Notes on Category Theory

Indrjo Dedej

Last revision: 5th July 2022.

Contents

0

Symbolism

0.1 Functions

If f is a name of a function, we write $f(x)$ the image of x. However, we may find ourselves writing fx or f_x to avoid an excessive usage of brackets.

A literary device — to be used sparingly, though — we often use is about functions $f : A \times B \to C$: for $x \in A$, we introduce the function

$$f(x, \quad) : B \to C$$
$$f(x, \quad)(y) := f(x, y).$$

The idea is to 'hold' the first variable to some value and let the second one vary: this is done by leaving a blank space to be filled with values from B. Obviously, for $y \in B$, we introduce the function

$$f(\quad, y) : A \to C$$
$$f(\quad, y)(x) := f(x, y).$$

If it does not create problems, symbols like $\bullet$, $\cdot$ and $-$ could be employed instead of leaving an empty space. You may find written $f(x, \bullet)$, $f(x, \cdot)$ or $f(x, -)$ for example.

Consider for instance the function from $\mathbb{R}$ to $\mathbb{R}$ that takes real numbers to their square may be denoted by $(\quad)^2$. Similarly, the function that takes $x \in \mathbb{R}$ to $e^x \in \mathbb{R}$ my be written as $e^\bullet$.

Another way to introduce functions is via *lambdas*: if you have a (well formed) formula Φ which may contain a variable x, you can provide the thing

$$\lambda x.\Phi$$

called *lambda abstraction*. You can 'pass values' to such things: denote by $(\lambda x.\Phi)(v)$ is the formula Φ with all the occurrences of x replaced by v. The notation $\lambda x.\Phi$ is exactly like the more familiar

$$\exists x : \Phi, \quad \lim_{x \to 0} f(x) \text{ or } \int_a^b f(x) \mathrm{d}x.$$

It is not important the choice of the letter x: you can replace x by another symbol, but remember to substitute the x-s occurring in Φ. However, this is not devoid of troubles: do not use a symbol that is already *free* in Φ, viz is not between the λ and the symbol of dot. Take $x + y$ which gives the abstraction $\lambda x.x + y$: we are sure you agree with us that $\lambda y.y + y$ is not the same thing.

So, what is the benefit for us? For example, $n + 1$ gives arise to the function

$$(\lambda n.n + 1) : \mathbb{N} \to \mathbb{N}$$
$$(\lambda n.n + 1)(k) := k + 1.$$

1

Categories

1.1 Definition

Let us start with some examples you should be familiar with.

Example 1.1.1 (Set Theory)**.** Here we have *sets* and *functions*. Whereas these of set and membership are assumed as primitive, the concept of function has a precise definition:

> For A and B sets, a function from A to B is any $f \subseteq A \times B$ such that for every $x \in A$ there exists one and only one $y \in B$ such that $(x, y) \in f$.

We write '$f : A \to B$' to mean 'f is a function from A to B'; for $x \in A$, we denote by $f(x)$ the element of B bound to x by f. Observe the following ways to introduce a function are equivalent:

- telling the pairs that make f;
- for every $x \in A$ saying which is the $y \in B$ such that $(x, y) \in f$; I'm sure you are pretty used to introduce functions by writing something like

$$f(x) := \text{formula that may contain } x.$$

That being said, let us deal with the operation of composing consecutive functions: for A, B and C sets and functions

$$A \xrightarrow{f} B \xrightarrow{g} C,$$

the *composite* of g and f is the function posed in this way:

$$g \circ f : A \to C$$
$$g \circ f(x) := g\big(f(x)\big).$$

Such operation has some remarkable properties.

1. For every set A the function $1_A : A \to A$ defined by $1_A(x) = x$ is such that for every set B and function $g : B \to A$ we have
$$1_A \circ g = g$$
and for every set C and function $h : A \to C$ we have
$$h \circ 1_A = h.$$
Here, 1_A is the *identity* for A.
2. $\circ$ is associative, that is for A, B, C and D sets and
$$A \xrightarrow{f} B \xrightarrow{g} C \xrightarrow{h} D$$
functions, we have the identity
$$(h \circ g) \circ f = h \circ (g \circ f).$$

Example 1.1.2 (Topology)**.** A *topological space* is a set where some of its subsets have the status of 'open sets'; *continuous functions* are set functions that care about such label: that is for X and Y topological spaces a function $f : X \to Y$ is said *continuous* iff for every open set U of Y the set $f^{-1}U$ is an open set of X. In general, for X, Y and Z sets and $f : X \to Y$ and $g : Y \to Z$ functions, we have
$$(g \circ f)^{-1}U = f^{-1}\left(g^{-1}U\right) \text{ for every } U \subseteq Z$$
Now, if X, Y and Z are topological spaces and f and g continuous, for if U is open, then so is $(g \circ f)^{-1}U$: that is $g \circ f$ is continuous as well. Being continuous functions functions, the associativity comes for free; moreover, the identity functions are continuous. Take the properties listed in the previous example and replace 'set' with 'topological space' and 'function' with 'continuous function' and notice how things work fine.

Now it's time to give a definition of what we have been highlighting so far.

Definition 1.1.3 (Categories)**.** A *category* amounts at assigning some things called *objects* and, for each couple of objects a and b, of some other things named *morphisms* from a to b. We write $f : a \to b$ to say that f is a morphism from a to b, where a is the *domain* of f and b the codomain. Besides, for a, b and c objects and $f : a \to b$ and $g : b \to c$ morphisms, there is associated the *composite morphism*
$$gf : a \to c.$$
All those things are regulated by the following axioms:

1. for every object x there is a morphism, 1_x, from x to x such that for every object y and morphism $g : y \to x$ we have
$$1_x g = g$$
and for every object z and morphism $h : x \to z$ we have
$$h 1_x = h;$$
2. for a, b, c and d objects and morphisms
$$a \xrightarrow{f} b \xrightarrow{g} c \xrightarrow{h} d$$
we have the identity
$$(hg)f = h(gf).$$

Sometimes, instead of 'morphism' you may find written 'map' or 'arrow'. The former is quite used outside Category Theory, whereas the latter refers to the fact that '$\to$' is employed.

Exercise 1.1.4 (Set Theory, again)**.** Take sets and relations. For A and B sets, by relation from A to B we mean a subset of $A \times B$. Define the composition of two relations — the first example may suggest to you the 'right' way to proceed. Can you stand some properties out akin to the case of sets and functions?

Exercise 1.1.5 (Topology, again)**.** This is a little variation of the previous example. Take topological spaces and *open functions*: now, can you stand out some pattern, like that of the previous two examples? Here, for X and Y topological spaces, a function $f : X \to Y$ is said open whenever for every open set $U \subseteq X$ the set $fU \subseteq Y$ is open too.

Exercise 1.1.6. At this early stage, one can provide a generous amount of examples of contexts presenting that leitmotif when things are built upon Set Theory. After all, groups are sets with some additional ingredients, homomorphisms are functions that cares about the group structure, composing two such functions yields a homomorphism; and composition complies the same laws highlighted in the previous two examples. The same applies to vector spaces, measure spaces, probability spaces, ... Make some examples by yourself.

The above definition has the aim to generalize the 'objects-morphisms-compositionality' pattern to a broader class of situations.

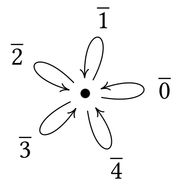


Figure 1.1. The group $\mathbb{Z}_5$ pictured as a category. Actually, you do not need to represent a monoid in this way; the picture is just to give a visual representation of the shift required by Category Theory.

Example 1.1.7 (Monoids are categories)**.** Consider a category $\mathcal{G}$ with a single object, that we indicate with a bare $\bullet$. All of its morphisms have $\bullet$ as domain and codomain: this fact implies the composite of two morphisms $\bullet \to \bullet$ is a morphism $\bullet \to \bullet$ too. This motivates us to proceed as follows: let G be the collection of the morphisms of $\mathcal{G}$ and consider the function

$$G \times G \to G,\ (x, y) \to xy,$$

that is the operation of composing morphisms. Being $\mathcal{G}$ a category implies this function is associative and $\mathcal{G}$ has the identity of $\bullet$, that is G has one element we call 1 and such that $f1 = 1f = f$ for every $f \in G$. In other words, we are saying G is a monoid.
Conversely, take a monoid G and any thing you want: make such thing acquire the status of object and the elements of G that of morphisms; in that case, the operation of G has the right to be called composition because the axioms of monoid say so.
The conclusion is: a monoid ‘is’ a category with a single object. $\bullet$ is something we cared of only because by definition morphisms need objects and it has no role other than this.

In Mathematics, a lot of things are monoids, so this is nice.

Example 1.1.8 (Preordered sets are categories)**.** A *preordered set* (sometimes contracted as *proset*) consists of a set A and a relation $\leq$ on A such that:

1. $x \leq x$ for every $x \in A$;
2. for every $x, y, z \in A$ we have that if $x \leq y$ and $y \leq z$ then $x \leq z$.

Now we do this: for $x, y \in A$, whenever $x \leq y$ take $(a, b) \in A \times A$. We operate with these couples as follows:

$$(y,z)(x,y) := (x,z), \tag{1.1.1}$$

where $x, y, z \in A$. This definition is perfectly motivated by (2): in fact, if $x \leq y$ and $y \leq z$ then $x \leq z$, and so there is (x, z). By (1), for every $x \in A$ we have the couple (x, x), which has the following property: for every $y \in A$

$$\begin{aligned} (x,y)(x,x) &= (x,y) \quad \text{for every } y \in A \\ (x,x)(z,x) &= (z,x) \quad \text{for every } z \in A. \end{aligned} \tag{1.1.2}$$

Another remarkable feature is that for every $x_1, x_2, x_3, x_4 \in A$

$$\big((x_3,x_4)(x_2,x_3)\big)(x_1,x_2) = (x_3,x_4)\big((x_2,x_3)(x_1,x_2)\big) \tag{1.1.3}$$

We have a category indeed: its objects are the elements of A, the morphisms are the couples (x, y) such that $x \leq y$ and (1.1.1) gives the notion of composition; (1.1.2) says what are identities while (1.1.3) tells the compositions are associative.

A lot of things are prosets, so this is nice.

Example 1.1.9 (Matrices). We need to clarify some terms and notations before. Fixed some field k, for m and n positive integers, a *matrix* of type $m \times n$ is a table of elements of k arranged in m rows and n columns:

$$\begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,n} \\ \vdots & \vdots & & \vdots \\ x_{m,1} & x_{m,2} & \cdots & x_{m,n} \end{pmatrix}$$

If A is the name of a matrix, then $A_{i,j}$ is the element on the intersection of the ith row and the jth column. Matrices can be multiplied: if A and B are matrices of type $m \times n$ and $n \times r$ respectively, then AB is the matrix of type $m \times r$ where

$$(AB)_{i,j} := \sum_{p=1}^{n} A_{i,p} B_{p,j}.$$

Our experiment is this: consider the positive integers in the role of objects and, for m and n integers, the matrices of type $m \times n$ as morphisms from n to m; now, take AB as the composition of A and B. Let us investigate whether categorial axioms hold.

- For n positive integer, we have the *identity matrix* I_n, the one of type $n \times n$ defined by

  $$(I_n)_{i,j} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases}$$

  One, in fact, can verify that such matrix is an 'identity' in categorial sense: for every positive integer m, an object, and for every matrix A of type $m \times n$, a morphism from n to m, we have

  $$AI_n = A,$$

  that is composing A with I_n returns A; similarly, for every positive integer r and for every matrix B of type $r \times n$ we have

  $$I_n B = B.$$

- For A, B and C matrices of type $m \times n$, $n \times r$ and $r \times s$ respectively, we have

  $$(AB)C = A(BC).$$

  Again, this identity can be regarded under a categorial light.

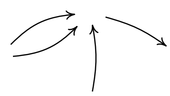
The category of matrices on a field k just depicted is written $\mathbf{Mat}_k$.

Remark 1.1.10. Though the previous example may seem quite useless, it really does matter. Just wait until we talk about equivalence of categories.

Example 1.1.11 (Chain complexes)**.** [...]

1.2 The language of diagrams

A diagram is a drawing made of 'nodes', that is empty slots, and 'arrows', that part from some nodes and head to other ones. Here is an example:



(1.2.4)

Nodes are the places where to put objects' names and arrows are to be labelled with morphisms' names. The next step is putting labels indeed, something like this:

$$\begin{tikzcd} & b \arrow[dr, bend left, "q"] & \\ a \arrow[ur, bend left, "f"] \arrow[ur, bend right, "g"'] & & d \\ & c \arrow[u, "h"'] & \end{tikzcd} \tag{1.2.5}$$

The idea we want to capture is: having a scheme of nodes and arrows, as in (1.2.4), and then assigning labels, as in (1.2.5). Since diagrams serve to graphically show some categorial structure, there should exist the possibility to 'compose' arrows: two consecutive arrows

$$\begin{tikzcd} {} \arrow[r] & {} \arrow[r] & {} \end{tikzcd} \tag{1.2.6}$$

naturally yields that one that goes from the first node and heads to the last one; if in (1.2.6) we label the arrows with f and g, respectively, then the composite arrow is to be labelled with the composite morphism gf. That operation shall be associative and there should exist identity arrows too, that is arrows that represent and behave exactly as identity morphisms. In other words, our drawings shall care of the categorial structure.

