but in this case set $\varepsilon_b = \varepsilon_c/2$ and use by transitivity $0 = |a||b_n - b| \leq 1 \cdot |b_n - b|$.

Now recall that we started with $n \geq N_c$ implies $|a_n - a| \leq \varepsilon_a$ and $|b_n - b| \leq \varepsilon_b$. These are now

$$|a_n - a| \leq \varepsilon_a = \frac{\varepsilon_c}{2B},$$
$$|b_n - b| \leq \varepsilon_b = \frac{\varepsilon_c}{2|a|},$$

or equivalently by dilations,

$$B|a_n - a| \leq \frac{\varepsilon_c}{2},$$
$$|a||b_n - b| \leq \frac{\varepsilon_c}{2}.$$

We'll skip the obvious step of translating these so that they appear in the obvious forms for a transitive sequence since we've done that a few times now, but needless to say we now have

$$|a_n b_n - ab| \leq B|a_n - a| + |a||b_n - b| \leq \frac{\varepsilon_c}{2} + \frac{\varepsilon_c}{2} = \varepsilon_c,$$
$$|a_n b_n - ab| \leq \varepsilon_c$$

which is exactly what we wanted to prove. Since this is true whenever $n \geq N_c$, we know that $a_n b_n \to ab$ when $n \to \infty$. We are done.

d. Part d follow trivially from part c so long as we first prove that $(d_n)_{n\in\mathbb{N}}$ defined $d_n = 1/b_n$ converges to $1/b$, since then we can say $a_n d_n \to a(1/b) = a/b$ as $n \to \infty$.

To do that, we will need to employ some unusual techniques in the proving of a limit. We will need to show that for all $\varepsilon_d > 0$ there exists some $N_d \in \mathbb{N}$ such that when $n \geq N_d$, $|1/b_n - 1/b| \leq \varepsilon_d$; first notice something about the term $|1/b_n - 1/b|$. We know that $(b_n)_{n\in\mathbb{N}}$ converges, and so we know that there exists some N_b for when $\varepsilon_b = |b|/2$ we will be able to write

$$|b_n - b| \leq \varepsilon_b = \frac{|b|}{2}.$$

Recall in lemma realnumsax.13.e we showed that $|a| \leq k$ satisfies $-k \leq a \leq k$ so we'll use that now on $|b|/2$ and $b_n - b$. This gives us

$$-\frac{|b|}{2} \leq b_n - b \leq \frac{|b|}{2}.$$

Now consider that $|b|$ is either b or $-b$ depending on whether b is positive or negative, and depending on those two cases, this sequence of inequalities is one of the following:

$$-\frac{b}{2} \leq b_n - b \leq \frac{b}{2},$$
$$\frac{b}{2} \leq b_n - b \leq -\frac{b}{2}.$$

We will see that despite not knowing which is true, the structure of the inequality means that we can say something regardless; in either case, add b to both sides to obtain one of the two

$$\frac{b}{2} \leq b_n \leq \frac{3b}{2},$$
$$\frac{3b}{2} \leq b_n \leq \frac{b}{2}.$$

Now consider what happens to the ordering if we take the absolute value: the absolute value operation $x \mapsto |x|$ is a monotonic increasing function $x \mapsto x$ when its argument is greater than or equal to zero, meaning that in the first case the order is preserved, but it is a monotonic decreasing function $x \mapsto -x$ when its argument is less than or equal to zero, which flips the orderings, meaning both cases result in

$$\frac{|b|}{2} \leq |b_n| \leq \frac{3|b|}{2}.$$

This entire inequality is of some interest to us, since it says that at some point, when $n \geq N_b$ for some N_b induced by a choice $\varepsilon_b = |b|/2$, we will have $|b_n$ less than $|b|/2$ away from $|b|$ but specifically we need the first part,

$$\frac{|b|}{2} \leq |b_n|.$$

Moreover, if we say $n \geq N_d$ for some N_d we specify shortly, and $N_d \geq N_b$, then we will have $n \geq N_d \geq N_b$, and thus $n \geq N_b$, implying that $|b|/2 \leq |b_n|$ also. Once again, this is all downstream of the fact that $|b|/2$ is itself $|b|/2$ away from $|b|$, and at some point $|b_n|$ must get close enough to $|b|$ that it cannot be more than $|b|/2$ away in the course of its convergence on b.

This may at first seem like a strange fact, since we have not previously explored setting ε for known limits to some specific value in order to obtain specific inequalities, but it remains a valid one, and is indeed vital for the rest of the proof.

Now say we are given some $\varepsilon_d > 0$ and we need to find $N_d \in \mathbb{N}$ such that when $n \geq N_d$, we have $|1/b_n - 1/b| \leq \varepsilon$. Set $\varepsilon_b = |b|^2\varepsilon_d/2$ so that it induces some N_b, but importantly, if this N_b induced by $\varepsilon_b = |b|^2\varepsilon_d/2$ is smaller than the N_b induced by $\varepsilon_b = |b|/2$, take the larger N_b to be our N_d so that it is always true that $|b_n| \geq |b|/2$.

Now observe we can do some manipulations on the fractions in $|1/b_n - 1/b|$,

$$\begin{aligned}\left|\frac{1}{b_n}-\frac{1}{b}\right| &= \left|\frac{b}{bb_n}-\frac{b_n}{bb_n}\right| \\ &= \left|\frac{b-b_n}{bb_n}\right| \\ &= \frac{|b-b_n|}{|b||b_n|}.\end{aligned}$$

We can now apply $|b|/2 \leq |b_n|$ on this by inverting the inequality, which we recall is a monotonic decreasing function $x \mapsto 1/x$ so we flip the ordering and write $2/|b| \geq 1/|b_n|$. This new inequality, dilated on both sides by $|b-b_n|/|b|$ is exactly the form above, and gives us

$$\left|\frac{1}{b_n}-\frac{1}{b}\right| = \frac{|b-b_n|}{|b||b_n|} \leq \frac{2|b_n-b|}{|b|^2}.$$

Presuming that $n \geq N_d$ and $N_d = N_b$ or $N_d > N_b$ depending on what we chose earlier, we know that $n \geq N_b$ and thus $|b_n - b| \leq \varepsilon_b$ is satisfied. Dilating this on both sides by $2/|b|^2$ it is

$$\frac{2|b_n-b|}{|b|^2} = \left(\frac{2}{|b|^2}\right)\left(|b_n-b|\right) \leq \left(\frac{2}{|b|^2}\right)(\varepsilon_b)$$

but we had set $\varepsilon_b = |b^2|\varepsilon_d/2$ so we have

$$\left|\frac{1}{b_n}-\frac{1}{b}\right| \leq \frac{2|b_n-b|}{|b|^2} \leq \left(\frac{2}{|b|^2}\right)\left(\frac{|b^2|\varepsilon_d}{2}\right).$$

Cancelling some terms in our fractions on the right and ignoring the middle of this transitive inequality, we have

$$\left|\frac{1}{b_n}-\frac{1}{b}\right| \leq \varepsilon_d$$

as we had desired.

I highly recommend looking over the above proof for any techniques which are unfamiliar, since there are a few there we haven't used before.

It must be noted that while all numbers can be compared in $\mathbb{R}$, this is not the case for sequences since it is entirely possible for two sequences to have, say $a_1 > b_1$ but $a_2 < b_2$ for different parts of the sequence $n = 1$ or $n = 2$ respectively. This is what it means to be partially ordered; in a partially ordered set, objects can be ordered but not reliably. Some objects will be incomparable with one another as is the case in sequences.

The point of the *Order Limit Theorem* will be that when such an ordering does exist, it is preserved under the limit.

Theorem seqlimsinR.13 — (Order Limit Theorem)

Let $(a_n)_{n\in\mathbb{N}}$ and $(b_n)_{n\in\mathbb{N}}$ be sequences converging to a and b respectively. If $a_n \leq b_n$ for all $n \in \mathbb{N}$ then $a \leq b$. That is,

$$\forall n \in \mathbb{N}, (a_n \leq b_n) \implies \left(\lim_{n\to\infty} a_n\right) \leq \left(\lim_{n\to\infty} b_n\right)$$

Proof.

