MA213

Second Year Essay

The Yoneda Lemma

2023-03-21

2111082

Contents

1 Introduction

Many structures in mathematics come alongside some notion of maps, which are used to relate different objects with those structures. For instance, groups come alongside group homomorphisms, vector spaces with linear transformations, probability spaces with measurable functions, and topological spaces with continuous maps, to name a few.

A *category* is any collection of objects with maps between those objects that compose associatively. All of the previous examples are thus specific types of categories, and many common constructions in those areas do not actually rely on anything specific to that area and can be carried out and unified together when performed at the level of a category. For instance, the Cartesian product, direct product of groups (rings, monoids...), product topology, disjoint union, and graph tensor product are all instances of a categorical product. If we can prove something about the categorical product, we'll have proved a result about all of these different types of objects.

Just as many properties in metric spaces are actually topological in nature, many mathematical objects can be reduced to purely categorical constructions: direct sums, kernels, quotient objects, compactifications and completions are all also categorical in nature.

A common theme in category theory is that maps between objects are more important than the objects themselves, and it will often be the case that it is easier to describe an object by the properties it satisfies or what relations the object has, rather than what the object itself actually is. Even in abstract algebra, this is often the case – we care that the elements of a group have group structure, not what the elements themselves actually are or how we label them.

In mathematics, we often come across statements of the form, $\exists! x : P(x)$, or, "There exists a unique x such that $P(x)$ holds." The property P is called a *universal property*, and it uniquely characterises the object x up to an isomorphism. For example,

Theorem 1.1. *Let* $\mathbf{1}$ *be a set with one element. Then, for all sets* X*, there exists a unique function from* X *to* $\mathbf{1}$*.*

Proof. For existence, we define a function that maps every element of X to the unique element of $\mathbf{1}$. Because every element of X only has one choice of destination, this function is unique. ■

So, the property "For all sets X, there exists a unique function from X to $\mathbf{1}$" uniquely characterises $\mathbf{1}$, up to relabelling of the element. Rather than describing an object itself, universal properties allow us to describe objects by how they relate to other objects in whatever universe we're working in.

The *Yoneda lemma* expands on this concept, suggesting that we may study a category $\mathscr{C}$ by examining the maps from $\mathscr{C}$ to the category of sets.

2 Categories

[Lei14] A *category* $\mathscr{C}$ consists of:

- A class $\mathrm{ob}(\mathscr{C})$ of *objects* in $\mathscr{C}$.
- For all (ordered) pairs of objects $A,B \in \mathrm{ob}(\mathscr{C})$, a class $\hom(A,B)$ of *maps* or *arrows* called *morphisms* from A to B, called the *hom-set* or *hom-class* of morphisms from A to B, also sometimes written $\mathscr{C}(A,B)$ or $\hom_{\mathscr{C}}(A,B)$ (particularly useful if multiple categories are in use). If $f \in \hom(A,B)$, we write $f : A \to B$ or $A \xrightarrow{f} B$. The collection of all of these classes is the hom-set of $\mathscr{C}$, and is written $\hom(\mathscr{C})$.
- For any three objects $A,B,C \in \mathrm{ob}(\mathscr{C})$, a binary operation, $\circ : \hom(A,B) \times \hom(B,C) \to \hom(A,C)$, $(g,f) \mapsto g \circ f$, called *composition*, such that,
  - (*associativity*) if $f : A \to B$, $g : B \to C$, and $h : C \to D$, then $h \circ (g \circ f) = (h \circ g) \circ f$;

- (*identity*) for every object $X \in \mathrm{ob}(\mathscr{C})$, there exists a morphism $\mathrm{id}_X : X \to X$ called the *identity morphism* on X, such that every morphism $f : A \to X$ satisfies $\mathrm{id}_X \circ f = f$, and every morphism $g : X \to B$ satisfies $g \circ \mathrm{id}_X = g$.

In the above definitions, we use the term *class*. This is because these collections of data generally do not count as sets under ZFC or related axiomatisations of set theory. For instance, the collection of all sets does not qualify as a set under ZFC, but such a totality is frequently required in categorical constructions, and one therefore appeals to classes instead.

The notion of a class is informal in ZFC, since ZFC axiomatises only sets. We will take NBG set theory as our ambient metatheory to have a formal notion of classes, as is standard in the area, but for practical purposes, the reader may proceed with the following informal description: a *class* is regarded as being any collection of *ZFC sets* specified unambiguously by a property all its members share, including properties such as "being a set". Any class which fails to satisfy the axioms characterising sets is a *proper class*, while a class that also satisfies the set axioms is a *small class.*

Many of the categories we will construct will be *locally small*, meaning that the class of morphisms between any pair of objects happens to be a set. That is, $\hom(A,B)$ is a set (in the sense that they can be constructed in our choice of ambient metatheory) for all $A,B \in \mathrm{ob}(\mathscr{C})$. If we also have that $\mathrm{ob}(\mathscr{C})$ is a set, then $\mathscr{C}$ is furthermore a *small* category.

Let us examine this data in more detail.

An object can really be anything we want, but many of the simplest and most familiar examples will begin with sets, often with additional structure, such as groups or rings. For any two objects, A and B, the category has a collection of morphisms from A to B, $\hom(A,B) = \{f,g,\ldots\}$. This doesn't really explain what a morphism actually *is*, but morphisms are so general that any more specificity is not particularly useful. Defining a morphism is somewhat like defining a vector – are vectors fundamentally geometric arrows embedded in some space, or are they fundamentally ordered lists of coordinates? – the answer being, neither; a vector is anything that obeys the vector space axioms. It's more helpful to define them by the properties they satisfy, rather than what they themselves are, and as we will soon see, this viewpoint will become a recurring pattern. In fact, we should note that objects are in bijection with identity morphisms, so it is possible to define categories entirely in terms of morphisms, and dispense with the objects entirely. We will not do so here, but it is yet another reminder that we will often care more about relations between objects than about objects themselves.

It may be helpful to view a morphism a type of directional relation rather than a function. There is a morphism f from A to B if A is related to B, but B does not have to be related to A, and we write $f : A \to B$, or draw an arrow from A to B on a diagram to represent this.

$$A \xrightarrow{\quad f \quad} B$$

It could be the case that A and B are not related at all, so the collection of morphisms from A to B is empty. We can also have multiple morphisms from A to B if A is related to B in several ways.

$$\begin{tikzcd} A \arrow[r, bend left, "f"] \arrow[r, bend right, "g"'] & B \end{tikzcd}$$

An object can also be related to itself, and in multiple ways at once.

Morphisms must also have a binary operation defined on them, called *composition* that obey the composition law. If there are morphisms $A \xrightarrow{f} B \xrightarrow{g} C$, then the category must also contain a morphism $A \xrightarrow{h=g\circ f} C$. Furthermore, any three morphisms must compose associatively: that is, $(h\circ g)\circ f = h\circ(g\circ f)$ for all morphisms f, g and h (with the appropriate domains and codomains). Categories also require identity morphisms – for every object A, there must exist a morphism $\mathrm{id}_A : A \to A$ such that all morphisms $f : A \to B$ and $g : B \to A$ satisfy $\mathrm{id}_A \circ g = g$ and $f \circ \mathrm{id}_A = f$.

Let $\mathrm{ob}(\mathscr{C}) = \{A\}$, and $\hom(A,A) = \{\mathrm{id}_A\}$. That is, $\mathscr{C}$ is a category containing only one object, and a single morphism from that object to itself. This morphism trivially satisfies the assocativity and identity requirements, so $\mathscr{C}$ is a category, called the *trivial category*, depicted below.

$$A \circlearrowleft \mathrm{id}_A$$

Apart from the trivial category, we usually omit the identity morphism from such diagrams. Conversely, a category which contains no morphisms apart from identity morphisms is called a *discrete category.*

Let $\mathrm{ob}(\mathscr{C}) = \{A,B\}$, and the non-identity morphisms be $\hom(A,B) = \{f\}$ and $\hom(B,A) = \emptyset$. That is, $\mathscr{C}$ is a category containing two objects, and a single non-identity morphism connecting them in one direction only. This is the *arrow category.*

$$A \xrightarrow{\quad f \quad} B$$

Now, let $(G, \cdot\ )$ be a group, $\mathrm{ob}(\mathscr{C}) = \{\bullet\}$, and $\hom(\bullet,\bullet) = G$. For any two morphisms, f and g, we define the composition $f \circ g$ to be $f \cdot g$, so the morphisms have group structure. Because groups require associativity and identities, the morphism axioms are satisfied, and we see that a group is really a one-object category. In fact, there's nothing specific to groups here – we could just have easily started with a monoid or any other algebraic structure with associativity and identities.

These categories are pretty simple, but they give us an idea of how basic categories can be. We will build a more complicated category next: **Set**. Unsurprisingly, the objects of **Set** are sets, while morphisms are ordinary set functions. Composition of morphisms is just regular function composition, and identity morphisms just identity functions which map elements of sets to themselves. The associativity and identity laws follow from elementary properties of function composition. So, **Set** is a category.