If we want to formalize the idea just outlined, the definition of diagram sounds something like this:

Definition 1.2.1 (Diagrams)**.** A *diagram* in a category $\mathcal{C}$ is having:

- a scheme of nodes and arrows, that is a category $\mathcal{I}$;
- labels for nodes, that is for every object i of $\mathcal{I}$ one object x_i of $\mathcal{C}$;
- labels for arrows, that is for every pair of objects i and j of $\mathcal{I}$ and morphism $\alpha : i \to j$ of $\mathcal{I}$, one morphism $f_\alpha : x_i \to x_j$ of $\mathcal{C}$

with all this complying the following rules:

1. $f_{1_i} = 1_{x_i}$ for every i object of $\mathcal{I}$;
2. $f_\beta f_\alpha = f_{\beta\alpha}$, for α and β two consecutive morphisms of $\mathcal{I}$.

[A finer formalisation of commutativity?] Consecutive arrows form a 'path'; in that case, we refer to the domain of its first arrow as the domain of the path and to the codomain of the last one as the codomain of the path. Two paths are said *parallel* when they share both domain and codomain. A diagram is said to be *commutative* whenever any pair of parallel paths yields the same composite morphism.

Example 1.2.2 (Identities via commutative diagrams)**.** Let $\mathcal{C}$ be a category and x an object of $\mathcal{C}$. The fact that 1_x the identity of x can be translated as follows: the diagrams

$$\begin{tikzcd} & x \arrow[dd, "1_x"] \\ a \arrow[ur, "f"] \arrow[dr, "f"'] & \\ & x \end{tikzcd} \qquad \begin{tikzcd} x \arrow[dd, "1_x"'] \arrow[dr, "g"] & \\ & b \\ x \arrow[ur, "g"'] & \end{tikzcd} \tag{1.2.7}$$

commute for every a and b objects and f and g morphisms of $\mathcal{C}$.

Example 1.2.3 (Associativity via commutative diagrams)**.** Associativity can be rephrased by saying:

$$\begin{tikzcd} a \arrow[r, "f"] \arrow[dr, "gf"'] & b \arrow[dr, "hg"] & \\ & c \arrow[r, "h"'] & d \end{tikzcd}$$

commutes for every a, b, c and d objects and f, g and h morphisms in $\mathcal{C}$.

1.3 Isomorphisms

[This section has to be rewritten.]

Let us step back to the origins. The categorial axioms state identities that deals with morphisms, since equality between morphisms is involved. For that reason, we shall regard these axioms as ones about morphisms, since objects barely appear as start/end point of morphisms.

Thus categories have a notion of sameness between morphisms, the equality, but nothing is said about objects. Of course, if a category has also the equality for objects, it is fine, but we can craft a better notion of sameness of objects. Not because equality is bad, but we shall look for something that can be stated solely in categorial terms.

As usual, simple examples help us to isolate this notion.

Example 1.3.1 (Set Theory, equinumerousity)**.** Cantor, the father of Set Theory, conducted its enquiry on cardinalities and not on equality of sets. For A and B sets, the following statements are equivalent:

1. there exists a bijective function $A \to B$;
2. there exist two functions $\begin{tikzcd} A \arrow[r, bend left, "f"] & B \arrow[l, bend left, "g"] \end{tikzcd}$ such that $gf = 1_A$ and $fg = 1_B$.

Though they are logically equivalent, they differ in some sense. In Set Theory, the adjective 'bijective' is defined by referring of the fact that sets are things that have elements:

for every $y \in B$ there is one and only one $x \in A$ such that $f(x) = y$.

In contrast, (2) is a statement written in terms of functions and compositions of functions: so (2) is written in a categorial language.

Exercise 1.3.2. Demonstrate the equivalence above.

This is enough to formulate a definition.

Definition 1.3.3 (Isomorphic objects)**.** Let $\mathcal{C}$ be a category and a and b two of its objects. A morphism $f : a \to b$ of $\mathcal{C}$ is an *isomorphism* whenever there is in the same category a morphism $g : b \to a$ such that $gf = 1_a$ and $fg = 1_b$. In that case, a is said *isomorphic* to b when there is an isomorphism $a \to b$ in $\mathcal{C}$.

Definition 1.3.4 (Skeletal categories)**.** [...]

Example 1.3.5 (Inverse matrices)**.** [Yet to be TEX-ed...]

Definition 1.3.6 (Skeleton)**.** [...]

1.4 Duality

For $\mathcal{C}$ a category, its *dual* (or *opposite*) category is denoted $\mathcal{C}^{\mathrm{op}}$ and is described as follows. Here, the objects are the same of $\mathcal{C}$ and 'being a morphism $a \to b$' exactly means 'being a morphism $b \to a$ in $\mathcal{C}$'. In other words, passing from a category to its dual leaves the objects unchanged, whereas the morphisms have their verses reversed. To dispel any ambiguity, by 'reversing' the morphisms we mean that morphisms $f : a \to b$ of $\mathcal{C}$ can be found among the morphisms $b \to a$ of $\mathcal{C}^{\mathrm{op}}$ and, vice versa, morphisms $a \to b$ of $\mathcal{C}^{\mathrm{op}}$ among the morphisms $b \to a$ of $\mathcal{C}$. Nothing is actually constructed out of the blue. Some authors suggest to write f^{op} to indicate that one f once it has domain and codomain interchanged, but we do not do that here, because they really are the same thing but in different places. So, if f is the name of a morphism of $\mathcal{C}$, the name f is kept to indicate that morphism as a morphism of $\mathcal{C}^{\mathrm{op}}$; obviously, the same convention applies in the opposite direction. It may seem we are going to nowhere, but it makes sense when it comes to define the compositions in $\mathcal{C}^{\mathrm{op}}$: for $f : a \to b$ and $g : b \to c$ morphsisms of $\mathcal{C}^{\mathrm{op}}$ the composite arrow is so defined

$$gf := fg.$$

This is not a commutative property, though. Such definition is to be read as follows. At the left side, f and g are to be intended as morphisms of $\mathcal{C}^{\mathrm{op}}$ that are to be composed therein. Then the composite gf is calculated as follows:

1. look at f and g as morphisms of $\mathcal{C}$ and compose them as such: so $f : b \to a$ and $g : c \to b$ and $fg : c \to a$ according to $\mathcal{C}$;
2. now regard fg as a morphism of $\mathcal{C}^{\mathrm{op}}$: this is the value gf is bound to.

Let us see now whether the categorial axioms are respected. For x object of $\mathcal{C}^{\mathrm{op}}$ there is 1_x, which is a morphism $x \to x$ in either of $\mathcal{C}$ and $\mathcal{C}^{\mathrm{op}}$. For every object y and morphism $f : y \to x$ of $\mathcal{C}^{\mathrm{op}}$ we have

$$1_x f = f 1_x = f.$$

Similarly, we have that

$$g 1_x = g$$

for every object z and morphism $g : x \to z$ of $\mathcal{C}^{\mathrm{op}}$. Hence, 1_x is an identity morphism in $\mathcal{C}^{\mathrm{op}}$ too. Consider now four objects and morphisms of $\mathcal{C}^{\mathrm{op}}$

$$a \xrightarrow{f} b \xrightarrow{g} c \xrightarrow{h} d$$

and let us parse the composition

$$h(gf).$$

In $h(gf)$ regard both h and gf as morphisms of $\mathcal{C}$. In that case, $h(gf)$ is exactly $(gf)h$, where gf is fg once f and g are taken as morphisms of $\mathcal{C}$ and composed there. So $h(gf) = (fg)h$, where at left hand side compositions are performed in $\mathcal{C}$: being the composition is associative, $h(gf) = (fg)h = f(gh)$. We go back to $\mathcal{C}^{\mathrm{op}}$, namely $f(gh)$ becomes $(gh)f$ and gh becomes hg, so that we eventually get the associativity

$$h(gf) = (hg)f.$$

It may seem hard to believe, but duality is one of the biggest conquest of Category Theory. [...]

1.5 Foundations

Let us return at the beginning, namely the definition of category. Why not formulate it in terms of sets? That is, why don't muster the objects into a set, for any pair of objects, the morphisms into a set and writing compositions as functions?

Let us analyse what happens if we do that. A basic and quite popular fact that fatally crushes our hopes is:

there is no set of all sets.[1]

The first aftermath is that the existence of **Set** would not be legal, because otherwise a set would gather all sets.

Another example comes from both Algebra and Set Theory. In general, it's not a so profound result, but it is interesting for our discourse:

every pointed set $(X, 1)$ has an operation that makes it a group.[2]

Viz there exists no set of all groups, and then neither **Grp** would be supported.

As if the previous examples were not enough, Topology provides another irreducible case. Any set has the corresponding powerset, thus any set gives rise to at least one topological space. Our efforts are doomed, again: there is no set of all topological spaces, and so also **Top** would not be allowed!

It seems that using Set Theory requires the sacrifice of nice categories; and we do not want that, of course. From the few examples above one could surmise it is a matter of *size*: sets sometimes are not appropriate for collecting all the stuff that makes a category. Luckily, there is not a unique Set Theory and, above all, there is one that could help us.

The *von Neumann-Bernays-Gödel approach*, usually shortened as NBG, was born to solve size problems, and may be a good ground for our purposes. In NBG we have *classes*, the most general concept of 'collection'. But not all classes are at the same level: some, the *proper classes*, cannot be element of any class, whilst the others are the *sets*. Here is how the definition of category would look like.

Definition 1.5.1 (Categories)**.** A category $\mathcal{C}$ consists of:

- a class of objects, denoted $|\mathcal{C}|$;
- for every $a, b \in |\mathcal{C}|$, a class of morphisms from a to b, written as $\mathcal{C}(a, b)$;
- for every $a, b, c \in |\mathcal{C}|$, a composition, viz a function

$$\mathcal{C}(b, c) \times \mathcal{C}(a, b) \to \mathcal{C}(a, c)\,, \ (g, f) \to gf$$

with the following axioms:

1 If we want a set X to be the set of all sets, then it has all its subsets as elements, which is an absurd. In fact, Cantor's Theorem states that for every set X there is no surjective function $f : X \to 2^X$.

2 Actually, this fact is equivalent to the Axiom of Choice.

1. for every $x \in |\mathcal{C}|$ there exists a $1_x \in \mathcal{C}(x,x)$ such that for every $y \in |\mathcal{C}|$ and $g \in \mathcal{C}(y,x)$ we have
$$1_x g = g$$
and for every $z \in |\mathcal{C}|$ and $h \in \mathcal{C}(x,z)$ we have
$$h 1_x = h;$$
2. for $a,b,c,d \in |\mathcal{C}|$ and $f \in \mathcal{C}(a,b)$, $g \in \mathcal{C}(b,c)$ and $h \in \mathcal{C}(c,d)$ we have the identity
$$(hg)f = h(gf).$$

How does this double ontology of NBG actually apply at our discourse? For example, in NBG the class of all sets is a legit object: it is a proper class, because it cannot be an actual set. Thus, **Set** exists on NBG, and so exists **Grp**, **Top** and other big categories. Which is nice.

Hence, it is sensible to introduce some terms that distinguish categories by the size of their class of objects. [...]

[What can go wrong if $\mathcal{C}(a,b)$ are proper classes?]

1.6 Monomorphisms and Epimorphisms

[This section has to be rewritten.]

Definition 1.6.1 (Monomorphisms and epimorphisms)**.** A morphism $f : a \to b$ of a category $\mathcal{C}$ is said to be:

- a *monomorphism* whenever if
$$\begin{tikzcd} c \arrow[r, bend left, "g_1"] \arrow[r, bend right, "g_2"'] & a \arrow[r, "f"] & b \end{tikzcd}$$
commutes for every object c and morphisms $g_1, g_2 : c \to a$ of $\mathcal{C}$, then $g_1 = g_2$;
- an *epimorphism* whenever if
$$\begin{tikzcd} a \arrow[r, "f"] & b \arrow[r, bend left, "h_1"] \arrow[r, bend right, "h_2"'] & c \end{tikzcd}$$
commutes for every object d and morphisms $h_1, h_2 : c \to a$ of $\mathcal{C}$, then $h_1 = h_2$;

[...]

Another way to express the things of the previous definition is this: $f : a \to b$ is a monomorphism whenever for every $c \in |\mathcal{C}|$ the function

$$\mathcal{C}(c,a) \to \mathcal{C}(c,b)\,,\ g \to fg \tag{1.6.8}$$

is injective. Similarly, $f : a \to b$ is an epimorphism when for every $d \in |\mathcal{C}|$ the function

$$\mathcal{C}(a,d) \to \mathcal{C}(b,d)\,,\ h \to hf \tag{1.6.9}$$

is injective. Category theorists call the functions (1.6.8) *precompositions* with f and (1.6.9) *postcompositions* with f.

2

Functors

2.1 Definition

Definition 2.1.1 (Functors)**.** A functor F from a category $\mathcal{C}$ to a category $\mathcal{D}$ is having the following functions, all indicated with F:

- a 'function on objects'

$$F : |\mathcal{C}| \to |\mathcal{D}| \ , \ x \to F(x)$$

- for every objects a and b, one 'function on morphisms'

$$F : \mathcal{C}(a,b) \to \mathcal{D}(F(a),F(b)) \ , \ f \to F(f)$$

such that

1. for every object x of $\mathcal{C}$ we have $F(1_x) = 1_{F(x)}$;
2. for every objects x, y, z and morphisms $f : x \to y$ and $g : y \to z$ of $\mathcal{C}$ we have $F(g)F(f) = F(gf)$.

To say that F is a functor from $\mathcal{C}$ to $\mathcal{D}$ we use $F : \mathcal{C} \to \mathcal{D}$, a symbolism that recalls that one of morphism in categories.