This time we are not proving a limit, but rather purely trying to get the inequality $a \leq b$ from knowing that $a_n \to a$ and $b_n \to b$ as $n \to \infty$ and $a_n \leq b_n$ for all $n \in \mathbb{N}$. First, set ε_a and ε_b both to the value $\varepsilon_a, \varepsilon_b = |a-b|/3$ and set $N = \max\{N_a, N_b\}$ the maximum of the two. And since $\varepsilon_a = \varepsilon_b$, it will become prudent to refer to them interchangably so let them both be written ε. So we have

$$|a_n - a| \leq \varepsilon = \frac{|a-b|}{3}$$
$$|b_n - b| \leq \varepsilon = \frac{|a-b|}{3}$$

when we assume $n \geq N$.
We start by applying lemma realnumsax.13.e to these inequalities, obtaining

$$-\varepsilon \leq a_n - a \leq \varepsilon$$
$$-\varepsilon \leq b_n - b \leq \varepsilon$$

from which we take the left inequality from the first line, multiplying by -1 and adding a_n to both sides, and the right inequality of the second line adding $b + \varepsilon$ to both sides.

$$a \leq a_n + \varepsilon$$
$$b_n + \varepsilon \leq b + 2\varepsilon$$

Since we have $a_n \leq b_n$ for all $n \in \mathbb{N}$, we also have $a_n + \varepsilon \leq b_n + \varepsilon$, which gives us the transitive sequence

$$a \leq a_n + \varepsilon \leq b_n + \varepsilon \leq b + 2\varepsilon$$
$$a \leq b + 2\varepsilon.$$

We can subtract b from both sides and divide by two to isolate ε before expanding to its form as $|a-b|/3$, obtaining

$$\frac{a-b}{2} \leq \frac{|a-b|}{3}.$$

Note at this point that since no value in this inequality depends on n, despite the fact that we implied it from $n \geq N$, it must be true always since it does not change when n is changed, and indeed it is always possible to set $n \geq N$ since $(a_n)_{n\in\mathbb{N}}$ and $(b_n)_{n\in\mathbb{N}}$ converge.
We now have three cases to consider. By the trichotomy of ordering, $a-b$ is either $a-b<0$, $a-b=0$ or $a-b>0$. We'll call these cases (1), (2) and (3).

In case (1), $a - b < 0$ implies $a < b$ by adding b to both sides, which is exactly what we wanted to prove.
In case (2), $a - b = 0$ implies $a = b$ by adding b to both sides, and this is a subcase of $a \leq b$ which is literally $(a < b) \wedge (a = b)$, so once again the theorem is proven in this case.
Case (3) gets a bit weirder, since assuming $a-b > 0$ and thus $a > b$ also implies $|a-b| = a-b$, the inequality we have above can be rewritten by manipulations on fractions,

$$\begin{aligned} \frac{a-b}{2} &\leq \frac{a-b}{3} \\ \frac{a-b}{2} - \frac{a-b}{3} &\leq 0 \\ \frac{3(a-b)}{6} - \frac{2(a-b)}{6} &\leq 0 \\ \frac{a-b}{6} &\leq 0 \\ a - b &\leq 0 \end{aligned}$$

but this is in contradiction with our assumption that $a-b > 0$, so we can assume that this case never occurs.
This means that only cases (1) or (2) never happen, and thus $a \leq b$ is always true when $a_n \leq b_n$ for all $n \in \mathbb{N}$.

In the following section, our discussion on regions of convergence will make it intuitively obvious that the 'regions' that $(a_n)_{n\in\mathbb{N}}$ and $(b_n)_{n\in\mathbb{N}}$ converge in are obviously non-overlapping eventually when $a < b$, and thus that $a < b$ has to be true.

As promised above, the following theorem will be our very rough analogy to the Archimedean principle for limits. That is, it says nothing about infinitessimals, however in the same way that showing no infinitessimals exists means that two numbers are presumed equal if you cannot find a number between them, this theorem will say that a sequence must converge if it is bounded on either side by sequences which together converge on the same limit between them. In this way, a sequence is *squeezed* into convergence.

When last I was taking graduate mathematics class, my peers and I still considered this theorem a relevant one which we would mention by name, a testament to its enduring usefulness.

Theorem seqlimsinR.14 — (Squeeze Theorem)

Let $(a_n)_{n\in\mathbb{N}}$, $(b_n)_{n\in\mathbb{N}}$ and $(c_n)_{n\in\mathbb{N}}$ be sequences, where $a_n \to a$ and $c_n \to a$ as $n \to \infty$. If $a_n \leq b_n \leq c_n$ for all $n \in \mathbb{N}$, then $(b_n)_{n\in\mathbb{N}}$ also converges to a.

Proof.

Say we are given some $\varepsilon_b > 0$ in the course of proving its limit. Set ε_a and ε_c to $\varepsilon_a, \varepsilon_c = \varepsilon_b$ and $N_b = \max\{N_a, N_c\}$ so that when $n \geq N_b$, we have

$$\begin{aligned} |a_n - a| &\leq \varepsilon_a = \varepsilon_b, \\ |c_n - a| &\leq \varepsilon_c = \varepsilon_b. \end{aligned}$$

We'll now apply a similar technique to the one from the order limit theorem proof, applying

lemma realnumsax.13.e both times to get

$$-\varepsilon_b \leq a_n - a \leq \varepsilon_b,$$
$$-\varepsilon_b \leq c_n - a \leq \varepsilon_b$$

where we take the left hand inequality of the first line and add a to both sides, and take the right hand inequality on the second line and add a to both sides, getting

$$a - \varepsilon_b \leq a_n,$$
$$c_n \leq a + \varepsilon_b.$$

Since by assumption we have $a_n \leq b_n \leq c_n$ for all $n \in \mathbb{N}$, we can add these inequalities to the transitive chain and obtain

$$a - \varepsilon_b \leq a_n \leq b_n \leq c_n \leq a + \varepsilon_b$$
$$a - \varepsilon_b \leq b_n \leq a + \varepsilon_b$$
$$-\varepsilon_b \leq b_n - a \leq \varepsilon_b$$

which now lets us use the other direction of lemma realnumsax.13.e to obtain

$$|b_n - a| \leq \varepsilon_b$$

as desired when $n \geq N_b$. We are done.

openlimsR Intervals in ℝ and Limit Characterizations

In this section it is pertinant to stop and examine the properties of real numbers again, as it turns out that the axiom of completeness has much more significant consequences than its mere statement might suggest. That is, we have mentioned many times in the previous section the notion of a 'region of convergence', and while we have formally discussed the idea of 'convergence', we have not yet formally discussed the notion of a 'region'.

Such a concept is deceptively complicated, as it cuts to the heart of what separates the discrete regime of numbers from continuous mathematics, and more importantly, is at the heart of why it is meaningful for us to speak of numbers as a model for geometry at all. The real numbers are constructed with the intention that they have *no gaps*. In mathematics, we speak about this property positively and instead say that "the real numbers are **complete**". There are multiple criteria for completeness (which in certain hairsplitting settings even disagree quite badly) but for the purposes of real analysis we are generally concerned with *Cauchy completeness*.

In fact Cauchy completeness is exactly the property that in a space, all Cauchy sequences converge, a property that we will return to in many settings since it is profoundly non-trivial. Consider for instance the set of rational numbers, the numbers formed from fractions of integers, $\mathbb{Q}$. Obviously numbers such as $\sqrt{2}$ are not included in such a set, and yet it is easy to construct a sequence of rational numbers that converges to an irrational. In fact it is relatively easy using the reasoning of the monotone convergence theorem to simply imagine a sequence of only rational numbers which is monotonic increasing but bounded above by $\sqrt{2}$ (since irrational numbers and rational numbers may be placed on the same ordered axis) which is at every point rational and yet always

approaching an irrational number. It is possible to speak of such a sequence converging and yet it must converge to something outside of the space it is defined in, that of the rationals.