Many other commonly used categories follow this formula – that is, their objects are sets with additional structure, and their morphisms are functions that respect that structure. For example, in the category **Grp**, objects are groups, and morphisms are group homomorphisms. Composition and identities are inherited from **Set**, because everything in **Grp** is just a specialised version of something in **Set**. Similarly, **Ring** is the category of rings and ring homomorphisms; **Top**, topological spaces and continuous maps; $R - \mathbf{Vect}$, modules over a ring R and R-linear maps; etc.

However, this doesn't have to be the case, and in general, categories need not have sets as objects and structure-preserving maps as morphisms. We construct a basic example of such a category as follows: the objects in our category will be the real numbers, and for any real numbers, x and y, we define a unique morphism from x to y to exist if and only if $x \leq y$. The problem here, as opposed to in **Set** or **Grp**, is that we can't really say what a morphism really *is*. Here, it's not something that acts on any elements like a function in **Set** or a group homomorphism in **Grp**. In fact, it doesn't really seem to do anything at all, other than exist whenever x is less than or equal to y.

If we have morphisms $f : x \to y$ and $g : y \to z$, then we know $x \leq y$ and $y \leq z$. By transitivity of $\leq$, we have $x \leq z$, giving a unique morphism $h : x \to z$ by definition, which we can assign to the composition $g \circ f$. Because this morphism is unique, this assignment is well-defined and determines the composition of any pair of morphisms. Because $x \leq x$ holds for all x by reflexivity, there is a unique morphism from any element to themselves, which we can use as the identity morphism and the associativity and identity laws follow easily. So, $(\mathbb{R}, \leq)$ is a category. It may be noted that we used nothing specific to the real numbers, so any set equipped with a non-strict preorder is in fact a small category.

We can also construct a new category from a pair of existing categories. Given categories $\mathscr{C}$ and $\mathscr{D}$, the *product category* $\mathscr{C} \times \mathscr{D}$ is defined by $\mathrm{ob}(\mathscr{C} \times \mathscr{D}) = \mathrm{ob}(\mathscr{C}) \times \mathrm{ob}(\mathscr{D})$, and $\hom_{\mathscr{C}\times\mathscr{D}}((A,B),(A',B')) = \hom_{\mathscr{C}}(A,A') \times \hom_{\mathscr{D}}(B,B')$, with compositions defined componentwise [Lan13]. That is, if $A \xrightarrow{f} A'$ and $B \xrightarrow{g} B'$ are objects and morphisms in categories $\mathscr{C}$ and $\mathscr{D}$ respectively, then we have the objects and morphism $(A,B) \xrightarrow{(f,g)} (A',B')$ in the product category $C \times D$. We just take pairs of objects in the constituent categories and pairs of corresponding morphisms between them.

The *principle of duality* states that every categorical definition and theorem has a *dual* definition and theorem, obtained by reversing the direction of all morphisms in the categories involved. We often prefix a dual notion with *co-*, such as in products and coproducts, or domains and codomains.

For instance, every category $\mathscr{C}$ has a *dual* or *opposite* category with the same class of objects, but with the domains and codomains of all morphisms interchanged, denoted $\mathscr{C}^{\mathrm{op}}$. That is, $\mathrm{ob}(\mathscr{C}) = \mathrm{ob}(\mathscr{C}^{\mathrm{op}})$, and $\hom_{\mathscr{C}}(A,B) = \hom_{\mathscr{C}^{\mathrm{op}}}(B,A)$ for all objects A and B. We note that this notion of duality for categories is involutive, so $(\mathscr{C}^{\mathrm{op}})^{\mathrm{op}} = \mathscr{C}$ for all categories $\mathscr{C}$.

Theorem 2.1 (Conceptual Duality)**.** *Let Σ be a statement that holds in all categories. Then the dual statement Σ^* holds for all categories.*

Proof. [Bor+94, adapted] If Σ holds in a category $\mathscr{C}$, then Σ^* holds in $\mathscr{C}^{\mathrm{op}}$. Every category is the dual of its dual, so Σ^* holds in all categories. ■

2.1 Commutative Diagrams

It is often helpful to depict categories visually. We have already been using arrows to show morphisms between objects, but we can do a lot more with these representations. If we take a selection of objects in a category and draw morphisms between them, we can compose morphisms by following a path through the diagram, and because of associativity, each path corresponds to a unique composition.

This is useful enough by itself, but certain diagrams have an additional helpful property. A diagram is *commutative* if, for every pair of objects in the diagram, all routes between them are equivalent. For example, this diagram is commutative if and only if $h = g \circ f$.

$$\begin{tikzcd} X \arrow[r, "f"] \arrow[dr, "h"'] & Y \arrow[d, "g"] \\ & Z \end{tikzcd}$$

Suppose we have objects A and B in a category, and morphisms f from A to B and g from B to A such that the following diagram is commutative.

$$\begin{tikzcd} A \arrow[loop left, "\mathrm{id}_A"] \arrow[r, shift left, "f"] & B \arrow[l, shift left, "g"] \arrow[loop right, "\mathrm{id}_B"] \end{tikzcd}$$

That is, $f \circ g = \mathrm{id}_B$ and $g \circ f = \mathrm{id}_A$. f and g are then *isomorphisms* – morphisms with inverses – and we alternatively label g as f^{-1}. If an isomorphism between A and B exists, we say that A and B are *isomorphic*, and we denote this relation as $A \cong B$.

In the context of **Set**, isomorphisms are exactly the bijections, which is equivalent to the statement that a function has a two-sided inverse if and only if it is bijective. With this, we see that two sets are isomorphic if and only if they contain the same number of elements, possibly labelled in different ways – that is, if their cardinalities are equal. The actual contents of the set, and any extra structure the set has, aren't important in **Set**.

The isomorphisms in **Grp** are group isomorphisms, as you'd might expect, but this result not immediate if the definition of a group isomorphism you use is a "bijective homomorphism"*. To show that group isomorphisms are isomorphisms in **Grp**, we need to prove that the inverse of a bijective homomorphism is also a homomorphism. Similarly, the isomorphisms in **Ring** are exactly the ring isomorphisms.

An isomorphism is the mathematical way of saying that we only care about some specific property of an object. If we're working with the natural numbers, it doesn't matter if we're using the Peano construction or the von Neumann construction, because there are isomorphisms between them that preserve the behaviour of 0, 1, + and $\cdot$, which are the only things that matter for natural numbers (when considered as a semiring). If we're studying groups, then we don't really care about what elements are in each group, only that these elements have group structure. From the point of view of the category, isomorphic elements look the same because they share the only properties that the category cares about.

*Which is not true for say, topological spaces, or the category **Top**, where homomorphisms are continuous functions. However, the inverse of a bijective continuous function is not necessarily continuous, so bijective homomorphisms in **Top** are not necessarily isomorphisms. The isomorphisms in **Top** are instead the *bicontinuous* bijections.

You've probably heard that a topologist cannot tell the difference between a coffee mug and a doughnut. This is because in **Top**, these two objects have the same number of holes (a topological invariant that *does* matter in **Top**), and they can be bicontinuously and bijectively deformed into each other.

2.2 Functors

One central theme of category theory is the idea of mappings between objects. Whenever we encounter a new type of mathematical object, we should always ask if there is a sensible notion of a map between these objects. Of course, categories themselves are mathematical objects we can ask this question on.

Let $\mathscr{C}$ and $\mathscr{D}$ be categories. A *functor*, $F : \mathscr{C} \to \mathscr{D}$, consists of two parts: a mapping on objects, and a mapping on morphisms, that follow two constraints. $F : \mathrm{ob}(\mathscr{C}) \to \mathrm{ob}(\mathscr{D})$ associates each object X in $\mathscr{C}$ to an object, $F(X)$, in $\mathscr{D}$.

$$X \mapsto F(X)$$

Similarly, the map $F : \hom(\mathscr{C}) \to \hom(\mathscr{D})$ associates each morphism $f : X \to Y$ in $\mathscr{C}$ to a morphism $F(f) : F(X) \to F(Y)$ in $\mathscr{D}$ such that:

- $F(\mathrm{id}_X) = \mathrm{id}_{F(X)}$ for every object X in $\mathscr{C}$;
- $F(g \circ f) = F(g) \circ F(f)$ for all morphisms $f : X \to Y$ and $g : Y \to Z$ in $\hom(\mathscr{C})$.

That is, the functor preserves identity morphisms and composition of morphisms.