Example 2.1.2 (Set functions)**.** Take a category whose class of objects is an actual set and devoid of morphisms. In that case, compositions are functions between empty sets and the categorial axioms are vacuous truths. So, for X and Y two sets, regarded as categories in the sense just outlined, functors $F : X \to Y$ are practically reduced to a function on objects, that is elements; in that case a functor between sets 'is' a function of sets. Observe that, in this — legal! — case not having any morphism yields that the functorial axioms are bare vacuous truths: so, also the converse holds, that is set funtions can be seen as functors.

Example 2.1.3 (Monotonic functions)**.** We have met before, how a preordered set is a category; recall also the pure set-theoretic definition of this notion. For $(A, \leq_A)$ and $(B, \leq_B)$ preordered sets, a function $f : A \to B$ is said *monotonic* whenever for every $x, y \in A$ we have $f(x) \leq_B f(y)$ provided that $x \leq_A y$. In bare set-theoretic terms, this can be rewritten as follows: for every $x, y \in A$ such that $(x, y) \in \leq_A$, then $(f(x), f(y)) \in \leq_B$, where we make explicit the pairs, that are morphisms of the preordered sets seen as categories.

Example 2.1.4 (Monoid homomorphsisms)**.** We have previously seen that a monoid 'is' a single-object category. Consider now two such categories, say $\mathcal{G}$ and $\mathcal{H}$, and a functor $f : \mathcal{G} \to \mathcal{H}$ is. Denoting by $\bullet_\mathcal{G}$ and $\bullet_\mathcal{H}$ the object of $\mathcal{G}$ and $\mathcal{H}$ respectively, there is a unique possibility: mapping $\bullet_\mathcal{G}$ to $\bullet_\mathcal{H}$. The functorial axioms in that case are:

$$f(xy) = f(x)f(y)$$

for every morphisms x and y of $\mathcal{G}$ and

$$f(1_\mathcal{G}) = 1_\mathcal{H},$$

with $1_\mathcal{G}$ and $1_\mathcal{H}$ being the identities of $\mathcal{G}$ and $\mathcal{H}$ respectively. These two properties say that f is a monoid homomorphism; in this case there is also an equation that about objects but these two are a mere subtlety that adds nothing. It is easy to do the converse: a monoid homomorphism is a functor.

Example 2.1.5 (The category **Eqv**)**.** A *setoids* [nlab uses this term...], that is sets together with an equivalence relation defined on it; if X is a set and $\sim$ an equivalence relation over X, the corresponding setoid is written simply as $(X, \sim)$. Any set X has naturally its equality relation $=_X$ defined on it[1], and for X and Y sets, a function $f : X \to Y$ respects this rule by definition:

for every $a, b \in X$, if $a =_X b$ then $f(a) =_Y b$.

We would like to replace the equalities above with equivalence relations: for if $(X, \sim_X)$ and $(Y, \sim_Y)$ are setoids, a *functoid* [ok, let me find/craft a nicer name...] from $(X, \sim_X)$ to $(Y, \sim_Y)$ is exactly a set function $f : X \to Y$ such that

for every $a, b \in X$, if $a \sim_X b$ then $f(a) \sim_Y f(b)$.

Functoids are certain type of functions, so composing two of them as such returns a funtoid. Categorial axioms hold for free, so we really have a *category of setoids and functoids*, **Eqv**.
There is a nice theorem:

1 In Set Theory, $=_X$ is the set $\{(a, a) \mid a \in X\}$.

Let X and Y be two sets with $\sim_X$ and $\sim_Y$ equivalence relations on X and Y respectively. Then for every $f : X \to Y$ such that $f(a) \sim_Y f(b)$ for every $a, b \in X$ such that $a \sim_X b$, there exists one and only one $\phi : X/\sim_X \to Y/\sim_Y$ that makes

$$\begin{tikzcd} X \arrow[r, "f"] \arrow[d, "\lambda a.[a]_X"'] & Y \arrow[d, "\lambda b.[b]_Y"] \\ X/\sim_X \arrow[r, "\phi"'] & Y/\sim_Y \end{tikzcd}$$

commute. (The vertical functions are the canonical projections.)

This underpins the functor

$$\pi : \mathbf{Eqv} \to \mathbf{Set}$$

that maps setoids $(X, \sim)$ to the sets $X/\sim$ and functoids $f : (X, \sim_X) \to (Y, \sim_Y)$ to functions

$$\pi_f : X/\sim_X \to Y/\sim_Y$$
$$\pi_f([a]_X) := [f(a)]_Y,$$

whose existence and uniqueness is claimed by the just mentioned Proposition.

Example 2.1.6 (Free group functor)**.** Suppose given a *group alphabet* S, that is a set of 'letters'; then *group words* with system S are strings obtained by juxtaposition of a finite amount of 'x^1' and 'x^{-1}', where $x \in S$. The *empty word* is obtained by writing no letter, and we shall denote it by something, say e; instead, the other words appear as

$$x_1^{\phi_1} \cdots x_n^{\phi_n},$$

with $x_1, \ldots, x_n \in S$ and $\phi_1, \ldots, \phi_n \in \{-1, 1\}$. [2] [3]
The length of a word is the number of letters it is made of. We define equality only on words having the same length: we say $x_1^{\alpha_1} \cdots x_n^{\alpha_n}$ is equal to $y_1^{\beta_1} \cdots y_n^{\beta_n}$ whenever $x_i = y_i$ and $\alpha_i = \beta_i$ for every $i \in \{1, \ldots, n\}$.
A group word $x_1^{\phi_1} \cdots x_n^{\phi_n}$ is called *irreducible* whenever $x_i^{\phi_i} \neq x_{i+1}^{-\phi_{i+1}}$ for every $i \in \{1, \ldots, n-1\}$; the empty word is irreducible by convention. Let us write

2 Something that may irk you is that our words can be redundant, being consecutive repetitions of the same letter allowed. If you want, you can let exponents range over all the integers, but this needs you to modify what comes after.

3 Here, we can choose any pair of symbols instead of -1 and 1. If we do so, we need a function that maps each of them into the other one. In this presentation we employ the function that takes one integer and returns its opposite.

$\langle S \rangle$ the set of all irreducible words written using the alphabet S. It is natural to join two words by bare juxtaposition, but the resulting word may not be irreducible; this issue has to be fixed:

$$\begin{aligned}
&\cdot : \langle S \rangle \times \langle S \rangle \to \langle S \rangle \\
&e \cdot w := w\,,\ w \cdot e := w \\
&(x_1^{\lambda_1} \cdots x_m^{\lambda_m}) \cdot (y_1^{\mu_1} \cdots y_n^{\mu_n}) := \begin{cases} (x_1^{\lambda_1} \cdots x_{m-1}^{\lambda_{m-1}}) \cdot (y_2^{\mu_2} \cdots y_n^{\mu_n}) & \text{if } x_m^{\lambda_m} = y_1^{-\mu_1} \\ x_1^{\lambda_1} \cdots x_m^{\lambda_m} y_1^{\mu_1} \cdots y_n^{\mu_n} & \text{otherwise.} \end{cases}
\end{aligned}$$

Let us define a function that either reverses the order of the letters and changes each exponent to the other one:

$$i : \langle S \rangle \to \langle S \rangle\,,\ i\left(x_1^{\xi_1} \cdots x_i^{\xi_i} x_{i+1}^{\xi_{i+1}} \cdots x_n^{\xi_n}\right) := x_n^{-\xi_n} \cdots x_{i+1}^{-\xi_{i+1}} x_i^{-\xi_i} \cdots x_1^{-\xi_1}.$$

It is immediate to show that $w \cdot i(w) = i(w) \cdot x = e$ for every $w \in \langle S \rangle$. Only the associativity of $\cdot$ is a a bit tricky to prove. At this point we have endowed $\langle S \rangle$ with a group structure.
Thus from a set S we are able to build a group $\langle S \rangle$, that is called *free group* with base S, or group generated by S. Now, if take two sets S and T and a function $f : S \to T$, we have the group homomorphism

$$\langle f \rangle : \langle S \rangle \to \langle T \rangle\,,\ \langle f \rangle (x_1^{\delta_1} \cdots x_n^{\delta_n}) := \big(f(x_1)\big)^{\delta_i} \cdots \big(f(x_n)\big)^{\delta_n}.$$

It is immediate to demonstrate that we ended up with having a functor

$$\langle\ \ \rangle : \mathbf{Set} \to \mathbf{Grp}.$$

Exercise 2.1.7. There is a plenty of 'free stuff' around that can give arise to functors like the one above. Find and illustrate some of them.

Traditionally, functors of Definition 2.1.1 above are called 'covariant', because there are *contra*variant functors too. However, there is no sensible reason to maintain these two adjectives; at least, almost everyone agrees to not use the first adjective, whilst the second one still survives.

For if $\mathcal{C}$ and $\mathcal{D}$ are categories, a *contravariant functor* from $\mathcal{C}$ to $\mathcal{D}$ is just a functor $\mathcal{C}^{\mathrm{op}} \to \mathcal{D}$. It is best that we say what functors $F : \mathcal{C}^{\mathrm{op}} \to \mathcal{D}$ do. They map objects to objects and morphisms $f : a \to b$ of $\mathcal{C}^{\mathrm{op}}$ to morphisms $F(f) : F(a) \to F(b)$ of $\mathcal{D}$. But, remembering how dual categories are defined, what F actually does is this:

> it maps objects of $\mathcal{C}$ to objects of $\mathcal{D}$, and morphisms $f : b \to a$ of $\mathcal{C}$ to morphisms $F(f) : F(a) \to F(b)$ of $\mathcal{D}$ (mind that a and b have their roles flipped).

Now, what about functoriality axioms? Neither with identities F does something different and the composite gf of $\mathcal{C}^{\mathrm{op}}$ is mapped to the composite $F(g)F(f)$ of $\mathcal{D}$. Again by definition of dual categories, this can be translated as follows:

> the composite fg of $\mathcal{C}$ is mapped to $F(g)F(f)$ (notice here how f and g have their places switched).

You can think of contravariant functors as a trick to do what we want.

Example 2.1.8. The set of natural numbers $\mathbb{N}$ has the order relation of divisibility, that we denote $|$: regard this poset as a category. From Group Theory, we know that for every $m, n \in \mathbb{N}$ such that $m \mid n$ there is a homomorphism

$$f_{m,n} : \mathbb{Z}/n\mathbb{Z} \to \mathbb{Z}/m\mathbb{Z}\,,\ f_{m,n}(a + n\mathbb{Z}) := a + m\mathbb{Z}.$$

In fact, $\mathbb{Z}/m\mathbb{Z}$ is the kernel of the homomorphism

$$\pi_m : \mathbb{Z} \to \mathbb{Z}/m\mathbb{Z}\,,\ \pi_m(x) := x + m\mathbb{Z}$$

and, because $m \mid n$, we have $n\mathbb{Z} \subseteq m\mathbb{Z}$. In that case, some Isomorphism Theorem[4] justifies the existence of $f_{m,n}$. This offers us a nice functor:

$$F : (\mathbb{N}, |)^{\mathrm{op}} \to \mathbf{Grp}$$

that maps naturals n to groups $\mathbb{Z}/n\mathbb{Z}$ and $m \mid n$ to the homomorphism $f_{m,n}$ defined above.

2.2 Category of categories?

Functors can be composed, and I think a this point it is not a secret. Take $\mathcal{C}$, $\mathcal{D}$ and $\mathcal{E}$ categories and functors

$$\mathcal{C} \xrightarrow{F} \mathcal{D} \xrightarrow{G} \mathcal{E}.$$

4 How theorems are named sometimes varies, so for sake of clarity let us explicit the statement we are referring to: Let G and H be two groups, $f : G \to H$ an homomorphism and N some normal subgroup of G. Consider also the homomorphism $p_N : G \to G/N$, $p_N(x) := xN$. If $N \subseteq \ker f$ then there exists one and only one homomorphism $\overline{f} : G/N \to H$ such that $f = \overline{f} p_N$. (Moreover, $\overline{f}$ is surjective if and only if so is f.)

The sensible way to define the composite functor $GF : \mathcal{C} \to \mathcal{E}$ is mapping the objects x of $\mathcal{C}$ to the objects $GF(x)$ of $\mathcal{E}$, and the morphisms $f : x \to y$ of $\mathcal{C}$ to the morphisms $GF(f) : GF(x) \to GF(y)$ of $\mathcal{E}$. That being set, the composition is associative and there is an identity functor too.

So what prevents us to consider a category — we can call **Cat** — that has categories as objects and functors as morphisms? If we work upon NBG, we can think of any proper class as a category, for this statement have a closer look at Example 2.1.2. What happens now is that the class of objects of **Cat** has an element that is a proper class, which isn't clearly legal in NBG.

Is a category of *locally small* categories and functors problematic? Take $\mathcal{C}$ such that $\mathcal{C}(a,b)$ is a proper class for some a and b objects: consider $\mathcal{C}/b$. In this case $|\mathcal{C}/b|$ is a proper class too.

So, now what? If we stick to NBG, this is a limit we have to take into account. From now on, **Cat** is the category of *small* categories and functors between small categories.

2.3 Equivalent categories

Let us give a definition that will motivate our discourse.

Definition 2.3.1 (Full- and faithfulness)**.** A functor $F : \mathcal{C} \to \mathcal{D}$ is said *full*, respectively *faithful*, whenever for every $a, b \in |\mathcal{C}|$ the functions

$$F : \mathcal{C}(a,b) \to \mathcal{D}(F(a), F(b))$$

are surjective, respectively injective; we say that F is *fully faithful* [how lame, lol...] whenever it is both full and faithful.