It is precisely because we have the axiom of completeness that we may point to a subset of the rational numbers, bounded above by a number which is not itself rational and thus a 'gap' as far as the rationals are concerned, and fill that gap with whatever needs to be there to *complete* the space with a supremum of the set. There are far greater elaborations on these concepts that we can make but they will have to wait for the coming sections of this chapter. For now, we should actually find out what a region is in our $\mathbb{R}$ conception of 'space'.

openlimsR.1 Intervals and some promised Theorems

We begin our study of 'regions' with the notion of an interval. Formally, the word 'line' refers to a line that does not end in either direction; if it stops in one direction but continues indefinitely in the other then we call it a 'ray', and if it ends in both directions and thus has finite length, we call it a 'segment'. Obviously, if we are concerned with the real number *line* then we may speak of real number *segments*, and that is exactly what is described by an interval. Just as you think of a segment of time as beginning at one specified moment and ending at another, we can set a beginning and an end for an interval to define a very basic space.

Definition openlimsR.1

A set $I \subset \mathbb{R}$ is called an **open interval** if there exists two numbers $a, b \in \mathbb{R}$ such that

$$I = \{x \in \mathbb{R} \mid a < x < b\}$$

i.e. the interval is all of the points *between* a and b but not including them. We then write as notation

$$(a, b) := \{x \in \mathbb{R} \mid a < x < b\}.$$

We may also use this notation on open intervals such as $\{x \in \mathbb{R} \mid x < b\}$ or $\{x \in \mathbb{R} \mid a < x\}$, writing

$$(-\infty, b) := \{x \in \mathbb{R} \mid x < b\}$$
$$(a, \infty) := \{x \in \mathbb{R} \mid a < x\}$$
$$(-\infty, \infty) := \mathbb{R}$$

which we shall think of as making sense since this notation *distinctly does not include* $-\infty$ or ∞, and thus only deals with real numbers.
A set $I \subset \mathbb{R}$ is called a **closed interval** if there exists two numbers $a, b \in \mathbb{R}$ such that

$$I = \{x \in \mathbb{R} \mid a \leq x \leq b\}$$

i.e. the intterval is all of the points between a and b *and including them.* We then write as notation

$$[a, b] := \{x \in \mathbb{R} \mid a \leq x \leq b\}$$

which thus allows one to write $[a, a] = \{a\}$ for a singleton set. There is also mixed interval notation

$$[a,b) := \{x \in \mathbb{R} \mid a \leq x < b\}$$
$$(a,b] := \{x \in \mathbb{R} \mid a < x \leq b\}$$

since the curved parentheses denote not including the endpoint and the square bracket denotes including the endpoint.

I should tell you if you become interested in other mathematics texts that many authors will write open intervals not as (a,b) but rather as $]a,b[$ to imply that those endpoints are taken by some imaginary closed intervals on either side of the open interval.

It is also worth noting that this notation comes along with a certain responsibility, which is that when using it, we have to be certain that (a,b) or $[a,b]$ indeed has $a < b$ or $a \leq b$ respectively. This does not necessarily mean that at every point we write an interval in a proof we pause to show that the interval is a valid one (unless for some reason that should be necessary) but it is a case where the tools of our reasoning can be fallable. In fact this particular responsibility has probably never lead any major particular error, but it remains a good example of things we have to keep track of that simultaneously validate our thought process but are written only because we are certain they are true. For instance, neither can we write $\lim_{n\to\infty} a_n$ if $(a_n)_{n\in\mathbb{N}}$ is not a sequence which converges, and there will be many instances where it is tempting to look at a sequence and simply think of $\lim_{n\to\infty}$ as a function on sequences, when it is only a function on convergent sequences.

But this is our first look at a *region*, and it is immediate that we can do some things with such a definition. In fact, since intervals can be thought of like segments, then we might consider their length; we have been using $|a-b|$ as the distance between two points a and b since it is the *magnitude* of their difference. Now recall what a limit describes: we say that the points in a sequence beyond some $N \in \mathbb{N}$ are $|a_n - a| \leq \varepsilon$ which is the same as

$$-\varepsilon \leq a_n - a \leq \varepsilon$$

by lemma realnumsax.13.e and thus also

$$a - \varepsilon \leq a_n \leq a + \varepsilon.$$

But the set of points x satisfying this condition in place of a_n would then be an interval $[a-\varepsilon, a+\varepsilon]$ since they are the same statement. This is our first observation about the role of 'regions' in convergence.

Lemma openlimsR.2

Let $a, b, c \in \mathbb{R}$ and $c \geq 0$. Then we have the following equivalences of statements.

- $|a-b| \leq c$ if and only if $a \in [b-c, b+c]$
- if $c \neq 0$ then $|a-b| < c$ if and only if $a \in (b-c, b+c)$

Proof.

Apply lemma realnumsax.13.e as mentioned above. Immediately our statements on absolute values are recognized as equivalent to

$$-c \leq a - b \leq c$$
$$-c < a - b < c$$

respectively. We can of course add b in both cases immediately to obtain

$$b - c \leq a \leq b + c$$
$$b - c < a < b + c$$

for the two cases. If we define sets I and $\bar{I}$ (distinguished by the presence of the bar above one) as

$$\bar{I} = [b - c, b + c] = \{x \in \mathbb{R} \mid b - c \leq x \leq b + c\}$$
$$I = (b - c, b + c) = \{x \in \mathbb{R} \mid b - c < x < b + c\}$$

then the statement $b-c \leq a \leq b+c$ is equivalent to $a \in \bar{I}$ and the statement $b-c < a < b+c$ is equivalent to $a \in I$. So we have showed the implication, however all of our steps of reasoning were reversible, so we have shown the implication in both directions.

Corollary openlimsR.3

Combining lemma openlimsR.2 and lemma realnumsax.13.e one immediately concludes that for all $a, b \in \mathbb{R}$, $|a| \leq c$ if and only if $a \in [-c, c]$, and $|a| < c$ if and only if $a \in (-c, c)$.

Corollary openlimsR.4

From lemma openlimsR.2 it follows that the statement of the limit for a sequence $(a_n)_{n\in\mathbb{N}}$
for all $\varepsilon > 0$ there exists $N \in \mathbb{N}$ such that $n \geq N$ implies $|a_n - a| \leq \varepsilon$
is equivalent to the statement
for all $\varepsilon > 0$ there exists $N \in \mathbb{N}$ such that $n \geq N$ implies $a_n \in [a - \varepsilon, a + \varepsilon]$.

With this mental tool in hand, it now becomes pertinant to ask certain questions. For instance, in the section appendix of the previous section we presented the order limit theorem, showing that two convergent sequences which retain an ordering for all $n \in \mathbb{N}$ will continue to satisfy that ordering in their limit. But we may also consider the interval between them. We can now finally ask about limits not of numbers but of sets, with infinite intersections or infinite unions.

Proposition openlimsR.5 — (Nested Interval Property)

Let $(a_n)_{n\in\mathbb{N}}$ and $(b_n)_{n\in\mathbb{N}}$ be sequences with the property that $a_n \leq b_n$ for all $n \in \mathbb{N}$ and the closed intervals defined $I_n = [a_n, b_n]$ satisfy $I_{n+1} \subseteq I_n$, i.e. $(a_n)_{n\in\mathbb{N}}$ is monotonic increasing and $(b_n)_{n\in\mathbb{N}}$ is monotonic decreasing such that each successive interval is smaller and contained in the previous one. Then, defining the appropriate notation as below, the set

$$\bigcap_{n\in\mathbb{N}} I_n := \lim_{n\to\infty} \bigcap_{i=1}^{n} I_i$$

is a nonempty set. In other words, the intersection of any sequence of successively nested intervals is non-empty.