A more concise way to phrase this is, for every pair of objects $A,B \in \mathrm{ob}(\mathscr{C})$, the functor F induces a mapping $F_{A,B} : \hom_{\mathscr{C}}(A,B) \to \hom_{\mathscr{D}}(F(A),F(B))$ that respects the structure of the categories.

$$\begin{tikzcd}
A \arrow[r, "f"] \arrow[d, phantom, ""] & B \\
F(A) \arrow[r, "F(f)"] & F(B)
\end{tikzcd}$$

$$\begin{array}{c} A \xrightarrow{\quad f \quad} B \\ \big\downarrow \\ F(A) \xrightarrow{F(f)} F(B) \end{array}$$

Theorem 2.2. *Functors preserve commutativity of diagrams.*

Proof. Because functors preserve composition of morphisms, for any two paths $a_1 \circ a_2 \circ \ldots \circ a_n$ and $b_1 \circ b_2 \circ \ldots \circ b_m$ connecting two objects in a commutative diagram of $\mathscr{C}$, we have,

$$\begin{aligned} F(a_1) \circ F(a_2) \circ \ldots \circ F(a_n) &= F(a_1 \circ a_2 \circ \ldots \circ a_n) \\ &= F(b_1 \circ b_2 \circ \ldots \circ b_m) \\ &= F(b_1) \circ F(b_2) \circ \ldots \circ F(b_m) \end{aligned}$$

so the corresponding paths in $\mathscr{D}$ are also equal, and hence the diagram of $\mathscr{D}$ commutes. ■

Corollary 2.2.1. *In particular, functors preserve isomorphism diagrams, so if f is an isomorphism in $\mathscr{C}$, then $F(f)$ is an isomorphism in $\mathscr{D}$.*

One of the most basic examples of a functor is the *constant functor* ΔX which associates every object in $\mathscr{C}$ to a single object $X \in \mathrm{ob}(\mathscr{D})$, and every morphism to id_X. Because every morphism is transformed into the identity morphism on X, composition and identities are trivially preserved, satisfying functoriality.

For a possibly more familiar example, let $(G, \cdot\,)$ and $(H, *\,)$ be groups, interpreted as categories $\mathscr{G}$ and $\mathscr{H}$. Any functor $F : \mathscr{G} \to \mathscr{H}$ must associate the only object in $\mathscr{G}$ to the only object in $\mathscr{H}$, and is thus determined only by its action on the morphisms. The functor must satisfy $F(\mathrm{id}_{\mathscr{G}}) = \mathrm{id}_{\mathscr{H}}$, and $F(g \cdot h) = F(g) * F(h)$ for all morphisms $g,h \in \hom(\mathscr{G})$. So, any functor $\mathscr{G} \to \mathscr{H}$ is just a group homomorphism $G \to H$ (and again, we haven't mentioned inverses, so this holds similarly for monoids).

One very important type of functor is the so-called *forgetful functor*. Forgetful functors do nothing to the objects and morphisms of a category apart from "forgetting" some additional structure that mattered in the original category. For instance, the forgetful functor U : **Grp** $\to$ **Set**.

Every object in **Grp** is a group – a set G with some extra structure in the form of a binary operation and a set of axioms. The forgetful functor U "forgets" this extra structure on objects, and gives $(G,\cdot) \mapsto G$, which is just a set – or rather, an object in **Set**. Similarly, morphisms in **Grp** are group homomorphisms, which are just set functions that happen to respect this extra structure. Forgetting that additional structure still leaves a normal set function – that is, a morphism in **Set**. Since morphisms are effectively unchanged, identity and composition morphisms still exist, so U is a well-defined functor.

Let $\mathscr{C}$ and $\mathscr{D}$ be categories. A *contravariant* functor from $\mathscr{C}$ to $\mathscr{D}$ is a functor $\mathscr{C}^{\mathrm{op}} \to \mathscr{D}$ (or equivalently, a functor $\mathscr{C} \to \mathscr{D}^{\mathrm{op}}$). In contrast, a *covariant* functor from $\mathscr{C}$ to $\mathscr{D}$ is an ordinary functor $\mathscr{C} \to \mathscr{D}$. Informally, a contravariant functor from $\mathscr{C}$ to $\mathscr{D}$ is just an ordinary functor $\mathscr{C}$ to $\mathscr{D}$ that "reverses all morphisms and compositions". This terminology is often used when a named category is involved – it is more convenient to say that a functor is contravariant, than to start writing $\mathbf{Set}^{\mathrm{op}}$ everywhere. However, contravariance can also arise naturally in some constructions:

For instance, the function that sends a set X to its power set $\mathcal{P}(X)$ defines the object mapping of a functor from **Set** to **Set**. We can define its action on morphisms $f : X \to Y$ by mapping f to the *direct image* function $\mathcal{P}(f) : \mathcal{P}(X) \to \mathcal{P}(Y)$ defined by $A \mapsto f(A)$, thus defining the *covariant* power set functor $\mathcal{P}(-) : \mathbf{Set} \to \mathbf{Set}$. However, we could alternatively define the morphism mapping by mapping f to the *inverse image* function $\mathcal{P}(f) : \mathcal{P}(Y) \to \mathcal{P}(X)$ defined by $A \mapsto f^{-1}(A)$. The inverse image function naturally reverses the direction of morphisms, thus defining the *contravariant* power set functor $\mathcal{P}(-) : \mathbf{Set}^{\mathrm{op}} \to \mathbf{Set}$.

2.3 Full and Faithful Functors

For set functions, it is often helpful to consider additional properties the function may satisfy, such as surjectivity and injectivity. There exists a similar notion for functors: let $\mathscr{C}$ and $\mathscr{D}$ be locally small* categories, and let $F : \mathscr{C} \to \mathscr{D}$ be a functor. If for every pair of objects, $A,B \in \mathrm{ob}(\mathscr{C})$, the induced function $F_{A,B} : \hom_{\mathscr{C}}(A,B) \to \hom_{\mathscr{D}}(F(A),F(B))$ is:

- surjective, then F is *full*;
- injective, then F is *faithful*;
- bijective, then F is *fully faithful*.

Note that faithfulness is distinct from injectivity, in that faithful functors are not necessarily injective on objects nor morphisms. For instance, let $\mathscr{C}$ be the discrete category on two objects, A and B, and let $\mathscr{D}$ be the trivial category on an object X. Any functor $F : \mathscr{C} \to \mathscr{D}$ will map the two objects in $\mathscr{C}$ to the unique object of $\mathscr{D}$, and similarly, the identity morphisms on A and B are both mapped to the identity morphism of X, so F is not injective on objects or morphisms. However, the functions $F_{A,A}$ and $F_{B,B}$ each map one morphism to one morphism, and are hence injective (in fact, bijective), while $F_{A,B}$ and $F_{B,A}$ are empty functions, and are hence vacuously injective (but not surjective, as $F(A) = F(B) = X$, and $\hom_{\mathscr{D}}(X,X)$ is non-empty). It follows that F is a faithful functor, but is injective on neither objects nor morphisms. Similarly, full functors are also not necessarily surjective on objects or morphisms, which can be shown by constructing a functor $G : \mathscr{D} \to \mathscr{C}$ in the previous example.

A fully faithful functor that is injective on isomorphism classes of objects is additionally said to be an *embedding*. If there exists an embedding F from $\mathscr{C}$ to $\mathscr{D}$, then F is said to *embed* $\mathscr{C}$ into $\mathscr{D}$.

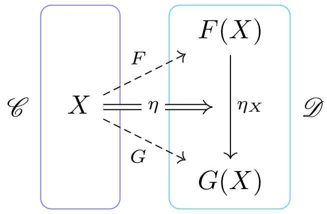
*Because the following definitions are in terms of properties of functions on hom-sets, we require that the hom-sets are indeed sets as these notions are set-theoretic in nature and do not extend readily to proper classes. For large categories, we extend the definition of full and faithful functors to left and right cancellative, respectively, instead.

3 Natural Transformations

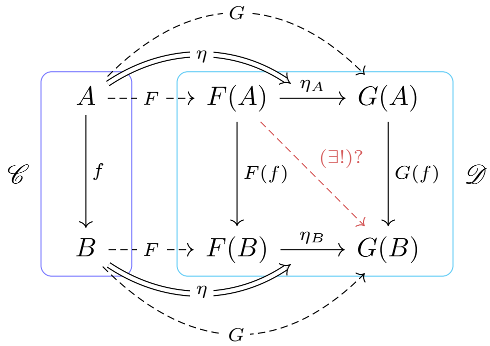
We have morphisms between categories in the form of functors, but the next obvious question to ask is, is there a notion of mappings between functors?

Fix categories $\mathscr{C}$ and $\mathscr{D}$, and let $\mathscr{C} \underset{G}{\overset{F}{\rightrightarrows}} \mathscr{D}$ be functors. A mapping $\begin{tikzcd} \mathscr{C} \arrow[r, bend left, "F"{name=U}] \arrow[r, bend right, "G"'{name=D}] \arrow[Rightarrow, from=U, to=D, "\eta"] & \mathscr{D} \end{tikzcd}$ or $\eta : F \Rightarrow G$ is then called a *natural transformation*.