What do we want 'two categories are the same' to mean? [Craft a nicer exposition... Let us try with categories being isomorphic first, and then with *essentially surjective* functors. Talk about *skeletons* of categories, and how can help to say whether two categories are equivalent.]

Example 2.3.2 (A functor $\mathbf{Mat}_k \to \mathbf{FDVect}_k$)**.** For k field, consider the functor

$$M : \mathbf{Mat}_k \to \mathbf{FDVect}_k$$

that maps $n \in |\mathbf{Mat}_k| = \mathbb{N}$ to $M(n) := k^n$ and $A \in \mathbf{Mat}_k(r,s)$ to the linear function

$$M_A : k^r \to k^s$$
$$M_A(x) = Ax.$$

(Here the elements of k^n are matrices of type $n \times 1$.) [...]

3

Natural Transformations

3.1 Definition

For $\mathcal{C}$ and $\mathcal{D}$ categories and $F, G : \mathcal{C} \to \mathcal{D}$ functors, a *transformation* from F to G amounts at having for every $x \in |\mathcal{C}|$ one morphism $F(x) \to G(x)$ of $\mathcal{D}$. In other words, a transformation is aimed to measure the difference of two parallel functor by the unique means we have, viz morphisms.

In general, we stick to the following convention: if η is the name of a transformation from F to G, then η_x indicates the component $F(x) \to G(x)$ of the transformation.

We are not interested in all transformations, of course.

Definition 3.1.1 (Natural transformations)**.** A transformation η from a functor $F : \mathcal{C} \to \mathcal{D}$ to a functor $G : \mathcal{C} \to \mathcal{D}$ is said to be *natural* whenever for every $a, b \in |\mathcal{C}|$ and $f \in \mathcal{C}(a, b)$ the square

$$\begin{tikzcd} F(a) \arrow[r, "\eta_a"] \arrow[d, "F(f)"'] & G(a) \arrow[d, "G(f)"] \\ F(b) \arrow[r, "\eta_b"'] & G(b) \end{tikzcd}$$

commutes. This property is the 'naturality' of η.

There are some notations for referring to natural transformations: one may write $\eta : F \Rightarrow G$ or even

$$\begin{tikzcd} \mathcal{C} \arrow[r, bend left, "F"] \arrow[r, bend right, "G"'] \arrow[r, phantom, "\Downarrow\eta"] & \mathcal{D} \end{tikzcd}$$

if they want to explicit also categories.

Natural transformations can be composed: taken two consecutive natural transformations

$$\begin{tikzcd} \mathcal{C} \arrow[r, bend left=60, "F"{name=U}] \arrow[r, "G"{name=M, description}] \arrow[r, bend right=60, "H"'{name=D}] & \mathcal{D} \arrow[Rightarrow, from=U, to=M, "\eta"] \arrow[Rightarrow, from=M, to=D, "\theta"] \end{tikzcd}$$

the transformation $\theta\eta$ that have the components $\theta_x\eta_x : F(x) \to H(x)$, for $x \in |\mathcal{C}|$ of $\mathcal{D}$ is natural. Such composition is associative. Moreover, for every functor $F : \mathcal{C} \to \mathcal{D}$ there is the natural transformation $1_F : F \Rightarrow F$ with components $1_{F(x)} : F(x) \to F(x)$, for $x \in |\mathcal{C}|$; they are identities in categorial sense:

$$\eta 1_F = \eta \text{ for every natural transformation } \eta : F \Rightarrow G$$
$$1_F \mu = \mu \text{ for every natural transformation } \mu : H \Rightarrow F.$$

All this suggests to, given two categories $\mathcal{C}$ and $\mathcal{D}$, form a category with functors $\mathcal{C} \to \mathcal{D}$ as objects and natural transformations as morphism, them being composable as explained above. [...]

[Consider `https://mathoverflow.net/q/39073`...]

3.2 Equivalent categories, again

3.3 The Yoneda Lemma

[Maybe, I should stick to *small* categories...]

[Use `cramped` for some `tikzcds`...]

We have the *evaluation functor*

$$\mathrm{ev}_{\mathcal{C}} : \mathcal{C} \times [\mathcal{C}, \mathbf{Set}] \to \mathbf{Set}$$

that on objects

$$\mathrm{ev}_{\mathcal{C}}(x, F) := F(x)$$

and on morphisms

$$\mathrm{ev}\left(\begin{array}{c} a \\ \big\downarrow f \\ b \end{array}, \begin{array}{c} F \\ \eta \Big\Downarrow \\ G \end{array} \right) := \eta_b F(f) = G(f)\eta_a.$$

Lemma 3.3.1 (A lemma for the Yoneda Lemma)**.** Let $\mathcal{C}$ be a locally small category. Then for every $x \in |\mathcal{C}|$ and functor $F : \mathcal{C} \to \mathbf{Set}$,

$$[\mathcal{C}, \mathbf{Set}](\mathcal{C}(x,-), F) \cong F(x).$$

In particular, the classes $[\mathcal{C}, \mathbf{Set}](\mathcal{C}(x,-), F)$ are actual sets.

Proof. For x and F as in the hypothesis, take functions

$$\lambda_{x,F} : [\mathcal{C}, \mathbf{Set}](\mathcal{C}(x,-), F) \to F(x)\,,\ \lambda_{x,F}(\alpha) := \alpha_x(1_x).$$

Now, for every $a \in F(x)$ we have the transformation $\mu_{x,F}(a)$ from $\mathcal{C}(x, \bullet)$ to F which has the components

$$\mathcal{C}(x,c) \to F(c)\,,\ f \to \big(F(f)\big)(a);$$

it is immediate to show that it is natural. Thus we have functions

$$\mu_{x,F} : F(x) \to [\mathcal{C}, \mathbf{Set}](\mathcal{C}(x,-), F).$$

We prove

$$\begin{aligned} \lambda_{x,F}\mu_{x,F} &= 1_{F(x)} \\ \mu_{x,F}\lambda_{x,F} &= 1_{[\mathcal{C},\mathbf{Set}](\mathcal{C}(x,-),F)}. \end{aligned}$$

In fact, for $a \in F(x)$ we have $\lambda_{x,F}\big(\mu_{x,F}(a)\big)$ is the component $\mathcal{C}(x,x) \to F(x)$ of $\mu_{x,F}(a)$ evaluated at 1_x, viz $1_{F(x)}(a) = a$. Besides, for if $\alpha : \mathcal{C}(x, \bullet) \Rightarrow F$ natural transformation we have $\mu_{x,F}\big(\lambda_{x,F}(\alpha)\big) = \mu_{x,F}\big(\alpha_x(1_x)\big)$ is the natural transformation $\mathcal{C}(x, \bullet) \Rightarrow F$ with components

$$\mathcal{C}(x,c) \to F(c)\,,\ f \to \big(F(f)\big)(\alpha_x(1_x)) = \alpha_c(f)$$

for $c \in |\mathcal{C}|$; that is $\mu_{x,F}\lambda_{x,F}(\alpha) = \alpha$. The proof is complete now. □

Let $\mathcal{C}$ be a locally small category. We have the functor

$$よ_{\mathcal{C}} : \mathcal{C} \times [\mathcal{C}, \mathbf{Set}] \to \mathbf{Set}$$

given on objects as follows

$$よ_{\mathcal{C}}(x, F) := [\mathcal{C}, \mathbf{Set}]\big(\mathcal{C}(x,-), F\big)$$

and on morphisms

$$\left[\text{よ}_{\mathcal{C}}\left(\begin{array}{ccc} a & & F\\ f\big\downarrow & , & \eta\Downarrow \\ b & & G\end{array}\right)\right]\left(\begin{array}{c}\mathcal{C}(a,-)\\ \Downarrow\alpha\\ F\end{array}\right) := \left\{\mathcal{C}(b,c)\xrightarrow{\eta_c\alpha_c(\_f)} G(c)\,\middle|\, c\in|\mathcal{C}|\right\}.$$

Observe that Lemma 3.3.1 solves annoying size issues in the definition of $\text{よ}_{\mathcal{C}}$ on objects. While the statement of this lemma is important for technical reasons, its proof guides us to the following completion.

Proposition 3.3.2 (Yoneda Lemma)**.** For $\mathcal{C}$ locally small category, $\text{よ}_{\mathcal{C}} \cong \mathrm{ev}_{\mathcal{C}}$.

Proof. The transformation $\lambda : \text{よ}_{\mathcal{C}} \Rightarrow \mathrm{ev}_{\mathcal{C}}$ having as components the functions $\lambda_{x,F}$ of the proof of Lemma 3.3.1 is natural, that is

$$\begin{tikzcd}
\text{よ}_{\mathcal{C}}(a,F) \arrow[r, "\lambda_{a,F}"] \arrow[d, "\text{よ}_{\mathcal{C}}(f,\eta)"'] & \mathrm{ev}_{\mathcal{C}}(a,F) \arrow[d, "\mathrm{ev}_{\mathcal{C}}(f,\eta)"] \\
\text{よ}_{\mathcal{C}}(b,G) \arrow[r, "\lambda_{b,G}"'] & \mathrm{ev}_{\mathcal{C}}(b,G)
\end{tikzcd}$$

commutes for every $f \in \mathcal{C}(a,b)$ and $\eta \in [\mathcal{C},\mathbf{Set}](F,G)$. In fact, for every natural transformation $\eta \in \text{よ}_{\mathcal{C}}(a,F)$ we have

$$\mathrm{ev}_{\mathcal{C}}(f,\eta)\big(\lambda_{a,F}(\alpha)\big) = \eta_b\alpha_b\mathcal{C}(a,f)(1_a) = \eta_b\alpha_b f;$$

besides,

$$\lambda_{b,G}\big(\text{よ}_{\mathcal{C}}(f,\eta)(\alpha)\big) = \eta_b\alpha_b(\_f)(1_b) = \eta_b\alpha_b f.$$

We can conclude λ is an isomorphism, as the proof of Lemma 3.3.1 tells us its components are isomorphisms. $\square$

4

Limits & colimits

4.1 Definition

Albeit this may cause uneasy nights, we present ex abrupto the general notion of (co)limits.

Definition 4.1.1 (Limits & colimits)**.** Let $\mathcal{I}$ and $\mathcal{C}$ be two categories. For every $v \in |\mathcal{C}|$ we have the *constant functor*

$$k_v : \mathcal{I} \to \mathcal{C}$$

where $k_v(i) := v$ for every $i \in |\mathcal{I}|$ and $k_v(f) := 1_v$ for every morphism f of $\mathcal{I}$. A *limit* of a functor $F : \mathcal{I} \to \mathcal{C}$ is some $v \in |\mathcal{C}|$ with a natural transformation $\lambda : k_v \Rightarrow F$ such that:

> for any $a \in |\mathcal{C}|$ and $\mu : k_v \Rightarrow F$ there exists one and only one $f \in \mathcal{C}(a, v)$ such that
>
> $$\begin{tikzcd} a \arrow[d, "f"'] \arrow[dr, "\mu_i"] & \\ v \arrow[r, "\lambda_i"'] & F(i) \end{tikzcd}$$
>
> commutes for every $i \in |\mathcal{I}|$.

A *colimit*, instead, is an object $u \in |\mathcal{C}|$ together with a natural transformation $\chi : F \Rightarrow k_u$ that has the property:

> for every $b \in |\mathcal{C}|$ and $\xi : F \Rightarrow k_b$ there exists one and only one $g \in \mathcal{C}(u, b)$ that makes
>
> $$\begin{tikzcd} u \arrow[d, "g"'] & \\ b & F(i) \arrow[ul, "\chi_i"'] \arrow[l, "\xi_i"] \end{tikzcd}$$
>
> commute for every $i \in |\mathcal{I}|$.

It may be of aid to expand a little bit some parts of the definition. For example, what is a natural transformation $\eta : k_v \Rightarrow F$? By definition of natural transformation, it is a collection $\{\eta_i : v \to F(i) \mid i \in |\mathcal{I}|\}$ of morphisms of $\mathcal{C}$ that has the property:

$$\begin{tikzcd} & F(i) \arrow[dd, "F(f)"] \\ v \arrow[ur, "\eta_i"] \arrow[dr, "\eta_j"'] & \\ & F(j) \end{tikzcd}$$

commutes for every $i, j \in |\mathcal{I}|$ and $f \in \mathcal{I}(i, j)$.

Yes, for a functor $\mathcal{I} \to \mathcal{C}$, the category $\mathcal{C}$ has its share, but it is $\mathcal{I}$ who has the last say in the research of (co)limits. The role of $\mathcal{I}$ is to give a ‘shape’ of the limits we are looking for, indeed.

Example 4.1.2. Let $\mathcal{C}$ be a category and $\mathbf{1}$ a category that has one object and one morphism, and take a functor $f : \mathbf{1} \to \mathcal{C}$, some $v \in \mathcal{C}$ and the corresponding constant functor $k_v : \mathbf{1} \to \mathcal{C}$. A natural transformation $\zeta : k_v \Rightarrow f$ amounts of a single morphism $v \to \widetilde{f}$ of $\mathcal{C}$, where $\widetilde{f}$ indicates the image of the unique object of $\mathbf{1}$ via f. Thus, a limit of f is some $v \in |\mathcal{C}|$ and a morphism $\lambda : v \to \widetilde{f}$ of $\mathcal{C}$ such that: for every object u and morphism $\mu : u \to \widetilde{f}$ in $\mathcal{C}$, there is a unique morphism $u \to v$ of $\mathcal{C}$ that makes

$$\begin{tikzcd} u \arrow[dr, "\mu"] \arrow[dd] & \\ & \widetilde{f} \\ v \arrow[ur, "\lambda"'] & \end{tikzcd}$$

commute.