Proof.

We may immediately notice that all of the requirements for the monotone convergence theorem are satisfied for both $(a_n)_{n\in\mathbb{N}}$ and $(b_n)_{n\in\mathbb{N}}$, so we know that they converge to values a and b in $\mathbb{R}$ respectively. Moreover, the order limit theorem then immediately tells us that these a and b satisfy $a \leq b$.

Now let us inspect the statement we wish to prove. Since the intervals are nested, we have that each intersection is equal only to the smallest interval in the intersection operation, which is the $I_\square$ with highest i. Recalling the big operator notation, for $\bigcap_{i=1}^{n} I_i$, this highest i is itself n, so $\bigcap_{i=1}^{n} I_i = I_n$. Another way of saying this is that since I_{i+1} is the set of points x which satisfy $a_{i+1} \leq x \leq b_{i+1}$ then by transitivity,

$$a_i \leq a_{i+1} \leq x \leq b_{i+1} \leq b$$

meaning $a_i \leq x \leq b$ are automatically satisfied by transitivity on both sides, i.e. $I_i \cap I_{i+1} = I_{i+1}$. And this remains inductively true, i.e. we may count up or down as many times as we want and still find that the deepest nested interval is the only one remaining from the intersection.

This means that

$$\begin{aligned}\bigcap_{n\in\mathbb{N}} I_n = \lim_{n\to\infty} \bigcap_{i=1}^{n} I_i &= \lim_{n\to\infty} I_n \\ &= \lim_{n\to\infty} [a_n, b_n] \\ &= \lim_{n\to\infty} \big\{x \in \mathbb{R} \mid a_n \leq x \leq b_n\big\}.\end{aligned}$$

It may here seem difficult to proceed, since we do not have an obvious rule that allows us to pass a limit inside a set. We do however have a way to pass a limit to an inequality via the order limit theorem, and we can apply this on $a_n \leq x$ and $x \leq b_n$ separately to produce a transitive chain. All that we require to do such a thing is to reinterpret each $x \in \mathbb{R}$ as a sequence $(x_n)_{n\in\mathbb{N}}$ for which $x_n = x$ for all $n \in \mathbb{N}$, i.e. each $(x_n)_{n\in\mathbb{N}}$ is a constant sequence which is always equal to the value of $x \in \mathbb{R}$ we care about. Consequently each such sequence converges trivially to its corresponding x. Then each $x \in \mathbb{R}$ may apply the order limit theorem and deduce that $a_n \leq x_n$ and $x_n \leq b_n$ implies $a \leq x$ and $x \leq b$, thus $a \leq x \leq b$. Since we can apply this for all $x \in \mathbb{R}$, we have in a sense performed a transformation on the condition that x is defined to satisfy in I_n. So just as in the order limit theorem we wrote

$$\forall n \in \mathbb{N}, (a_n \leq b_n) \implies \Big(\lim_{n\to\infty} a_n\Big) \leq \Big(\lim_{n\to\infty} b_n\Big)$$

we will now write

$$\begin{aligned}\lim_{n\to\infty} \big\{x \in \mathbb{R} \mid a_n \leq x \leq b_n\big\} &= \big\{x \in \mathbb{R} \mid \lim_{n\to\infty} a_n \leq x \leq \lim_{n\to\infty} b_n\big\} \\ &= \big\{x \in \mathbb{R} \mid a \leq x \leq b\big\} \\ &= [a, b].\end{aligned}$$

This alone is not the statement that $\bigcap_{n\in\mathbb{N}} I_n$ is non-empty, however $a \leq b$ is either $a < b$ or $a = b$. In the former case $[a, b]$ is clearly non-empty since it includes $(a + b)/2$ or any other number between. In the latter case it remains non-empty as we have effectively applied the squeeze theorem, concluding $\bigcap_{n\in\mathbb{N}} I_n = [a, a]$ which is the singleton set $\{a\}$, containing itself and thus non-empty.

In fact this property, which once elaborated even a little seems obvious, is exactly what we need to complete two proofs promised in the previous section.

We had said that a sequence which is bounded has an infinite number of points to place and only a finite amount of space to put them, and so it should make sense that at least some subsequence of the set does converge. We had also said that it should make sense that Cauchy sequences converge, a fact which we will then be able to prove.

Theorem openlimsR.6 — (Bolzano Weierstraß)

Every bounded sequence has a converging subsequence.

It is my understanding that when you are unable to type the character "ß", the second most correct way to refer to this theorem is as Bolzano-Weierstrass. Not even wikipedia refers to it with this character, however it is common amongst mathematical texts to spell the theorem's name with the proper character.

Proof.

Let $(a_n)_{n\in\mathbb{N}}$ be our bounded seequence, and since it is bounded, there exists some $b \in \mathbb{R}$ such that $a_n \in [-b, b]$ for all $n \in \mathbb{N}$; this is not to imply that each a_n is particularly close to zero, but rather in the way that you would say a bounded sequence has $b_- \leq a_n \leq b_+$ for some much tighter bound, we simply take $b = \max\{|b_-|, |b_+|\}$ and in the same way $|a_n| \leq b$, we say $a_n \in [-b, b]$ by corollary openlimsR.3.

We now proceed by binary search, much in the same way one would search an ordered list, to find some infinite collection of points which can form a converging subsequence. Since $(a_n)_{n\in\mathbb{N}}$ is an infinite sequence, there are infinite elements of the sequence in $[-b, b]$ since all $a_n \in [-b, b]$. We call $I_0 = [-b, b]$.

Now one of the two intervals $[-b, 0]$ or $[0, b]$ will have the property that there does not exist a $N \in \mathbb{N}$ with the property that $n \geq N$ implies a_n is not in the interval, and we call this interval I_1. In other words, this interval I_1 which is either $[-b, 0]$ or $[0, b]$, has the property that there is no *final* $a_n \in I_1$, and so there does not exist a $N \in \mathbb{N}$ that would allow $n \geq N$ to imply $a_n \notin I_1$. This is since $(a_n)_{n\in\mathbb{N}}$ is infinite and must put its sequence elements somewhere, so I_1 is the interval out of the two that does continue indefinitely to contain elements a_n. It is our narrowing region of convergence.

Repeating this operation, assume we had picked $I_1 = [0, b]$ (although of course the operation is very similar if we had not), we then pick I_2 as either $[0, b/2]$ or $[b/2, b]$, depending on which interval has the property that there does not exist a rank $N \in \mathbb{N}$ after which $n \geq N$ fails to find any $a_n \in I_2$.

We are thus inductively defining a sequence of nested intervals $(I_n)_{n\in\mathbb{N}}$ where each I_n always fails to find a 'final' sequence element within it, and thus has infinite sequence elements within. Writing these intervals as $I_n = [x_n, y_n]$ we define the sequence of bounds $(x_n)_{n\in\mathbb{N}}$ and $(y_n)_{n\in\mathbb{N}}$, which are for example $x_1 = 0$, $x_2 = b/2$ and $y_1 = b$, $y_2 = b$, etc. with $x_n \leq y_n$ for all $n \in \mathbb{N}$. By the nested interval property, we know that $\lim_{n\to\infty} I_n$ is a non-empty set, so we choose some

$$c \in \left[\lim_{n\to\infty} x_n, \lim_{n\to\infty} y_n\right]$$

and define a corresponding subsequence $(c_n)_{n\in\mathbb{N}}$ such that each c_n is equal to the $a_m \in I_n$ with the smallest $m \geq n$ possible to satisfy $a_m \in I_n$. In this way we respect the subsequence property that the $\varphi\colon \mathbb{N} \to \mathbb{N}$ that defines $c_n = a_{\varphi(n)}$ is monotonic increasing, since the smallest $m \geq n$ such that $a_m \in I_n$ will either be excluded in I_{n+1}, be excluded since $m \geq n$ fails to be $m \geq n+1$, or simply remain in place until it is displaced as described.
Our goal now becomes to show that $c_n \to c$ as $n \to \infty$, since we have identified a subsequence and must now show that it is convergent. We will do this using the squeeze theorem, squeezing $(c_n)_{n\in\mathbb{N}}$ between $(x_n)_{n\in\mathbb{N}}$ and $(y_n)_{n\in\mathbb{N}}$.
Observe first that by construction, each I_{n+1} has length half that of I_n, as we saw that $I_0 = [-b, b]$ and we split the interval in half so that I_1 is either $[-b, 0]$ or $[0, b]$, etc. and since $I_n = [x_n, y_n]$, this means we may write