F and G map objects and morphisms in $\mathscr{C}$ to objects and morphisms in $\mathscr{D}$, respectively. To define a mapping from F to G, we would like to associate objects and morphisms in $\mathscr{D}$ mapped by F to corresponding objects and morphisms in $\mathscr{D}$ mapped by G. For objects, this just means that if X is an object in $\mathscr{C}$, then we wish to associate $F(X)$ with $G(X)$. However, $F(X)$ and $G(X)$ are objects in $\mathscr{D}$, so a relation between them is just a morphism in $\hom_{\mathscr{D}}(F(X),G(X))$. So, η just maps each X in $\mathscr{C}$ to a morphism $F(X) \xrightarrow{\eta_X} G(X)$ called a *component* of η.



However, $\hom_{\mathscr{D}}(F(X),G(X))$ possibly contains many morphisms we could assign to η_X. To help us decide which one to use, consider a morphism $f : A \to B$ in $\mathscr{C}$. Under F and G, f gives the two morphisms $F(f) : F(A) \to F(B)$ and $G(f) : G(A) \to G(B)$. It would seem sensible for $F(f)$ to be related to $G(f)$ under η. From the mapping on objects, we also have $\eta_A : F(A) \to G(A)$ and $\eta_B : F(B) \to G(B)$, giving a square diagram of morphisms.



In this diagram, there are two paths from $F(A)$ to $G(B)$: $\eta_B \circ F(f)$, and $G(f) \circ \eta_A$. Because categories require compositions, these morphisms always exist, but if η_A and η_B were assigned without any other constraints, these compositions are not necessarily equal and there could be multiple distinct morphisms from $F(A)$ to $G(B)$. However, we can use this to relate $F(f)$ with $G(f)$ by enforcing that these compositions are equal, or equivalently, that the diagram commutes. This requirement is the *naturality* condition.

So, for categories $\mathscr{C}$ and $\mathscr{D}$, and functors $\mathscr{C} \underset{G}{\overset{F}{\rightrightarrows}} \mathscr{D}$, a natural transformation $\begin{tikzcd} \mathscr{C} \arrow[r, bend left, "F"{name=U}] \arrow[r, bend right, "G"'{name=D}] \arrow[Rightarrow, from=U, to=D, "\eta"] & \mathscr{D} \end{tikzcd}$ is a collection of morphisms $\left(F(X) \xrightarrow{\eta_X} G(X)\right)_{X \in \mathrm{ob}(\mathscr{C})}$ such that the following diagram commutes:

$$\begin{tikzcd} A \arrow[d, "f"] & F(A) \arrow[r, "\eta_A"] \arrow[d, "F(f)"] & G(A) \arrow[d, "G(f)"] \\ B & F(B) \arrow[r, "\eta_B"] & G(B) \end{tikzcd}$$

That is, $\eta_B \circ F(f) = G(f) \circ \eta_A$ for all $f : A \to B$ in $\hom(\mathscr{C})$.

We next need to verify that these natural transformations actually function as categorical morphisms. That is, that there always exists an identity, and that natural transformations compose associatively.

Following the component definition, the identity natural transformation on a functor $F : \mathscr{C} \to \mathscr{D}$ is a natural transformation $\mathrm{id}_F : F \Rightarrow F$ that maps each $X \in \mathrm{ob}(\mathscr{C})$ to a morphism $F(X) \xrightarrow{(\mathrm{id}_F)_X} F(X)$. This is just the identity morphism on $F(X)$, which always exists, so every component of id_F also always exists. The diagram, consisting of a single morphism and two identities, then trivially commutes, satisfying naturality, and hence id_F always exists. Identities, however, need to compose with other morphisms, and leave them unchanged. How do natural transformations compose?

3.1 Vertical Composition

Fix categories $\mathscr{C}$ and $\mathscr{D}$, and let F,G,$H : \mathscr{C} \to \mathscr{D}$ be functors. Consider the natural transformations $\alpha : F \Rightarrow G$ and $\beta : G \Rightarrow H$.

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

From the diagram, it would seem sensible to define the composition $\beta \circ \alpha$ to be a map from F to H. Such a composition of natural transformations is called a *vertical composition*.

Consider an object X in $\mathscr{C}$. The two components of α and β at X are then $\alpha_X : F(X) \to G(X)$ and $\beta_X : G(X) \to H(X)$. Because these are just morphisms in $\mathscr{D}$, they can be composed according to regular morphisms composition rules, and so, we can define the component $(\beta \circ \alpha)_X$ to be $\beta_X \circ \alpha_X : F(X) \to H(X)$. Because identity natural transformations map objects to identity morphisms, this also verifies that they do in fact function as identities with respect to vertical composition.

However, it remains to show that these components satisfy the naturality requirement.

$$\begin{tikzcd}
A \arrow[d, "f"] & F(A) \arrow[r, "\alpha_A"] \arrow[d, "F(f)"] & G(A) \arrow[r, "\beta_A"] \arrow[d, "G(f)"] & H(A) \arrow[d, "H(f)"] \\
B & F(B) \arrow[r, "\alpha_B"] & G(B) \arrow[r, "\beta_B"] & H(B)
\end{tikzcd}$$

Because α and β are natural transformations, they individually satisfy the naturality requirement, so each square commutes individually, and hence the diagram as a whole also commutes.

For any two categories, we can now define functors between them, and natural transformations between those functors that obey the morphism axioms. This suggests the construction of a new category, where the objects are functors, and the morphisms are natural transformations.

Let $\mathscr{C}$ and $\mathscr{D}$ be categories. We construct the *functor category* $[\mathscr{C},\mathscr{D}]$ by taking objects to be functors from $\mathscr{C}$ to $\mathscr{D}$, morphisms to be natural transformations, composition of morphisms to be vertical composition of natural transformations, and identity morphisms to be identity natural transformations.

Given the name of vertical composition, it is unsurprising that we have a notion of *horizontal composition*, but its discussion is relegated to § 6.2 in the addendum.

3.2 Natural Isomorphisms

Fix categories $\mathscr{C}$ and $\mathscr{D}$. A *natural isomorphism* between functors from $\mathscr{C}$ to $\mathscr{D}$ is an isomorphism in the functor category $[\mathscr{C},\mathscr{D}]$.

That is, $\eta : F \Rightarrow G$ is an natural isomorphism if η is a natural transformation and there exists a natural transformation $\vartheta : G \Rightarrow F$ such that $\vartheta \circ \eta = \mathrm{id}_F$ and $\eta \circ \vartheta = \mathrm{id}_G$, and we write η^{-1} for ϑ.

$$\begin{tikzcd} A \arrow[loop left, "\mathrm{id}_A"] \arrow[r, shift left, "f"] & B \arrow[l, shift left, "f^{-1}"] \arrow[loop right, "\mathrm{id}_B"] \end{tikzcd} \qquad \leftarrow \text{cf.} \rightarrow \qquad \begin{tikzcd} & \mathscr{C} \arrow[ddl, bend right] \arrow[ddr, bend left] & \\ F \arrow[loop left, Rightarrow, "\mathrm{id}_F"] \arrow[rr, Rightarrow, shift left, "\eta"] & & G \arrow[ll, Rightarrow, shift left, "\eta^{-1}"] \arrow[loop right, Rightarrow, "\mathrm{id}_G"] \\ & \mathscr{D} & \end{tikzcd}$$

In this case, we say F and G are *naturally isomorphic*, and because natural isomorphisms are just isomorphisms in a specific type of category, we reuse notation and write $F \cong G$, or we say that $F(X) \cong G(X)$ *naturally in* X whenever we need to bind a variable.

The next theorem gives an alternative characterisation of natural isomorphisms.

Theorem 3.1. *Let* $\begin{tikzcd} \mathscr{C} \arrow[r, bend left, "F"{name=U}] \arrow[r, bend right, "G"'{name=D}] & \mathscr{D} \arrow[from=U, to=D, Rightarrow, "\eta"] \end{tikzcd}$ *be a natural transformation. Then, η is a natural isomorphism if and only if $\eta_X : F(X) \to G(X)$ is an isomorphism for all $X \in \mathrm{ob}(\mathscr{C})$.*

Proof. Suppose η is a natural isomorphism, so there exists ϑ such that $\vartheta \circ \eta = \mathrm{id}_F$. Then, $(\vartheta \circ \eta)_X = \vartheta_X \circ \eta_X = (\mathrm{id}_F)_X$ for all $X \in \mathrm{ob}(\mathscr{C})$, so every component is an isomorphism, completing the forward implication.