Exercise 4.1.3. What are colimts of functors $\mathbf{1} \to \mathcal{C}$?

Sandbox 4.1.4. Consider a monoid (viz a single object category) $\mathcal{G}$: for the scope of this example we write G for the set of the morphisms of $\mathcal{G}$. Let $F : \mathcal{G} \to \mathbf{Set}$ be a functor, and let $\widehat{F}$ indicate the F-image of the unique object of $\mathcal{G}$ whilst, for $f \in G$, $\widehat{f}$ the function $F(f) : \widehat{F} \to \widehat{F}$. Now, being $k_X : \mathcal{G} \to \mathbf{Set}$ the functor constant at X, with X a set, a natural transformation $\lambda : F \Rightarrow k_X$ is a morphism $\lambda : \widehat{F} \to X$ such that $\lambda = \lambda\widehat{f}$ for every $f \in G$. These two things, the set X and the function λ, together are a colimit of F whenever

> for every set Y and function $\mu : \widehat{F} \to Y$ such that $\mu = \mu\widehat{f}$ for every $f \in G$ there exists one and only one function $h : X \to Y$ such that $\mu = h\lambda$.

[Is that thing even interesting?] [Write about functors $\mathcal{G} \to \mathbf{Set}$...]

Proposition 4.1.5. Let $\mathcal{C}$ be a category and $F : \mathcal{I} \to \mathcal{C}$ a functor. If $\eta : k_a \Rightarrow F$ and $\theta : k_b \Rightarrow F$ are limits (or colimits) of F, then a is isomorphic to b.

Proof. We prove only the part regarding limits. Being η and θ limits of F, we a have a unique $f : a \to b$ and a unique $g : b \to a$ that make the two triangles in

$$\begin{tikzcd} a \arrow[dr, "\eta_i"] \arrow[dd, shift left, "f"] & \\ & F(i) \\ b \arrow[ur, "\theta_i"'] \arrow[uu, shift left, "g"] & \end{tikzcd}$$

commute for every object i of $\mathcal{I}$. In this case,

$$\eta_i = \theta_i f = \eta_i(gf)$$
$$\theta_i = \eta_i g = \theta_i(fg)$$

Invoking again the universal property of limits, $gf = 1_a$ and $fg = 1_b$. □

Fortunately, there are few shapes that are both ubiquitous and simple. This section is dedicated to them, while in the successive one we will prove (Proposition 4.3.2) that if some simple functors have limits, then all the functors do have limits.

4.2 Noteworthy limits and colimits

[Make subsections sections?]

4.2.1 Terminal and initial objects

Definition 4.2.1. For $\mathcal{C}$ category, the limits of the empty functor $\varnothing \to \mathcal{C}$ are called *terminal objects* of $\mathcal{C}$, whereas the colimits *initial objects*.

Let us expand the definition above so that we can can look inside things. A cone over the empty functor $\varnothing \to \mathcal{C}$ with vertex $a \in |\mathcal{C}|$ is a natural transformation

$$\begin{tikzcd} \varnothing \arrow[r, bend left, "k_a"] \arrow[r, bend right] \arrow[r, phantom, "\Downarrow"] & \mathcal{C} \end{tikzcd} .$$

Here, the empty functor is k_a because there is at most one functor $\varnothing \to \mathcal{C}$. Again, because there must be a unique one, our natural transformation is the empty transformation, viz the one devoid of morphisms. A similar reasoning leads us to the following explicit definition of terminal and initial object.

Definition 4.2.2. Let $\mathcal{C}$ be a category. Then

- a terminal object of $\mathcal{C}$ is an $a \in |\mathcal{C}|$ such that for every $x \in |\mathcal{C}|$ there exists one and only one $f \in \mathcal{C}(x, a)$;
- an initial object of $\mathcal{C}$ is an $a \in |\mathcal{C}|$ such that for every $x \in |\mathcal{C}|$ there exists one and only one $f \in \mathcal{C}(a, x)$.

Examples time.

Example 4.2.3 (Empty set and singletons)**.** It may sound weird, but for every set X there does exist a function $\varnothing \to X$; moreover, it is the unique one. To get this, think set-theoretically: a function is any subset of $\varnothing \times X$ that has the property we know. But $\varnothing \times X = \varnothing$, so its unique subset is $\varnothing$. This set is a function from $\varnothing$ to X since the statement

for every $a \in \varnothing$ there is one and only one $b \in X$ such that $(a, b) \in \varnothing$

is a ‘vacuous truth’. So $\varnothing$ is an initial object of **Set**. This case is quite particular, since the initial objects of **Set** are actually equal to $\varnothing$.
Now let us look for terminal objects in **Set**. Take an arbitrary set X: there is exactly one function from X to any singleton, that is singletons are terminal object of **Set**. Conversely, by Proposition 4.1.5, the terminal objects of **Set** must be singletons.

Exercise 4.2.4. Trivial groups — there is a unique way a singleton can be a group — are either terminal and initial objects of **Grp**.

We can present a lot of stupid examples, but we can head straight toward more interesting things.

Construction 4.2.5. Let us introduce a nice category that allows us to express some nice and simple facts in Mathematics. Let $\mathcal{C}$ and $\mathcal{J}$ two categories, a one of its objects and take a functor $F : \mathcal{J} \to \mathcal{C}$. We have the category $(a \downarrow F)$, this made:

- the objects are the morphisms $a \to F(x)$ of $\mathcal{C}$, for $x \in |\mathcal{J}|$;

- the morphisms from $f : a \to F(x)$ to $g : a \to F(y)$ are the morphisms $h : x \to y$ of $\mathcal{J}$ such that

$$\begin{tikzcd} & F(x) \arrow[dd, "F(h)"] \\ a \arrow[ur, "f"] \arrow[dr, "g"'] & \\ & F(y) \end{tikzcd}$$

  commutes;
- the composition is that of $\mathcal{J}$.

Example 4.2.6. In Linear Algebra (or when dealing with free modules) we have a nice theorem:

> Let V be a vector space over a field k and $S \subseteq V$ a base. For every vector space W over k and function $\phi : S \to W$ there exists a unique linear function $f : V \to W$ such that
>
> $$\begin{tikzcd} S \arrow[r, hook, "i"] \arrow[dr, "\phi"'] & V \arrow[d, "f"] \\ & W \end{tikzcd}$$
>
> commutes.

In other words, this statement says that a linear function is completely determined by what it does with the vectors of S. We will consider now two functors

$$\mathbf{Set} \xrightarrow{\langle \cdot \rangle} \mathbf{Vect}_k \xrightarrow{U} \mathbf{Set}.$$

The first one takes a set S and produces the vector space on k

$$\langle S \rangle := \left\{ \sum_{x \in S} \lambda_x x \,\middle|\, \lambda : S \to k,\ \lambda_x \neq 0 \text{ for finitely many times} \right\}$$

(considered with two obvious operations). Furthermore, a function of sets $f : S \to T$ induces a linear function $\langle f \rangle : \langle S \rangle \to \langle T \rangle$ defined by

$$\langle f \rangle \left(\sum_{x \in S} \lambda_x x \right) := \sum_{x \in S} \lambda_x f(x)$$

where $\lambda : S \to k$ is almost always null. The functoriality of $\langle \cdot \rangle$ is just a matter of quick controls. [Do we really need all that machinery?] The functor U instead takes vector spaces and returns the correspondent set of vectors; we write $U(V) := V$, but observe that in **Set** we don't care anymore of the vector structure of V. Similarly, it takes linear functions and the return them: but, since U lands onto **Set**, who cares about linearity there? (We may say that U is the 'inclusion' of $\mathbf{Vect}_k$ into **Set**.) [Talk about forgetful functors elsewhere...] All this words allow us restate the aforementioned theorem as:

> For if S is a set, the inclusion $S \hookrightarrow \langle S \rangle$ is an initial object of $(S \downarrow U)$.

Exercise 4.2.7. In the previous example some details are omitted: you can be more talkative, though. However, it is really worth to think such examples — not only because we will meet such pattern later under the vest of adjunctions. You may also look for another examples of similar kind, I'm sure you will find some.

Example 4.2.8 (Isomorphism Theorem for Set Theory)**.** We have defined **Eqv** earlier, recall it here. We have the functor

$$j : \mathbf{Set} \to \mathbf{Eqv}$$

that maps sets X to setoids X together with the equality relation, and functions $f : X \to Y$ to themselves. To get the mood for this example, sets *are* setoids where the equivalence relation is equality and functions *are* functoids between such setoids. In this case, the classical theorem

> Let X and Y be two sets and $\sim$ an equivalence relation on X. For every function $f : X \to Y$ such that $f(a) = f(b)$ for every $a, b \in X$ with $a \sim b$, there exists one and only one function $\overline{f} : X/\sim \to Y$ such that
>
> $$\begin{tikzcd} X \arrow[rr, "f"] \arrow[dr, "p"'] & & Y \\ & X/\sim \arrow[ur, "\overline{f}"'] & \end{tikzcd}$$
>
> commutes, where $p : X \to X/\sim$ is the canonical projection.

can be restated as follows

> the canonical projection $p : (X, \sim) \to X/\sim$ is initial in $(X, \sim) \downarrow j$.

Example 4.2.9 (Recursion). In Set Theory, there is a nice theorem, the *Recursion Theorem*:

> Let $(\mathbb{N}, 0, s)$ be a Peano Model, where $0 \in \mathbb{N}$ and $s : \mathbb{N} \to \mathbb{N}$ is its successor function. For every pointed set X, $a \in X$ and $f : X \to X$ there exists one and only one function $x : \mathbb{N} \to X$ such that $x_0 = a$ and $x_{s(n)} = f(x_n)$ for every $n \in \mathbb{N}$.

Here, by Peano Model we mean a set $\mathbb{N}$ that has one element, we write 0, stood out and a function $s : \mathbb{N} \to \mathbb{N}$ such that, all this complying some rules:

1. s is injective;
2. $s(x) \neq 0$ for every $x \in \mathbb{N}$;
3. for if $A \subseteq \mathbb{N}$ has 0 and $s(n) \in A$ for every $n \in A$, then $A = \mathbb{N}$.

We show now how we can involve Category Theory in this case. First of all, we need a category where to work.
The statement is about things made as follows:

> a set X, one distinguished $x \in X$ and one function $f : X \to X$.

[Is there a name for these things?] We may refer to such new things by barely a triple (X, a, f), but we prefer something like this:

$$1 \xrightarrow{x} X \xrightarrow{f} X,$$

where 1 is any singleton, as usual. Peano Models are such things, with some additional properties. It is told about the existence and the uniqueness of a certain function. We do not want mere functions, of course: given

$$1 \xrightarrow{x} X \xrightarrow{f} X \text{ and } 1 \xrightarrow{y} Y \xrightarrow{g} Y,$$

we take the functions $r : X \to Y$ such that

$$\begin{tikzcd} & X \arrow[r, "f"] \arrow[d, "r"] & X \arrow[d, "r"] \\ 1 \arrow[ur, "x"] \arrow[dr, "y"'] & & \\ & Y \arrow[r, "g"'] & Y \end{tikzcd}$$

commutes and nothing else. [Is there a name for such functions?] These ones are the things we want to be morphisms. Suppose given

$$\begin{tikzcd}
 & X \arrow[r, "f"] \arrow[d, "p"] & X \arrow[d, "p"] \\
1 \arrow[ur, "x"] \arrow[r, "y"] \arrow[dr, "z"'] & Y \arrow[r, "g"] \arrow[d, "q"] & Y \arrow[d, "q"] \\
 & Z \arrow[r, "h"'] & Z
\end{tikzcd}$$

where all the squares and triangles commute: thus we obtain the commuting

$$\begin{tikzcd}
 & X \arrow[r, "f"] \arrow[d, "qp"] & X \arrow[d, "qp"] \\
1 \arrow[ur, "x"] \arrow[dr, "z"'] & & \\
 & Z \arrow[r, "h"'] & Z
\end{tikzcd}$$

This means that composing two morphisms as functions in **Set** produces a morphism. This is how we want composition to defined in this context. This choice makes the categorial axioms automatically respected. We call this category **Peano**. [Unless there is a better naming, of course.]
Being the environment set now, the Recursion Theorem becomes more concise:

Peano Models are initial objects of **Peano**.

By Proposition 4.1.5, any other initial object of **Peano** are isomorphic to some Peano Model: does this mean its initial objects are Peano Models? (Exercise.)

Exercise 4.2.10 (Induction $\Leftrightarrow$ Recursion)**.** In **Set**, suppose you have $1 \xrightarrow{0} \mathbb{N} \xrightarrow{s} \mathbb{N}$, where s is injective and $s(n) \neq 0$ for every $n \in \mathbb{N}$. Demonstrate that the following statements are equivalent:

1. for if $A \subseteq \mathbb{N}$ has 0 and $s(n) \in A$ for every $n \in A$, then $A = \mathbb{N}$;
2. $1 \xrightarrow{0} \mathbb{N} \xrightarrow{s} \mathbb{N}$ is an initial object of **Peano**.

(1) $\Rightarrow$ (2) proves the Recursion Theorem, whereas (2) $\Rightarrow$ (1) requires you to codify a proof by induction into a recursion. Try it, it could be nice. [Prepare hints...]