$$|x_n - y_n| = \frac{b}{2^{n-1}}$$

since $[-b, 0]$ and $[0, b]$ both have interval length b at I_1, we see as well here that setting $n = 1$ produces a length $|x_1 - y_1| = b$. But this is also

$$|x_n - y_n| = (2b)2^{-n}$$

and the right hand side should be familiar to us. In example **??** we showed that $2^{-n} \to 0$ as $n \to \infty$. By the algebraic limit theorem, we also know that $(2b)2^{-n} \to 0$ as $n \to \infty$ since we are merely scaling the limit by $2b$. This tells us that the distance between x_n and y_n converges to zero, or that in the limit they should be the same number, and thus $[x, y]$ is a singleton set, $\{c\}$, but we should show that explicitly.
We know that $x_n \leq y_n$ for all $n \in \mathbb{N}$ so $|x_n - y_n| = y_n - x_n$, and we know that y_n by construction is monotonic decreasing and x_n is monotonic increasing (once again, consider that the intervals are nested) so with $c \in [x, y]$, the limit of the interval-nesting, we may say $x_n \leq x \leq c \leq y \leq y_n$. This transitive chain implies that $x_n - c \geq 0$ and $y_n - c \leq 0$ (which should be obvious, since c is between x_n and y_n) but we may also consider this a different way:

$$0 \leq c - x_n \leq y_n - x_n$$
$$0 \leq y_n - c \leq y_n - x_n$$

and we know this to be true since $c \geq x_n$ (as above, implying the $0 \leq c - x_n$) and we may merely subtract x_n from both sides of $c \leq y_n$ (as above), and for the second inequality chain we may take $c \geq x_n$ and derive $-c \leq -x_n$, adding y_n to both sides. However we know that the expressions on the right $y_n - x_n$ converge to zero, so by squeeze theorem, we have $c - x_n \to 0$ and $y_n - c \to 0$ as $n \to \infty$, which by the algebraic limit theorem is the same as saying $x_n \to c$ and $y_n \to c$ as $n \to \infty$.
Now since $(c_n)_{n\in\mathbb{N}}$ is defined such that each c_n is chosen from $[x_n, y_n]$, we have by construction

$$x_n \leq c_n \leq y_n$$

for all $n \in \mathbb{N}$ with both $x_n \to c$ and $y_n \to c$ as $n \to \infty$. This satisfies the requirements for the squeeze theorem to say $c_n \to c$ as $n \to \infty$.

With this in hand, we will be able to apply Bolzano-Weierstraß together with the property of preservation of limits under subsequences and the boundedness of Cauchy sequences to say, hey,

a cauchy sequence is bounded so it has a convergent subsequence with some limit, and the Cauchy sequence must get infinitely close to the rest of itself and thus to its subsequence, which also converges. That means it gets infinitely close to the same limit as its subsequence and converges!

Theorem openlimsR.7

All Cauchy sequences converge.

Proof.

Let $(a_n)_{n\in\mathbb{N}}$ be our Cauchy sequence. By proposition seqlimsinR.11, it is bounded, and this satisfies the requirements for Bolzano-Weierstraß implying there is a subsequence $(b_n)_{n\in\mathbb{N}}$ of $(a_n)_{n\in\mathbb{N}}$ and we call this limit $b \in \mathbb{R}$. Our goal becomes to show that for all $\varepsilon_L > 0$ (L for limit) there exists $N_L \in \mathbb{N}$ such that $n \geq N_L$ implies $|a_n - b| \leq \varepsilon_L$.
To show this, let us start with the assumption that $(a_n)_{n\in\mathbb{N}}$ is Cauchy, and so for any $\varepsilon_C > 0$ (C for Cauchy) there exists $N_C \in \mathbb{N}$ such that $n, m \geq N_C$ implies $|a_n - a_m| \leq \varepsilon_C$, and the assumption that $(b_n)_{n\in\mathbb{N}}$ converges, for all $\varepsilon_b > 0$ there exists $N_b \in \mathbb{N}$ such that $n \geq N_b$ implies $|b_n - b| \leq \varepsilon_b$.
So assuming we are given some ε_L, set $\varepsilon_b, \varepsilon_C = \varepsilon_L/2$ and set $N_L = \max\{N_b, N_C\}$. This means that when $n, m \geq N_L$, we have both

$$|b_n - b| \leq \varepsilon_b = \frac{\varepsilon_L}{2},$$
$$|a_n - a_m| \leq \varepsilon_C = \frac{\varepsilon_L}{2}.$$

Moreover, since we may choose $m, n \geq N_L$, let us say that $m = \varphi(n)$ where $\varphi\colon \mathbb{N} \to \mathbb{N}$ is the monotonic increasing map that defines the subsequence $b_n = a_{\varphi(n)} = a_m$. This means that the latter inequality above is

$$|a_n - b_n| \leq \varepsilon_C = \frac{\varepsilon_L}{2}.$$

This is exactly what we need to apply the triangle inequality on $a_n, b_n, b \in \mathbb{R}$, yielding

$$|a_n - b| \leq |a_n - b_n| + |b_n - b|.$$

We'll skip the bells and whistles as we have gotten more familiar with them in previous proofs: since we know $|a_n - b_n| \leq \varepsilon_C$ and we know $|b_n - b| \leq \varepsilon_b$, we'll replace them both in one transitive step which is obvious to us now.

$$|a_n - b_n| + |b_n - b| \leq \varepsilon_C + \varepsilon_b.$$

But recall we set ε_C and ε_b to $\varepsilon_L/2$ so they sum together to form ε_L. This gives us the transitive chain

$$|a_n - b| \leq |a_n - b_n| + |b_n - b| \leq \varepsilon_C + \varepsilon_b = \varepsilon_L$$
$$|a_n - b| \leq \varepsilon_L$$

but this is exactly the form of the limit we wanted, so we are done.

It is worth noting how short this proof is, considering how important it will be to us and how we had first suspected this theorem might be true a whole section ago. In fact, most of the heavily lifting in this proof was done by applying theorems which were proved in the previous section, with the exception of Bolzano-Weierstraß. Our theory is developing to the point where we are able to rapidly make meaningful statements about our constructions using the framework we have built up rather than constantly resorting to first principles. This is in many ways the pattern of a mathematical theory. The pattern we see here will become less starkly apparent as we move on however, since as we build up different theories they will begin to refer to one another in a much more interconnected way rather than merely a single theory building on itself.

openlimsR.2 Open and Closed Sets and the Topological Limit

We have however only scratched the surface of our discussion of space, and we have tiptoed into it using the notion of an interval to describe the idea of a region. But the way in which we have called our intervals *open* or *closed* is a far more general property of regions than mere intervals. In fact they will continue to prove fundamental for many chapters to come.

Our exploration of openness and closedness will remain in $\mathbb{R}$ in this section as we use $\mathbb{R}$ as an example to demonstrate their properties and their uses. Quickly it will become apparent that the exclusion or inclusion of end-points of an interval are perhaps the least interesting things about open and closed sets. In fact, if we are to ask what are motivations are here, why we are investigating this notion of "openness" or "closedness" at all, it is because these properties turn out to be deeply intrinsically linked with the features of limits.