Now, suppose that $\eta_X : F(X) \to G(X)$ is an isomorphism for all $X \in \mathrm{ob}(\mathscr{C})$. Define $\vartheta : G \Rightarrow F$ by $\vartheta_X = (\eta_X)^{-1}$. Because η is a natural transformation, we have $\eta_B \circ F(f) = G(f) \circ \eta_A$. Left and right multiplying by ϑ_B and ϑ_A respectively, we have, $F(f) \circ \vartheta_A = \vartheta_B \circ G(f)$ which is exactly the naturality condition, and hence ϑ is a natural transformation. Then, $\vartheta \circ \eta = \left(F(X) \xrightarrow{\vartheta_X \circ \eta_X} F(X)\right)_{X \in \mathrm{ob}(\mathscr{C})} = \mathrm{id}_F$, and $\eta \circ \vartheta = \left(G(X) \xrightarrow{\eta_X \circ \vartheta_X} G(X)\right)_{X \in \mathrm{ob}(\mathscr{C})} = \mathrm{id}_G$, and hence η is a natural isomorphism, completing the backward implication. ∎

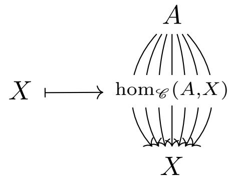
In the reverse direction, we used that η is a natural transformation to obtain naturality for ϑ. Without this, it could still be the case that $F(X) \cong G(X)$ for all X, but there does not exist a natural transformation from F to G at all, so "$F(X) \cong G(X)$ naturally in X" is a much stronger condition than just "$F(X) \cong G(X)$ for all X".

4 Hom-Functors

Suppose we wish to study the properties of an object A in a locally small category $\mathscr{C}$. One way to do so is to look at A from a different object, X. Then, look at A from another object, Y, and repeat. By looking at how A is seen by other objects, we can obtain a lot of information about A. The relationships an object X has with A are exactly the hom-sets $\hom(A,X)$ and $\hom(X,A)$, but these sets are different for each X. In fact, in locally small categories, this assignment of hom-sets with respect to a fixed A is functorial in X. That is, given a fixed A, every morphism $X \to Y$ induces a function $\hom_{\mathscr{C}}(A,X) \to \hom_{\mathscr{C}}(A,Y)$.

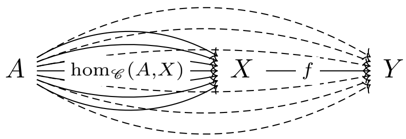
Let $\mathscr{C}$ be a locally small category, and fix an object $A \in \mathrm{ob}(\mathscr{C})$. We define the (covariant) *hom-functor*, $\hom(A,-) : \mathscr{C} \to \mathbf{Set}$, also denoted h_A, as follows.

For each object $X \in \mathrm{ob}(\mathscr{C})$, we define $\hom(A,-)(X) = \hom_{\mathscr{C}}(A,X)$, so each object is mapped to the set of maps from A to that object.



We can interpret this as h_A mapping objects to how they are "seen" by A.

For each morphism $f : X \to Y$, we define $\hom(A,-)(f)$ to be the function $\hom(A,f) : \hom_{\mathscr{C}}(A,X) \to \hom_{\mathscr{C}}(A,Y)$, also denoted $h_A(f)$, defined by the postcomposition $g \mapsto f \circ g$.



[nLa23, adapted]

That is, we map each morphism $X \xrightarrow{f} Y$ in $\hom(\mathscr{C})$ to the function $h_A(f)$ that maps each morphism $A \xrightarrow{g} X$ to the composite morphism $A \xrightarrow{g} X \xrightarrow{f} Y$. In the above diagram, the morphisms on the left are "combed" through to Y through f, and we can interpret this as h_A mapping morphisms $X \xrightarrow{f} Y$ to how A "sees" the object Y through f.

The contravariant hom-functor $\hom(-,B)$, also denoted* h^B, is defined dually, with h^B mapping objects and morphisms to how they see B, rather than how they are seen from B.

$$\begin{array}{ccc}
\mathscr{C} & \xrightarrow{h_A} & \mathbf{Set} \\
\\
X & \mapsto & \hom_{\mathscr{C}}(A,X) \\
{\scriptstyle f}\downarrow & \mapsto & \downarrow{\scriptstyle h_A(f)} \\
Y & \mapsto & \hom_{\mathscr{C}}(A,Y)
\end{array}
\qquad
\begin{array}{ccc}
\mathscr{C}^{\mathrm{op}} & \xrightarrow{h^B} & \mathbf{Set} \\
\\
X & \mapsto & \hom_{\mathscr{C}}(X,B) \\
{\scriptstyle f}\downarrow & \mapsto & \uparrow{\scriptstyle h^B(f)} \\
Y & \mapsto & \hom_{\mathscr{C}}(Y,B)
\end{array}$$

[Rie17, adapted]

Theorem 4.1. *h_A is a functor.*

Proof. We verify the functor axioms.

- Let $A \xrightarrow{f} X$ be a morphism. Then,

$$\begin{aligned}\big[h_A(\mathrm{id}_X)\big](f) &= \mathrm{id}_X \circ f \\ &= \mathrm{id}_{h_A(X)}(f)\end{aligned}$$

  so h_A preserves identities.

- Let $A \xrightarrow{h} X \xrightarrow{g} Y \xrightarrow{f} Z$ be morphisms. Then,

$$\begin{aligned}\big[h_A(g \circ f)\big](h) &= (g \circ f) \circ h \\ &= g \circ (f \circ h) \\ &= \big[h_A(g)\big](f \circ h) \\ &= \big[h_A(g)\big]\Big(\big[h_A(f)\big](h)\Big) \\ &= \big[h_A(g) \circ h_A(f)\big](h)\end{aligned}$$

  and h_A preserves compositions. ■

*The usage of h_X for the covariant hom-functor and h^X for the contravariant hom-functor is not standardised. Some texts – notably [Lei14] – reverse the labelling, or use different notation entirely.

Corollary 4.1.1. *By duality, h^B is also a functor.*

For each object A, we have assigned a functor h_A, encapsulating how the category is seen from A, and as A varies, this view varies. However, it is the same category being seen from all objects, so it wouldn't be unusual for us to expect that this assignment has some internal consistency.

As it turns out, any morphism $f : A \to B$ induces a natural transformation $h_f : h_B \Rightarrow h_A$. Note the change in direction here! A collection of covariant functors come together to define a contravariant natural transformation. And, if we started with the contravariant hom-functors, they would all come together to define a covariant natural transformation.

Consider the component $h_B(X) \to h_A(X)$ of h_f at an object X. Recall that a map $h_B(X) \to h_A(X)$ just sends morphisms $B \to X$ to $A \to X$. We can interpret these hom-sets as contravariant hom-functors at a fixed X, so we're really just looking for a morphism $h^X(B) \to h^X(A)$, which is given exactly by precomposition by f. That is, each morphism $g : B \to X$ is mapped to the morphism $g \circ f : A \to X$.

In fact, there's no reason why we should have to fix one argument at a time. The notation hom$(A,-)$ and hom$(-,B)$ suggests that we may take both inputs to the hom-functor to be variable. Let $f : X \to Y$ and $h : B \to A$ be morphisms, and consider the following diagram:

$$\begin{tikzcd} \hom(A,X) \arrow[r, "\hom(h,X)"] \arrow[d, "\hom(A,f)"'] & \hom(B,X) \arrow[d, "\hom(B,f)"] \\ \hom(A,Y) \arrow[r, "\hom(h,Y)"'] & \hom(B,Y) \end{tikzcd}$$

Consider a morphism $g \in \hom(A,X)$. We will follow how it is mapped under this square along the two different paths, in a technique called *diagram chasing*.

Along the upper path, we have $g \mapsto \hom(h,X)(g) = g \circ h \mapsto \hom(B,f)(g \circ h) = f \circ (g \circ h)$. Along the lower path, we have $g \mapsto \hom(A,f)(g) = f \circ g \mapsto \hom(h,Y)(f \circ g) = (f \circ g) \circ h$. But, by associativity of morphism composition, these paths are equal, and we see that this diagram commutes for any choice of f, g, and h, implying that $\hom(-,-)$ is a functor $\mathscr{C}^{\text{op}} \times \mathscr{C} \to \mathbf{Set}$.

A functor $F : \mathscr{C} \to \mathbf{Set}$ is *representable* if $F \cong h_X$ (or h^X) for at least one choice of $X \in \text{ob}(\mathscr{C})$. The object X along with the natural transformation $F \Rightarrow h_X$ are then a *representation* of F. As it turns out, the object X is determined uniquely up to isomorphism in $\mathscr{C}$. We often call representable functors just *representables*.

As an example of a representable, the identity functor $\text{id}_{\mathbf{Set}} : \mathbf{Set} \to \mathbf{Set}$ is represented by the singleton set $\mathbf{1}$. Any function $\mathbf{1} \to X$ just picks elements from the set X, so there are exactly as many functions $\mathbf{1} \to X$ as there are elements of X, giving $\hom_{\mathbf{Set}}(\mathbf{1},X) \cong X = \text{id}_{\mathbf{Set}}(X)$, as required. Naturality also follows trivially as half of the functions to be considered are identities.