Remark 4.2.11. In general, suppose you have a category $\mathcal{C}$ that has a terminal object, that we denote by 1. [Explain why a terminal object is required.] We have then the category $\mathbf{Peano}_{\mathcal{C}}$, that is **Peano** with objects and morphisms picked from $\mathcal{C}$ and compositions performed in $\mathcal{C}$ — really, there is nothing special in **Set** that hinders us to do so. A *natural number object* of $\mathcal{C}$ is any of the initial objects of $\mathbf{Peano}_{\mathcal{C}}$, that is an assignment of one object $\mathbb{N}$, a morphism $0 : 1 \to \mathbb{N}$ and a morphism $f : X \to X$ of $\mathcal{C}$ such that they satisfy the *recursion property*, the Recursion Theorem but for $\mathcal{C}$. **Set** has natural number objects, as we have seen in the previous example, but another categories may have none. [For instance?] [This is quite interesting... An 'arithmetic' for categories, perhaps...]

Construction 4.2.12 (Category of cones)**.** For $\mathcal{C}$ category, let $F : \mathcal{I} \to \mathcal{C}$ be a functor. Then we define the *category of cones* over F as follows.

- The objects are the cones over F.
- For $\alpha := \left\{a \xrightarrow{\alpha_i} F(i)\right\}_{i \in |\mathcal{I}|}$ and $\beta := \left\{b \xrightarrow{\beta_i} F(i)\right\}_{i \in |\mathcal{I}|}$ two cones, the morphisms from α to β are the morphisms $f : a \to b$ of $\mathcal{C}$ such that

$$\begin{tikzcd} a \arrow[dr, "\alpha_i"] \arrow[dd, "f"'] & \\ & F(i) \\ b \arrow[ur, "\beta_i"'] & \end{tikzcd}$$

  commutes for every $i \in |\mathcal{I}|$.
- The composition of morphisms here is the same as that of $\mathcal{C}$.

We write such category as $\mathbf{Cn}_F$. We define also the *category of cocones* over F, written as $\mathbf{CoCn}_F$.

- The objects are the cocones over F.
- For $\alpha := \left\{F(i) \xrightarrow{\alpha_i} a\right\}_{i \in |\mathcal{I}|}$ and $\beta := \left\{F(i) \xrightarrow{\beta_i} b\right\}_{i \in |\mathcal{I}|}$ cocones, the morphisms from α to β are the morphisms $f : a \to b$ of $\mathcal{C}$ such that

$$\begin{tikzcd} & a \arrow[dl, "\alpha_i"'] \arrow[dd, "f"] \\ F(i) & \\ & b \arrow[ul, "\beta_i"] \end{tikzcd}$$

  commutes for every $i \in |\mathcal{I}|$.

- The composition of morphisms here is the same as that of $\mathcal{C}$.

It is quite immediate in either of the cases to show that categorial axioms are verified.

Proposition 4.2.13. For $\mathcal{C}$ category and $F : \mathcal{I} \to \mathcal{C}$ functor,

- limits of F are terminal objects of $\mathbf{Cn}_F$ and viceversa.
- colimits of F are initial objects of $\mathbf{CoCn}_F$ and viceversa.

Proof. Simple. □

Example 4.2.14 (Elements are morphisms from terminal objects)**.** A set is an objects made of elements: if X denotes a set and x some thing, we can formulate the sentence $x \in X$. However, we have seen cases — like $\mathbf{Mat}_k$ — where objects are 'atoms', that is do not have an internal structure as sets do. In some categories, objects have elements, in others the objects do not.
Let us take advantage of a basic fact about sets:

$$X \cong \mathbf{Set}(1, X) \text{ for every set } X.$$

In general, the isomorphism relation above is not made possible by a unique bijection, but there is one really meaningful for us: the function that takes $x \in X$ to the function $\widehat{x} : 1 \to X$ mapping the unique element of 1 into x. The great deal here is that functions $1 \to X$ inspect X and this isomorphism just outlined identifies every x to $\widehat{x}$.
Let us step back for a moment and turn our attention to the act of defining functions. To define a function $f : X \to Y$, one write an expression like

$$f(x) := \Gamma \tag{4.2.1}$$

with Γ being a formula that may contain the symbol x or not. By writing something like (4.2.1), you are prescribing the images of each element of the domain. This deeply relies on these two facts:

1. Sets are things you can look inside.
2. We have a principle that guarantees the function we are defining in such manner is uniquely determined:

   > Given two functions $f_1, f_2 : X \to Y$, if we have $f_1(a) = f_2(a)$ for every $a \in X$, then it must be $f_1 = f_2$.

This is crucial, since once you have assigned a function as in (4.2.1), it cannot behave any different from what prescribed.

How this can be interesting to us at this point? First of all, 1 is a terminal object. We have showed earlier how elements of X can be thought as functions $1 \to X$. In this case, the application of f to x is the mere composition

$$fx : 1 \xrightarrow{x} X \xrightarrow{f} Y.$$

The principle aforementioned can be restated as:

If the diagram

$$\begin{tikzcd} 1 \arrow[r, "x"] & X \arrow[r, shift left, "f_1"] \arrow[r, shift right, "f_2"'] & Y \end{tikzcd}$$

commutes for every $x : 1 \to X$, then $f_1 = f_2$.

[Continue. Do not forget!]

4.2.2 Products and coproducts

Definition 4.2.15. Let $\mathcal{C}$ be a category and $\mathcal{S}$ a discrete category. Then (co)limits of the functors $X : \mathcal{S} \to \mathcal{C}$ are called *(co)products*.

Again, let us put this definition into more explicit terms. First of all, what are cones over $X : \mathcal{S} \to \mathcal{C}$? For $p \in |\mathcal{C}|$ and $k_p : \mathcal{S} \to \mathcal{C}$ the functor constant at p, a natural transformation

$$\begin{tikzcd} \mathcal{S} \arrow[r, bend left, "k_p"{name=U}] \arrow[r, bend right, "X"'{name=D}] & \mathcal{C} \arrow[from=U, to=D, Rightarrow, "\pi"] \end{tikzcd}$$

is exactly a collection

$$\left\{p \xrightarrow{\pi_x} X_i\right\}_{i \in |\mathcal{S}|}.$$

In this fortunate case, the naturality condition automatically holds because $\mathcal{S}$ has no morphism other than the identities. Similarly, one can easily deduce what cocones are. Thus the explicit definition is:

Definition 4.2.16 (Explicit). [Introduce some new locutions...]

Example 4.2.17 (Cartesian product)**.** Given a family of sets $\{X_i \mid i \in I\}$, we have the *Cartesian product*

$$\prod_{i \in I} X_i := \left\{ f : I \to \bigcup_{i \in I} X_i \,\middle|\, f(i) \in X_i \text{ for every } i \in I \right\}.$$

with the *component functions*, one for each $j \in I$,

$$p_j : \prod_{i \in I} X_i \to X_j\,,\ p_j(f) := f(j).$$

(In other words, p_j takes an f and evaluates it at j.) Taken now any set A and functions $f_i : A \to X_i$ for $i \in I$, we have

$$\prod_{i \in I} f_i : A \to \prod_{i \in I} X_i$$

with $(\prod_{i \in I} f_i)(a)$ mapping $k \in I$ to $f_k(a)$. [Ok, the notation here is becoming cumbersome... Using $f_\bullet$ instead of $\prod_{i \in I} f_i$? The idea is: $f_\bullet(a)$ takes i as input by putting it in place of $\bullet$.] It is simple to show that

$$\begin{tikzcd}
A \arrow[dr, "f_\alpha"] \arrow[dd, "\prod_{i \in I} f_i"'] & \\
& X_\alpha \\
\prod_{i \in I} X_i \arrow[ur, "p_\alpha"'] &
\end{tikzcd}$$

commutes for every $\alpha \in I$. Moreover, $\prod_{i \in I} f_i$ is the only one that does this. Consider any function $g : A \to \prod_{i \in I} X_i$ with $f_\alpha = p_\alpha g$ for every $\alpha \in I$: then for every $x \in A$ we have

$$\big(g(x)\big)(\alpha) = p_\alpha\big(g(x)\big) = f_\alpha(x) = p_\alpha\left(\left(\prod_{i \in I} f_i\right)(x)\right) = \left(\left(\prod_{i \in I} f_i\right)(x)\right)(\alpha).$$

Hence, we can conclude $\prod_{i \in I} X_i$ with the collection of functions $\{p_i \mid i \in I\}$ is a product in **Set**. [Ok, where is all the machinery developed? Be more explicit.]

4.2.3 Equalizers and coequalizers

4.2.4 Pullbacks and pushouts

Example 4.2.18 (Seifert-van Kampen theorem)**.** Suppose given a topological space X, $A, B \subseteq X$ open sets such that $A \cup B = X$ and a point x_0 of $A \cap B$. Let us denote by i_A, i_B, j_A and j_B the group morphisms induced by the inclusions $A \cap B \hookrightarrow A$, $A \cap B \hookrightarrow B$, $A \hookrightarrow X$ and $B \hookrightarrow X$, respectively. If A, B and $A \cap B$ are path-connected then,

$$\begin{tikzcd} \pi_1(A \cap B, x_0) \arrow[r, "i_A"] \arrow[d, "i_B"'] & \pi_1(A, x_0) \arrow[d, "j_A"] \\ \pi_1(B, x_0) \arrow[r, "j_B"'] & \pi_1(X, x_0) \end{tikzcd}$$

is a pushout square of **Grp**.

4.3 (Co)Completeness

[No concerns about the size of $\mathcal{I}$? Start an enquiry about this soon!]

Definition 4.3.1. A category $\mathcal{C}$ is said *(co)complete* whenever any functor $\mathcal{I} \to \mathcal{C}$ has a (co)limit.

[**Set** is a complete category; so why why not deal with this — in a fine detailed way — before and postpone the proof of the Completeness Theorem?]

Proposition 4.3.2 (Completeness Theorem)**.** Categories that have products and equalizers are complete.

Proof. Take $\mathcal{I}$ and $\mathcal{C}$ to be two categories, and $X : \mathcal{I} \to \mathcal{C}$ any functor; just for convenience, let us write I for $|\mathcal{I}|$. $\mathcal{C}$ has all products, so let us write

$$\left\{ p \xrightarrow{\phi_i} X_i \,\middle|\, i \in I \right\}$$

for one of — it does not matter which one, right? — the products of $\{X_i \mid i \in I\}$. The class

$$H := \{ j \in I \mid \mathcal{I}(i, j) \neq \varnothing \text{ for some } i \in I \}$$

will be useful for the constructions to come. For $i \in I$, $j \in H$ and $f \in \mathcal{I}(i, j)$ we can draw this

$$\begin{tikzcd} & X_i \arrow[dr, bend left, "X_f"] & \\ p \arrow[ur, bend left, "\phi_i"] \arrow[rr, bend right, "\phi_j"'] & & X_j \end{tikzcd}$$

Again by the fact that $\mathcal{C}$ has products, let us write

$$\left\{ q \xrightarrow{\xi_j} X_j \,\middle|\, j \in H \right\}$$

for one of the products of $\{X_j \mid j \in H\}$. Now, the universal property of products yields two morphisms

$$p \underset{\theta}{\overset{\delta}{\rightrightarrows}} q \tag{4.3.2}$$

of $\mathcal{C}$ obtained as follows:

(1) θ is the one that factors ϕ_j through ξ_j for $j \in H$, viz $\phi_j = \xi_j\theta$.

(2) δ is the unique that factors $X_f\phi_i$ through ξ_j for every $i \in I$, $j \in H$ and $f \in \mathcal{I}(i, j)$, that is $X_f\phi_i = \xi_j\delta$

$\mathcal{C}$ has equalizers too, so let $\epsilon : e \to p$ be one of the equalizers of the parallel morphisms in (4.3.2). Now that everything is arranged, the rest of the proof is to prove that

$$\left\{ e \xrightarrow{\phi_i\epsilon} X_i \,\middle|\, i \in I \right\}$$

is a limit of X. It is important, however, to check preliminarily that it is a natural transformation. Take

$$\begin{tikzcd} & X_i \arrow[dd, "X_f"] \\ e \arrow[ur, "\phi_i\epsilon"] \arrow[dr, "\phi_j\epsilon"'] & \\ & X_j \end{tikzcd}$$

with $i, j \in I$ and $f \in \mathcal{I}(i, j)$. If $j \notin H$, then the commutativity of the diagram is a vacuous truth; otherwise,

$$X_i\phi_i\epsilon = \underbrace{\xi_j\delta\epsilon = \xi_j\theta\epsilon}_{\epsilon \text{ is equalizer}} = \phi_j\epsilon.$$

So, let us conclude the proof: provided a natural transformation

$$\left\{ a \xrightarrow{\sigma_i} X_i \,\middle|\, i \in I \right\},$$

we show how to construct $a \to e$ that makes

$$\begin{tikzcd} a \arrow[dd] \arrow[dr, "\sigma_i"] & \\ & X_i \\ e \arrow[ur, "\phi_i\epsilon"'] & \end{tikzcd}$$

commute. By universal property of product, there is a unique $\mu : a \to p$ such that $\sigma_i = \phi_i\mu = \xi_i\theta\mu$ for every $i \in I$. In particular, for $j \in H$, $i \in I$ and $f \in \mathcal{I}(i,j)$

$$\sigma_j = \begin{cases} \phi_j\mu = \xi_j\theta\mu & \text{by (1)} \\ X_f\sigma_i = X_f\phi_i\mu = \xi_j\delta\mu & \text{because } \sigma \text{ is a natural transformation and (2)} \end{cases}$$

As a consequence of the universal property of product of ξ, we must have $\theta\mu = \delta\mu$. Moreover, being $\epsilon : e \to p$ an equalizer of (4.3.2), then $\mu = \epsilon\psi$ for exactly one $\psi : a \to e$ of $\mathcal{C}$. Thus $\sigma_i = \phi_i\mu = \phi_i\epsilon\psi$, so ψ is what we are are looking for; at this point you can observe the uniqueness of ψ as well. That's all. □

Lemma 4.3.3. A category have finite products if and only if it has a terminal object and all binary products.