Definition openlimsR.8 — (Open and Closed Sets in $\mathbb{R}$)

Let $A \subseteq \mathbb{R}$ be some set. We say that A is an **open set** if for any point $a \in \mathbb{R}$, there exists $\varepsilon > 0$ such that the open interval $I = (a - \varepsilon, a + \varepsilon)$ is contained in A, $I \subseteq A$. When a set $B \subset \mathbb{R}$ has the property that there exists some open set $A \subseteq \mathbb{R}$ which is its set compliment, i.e.

$$B = \mathbb{R} \setminus A = \{b \in \mathbb{R} \mid b \notin A\}$$

then we call B a **closed set**.

Since closed sets here are defined by the space being absent an open set (indeed this was also noted when we mentioned the $]a, b[$ open interval notation), we shall focus on open sets first. Immediately we should look at this definition and ask if it generalizes what we want it to, i.e. are open intervals actually open sets? Well let's look closely at the definition.

The property stated is that an open set only contains points which it can place an open interval around within itself. At first this statement might sound strange, but remember that open intervals do not contain their end points. If we are to assume for a moment that open sets also do not contain endpoints, then any point we pick in an open interval is not itself an endpoint and thus not on the boundary. If it is within the boundary, then there are points between itself and the boundary which can itself form a bound for an interval which is contained in the set. Thus the intuition is as in the diagram below.

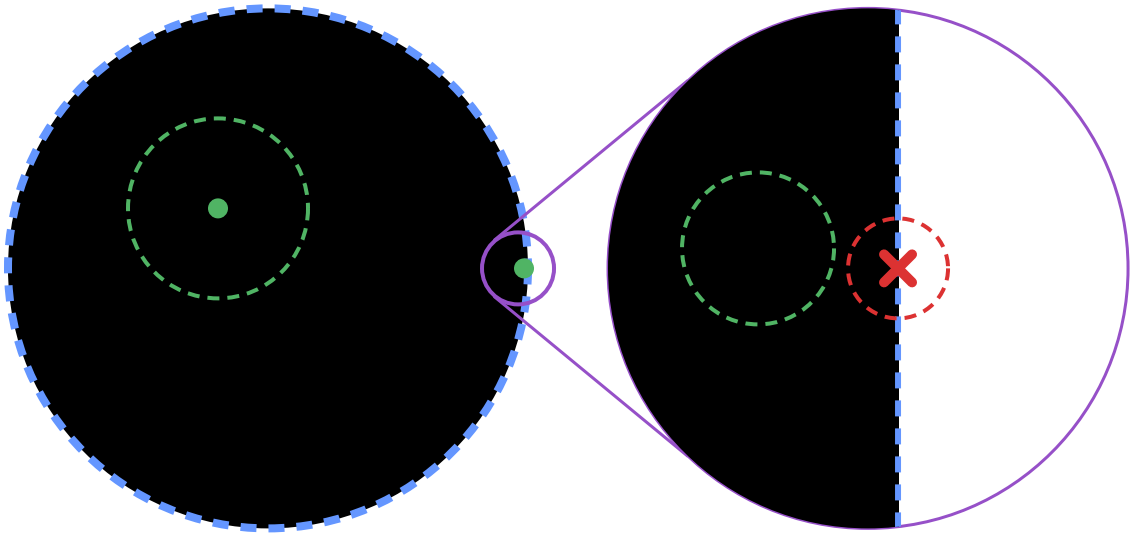

Figure 2.2: Our open set (left) contains only points for which we can find an open interval within the open set containing the
point. We also see inside it a point which looks very close to the boundary; zooming in there (right), we see that the point cannot be *on* the boundary, or else any open interval containing it would escape the open set. The point must be within the boundary, but it can be arbitrarily close to the edge, just never on the edge itself, and still have a contained open interval.

But we must not let ourselves think that this means open sets are merely open intervals. In $\mathbb{R}$ (and particular in only one dimension) this is *almost* true. The particular caveat we make is that, since an open set must only have the property above, that every point in the open set has an interval containing it which is itself contained in the open set, there is nothing stopping us from saying that an open set contains disconnected components. For instance, if we have four numbers $a, b, c, d \in \mathbb{R}$ with $a < b < c < d$ we may take the open intervals (a, b) and (c, d) and establish the open set $(a, b) \cup (c, d)$, the set containing all elements in either interval, and this is an example of an open set. The trick with this however is that open sets will sometimes be much stranger foam-like sets with perhaps infinite open intervals. The magic of this construction is that no matter how strange an open set we select, this particular statement of openness allows us to prove a lot of things without knowing the finer details of whatever infinite open-foam is going on in the open set. We simply know that every point in the set has an open interval around it.

We will be able to say much much more about closed sets shortly, but for now, since we have defined closed sets in $\mathbb{R}$ as the absence of an open set in $\mathbb{R}$, it is immediately apparent that since $(-\infty, a) \cup (b, \infty)$ is an open set, we also have its compliment $[a, b]$ the closed interval, as a closed set. Very roughly we can say that closed sets contain their end points in the way that closed intervals do, but the particular manner in which this works will have to be proven shortly.

Lemma openlimsR.9 — (Open Intervals are Open Sets and Closed Intervals are Closed Sets)

Let $a, b \in \mathbb{R}$ with $a < b$. Then (a, b) is an open set and $[a, b]$ is a closed set.

Proof.

A point $c \in (a,b)$ allows us to choose $\varepsilon = \min\{c-a, b-c\}$, i.e. the distance between c and the endpoint of the interval closer to c, thus having $(c-\varepsilon, c+\varepsilon) \subseteq (a,b)$. We should show this formally however, and to do that, we take the property of being in $(c-\varepsilon, c+\varepsilon)$, i.e. all $z \in (c-\varepsilon, c+\varepsilon)$ satisfy $c-\varepsilon < z < c+\varepsilon$ and show that it satisfies $a < z < b$. Let us treat the inequalities separately, as $c-\varepsilon < z$ and $z < c+\varepsilon$. Then we manipulate these inequalities as follows.

$$\begin{aligned} c-\varepsilon < z &\implies c-z < \varepsilon \\ z < c+\varepsilon &\implies z-c < \varepsilon. \end{aligned}$$

Using $\varepsilon = \min\{c-a, b-c\}$, we know $\varepsilon \leq c-a$ and $\varepsilon \leq b-c$, since ε is the minimum of the two and thus either equal in the inequality or less than it, we may apply transitivity using the two different inequalities,

$$\begin{aligned} c-z < \varepsilon \leq c-a \\ z-c < \varepsilon \leq b-c \end{aligned}$$

deriving

$$\begin{aligned} c-z < c-a &\implies a < z \\ z-c < b-c &\implies z < b \end{aligned}$$

as desired. So we have shown that $c-\varepsilon < z < c+\varepsilon$ implies $a < z < b$, meaning $(c-\varepsilon, c+\varepsilon) \subseteq (a,b)$.
Every closed interval $[a,b]$ also has the open set $A = (-\infty, a) \cup (b, \infty)$ so that $[a,b] = \mathbb{R} \setminus A$.

We must also acknowledge a sleight of hand. Our epsilon-N notion of a limit in the way we have been showing limits so far will begin to fail as we move deeper into mathematics, and that is because, in some sense, our definition is not the *true* one. We have relaxed a requirement, which is that we have said a limit is "for all $\varepsilon > 0$, there exists $N \in \mathbb{N}$ such that when $n \geq N$, we have $|a_n - a| \leq \varepsilon$" and this relaxation is to be found in our use of $\geq$ and $\leq$. In more general spaces, limits are proven using $>$ and $<$, so that the statement becomes "for all $\varepsilon > 0$, there exists $N \in \mathbb{N}$ such that when $n > N$, we have $|a_n - a| < \varepsilon$", and indeed the statement $|a_n - a| < \varepsilon$ is then the same as $a_n \in (a-\varepsilon, a+\varepsilon)$. We have *relaxed* this definition thus far because *we can*, that is, in $\mathbb{R}$ at least it turns out that these definitions are *equivalent*, and because being able to do this makes proofs much much easier. However this will begin to fail as we move into other spaces, and as it fails and we are forced to narrate limits not as $a_n \in [a-\varepsilon, a+\varepsilon]$ but as $a_n \in (a-\varepsilon, a+\varepsilon)$, excluding endpoints, a fundamental property of open sets will become apparent very slowly. That is, the idea that limits converge at all and what they converge to is ultimately a property *defined by* open sets. In fact this is what topology, absent any algebraic concerns, priniciply does for us in analysis, but it will take many more sections before we are able to see this in its full glory.