For a more interesting example, the forgetful functor $U : \mathbf{Grp} \to \mathbf{Set}$ is represented by the group $\mathbb{Z}$. Let G be a group. Because group homomorphisms send identities to identities, $0 \in \mathbb{Z}$ is always sent to the identity in G, so any homomorphism $\phi : \mathbb{Z} \to G$ is determined entirely by the image of 1 with the rest of the map following from the cyclic nature of $\mathbb{Z}$. This suggests that we send each homomorphism $\phi : \mathbb{Z} \to G$ to its determining value $\phi(1)$, giving us the components of a map $\alpha : \hom_{\mathbf{Grp}}(\mathbb{Z},-) \Rightarrow U$. The inverse map is then given by sending each $g \in U(G)$ to the homomorphism $z \mapsto g^z$. But, we still need naturality of this isomorphism.

Let $f : G \to H$ be a group homomorphism.

$$\begin{tikzcd} G \arrow[d, "f"] & \hom_{\mathbf{Grp}}(\mathbb{Z},G) \arrow[r, "\alpha_G"] \arrow[d, "h_{\mathbb{Z}}(f)"'] & U(G) \arrow[d, "U(f)"] \\ H & \hom_{\mathbf{Grp}}(\mathbb{Z},H) \arrow[r, "\alpha_H"'] & U(H) \end{tikzcd}$$

We will chase a homomorphism $\phi : \mathbb{Z} \to G$ through the diagram. Along the upper path, we have $\big(U(f) \circ \alpha_G\big)(\phi) = (f \circ \alpha_G)(\phi) = f(\alpha_G(\phi)) = f(\phi(1)) = (f \circ \phi)(1)$, and along the lower, we have, $\big(\alpha_H \circ h_{\mathbb{Z}}(f)\big)(\phi) = \alpha_H\big(h_{\mathbb{Z}}(f)(\phi)\big) = \big(h_{\mathbb{Z}}(f)(\phi)\big)(1) = (f \circ \phi)(1)$, so the diagram commutes, and $\hom_{\mathbf{Grp}}(\mathbb{Z},G) \cong U(G)$ naturally in G, as required. Through similar arguments, the forgetful functor $U : \mathbf{Ring} \to \mathbf{Set}$ is represented by the polynomial ring $\mathbb{Z}[x]$, and $U : \mathbf{Mon} \to \mathbf{Set}$ by the monoid $\mathbb{N}_0$ (you might notice that these are all free algebras on single generators – this is not a coincidence, § 6.3).

As another example, the contravariant power set functor $\mathcal{P} : \mathbf{Set}^{\text{op}} \to \mathbf{Set}$ sending sets to their power sets and functions to their inverse image is represented by the two element set $\mathbf{2}$, often depicted as $\{\top,\bot\}$ or $\{0,1\}$ with morphisms interpreted as an indicator functions of elements [Rie17].

For an example of a non-representable functor [Dot23], consider the functor $F : \mathbf{Set} \to \mathbf{Set}$ defined on objects by $X \mapsto X \coprod X$. Suppose there exists a set Y such that $\hom_{\mathbf{Set}}(Y,X) \cong X \coprod X$. If $X = \mathbf{1}$, then $\hom_{\mathbf{Set}}(Y,\mathbf{1}) \cong \{\mathbf{1}\} \cong \mathbf{1}$ is a singleton set, while $\mathbf{1} \coprod \mathbf{1} = \{\{0,1\},\{1,1\}\} \cong \mathbf{2}$ is a set with two elements, so they are not isomorphic and hence no such Y exists.

5 The Yoneda Lemma

It is an almost universal meta-problem in all of mathematics to describe and classify collections of mathematical objects [Hal20]. While a mathematical axiomatic definition of an object certainly distinguishes that object away from any others, this doesn't tell us much about the collection of all those objects as a whole. For example, while we can define a group in four short axioms, classifying all groups is a much harder problem. For a simpler example, imagine we are tasked with classifying the real numbers. The real number line is a classification of all real numbers by embedding them in some space that has more properties than the real numbers had alone. For instance, the number line is a metric space, a topological space, etc.

While we can define real number with Dedekind cuts, or with completeness axioms, this kind of embedding gives a lot of additional useful information that isn't visible from the axioms alone. Importantly, there is a bijection between the points on the number line and real numbers, but we also have the new information in that real numbers near each other on the number line are similar in magnitude. We can try extend this idea of a classifying space to other kinds of objects, where "nearby" objects have more similar properties than "distant" objects, and more generally, these spaces are called *moduli spaces* [Hal20]. Unfortunately, the moduli space for any kind of useful object is often completely unrecognisable, and has very few properties we can leverage to our advantage.

However, we can attempt to examine these spaces by looking at the maps from other spaces to them. Let $\mathbf{1}$ be the set with one element. Any map from $\mathbf{1}$ to $\mathbb{R}$ effectively amounts to picking an element from $\mathbb{R}$, so there is a bijection between the functions $\mathbf{1} \to \mathbb{R}$ and the points in $\mathbb{R}$. In fact, there's nothing specific about $\mathbb{R}$ here. More generally, the maps from the one-point space $\mathbf{1}$ to any space X amount to picking points from X. If X is, for example, a metric or topological space, then it is a set equipped with some extra structure in the form of a metric or a topology. By examining the maps from $\mathbf{1}$ to X, we can recover half of that information: just by looking at X from the simplest possible (non-empty) space, we recover all the points of X.

What if we look at the maps from a more complicated space? A map from the interval $[0,1]$ to X is just some parametrisation of a curve in X, so the maps $[0,1] \to X$ recover the paths in X, while a map from the circle S^1 to X is just a topological loop, so the maps $S^1 \to X$ recover the homotopy classes of loops on X. The point is, we get more and more information about X by examining how it appears from different choices of domains.

But exactly how much information can we recover? Is it always possible to obtain as much data from looking at maps as we would from just analysing the space itself? After all, we have no reason to expect that the entire structure of the space is always captured by these maps.

Except, it always is – and that, is the Yoneda lemma.*

*Or at least, part of it – it says a lot of things. The Yoneda lemma is very powerful in more advanced category theory, The remarkable thing is that the Yoneda lemma is a proof at the level of categories, so it holds for any category of spaces.

We begin the lemma by asking what information representables recover. More precisely, let $\mathscr{C}$ be a locally small category, and fix an object $A \in \mathrm{ob}(\mathscr{C})$, which induces the representable covariant functor h_A. For each covariant functor F, what are the natural transformations $h_A \Rightarrow F$ in the functor category $[\mathscr{C},\mathbf{Set}]$?

Lemma 5.1 (Yoneda)**.** *Let $\mathscr{C}$ be a locally small category. Then,*

$$\hom_{[\mathscr{C},\mathbf{Set}]}(h_A,F) \cong F(A)$$

naturally in $F \in \mathrm{ob}([\mathscr{C},\mathbf{Set}])$ and $A \in \mathrm{ob}(\mathscr{C})$.

Before we proceed with the proof, we should unwrap what this is saying, in exact terms. Firstly, there is an isomorphism of sets, so there is a bijective function between $\hom_{[\mathscr{C},\mathbf{Set}]}(h_A,F)$ and $F(A)$ – there are as many natural transformations from h_A to F as there are elements of $F(A)$. Moreover, the collection of natural transformations between two functors isn't guaranteed to be a set, even if the two associated categories are (locally) small, so the lemma also shows that hom-sets of this form can be put into bijection with proper sets.

Next, the isomorphism is said to be natural in F and A, suggesting that both sides are functorial in *both F and A* – any morphisms $F \Rightarrow G$ and $A \to B$ must induce maps

$$\hom_{[\mathscr{C},\mathbf{Set}]}(h_A,F) \to \hom_{[\mathscr{C},\mathbf{Set}]}(h_B,G) \quad \text{and} \quad F(A) \to G(B)$$

and not only does the isomorphism hold for every F and A, there exist isomorphisms $\hom_{[\mathscr{C},\mathbf{Set}]}(h_A,F) \to F(A)$ and $\hom_{[\mathscr{C},\mathbf{Set}]}(h_B,G) \to G(B)$ such that the induced square commutes for any choice of F and A.

More precisely, we can regard the left and right sides of the expression as bifunctors $[\mathscr{C},\mathbf{Set}] \times \mathscr{C} \to \mathbf{Set}$, mapping (F,A) to $\hom_{[\mathscr{C},\mathbf{Set}]}(h_A,F)$ and $F(A)$, respectively (in particular, this latter functor is known as the *evaluation functor*), and the Yoneda lemma states that these functors are naturally isomorphic.

Proof. Let $\eta : h_A \Rightarrow F$ be a natural transformation. Consider the following diagram:

$$\begin{tikzcd} A \arrow[d, "f"] & h_A(A) \arrow[r, "\eta_A"] \arrow[d, "h_A(f)"'] & F(A) \arrow[d, "F(f)"] \\ B & h_A(B) \arrow[r, "\eta_B"'] & F(B) \end{tikzcd}$$

We chase the identity $\mathrm{id}_A \in \hom(A,A) = h_A(A)$ through the diagram. Along the upper path, we have $\mathrm{id}_A \mapsto \eta_A(\mathrm{id}_A) \mapsto F(f)(\eta_A(\mathrm{id}_A))$. Along the lower path, we have $\mathrm{id}_A \mapsto h_A(f)(\mathrm{id}_A) = f \circ \mathrm{id}_A = f$, followed by $f \mapsto \eta_B(f)$. From naturality of η, this diagram is commutative, so these two paths must be equal, giving $\eta_B(f) = F(f)(\eta_A(\mathrm{id}_A))$.