Proof. One implication is trivial. Let $\mathcal{C}$ be a category that has a terminal object, say 1, and all binary products [write what this would mean]. Let $\{x_1, \dots, x_n\} \subseteq |\mathcal{C}|$ and construct one product for them. We proceed by induction on n. If $n = 0$, we have a product: the terminal object 1. Assume now $\{x_1, \dots, x_n\}$ has a product, say the set of morphisms of $\mathcal{C}$

$$\{\alpha_i : p \to x_i \mid i = 1, \dots, n\}\,.$$

Take one $x_{n+1} \in |\mathcal{C}|$. By assumption, there is a product of p and x_{n+1}, a pair of morphisms

$$p \xleftarrow{\beta} q \xrightarrow{\gamma} x_{n+1}.$$

The question is: do the morphisms $\alpha_i\beta : q \to x_i$, for $i = 1, \dots, n$, and γ form a product of $\{x_1, \dots, x_n, x_{n+1}\}$? Yes, **exercise 4.3.4** (the drawing below is a hint).

$$\begin{tikzcd}
 & & r \arrow[dll, bend right, "f_i"'] \arrow[dl, dashed, bend right] \arrow[d, dotted] \arrow[dr, bend left, "f_{n+1}"] & \\
 & & q \arrow[dl, "\beta"'] \arrow[dr, "\gamma"] & \\
x_i & p \arrow[l, "\alpha_i"] & & x_{n+1}
\end{tikzcd}$$

□

Proposition 4.3.5 (Finite Completeness Theorem I)**.** Categories having terminal objects, binary products and equalizers are finitely complete. [Write a definition for 'finitely complete'.]

Proof. Use the last Lemma and the argument employed to prove the Completeness Theorem. □

Lemma 4.3.6. If a category has a terminal object and pullbacks [you haven't written about pullbacks yet], then it has binary products and equalizers.

Proof. Call $\mathcal{C}$ the category of the assumptions and 1 one of its terminal objects. It is simple that show that pullbacks of

$$\begin{tikzcd} & a \arrow[d] \\ b \arrow[r] & 1 \end{tikzcd}$$

are products of a and b. If it is not trivial, **exercise 4.3.7**. Now we show how to get an equalizer out of a terminal object and an appropriate pullback. Consider two parallel morphisms

$$\begin{tikzcd} a \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & b \end{tikzcd}$$

of $\mathcal{C}$. As we have just seen, there exists a product of a and b:

$$a \xleftarrow{\alpha} p \xrightarrow{\beta} b \ .$$

By universal property of product, define $\overline{f} : a \to p$ to be the morphisms such that $\alpha\overline{f} = 1_a$ and $f = \beta\overline{f}$. Similarly, let $\overline{g} : a \to p$ be the morphism such that $\alpha\overline{g} = 1_a$ and $g = \beta\overline{g}$. By assumption, $\mathcal{C}$ ha the pullback square

$$\begin{tikzcd} q \arrow[r, "m"] \arrow[d, "n"'] & a \arrow[d, "\overline{f}"] \\ a \arrow[r, "\overline{g}"'] & p \end{tikzcd}$$

Here $m = \alpha\overline{f}m = \alpha\overline{g}n = n$, so we end up having the fork

$$\begin{tikzcd} q \arrow[r, "m"] & a \arrow[r, shift left, "\overline{f}"] \arrow[r, shift right, "\overline{g}"'] & p \end{tikzcd} \ .$$

It should be easy to show that m is an equalizer: this is **exercise 4.3.8**. □

Proposition 4.3.9 (Finite Completeness Theorem II)**.** Categories that have terminal objects and pullbacks are finitely complete.

Proof. Use the previous Lemma and the Finite Completeness Theorem I. □

[Ok, now what? Topoi?]

5

Adjointness

5.1 Definition

Example 5.1.1 (Defining linear functions)**.** In Linear Algebra, we have a theorem which states linear functions are determined by the images of the elements of any base of the domain:

> Let V and W two vector spaces on some field k, with the first one having a base S; let us write i for the inclusion function $S \hookrightarrow V$. Then for every function $\phi : S \to W$ there exists one and only one linear function $f : V \to W$ such that
>
> $$\begin{tikzcd} S \arrow[r, hook, "i"] \arrow[dr, "\phi"'] & V \arrow[d, "f"] \\ & W \end{tikzcd}$$
>
> commutes.

That is the function

$$\begin{aligned} \mathbf{Vect}_k(V,W) &\to \mathbf{Set}(S,W) \\ f &\to fi \end{aligned} \tag{5.1.1}$$

is a bijection; in other words, this function post-composes linear functions with the inclusion of the base of the domain into the domain itself. Here, we write $\mathbf{Vect}_k$ and **Set** on purpose, because we want to walk a precise path. We have a function pointing to a vector space W, but set functions do not care about W being a vector space; instead, linear functions do! In some sense, in this example we see W ‘downgraded’ from the status of vector space to the one of barren set. On the other hand, from a set we construct an actual vector space — this is what being a base means. That being said, let us rearrange the diagram above:

$$\begin{tikzcd} S \arrow[r, "\text{generating}"] \arrow[d, "\phi"'] & V \arrow[d, "f"] \\ W & W \arrow[l, "\text{downgrading}"] \end{tikzcd}$$

where the horizontal arrows start from some category and land onto another one. The drawing just made is not meant to be a diagram in a strict sense, but an illustration on what is happening. If you are thinking the horizontal arrows are just a piece of a bigger picture, you are right: behind the scenes, two functors

$$\mathbf{Vect}_k \underset{G}{\overset{F}{\rightleftarrows}} \mathbf{Set}$$

are acting, where F is the functor that forgets and G the functor that takes sets and crafts a vector space from it and takes functions and gives linear functions. [Do I need to be more specific here? Perhaps, I may talk about such things elsewhere.] However, functors are a matter of morphisms too, so let us involve them too into this discourse. Let us call $\xi_{S,W}$ the function (5.1.1), as we will soon need thiss notation.
Take a function $\phi : S' \to S$ and a linear function $f : W \to W'$: in this case we have the function

$$\begin{aligned} \mathbf{Vect}_k(G(S), W) &\to \mathbf{Vect}_k(G(S'), W') \\ g &\to \left(fgG(\phi) : \begin{tikzcd} G(S) \arrow[r, "g"] & W \arrow[d, "f"] \\ G(S') \arrow[u, "G(\phi)"] & W' \end{tikzcd} \right) \end{aligned}$$

The point is that we have a functor

$$\mathbf{Vect}_k(G(\ \ ),\ \ ) : \mathbf{Set}^{\mathrm{op}} \times \mathbf{Vect}_k \to \mathbf{Set}$$

that maps pairs (S, V) to sets $\mathbf{Vect}_k(G(S), V)$ and with respect to morphisms acts as just described above. It is not enough, giving the same ϕ and f the function

$$\begin{aligned} \mathbf{Set}(S, F(W)) &\to \mathbf{Set}(S', F(W')) \\ \delta &\to \left(F(f)\delta\phi : \begin{tikzcd} S \arrow[r, "\delta"] & F(W) \arrow[d, "F(f)"] \\ S' \arrow[u, "\phi"] & F(W') \end{tikzcd} \right) \end{aligned}$$

and so a functor

$$\mathbf{Set}(\ \ , F(\ \ )) : \mathbf{Set}^{\mathrm{op}} \times \mathbf{Vect}_k \to \mathbf{Set}$$

mapping pairs (S, V) to sets $\mathbf{Set}(S, F(V))$ this time. That is we ended up with two functors

$$\mathbf{Set}^{\mathrm{op}} \times \mathbf{Vect}_k \to \mathbf{Set}.$$

Let us push our discourse a little further: the functions $\xi_{S,W}$, for S varying on sets and W on vector spaces over k, does form a natural isomorphism

$$\begin{tikzcd} \mathbf{Set}^{\mathrm{op}} \times \mathbf{Vect}_k \arrow[r, bend left, "\mathbf{Vect}_k(G(\ \ ),\ \ )"{name=U}] \arrow[r, bend right, "\mathbf{Set}(\ \ ,F(\ \ ))"'{name=D}] & \mathbf{Set} \arrow[Rightarrow, from=U, to=D, "\xi"] \end{tikzcd}$$

Observe, that being the $\xi_{S,W}$-s all isomorphisms, then we are done if we show that ξ is a natural transformation, viz

$$\begin{tikzcd} \mathbf{Vect}_k(G(S), W) \arrow[r, "\xi_{S,W}"] \arrow[d, "\lambda h.fhG(\phi)"'] & \mathbf{Set}(S, F(W)) \arrow[d, "\lambda h.F(f)h\phi"] \\ \mathbf{Vect}_k(G(S'), W') \arrow[r, "\xi_{S',W'}"'] & \mathbf{Set}(S', F(W')) \end{tikzcd}$$

commutes for every set S and S', vector space W and W', function $\phi : S' \to S$ and linear function $f : W \to W'$. To prove this, consider the inclusions $i : S \hookrightarrow G(S)$ and $i' : S' \hookrightarrow G(S')$. Thus

$$\left.\begin{matrix} (F(f)(hi)\phi)(v) \\ (fhG(\phi)i')(v) \end{matrix}\right\} = (f(h(\phi(v)))) \text{ for every } v \in S'$$

and we have concluded.

I'd love you to experiment by yourself a bit more before reading the definition of adjunction. The following exercise is already started, so you are 'just' required to finish it. Do not worry if you do not know how to go, read the definition and examine the other examples and return to this exercise. Any choice you do, it is worth to give it it a try now.

Exercise 5.1.2 (Inspired by Haskell[1]). [Borrow some Haskell notation?] We define the category of partial functions, written as **Par**. Here objects are sets and morphisms are partial functions. For A and B sets, a *partial function* from A to B is relation $f \subseteq A \times B$ with this property:

1 A programming language. It is not bad you know something about it.

$$\text{for every } x \in A \text{ and } y_1, y_2 \in B, \text{ if } (x, y_1) \in f \text{ and } (x, y_2) \in f \text{ then } y_1 = y_2.$$

We want to compose partial functions as well: provided $f \in \mathbf{Par}(A, B)$ and $g \in \mathbf{Par}(B, C)$,

$$gf := \{(x, y) \in A \times C \mid (x, z) \in f \text{ and } (z, y) \in g \text{ for some } z \in B\}.$$

It is immediate to verify **Par** complies the rules that make it a category. The thing important here is this: suppose given a partial function $f : A \to B$, every $x \in A$ may have one element of B bound — in this case, we write it $f(x)$ — or none. The key of the exercise is: what if we considered 'no value' as an admissible output value? Provided two sets A and B and a partial function $f : A \to B$, we assign an actual function

$$\overline{f} : A \to B + 1, \ \overline{f}(x) := \begin{cases} f(x) & \text{if } x \text{ has an element of } B \text{ bound} \\ * & \text{otherwise} \end{cases}$$

where $1 := \{*\}$ with $*$ designating the absence of output. It is quite simple to show that

$$\mathbf{Par}(A, B) \to \mathbf{Set}(A, B + 1), \ f \to \overline{f}$$

is a bijection for every couple of sets A and B. Now it's up to you to categorify this by considering two suitable functors

$$\begin{tikzcd} \mathbf{Set} \arrow[r, shift left, "I"] & \mathbf{Par} \arrow[l, shift left, "\texttt{Maybe}"] \end{tikzcd}.$$

It should be simple to guess how is defined I. As for `Maybe`, you do not need to know Haskell: if you do, fine, otherwise you are learning something new.