Theorem openlimsR.10 — (Characterizations of the Limit in $\mathbb{R}$)

Let $(a_n)_{n\in\mathbb{N}}$ be a sequence and $a \in \mathbb{R}$ a number. Then the following statements are equivalent.

- $\lim_{n\to\infty} a_n = a$
- for all $\varepsilon > 0$, there exists $N \in \mathbb{N}$ such that for all $n \geq N$ we have $|a_n - a| \leq \varepsilon$ (we will call this the *relaxed limit*)
- for all $\varepsilon > 0$, there exists $N \in \mathbb{N}$ such that for all $n > N$ we have $|a_n - a| < \varepsilon$ (we will call this the *proper limit*)
- for all open intervals (x, y) such that $a \in (x, y)$, there exists some $N \in \mathbb{N}$ such that for all $n > N$, we have $a_n \in (x, y)$ (we will call this the *region of convergence limit*)
- for all open sets $A \subseteq \mathbb{R}$ such that $a \in A$, there exists some $N \in \mathbb{N}$ such that for all $n > N$, we have $a_n \in A$ (we will call this the *topological limit*)

Proof.

The first statement of the list is purely notational, and is defined as equivalent to the relaxed limit. Indeed, once the proof is complete, we will interpret this notation as describing any of the other definitions of convergence. We will now need to move through this list proving that each one implies another and vice versa until we have formed a full chain of bidirectional implications. Note that this proof will use heavy implicit use of corollary openlimsR.3.

We'll begin by assuming the relaxed limit and trying to show the proper limit. That is, we presume that for all $\varepsilon_1 > 0$, there exists N_1 such that for all $n \geq N_1$ we have $|a_n - a| \leq \varepsilon_1$. We wish to prove that for all $\varepsilon_2 > 0$, there exists N_2 such that for all $n > N_2$, we have $|a_n - a| < \varepsilon_2$. So say we are given some ε_2. Set $\varepsilon_1 = \varepsilon_2/2$ so that it implies the existence of N_1, and choose $N_2 = N_1$. Since $\varepsilon_2 > 0$, it has the property that $\varepsilon_2/2 < \varepsilon_2$, that is, it is positive so its half is smaller than its whole. When we assume $n \geq N_2$, since $N_2 = N_1$, we have $|a_n - a| \leq \varepsilon_1$, but by construction this is

$$|a_n - a| \leq \varepsilon_1 = \frac{\varepsilon_2}{2} < \varepsilon_2$$
$$|a_n - a| < \varepsilon_2$$

as desired.

To prove the opposite direction, let us reverse our assumptions. Say we are given ε_1 and must find N_1 with the appropriate properties to satisfy the relaxed limit. Set $\varepsilon_2 = \varepsilon_1$ so that it induces N_2 and set $N_1 = N_2 + 1$. This means that $N_1 > N_2$, and in particular the property $n \geq N_1$ is $n \geq N_1 > N_2$ so $n > N_2$ is satisfied. When $n > N_2$ is satisfied, we have $|a_n - a| < \varepsilon_2 = \varepsilon_1$, but this is the same as $a_n \in (a - \varepsilon_1, a + \varepsilon_1)$. Obviously, this open interval is a subset of the corresponding closed interval, since the interval without endpoints fits inside the interval with its endpoints. This means we have implied $a_n \in [a - \varepsilon_1, a + \varepsilon_1]$, but that is the same as saying $|a_n - a| \leq \varepsilon_1$, which is what we wanted to prove.

For the next step, we show that the the proper limit is equivalent to the region of convergence limit. We'll reuse our numbers, saying that we wish to show "for all $\varepsilon_1 > 0$ there exists $N_1 \in \mathbb{N}$ such that for all $n > N$ we have $|a_n - a| < \varepsilon_1$" is the same as "for all open intervals (x, y) such that $a \in (x, y)$, there exists some $N_2 \in \mathbb{N}$ such that for all $n > N_2$, we have $a_n \in (x, y)$". So presume the former and we will show the latter. That is, assume we are given some (x, y) satisfying $a \in (x, y)$. Now choose $\varepsilon_1 = \min\{a - x, y - a\}$, since a is $x < a < y$, so these numbers $a - x$ and $y - a$ are both positive, and in particular

choosing the smaller of them in this way will mean that

$$(a-\varepsilon_1, a+\varepsilon_1) \subseteq (x,y)$$

We should prove this however. And the particular form of this proof will be that all $z \in (a-\varepsilon_1, a+\varepsilon_1)$ must also be $z \in (x,y)$. That is, we prove that any number satisfying the condition defining the subset satisfies the condition defining the proposed superset. First consider that

$$(a-\varepsilon_1, a+\varepsilon_1) = (a-\varepsilon_1, \infty) \cap (-\infty, a+\varepsilon_1)$$

in the sense that the restriction $a-\varepsilon_1 < z < a+\varepsilon_1$ may be decomposed into the two restrictions $a-\varepsilon_1 < z$ and $z < a+\varepsilon_1$. Since we define $\varepsilon_1 = \min\{a-x, y-a\}$, we have $\varepsilon_1 \leq a-x$ and $\varepsilon_1 \leq y-a$, since it is either equal to either of them or less than either one because it is the other one. So manipulating these individual conditions,

$$\begin{aligned} a-\varepsilon_1 < z &\implies a-z < \varepsilon_1 \\ z < a+\varepsilon_1 &\implies z-a < \varepsilon_1 \end{aligned}$$

we may apply transitivity using the two different inequalities,

$$\begin{aligned} a-z < \varepsilon_1 \leq a-x \\ z-a < \varepsilon_1 \leq y-a \end{aligned}$$

deriving

$$\begin{aligned} a-z < a-x &\implies x < z \\ z-a < y-a &\implies z < y \end{aligned}$$

and thus putting them together, we have shown $a-\varepsilon_1 < z < a+\varepsilon_1$ implies $x < z < y$. The corresponding statement on sets is that $(a-\varepsilon_1, a+\varepsilon_1) \subseteq (x,y)$ as desired. This means that $a_n \in (a-\varepsilon_1, a+\varepsilon_1)$, the same statement as $|a_n - a| < \varepsilon_1$ is the same as $a_n \in (x,y)$. But then if we choose the N_2 that we need to show exists to be $N_2 = N_1$, we have that for all $n > N_1 = N_2$, we have $|a_n - a| < \varepsilon_1$ which also implies $a_n \in (x,y)$ as desired. To do the proof in the converse case is even easier, since we may pick any interval (x,y) containing a and find N_2 such that $n > N_2$ implies $a_n \in (x,y)$. So say we are given ε_1, we may simply pick $x = a-\varepsilon_1$ and $y = a+\varepsilon_1$. Choosing $N_1 = N_2$, the statement $n > N_1$ implies $|a_n - a| < \varepsilon_1$ is now immediately the same as the statement we already know to be true, that $n > N_2$ implies $a_n \in (x,y) = (a-\varepsilon_1, a+\varepsilon_1)$ by corollary openlimsR.3.
For the final step of the proof, we must show that the region of convergence limit is the same as the topological limit. So we want to show that "for all open intervals (x,y) such that $a \in (x,y)$, there exists some $N_1 \in \mathbb{N}$ such that for all $n > N_1$, we have $a_n \in (x,y)$" is the same as "for all open sets A such that $a \in A$, there exists some $N_2 \in \mathbb{N}$ such that for all $n > N_2$, we have $a_n \in (x,y)$". So we'll presume that the sequence converges under the region of convergence limit and show that it converges under the topological limit. So say we are given some open set A with $a \in A$; since it is an open set, there exists an open interval with $a \in I$ and $I \subseteq A$. So we'll take this interval $(x,y) = I$ to be the open interval for the region of convergence limit, and thus imply a N_1 such that $n > N_1$

implies $a_n \in (x, y) = I \subseteq A$, and thus $a_n \in A$. So setting $N_2 = N_1$, we have $n > N_2 = N_1$ implies $a_n \in A$ exactly as we wanted.
In the other direction, our proof is similarly simple. Since all open intervals are open sets, when we are given some (x, y) with $a \in (x, y)$, we immediately set $A = (x, y)$ to induce some N_2 where $n > N_2$ implies $a_n \in A$. Setting $N_1 = N_2$, we have $n > N_1$ implying $a_n \in (x, y)$ since $a_n \in A = (x, y)$.