Remarkably, the input to the function on the right side is always $\eta_A(\mathrm{id}_A)$. This implies that any natural transformation $h_A \Rightarrow F$ is completely determined by its value at id_A. This naturally induces a function $\hom_{[\mathscr{C},\mathbf{Set}]}(h_A,F) \to F(A)$ defined by $\eta \mapsto \eta_A(\mathrm{id}_A)$, and moreover, this function is a bijection, as every value in $F(A)$ conversely extends to a unique natural transformation.

This establishes the required isomorphism, but we still need to show naturality.

First, we write both sides as functors $\vartheta, \mathrm{ev} : [\mathscr{C},\mathbf{Set}] \times \mathscr{C} \to \mathbf{Set}$. As mentioned before, the action of the two functors on objects is given by,

$$\vartheta(F,A) = \hom_{[\mathscr{C},\mathbf{Set}]}(h_A,F) \quad \text{and} \quad \mathrm{ev}(F,A) = F(A)$$

respectively. Now, we need to define their action on morphisms.

but this is one elementary application of it.

Being a product category, every morphism $(F,A) \to (G,B)$ in $[\mathscr{C},\mathbf{Set}] \times \mathscr{C}$ is of the form (α,f), where $\alpha : F \Rightarrow G$ is a morphism in $[\mathscr{C},\mathbf{Set}]$, and $f : A \to B$ is a morphism in $\mathscr{C}$. Fix two such morphisms, $\alpha : F \Rightarrow G$ and $f : A \to B$.

The first functor, ϑ, sends (α,f) to a function $\vartheta(\alpha,f) : \hom_{[\mathscr{C},\mathbf{Set}]}(h_A,F) \to \hom_{[\mathscr{C},\mathbf{Set}]}(h_B,G)$ defined by mapping each $\varphi : h_A \Rightarrow F$ to the composition $h_B \xrightarrow{h_f} h_A \xrightarrow{\varphi} F \xrightarrow{\alpha} G$. That is, $[\vartheta(\alpha,f)](\varphi) = \alpha \circ \varphi \circ h_f$.

The second functor, ev, sends the morphism (α,f) to a function $\mathrm{ev}(\alpha,f) : F(A) \to G(B)$. At this point, we should recall that α is a natural transformation, so the following diagram commutes:

$$\begin{tikzcd}
A \arrow[d, "f"] & F(A) \arrow[r, "\alpha_A"] \arrow[d, "F(f)"] & G(A) \arrow[d, "G(f)"] \\
B & F(B) \arrow[r, "\alpha_B"] & G(B)
\end{tikzcd}$$

From this, we see that there are two paths from $F(A)$ to $G(B)$, namely, $F(A) \xrightarrow{\alpha_A} G(A) \xrightarrow{G(f)} G(B)$, and $F(A) \xrightarrow{F(f)} F(A) \xrightarrow{\alpha_B} G(B)$. But, from naturality, these compositions are equal, so either choice yields the desired map. Next, we verify the functor axioms for ϑ and ev. First, the identity law:

$$\begin{aligned}
\vartheta(\mathrm{id}_F\,,\mathrm{id}_A)(\varphi) &= \mathrm{id}_F \circ \varphi \circ h_{\mathrm{id}_A} & \qquad \mathrm{ev}(\mathrm{id}_F\,,\mathrm{id}_A) &= F(\mathrm{id}_A) \circ (\mathrm{id}_A)_A \\
&= \varphi & &= \mathrm{id}_{F(A)} \circ \mathrm{id}_{F(A)} \\
&= \mathrm{id}_{[\hom(H_A,F)]}(\varphi) & &= \mathrm{id}_{F(A)}
\end{aligned}$$

where the first term on the right follows from the functoriality of F. So, ϑ and ev preserve identities.

Now, let $(F,A) \xrightarrow{(\alpha,f)} (G,B) \xrightarrow{(\beta,g)} (H,C)$ be morphisms.

$$\begin{aligned}
\vartheta\big((\beta,g) \circ (\alpha,f)\big)(\varphi) &= \vartheta(\beta \circ \alpha, g \circ f)(\varphi) & \qquad \mathrm{ev}\big((f,g) \circ (\alpha,f)\big) &= \mathrm{ev}(\beta \circ \alpha, g \circ f) \\
&= (\beta \circ \alpha) \circ \varphi \circ h_{g \circ f} & &= H(g \circ f) \circ (\beta \circ \alpha)_A \\
&= (\beta \circ \alpha) \circ \varphi \circ (h_f \circ h_g) & &= H(g) \circ H(f) \circ \beta_A \circ \alpha_A \\
&= \beta \circ (\alpha \circ \varphi \circ h_f) \circ h_g & &= H(g) \circ \beta_B \circ G(f) \circ \alpha_A \\
&= \beta \circ \big(\vartheta(\alpha,f)(\varphi)\big) \circ h_g & &= \big(H(g) \circ \beta_B\big) \circ \big(G(f) \circ \alpha_A\big) \\
&= \big[\vartheta(\beta,g) \circ \vartheta(\alpha,f)\big](\varphi) & &= \mathrm{ev}(\beta,g) \circ \mathrm{ev}(\alpha,f)
\end{aligned}$$

On the left, the expansion of $h_{g \circ f}$ follows from functoriality, with the reversal of the components resulting from contravariance. On the right, the expansion of $(\beta \circ \alpha)_A$ follows from the definition of vertical composition, and the replacement of $H(f) \circ \beta_A$ with $\beta_B \circ G(f)$ follows from the naturality of β. This last point is perhaps clearer as a diagram chase:

$$\begin{tikzcd}
F(A) \arrow[r, "(\beta\circ\alpha)_A"] \arrow[d, "F(g\circ f)"] \arrow[dr, "\mathrm{ev}((\beta\circ\alpha),(g\circ f))"] & H(A) \arrow[d, "H(g\circ f)"] \\
F(C) \arrow[r, "(\beta\circ\alpha)_A"] & H(C)
\end{tikzcd}
\qquad
\begin{tikzcd}
F(A) \arrow[r, "\alpha_A"] \arrow[d, "F(f)"] \arrow[dr, "\mathrm{ev}(\alpha,f)"] & G(A) \arrow[r, "\beta_A"] \arrow[d, "G(f)"] & H(A) \arrow[d, "H(f)"] \\
F(B) \arrow[r, "\alpha_B"] \arrow[d, "F(g)"] & G(B) \arrow[r, "\beta_B"] \arrow[d, "G(g)"] \arrow[dr, "\mathrm{ev}(\beta,g)"] & H(B) \arrow[d, "H(g)"] \\
F(C) \arrow[r, "\alpha_C"] & G(C) \arrow[r, "\beta_C"] & H(C)
\end{tikzcd}$$

The first two lines of the equation correspond to taking the upper path along the left diagram. The expansion in the third corresponds the uppermost path through the right diagram that passes through

$H(A)$. But the upper right square commutes by the naturality of β, so we may route through $G(B)$ instead of $H(A)$. But then, this is just the route created by taking $\mathrm{ev}(\alpha,f)$ followed by $\mathrm{ev}(\beta,g)$, as required. If we take the other definition of the evaluation functor, we similarly use the functoriality of α along the lower path.

So, ϑ and ev preserve composition, finally verifying functoriality.