Now it is time to isolate the concept we have envisaged in the previous examples. Let $\mathcal{C}$ and $\mathcal{D}$ be two locally small [or something more?] categories and

$$\begin{tikzcd} \mathcal{C} \arrow[r, bend left, "L"] & \mathcal{D} \arrow[l, bend left, "R"] \end{tikzcd}$$

two functors. We have then the functor

$$\hom_{\mathcal{C}}(\ \ , R(\ \ )) : \mathcal{C}^{\mathrm{op}} \times \mathcal{D} \to \mathbf{Set}$$

that maps objects (x, y) to $\hom_{\mathcal{C}}(x, R(y))$ and pairs of morphisms

$$\begin{pmatrix} (x,y) \\ {\scriptstyle (f,g)}\big\downarrow \\ (x',y') \end{pmatrix} = \begin{pmatrix} x & & y \\ {\scriptstyle f}\big\uparrow & , & \big\downarrow{\scriptstyle g} \\ x' & & y' \end{pmatrix}$$

to functions

$$\begin{gathered} \hom_{\mathcal{C}}(x, R(y)) \to \hom_{\mathcal{C}}(x', R(y')) \\ h \to R(g)hf \end{gathered}$$

We have also the functor

$$\hom_{\mathcal{D}}(L(\ \ ), \ \ ) : \mathcal{C}^{\mathrm{op}} \times \mathcal{D} \to \mathbf{Set}$$

that maps (x, y) to $\hom_{\mathcal{C}}(L(x), y)$ and pairs of morphisms

$$\begin{pmatrix} (x,y) \\ {\scriptstyle (f,g)}\big\downarrow \\ (x',y') \end{pmatrix} = \begin{pmatrix} x & & y \\ {\scriptstyle f}\big\uparrow & , & \big\downarrow{\scriptstyle g} \\ x' & & y' \end{pmatrix}$$

to functions

$$\begin{gathered} \hom_{\mathcal{D}}(L(x), y) \to \hom_{\mathcal{D}}(L(x'), y') \\ h \to ghL(f). \end{gathered}$$

Exercise 5.1.3. The functors just mentioned can be defined as composition of others, one of which is already known. We recall it here. For if $\mathcal{C}$ is a locally small category, the functor

$$\hom_{\mathcal{C}} : \mathcal{C}^{\mathrm{op}} \times \mathcal{C} \to \mathbf{Set}$$

described as follows:

- objects (x, y) are mapped into sets $\hom_{\mathcal{C}}(x, y)$;
- morphisms $(f, g) : (x, y) \to (x', y')$ of $\mathcal{C}^{\mathrm{op}} \times \mathcal{C}$, viz pairs

$$\begin{pmatrix} x & & y \\ {\scriptstyle f}\big\uparrow & , & \big\downarrow{\scriptstyle g} \\ x' & & y' \end{pmatrix}$$

  with the first morphism regarded as one of $\mathcal{C}$, into functions

$$\begin{gathered} \hom_{\mathcal{C}}(x, y) \to \hom_{\mathcal{C}}(x', y') \\ h \to ghf \end{gathered}$$

Now, suppose given two functors

$$F_1 : \mathcal{C}_1 \to \mathcal{D}_1 \text{ and } F_2 : \mathcal{C}_2 \to \mathcal{D}_2.$$

We define the *product functor*

$$F_1 \times F_2 : \mathcal{C}_1 \times \mathcal{C}_2 \to \mathcal{D}_1 \times \mathcal{D}_2$$

as follows: it maps objects (a_1, a_2) to $(F_1(a_1), F_2(a_2))$ and morphisms

$$\left(\begin{array}{c} x_1 \\ {\scriptstyle f_1}\big\downarrow \\ y_1 \end{array} , \begin{array}{c} x_2 \\ {\scriptstyle f_2}\big\downarrow \\ y_2 \end{array} \right)$$

into

$$\left(\begin{array}{c} F_1(x_1) \\ {\scriptstyle F_1(f_1)}\big\downarrow \\ F_1(y_1) \end{array} , \begin{array}{c} F_2(x_2) \\ {\scriptstyle F_2(f_2)}\big\downarrow \\ F_2(y_2) \end{array} \right)$$

Definition 5.1.4 (Adjunctions)**.** Consider two locally small categories $\mathcal{C}$ and $\mathcal{D}$ and two functors $\mathcal{C} \underset{R}{\overset{L}{\rightleftarrows}} \mathcal{D}$. An *adjunction* from L to R is a natural isomorphism

$$\begin{tikzcd} \mathcal{C}^{\mathrm{op}} \times \mathcal{D} \arrow[r, bend left, "\hom_{\mathcal{C}}(-,R(\cdot))"] \arrow[r, bend right, "\hom_{\mathcal{D}}(L(-),\cdot)"'] \arrow[r, phantom, "\Downarrow\alpha"] & \mathbf{Set} \end{tikzcd} .$$

We write such natural isomorphism as $\alpha : L \dashv R$, and say L is the *left adjoint*, whereas R is the *right* one.

[This part needs to be fixed.] Keeping the notation of the definition above, for $a \in |\mathcal{C}|$, $b \in |\mathcal{D}|$ and $f \in \hom_{\mathcal{C}}(a, R(b))$ we write $\alpha(f)$ for that element of $\hom_{\mathcal{D}}(L(a), b)$ which corresponds to f via the adjunction α. Since α is an isomorphism, it has the inverse $\alpha^{-1} : \hom_{\mathcal{D}}(L(a), b) \to \hom_{\mathcal{C}}(a, R(b))$: again, here $\alpha^{-1}(g)$ for $g \in \hom_{\mathcal{D}}(L(a), b)$ denotes the corresponding morphism via α^{-1}. Actually, the point is there is no preferred direction between the twos mentioned. In fact, for the most general purposes, more drastic conventions can be taken, that is using the same symbol for both α and α^{-1}:

$$\overline{a \xrightarrow{f} R(b)} = \left(L(a) \xrightarrow{\overline{f}} b \right), \quad \overline{L(a) \xrightarrow{g} b} = \left(a \xrightarrow{\overline{g}} R(b) \right).$$

Just pay attention about where things come from and where they are to go! Albeit this requires some caution, this notation has the great advantage that

$$\overline{\overline{p}} = p \text{ for every } p.$$

Now, we can explicitly express the naturality of the adjunction:

$$\overline{R(g)pf} = g\overline{p}L(f) \tag{5.1.2}$$

where $x, x' \in |\mathcal{C}|$, $y, y' \in |\mathcal{D}|$, $f \in \hom_{\mathcal{C}}(x', x)$, $g \in \hom_{\mathcal{D}}(y, y')$ and $p \in \hom_{\mathcal{C}}(x, R(y))$; equivalently,

$$\overline{gqL(f)} = R(g)\overline{q}f \tag{5.1.3}$$

where here $q \in \hom_{\mathcal{D}}(L(x), y)$.

5.2 Adjunctions, units and co-units

In the first example of the introduction, we isolated the concept of adjunction from that one of initial objects of certain categories. We now isolate and formalize this process; we will do the converse too. As result, we end up having two equivalent ways to work with adjointness.

Proposition 5.2.1. Suppose given two locally small categories $\mathcal{C}$ and $\mathcal{D}$, two functors $\mathcal{C} \underset{R}{\overset{L}{\rightleftarrows}} \mathcal{D}$ and a natural transformation $\eta : 1_{\mathcal{C}} \Rightarrow RL$ such that $\eta_x : x \to RL(x)$ is initial in $x{\downarrow}R$ for every $x \in |\mathcal{C}|$. Then, for $x \in |\mathcal{C}|$ and $y \in |\mathcal{D}|$, the functions

$$\mathcal{D}(Lx, y) \to \mathcal{C}(x, Ry)\,,\ f \to R(f)\eta_x$$

form a natural transformation $\mathcal{D}(L(\ \ ),\ \ ) \Rightarrow \mathcal{C}(\ \ , R(\ \ ))$.

Proof. The fact that η_x is initial object implies that these function are all bijective. Now, we just need to verify the transformation is natural. Take $x, x' \in |\mathcal{C}|$, $y, y' \in |\mathcal{D}|$, $f \in \mathcal{C}(x', x)$ and $g \in \mathcal{D}(y, y')$ and examine the square

$$\begin{tikzcd}
\mathcal{D}(L(x), y) \arrow[r, "u \to R(u)\eta_x"] \arrow[d, "u \to guL(f)"'] & \mathcal{C}(x, R(y)) \arrow[d, "v \to R(g)vf"] \\
\mathcal{D}(L(x'), y') \arrow[r, "v \to R(v)\eta_{x'}"'] & \mathcal{C}(x', R(y'))
\end{tikzcd}$$

Taken $u \in \mathcal{D}(L(x), y)$, we perform the following calculations

$$R(g)R(u)\eta_x f = R(gu)\eta_x f$$
$$R(guL(f))\eta_{x'} = R(gu)RL(f)\eta_{x'}$$

By naturality $\eta_x f = RL(f)\eta_{x'}$, so the construction ends here. □

Proposition 5.2.2. Suppose given locally small categories $\mathcal{C}$ and $\mathcal{D}$, two functors $\mathcal{C} \underset{R}{\overset{L}{\rightleftarrows}} \mathcal{D}$ and an adjunction $\alpha : L \dashv R$. Then $\eta : 1_{\mathcal{C}} \Rightarrow RL$ is natural and $\eta_x : x \to RL(x)$ is initial in $x{\downarrow}R$ for every $x \in |\mathcal{C}|$.

Proposition 5.2.3. Suppose now you have locally small categories $\mathcal{C}$ and $\mathcal{D}$, functors $\mathcal{C} \underset{R}{\overset{L}{\rightleftarrows}} \mathcal{D}$ and an adjunction $\alpha : L \dashv R$.Then the morphisms of $\mathcal{C}$

$$\eta_x : x \to RL(x)\,,\ \eta_x := \alpha\left(1_{L(x)}\right) \quad \text{where } x \in |\mathcal{C}|$$

do form a natural transformation $\eta : 1_{\mathcal{C}} \Rightarrow RL$; moreover, η_x is initial in $x{\downarrow}R$.

Proof. Yet to be TEX-ed... □

5.3 From units, counits and triangle identities to adjunctions

5.4 Limits from the view of adjunctions

Let $\mathcal{I}$ and $\mathcal{C}$ be two categories. For every $v \in |\mathcal{C}|$ we have the *constant functor*

$$k_v : \mathcal{I} \to \mathcal{C}$$

where $k_v(i) := v$ for every $i \in |\mathcal{I}|$ and $k_v(f) := 1_v$ for every morphism f of $\mathcal{I}$. Recall that being $\lambda : k_v \Rightarrow F$ a limit of a functor $F : \mathcal{I} \to \mathcal{C}$ means:

> for every $\mu : k_v \Rightarrow F$ there exists one and only one $f : a \to v$ of $\mathcal{C}$ such that $\mu_i = \lambda_i f$ commutes for every object i of $\mathcal{I}$.

That is, if you put it in other words, it sounds like:

> there is a bijection
> $$\mathcal{C}(a,v) \to [\mathcal{I},\mathcal{C}](k_a, F)$$
> taking $f : a \to v$ to the natural transformation
> $$\lambda_\bullet f : k_a \Rightarrow F\,,\ \lambda_\bullet f(i) := \lambda_i f.$$

There is a smell of adjunction situation here. Let us start with finding an appropriate pair of functors

$$\mathcal{C} \rightleftarrows [\mathcal{I}, \mathcal{C}] \ .$$

One functor is already suggested:

$$\Delta : \mathcal{C} \to [\mathcal{I}, \mathcal{C}]$$

takes $x \in |\mathcal{C}|$ to the functor $\mathcal{I} \to \mathcal{C}$ that maps every object to x and every morphism to 1_x; then for $i \in |\mathcal{I}|$ define

$$\Delta\left(x \xrightarrow{f} y\right)$$

to be the natural transformation $\Delta(x) \Rightarrow \Delta(y)$ amounting uniquely of f.

From now on, assume $\mathcal{I}$ is small and every functor $\mathcal{I} \to \mathcal{C}$ has a limit. Now, in spite of not being strictly unique ['strictly unique'... huh?], all the limits of a given functor are isomorphic, so are the vertices: let us indicate by $\lim F$ the vertex of any of the limits of F. Now, take a natural transformation

$$\begin{tikzcd} \mathcal{I} \arrow[r, bend left, "F"{name=U}] \arrow[r, bend right, "G"'{name=D}] & \mathcal{C} \arrow[from=U, to=D, Rightarrow, "\xi"] \end{tikzcd} ;$$

$\lim F$ is the vertex of some limit

$$\left\{ \lim F \xrightarrow{\lambda_i} F(i) \,\middle|\, i \in |\mathcal{I}| \right\}$$

and $\lim G$ is the vertex of a certain limit

$$\left\{ \lim G \xrightarrow{\mu_i} G(i) \,\middle|\, i \in |\mathcal{I}| \right\}.$$

If we display all the stuff we have gathered so far, we have for $i \in |\mathcal{I}|$

$$\begin{tikzcd} F(i) \arrow[r, "\xi_i"] & G(i) \\ \lim F \arrow[u, "\lambda_i"] & \lim G \arrow[u, "\mu_i"'] \end{tikzcd} \tag{5.4.4}$$

The universal property of limits ensures that there is one and only one morphism $\lim F \to \lim G$ making the above diagram a commuting square. Let us call this morphism $\lim \xi$. We have a functor

$$\lim : [\mathcal{I}, \mathcal{C}] \to \mathcal{C}$$

indeed. If you take $F = G$ and η the identity of the functor F in (5.4.4), then

$$\lim 1_F = 1_{\lim F},$$

obtained by uniquely employing the universal property of limit. Now take three functors $F, G, H : \mathcal{I} \to \mathcal{C}$ and two natural transformations $F \overset{\alpha}{\Longrightarrow} G \overset{\beta}{\Longrightarrow} H$. To these functors are associated the respective limits

$$\left\{ \lim F \xrightarrow{\lambda_i} F(i) \middle| i \in |\mathcal{I}| \right\}$$
$$\left\{ \lim G \xrightarrow{\mu_i} G(i) \middle| i \in |\mathcal{I}| \right\}$$
$$\left\{ \lim H \xrightarrow{\eta_i} H(i) \middle| i \in |\mathcal{I}| \right\}$$

so that we have commuting squares glued together:

$$\begin{tikzcd}
F(i) \arrow[r, "\alpha_i"] & G(i) \arrow[r, "\beta_i"] & H(i) \\
\lim F \arrow[u, "\lambda_i"] \arrow[r, "\lim \alpha"'] & \lim G \arrow[u, "\mu_i"] \arrow[r, "\lim \beta"'] & \lim H \arrow[u, "\eta_i"']
\end{tikzcd}$$

We have for every $i \in |\mathcal{I}|$

$$\eta_i \lim \beta \lim \alpha = \beta_i \mu_i \lim \alpha = \beta_i \alpha_i \lambda_i;$$

then, by how it is defined the limit of a natural transformation, it must be

$$\lim(\beta\alpha) = \lim \beta \lim \alpha.$$

The following proposition pushes all this discourse to a conclusion.

Proposition 5.4.1. There is an adjunction

$$\begin{tikzcd}
\mathcal{C} \arrow[r, bend left, "\Delta"] \arrow[r, phantom, "\perp"] & {[\mathcal{I}, \mathcal{D}]} \arrow[l, bend left, "\lim"]
\end{tikzcd}$$

Proof. [Yet to be TEX-ed...] □