This proof in some sense shows how all along our use of ε in proofs as a bound, a way of saying 'the distance between a_n and a is at least ε small' was in fact a proxy for speaking about open sets. In the region of convergence limit and topological limit we see that we can similarly define limits not with a notion of 'small' defined by *distance* but rather by choosing our own *small set*. Recalling our earlier analogy about limits as checking a property with a measurement instrument at some resolution corresponding to ε, we lose the more obvious manner of improving our precision where we simply make ε *smaller* (although indeed we can still simply scale down an open set) and instead gain the ability to check the precision in arbitrary other ways. For instance, where previously we had $|a_n - a| \leq \varepsilon$ meaning that $a_n \in (a - \varepsilon, a + \varepsilon)$, this region of convergence could only be defined as centered around a, buffered by ε on either side, but our region of convergence limit shows that in fact a does not need to be the center of the interval, it only needs to be in the interval.

But most of all, this proof should perhaps inform us what is meant by an open set in general. In some sense, what we have shown above, the notion of open sets as a region within which a sequence converges, could be argued in a sense to be what open sets are *for*. Indeed we will show in a later section that in other kinds of spaces where notions of ordering or other ways of structuring the space no longer make sense, the notion of an open set still exists such that it remains meaningful to take limits. If someone asks you 'what *is* an open set' and you simply must reply in only two seconds, your mind should go to 'a set that can be used to define convergence'. And even when our notions of what sets count as open are radically distorted in future exampls, our definition of which sequences converge and to what limits will still be defined by those distorted notions of open sets, precisely because which sets count as open define limits in abstract.

Now, if we are to say that what we have just done is demonstrated the fundamental purpose of open sets, then it should make sense to immediately follow with the fundamental purpose of clsoed sets as we promised earlier. More specifically, if open sets are sets containing some limit point and the tail of an infinite sequence converging to that limit point, closed sets are sets which contain the limits corresponding to each and every sequence contained within them. If we take this to be what we think closed sets *are*, it immediately follows why closed intervals must contain their endpoints, since any monotone increasing sequence in the interval bounded by it must converge to the supremum, which must then be included, and vice versa for the infimum. You could also say that this is the reason why these sets are called *closed*, not in opposition to open sets, but in the same sense that a set A may be *closed under addition*, meaning that $a, b \in A$ implies $a + b \in A$, closed sets are *closed under limits*, in that any sequences in the set converges to a point in the limit.

Proposition openlimsR.11 — (Closed Sets are Closed Under Limits in $\mathbb{R}$)

Let $B \subset \mathbb{R}$ be a set. We say that a point $a \in B$ is a **limit point** of B if there exists a sequence $(a_n)_{n \in \mathbb{N}}$ with $a_n \in B$ for all $n \in \mathbb{N}$ with $a = \lim_{n \to \infty} a_n$.
B is then closed if and only if it contains all of its limit points.

Proof.

To prove that closed sets contain their limit points, we must proceed by contradiction. If B is closed, then there exists an open set $A = \mathbb{R} \setminus B$ which is open by definition. what we want to prove is that all sequences $(a_n)_{n\in\mathbb{N}}$ which are $a_n \notin A$ (and thus $a_n \in B$) have limits $a \notin A$ (and thus $a \in B$, meaning it contains its limit points) so our proof by contradiction will presume that there exists a sequence $(a_n)_{n\in\mathbb{N}}$ with $a_n \in B$ converging to $a \in A$ (a limit point which is not in B).

In fact the violation of a known fact occurs immediately. When we described the topological limit it was that for any open set A containing the limit a, there exists a $N \in \mathbb{N}$ such that for all $n > N$, we have $a_n \in A$. And yet we have presumed that $a_n \notin A$ for all $n \in \mathbb{N}$, so there cannot exist any $N \in \mathbb{N}$ with that property, violating the assumption that $a_n \to a$ as $n \to \infty$. So such a sequence cannot exist, and all sequences in B must converge in B.

For the converse, we proceed by proof of contradiction again. That is, we are trying to prove that if B contains all its limit points, then the set $A = \mathbb{R} \setminus B$ is an open set. So we'll presume for the sake of contradiction that A is not open, i.e. there exists a point $z \in A$ for which no open interval centered on z is contained in A, or equivalently, for every interval $I = (z - \varepsilon, z + \varepsilon)$, the intersection $I \cap B$ is nonempty.

Let $(x_n)_{n\in\mathbb{N}}$ and $(y_n)_{n\in\mathbb{N}}$ be sequences that both converge to z with $z - x_n = y_n - z$, in effect defining by proxy a sequence $(\varepsilon_n)_{n\in\mathbb{N}}$ defining the nested intervals $(z-\varepsilon_n, z+\varepsilon_n) = (x_n, y_n)$, each candidate intervals which we would say are in A if A were open. As above, we stated that every interval containing z intersects B, so every $(x_n, y_n) \cap B$ is non-empty, but this means that from each non-empty set $(x_n, y_n) \cap B$ we may choose some $a_n \in (x_n, y_n) \cap B$ to define a sequence $(a_n)_{n\in\mathbb{N}}$ which must also converge to z due to the squeeze theorem (namely since $x_n < a_n < y_n$ and $x_n, y_n \to z$ as $n \to \infty$). What we have constructed then is a sequence $(a_n)_{n\in\mathbb{N}}$ which is $a_n \in B$ (since $a_n \in (x_n, y_n) \cap B$, i.e. $\big(a_n \in (x_n, y_n)\big) \wedge (a_n \in B)$) for all $n \in \mathbb{N}$ yet converges to $z \in A = \mathbb{R} \setminus B$, i.e. $z \notin B$, contradicting the assumption that B contains all its limit points. We conclude that B must be closed.

Understanding this property, that closed sets are sets which are *closed under limits*, and that open sets are sets which sequences converge into and that contain intervals around their points, we can now ask if open and closed sets are even opposites. Obviously they are defined in such a way that a closed set is the absence of an open set, and yet these properties seem somewhat disconnected from each other. It is worth noting that it is indeed possible for a set to be either open and closed or neither open nor closed; the archetypal example of an open and closed set (colloquially **clopen**) is $\mathbb{R}$ itself: just as we think of $\{a\} = [a, a]$ as a closed interval, we think of $\{\} = \emptyset = (a, a)$, the empty set, as open, thus $\mathbb{R} = \mathbb{R} \setminus \emptyset$ must be closed. Yet we also think of $\mathbb{R} = (-\infty, \infty)$ so we think of it as open. Both $\emptyset$ and $\mathbb{R}$ are then clopen sets. For a set which is neither open nor closed, take the rational numbers as a subset of the real numbers. The rational numbers are a collection of infinite *points* with no intervals, and so it cannot be open, and yet as discussed earlier, it does not contain limit points such as irrational numbers, so it is neither open nor closed.

openlimsR.3 Section Appendix: Some musings on Countability and Uncountability

(rewrite first draft here)