Now, we define a natural transformation $\Phi : \vartheta \Rightarrow \mathrm{ev}$. As stated earlier, we will map a natural transformation $\eta \in \vartheta(F,A)$ to its determining value $\eta_A(\mathrm{id}_A) \in \mathrm{ev}(F,A)$, giving us our definition of the component $\Phi_{(F,A)}$. All that remains is to show naturality:

$$\begin{tikzcd}
(F,A) \arrow[d, "{(\alpha,f)}"] & \vartheta(F,A) \arrow[r, "\Phi_{(F,A)}"] \arrow[d, "{\vartheta(\alpha,f)}"] & \mathrm{ev}(F,A) \arrow[d, "{\mathrm{ev}(\alpha,f)}"] \\
(G,B) & \vartheta(G,B) \arrow[r, "\Phi_{(G,B)}"] & \mathrm{ev}(G,B)
\end{tikzcd}$$

$$\begin{aligned}
\big(\mathrm{ev}(\alpha,f) \circ \Phi_{(F,A)}\big)(\eta) &= \mathrm{ev}(\alpha,f)\big(\eta_A(\mathrm{id}_A)\big) \\
&= \big(\alpha_B \circ F(f)\big)\big(\eta_A(\mathrm{id}_A)\big) \\
&= (\alpha_B \circ \eta_B \circ h_A(f))(\mathrm{id}_A) \\
&= (\alpha_B \circ \eta_B)\big(h_A(f)(\mathrm{id}_A)\big) \\
&= (\alpha_B \circ \eta_B)(f \circ \mathrm{id}_A) \\
&= (\alpha_B \circ \eta_B)(f) \\
&= (\alpha_B \circ \eta_B)(\mathrm{id}_B \circ f) \\
&= (\alpha_B \circ \eta_B)\big((h_f)_B(\mathrm{id}_B)\big) \\
&= (\alpha_B \circ \eta_B \circ (h_f)_B)(\mathrm{id}_B) \\
&= (\alpha \circ \eta \circ h_f)_B(\mathrm{id}_B) \\
&= \Phi_{(G,B)}(\alpha \circ \eta \circ h_f) \\
&= \big(\Phi_{(G,B)} \circ \vartheta(\alpha,f)\big)(\eta)
\end{aligned}$$

so the diagram commutes, as required. ■

5.1 The Yoneda Embedding

An important case of the Yoneda lemma is when the functor F is another hom-functor, h_B:

$$\hom_{[\mathscr{C},\mathbf{Set}]}(h_A,h_B) \cong \hom(B,A)$$

That is, the natural transformations between the two covariant hom-functors induced by A and B are in bijection with the morphisms between A and B in reverse direction: this is a contravariant(!) functor $\mathscr{C}^{\mathrm{op}} \to [\mathscr{C},\mathbf{Set}]$. This functor is denoted $h_\bullet$, defined on objects A by $h_\bullet(A) = h_A$, and on morphisms f by $h_\bullet(f) = h_f$. Similarly, applying the contravariant version of the Yoneda lemma to a contravariant hom-functor naturally gives rise to the covariant functor $h^\bullet : \mathscr{C} \to [\mathscr{C}^{\mathrm{op}},\mathbf{Set}]$.

In this context, the Yoneda lemma simply says that the functor $h_\bullet$ gives an embedding of $\mathscr{C}^{\mathrm{op}}$ into $[\mathscr{C},\mathbf{Set}]$. These functors are called the *Yoneda embeddings*, and are often denoted よ (hiragana *yo*) for Yoneda (from this point onwards, we will use よ wherever a proof applies to either functor).

Theorem 5.2 (Yoneda Embedding)**.** *Let $\mathscr{C}$ be a locally small category. Then, the Yoneda embeddings よ $: \mathscr{C} \hookrightarrow [\mathscr{C}^{\mathrm{op}},\mathbf{Set}]$ and よ $: \mathscr{C}^{\mathrm{op}} \hookrightarrow [\mathscr{C},\mathbf{Set}]$ are embeddings – that is, よ is fully faithful, and injective on objects up to isomorphism.*

Proof. よ is fully faithful if the induced mapping $よ_{A,B} : \hom(A,B) \to \hom(よ(A), よ(B))$ is a bijection for all objects $A,B \in \mathrm{ob}(\mathscr{C})$. But this is just the statement of the Yoneda lemma applied to hom-functors. Injectivity on objects up to isomorphism is proved in the corollary. ∎

Corollary 5.2.1. *If* $\hom(X,-) \cong \hom(Y,-)$ *or* $\hom(-,X) \cong \hom(-,Y)$*, then* $X \cong Y$*.*

Proof. By the Yoneda lemma, any natural transformation $\eta : h_\bullet(X) \to h_\bullet(Y)$ (dually, $h^\bullet$) is induced by a morphism $Y \to X$ (resp. $X \to Y$). If η is an isomorphism, it follows that η and η^{-1} are both induced by inverse morphisms between X and Y, so $X \cong Y$. ∎

At the beginning of this section, we asked how much information we get when we examine how an object looks from all other possible viewpoints. This corollary states that we recover the object, up to isomorphism – that is, the maps into or maps out of an object contain exactly as much information as that object itself.

6 Addendum

6.1 Set-Theoretic Problems

As mentioned before, the collections of objects and morphisms in a category do not generally form a set. The four main solutions [Bor21] to this are as follows:

- Ignore the problem;
- Use classes;
- Bounding the size of objects by some cardinal number, κ;
- Use Grothendieck universes (or other axiomatic solutions).

Here, we mainly use a combination of the first two options: while we have recognised that these collections do not necessarily form sets, we also do not address the problem any further.

In our usage, this is acceptable as the categories we encounter are generally (locally) small and the classes we use are, for all intents and purposes, always sets wherever the distinction could matter. It is only in more advanced categorical constructions that the difference between sets and classes is of importance, but it is notable that many theorems in category theory are deeply intertwined with set-theoretic questions of size [Shu08] unlike in many other areas of mathematics. For instance, the Yoneda lemma demands that the categories used are locally small, while Freyd's celebrated *adjoint functor theorem* explicitly depends on a set of morphisms actually being a set and not a class.

One problem with our informal formulation of classes, and also of NBG set theory, is that classes cannot contain other classes, or else we encounter problems when attempting to form the class of all classes. This causes some issues with constructing certain large categories which require collections of classes of objects or morphisms. The solution to this to use *conglomerates*, as in [Lan13] and [AHS90], which are to classes what classes are to sets. Since we mainly work with categories that have at most classes of objects and morphisms, conglomerates are generally a satisfactory solution to this problem, but we still run into issues when forming things like the category of all categories with this approach.

The third option, which we have opted not to use, turns the object and hom-classes of a category into sets by bounding the sizes of objects available. The hom-class is then bounded by the size of the power set of the object class, which is a set by the axiom of the power set. For instance, instead of considering the class of all sets to be the object class of **Set**, we pick a cardinal number κ, and only consider the set of sets of cardinality at most κ.

However, this is somewhat clumsy and artificial, as we need to keep track of extra data for every category we work with, and moreover, it involves making an arbitrary choice, with runs counter to the working principles of naturality.

The fourth option is to use a *Grothendieck universe* (or to resolve these problems in other axiomatic ways). Before we discuss Grothendieck universes, we take a quick aside to introduce some basic model theory.

In order to mathematically encapsulate some concept, we begin with a list of *axioms* and a list of *inference rules* that let us derive new statements from existing statements. Together, axioms and inference rules generate a *theory* consisting of all the statements that can be constructed from the axioms by applying inference rules to them. All the statements within a theory that are not axioms are called *theorems*.

For instance, we could have,

- All men are mortal (axiom);
- Socrates is a man (axiom);
- If "all A are B" and "X is A", then "X is B" (inference rule);
- Therefore, Socrates is mortal (theorem).

We can't do anything further with these axioms using our inference rule, so these three statements form our entire theory about Socrates, men, and mortality.

A *model* is any collection of objects that is consistent with a given theory. For instance, while our theory requires for us to have a mortal Socrates, it does not preclude the possibility of our model containing an immortal Cerberus, because the theory does not say anything about Cerberus, or about things that are not men.

For a more practical mathematical example, suppose we are trying to axiomatise the natural numbers. We begin by asserting that 0 is a number, then by saying that every number x has a successor, $S(x)$. The natural numbers are clearly a model of these two axioms, but they aren't the only model. For instance, a model consisting of a single number, 0, such that $S(0) = 0$, is consistent with our theory. The real numbers, or complex numbers are also consistent with our theory. So, the goal is to add just enough axioms to sufficiently constrain the possible models for our theory to be useful.

In much the same way, the axioms of ZFC are not assertions about "the real" universe of sets, because they are satisfied by many possible "universes of sets" [Shu08]. In fact, the Löwenheim-Skolem theorem states that any countable theory of first-order logic that admits an infinite model cannot have a unique model (up to isomorphism).

A Grothendieck universe U is a set that is transitive, closed under pairing, power sets, and indexed unions. That is,

- (transitive) $x \in U \land y \in x \to y \in U$;
- (pairing) $x \in U \land y \in U \to \{x,y\} \in U$;
- (power set) $x \in U \to (x) \in U$;
- (indexed unions) $I \in U \land \{x_i\}_{i \in I} \subseteq U \to \left(\bigcup_{i \in I} x_i\right) \in U$.

You may notice that several of these properties closely mirror axioms of ZFC, and as such, U will behave much like a "universal set" with respect to any element it contains. That is, for any element $x \in U$, U will contain all subsets of x, $\mathcal{P}(x)$, $\mathcal{P}(\mathcal{P}(x))$, etc., and it turns out that any uncountable Grothendieck universe is a model of ZFC itself.

Furthermore, the existence of non-trivial Grothendieck universes is not provable from within ZFC, as it would imply the existence of strongly inaccessible cardinals that are known to be not provable from ZFC, and in fact, it is possible to formulate Grothendieck universes as a type of inaccessible cardinal, as is done in [Shu08]. We can then add an axiom stating the existence of a Grothendieck universe.

Another popular extension of ZFC is *Tarski-Grothendieck set theory*, which is ZFC with an additional axiom that roughly says "for every set x, there exists a Grothendieck universe it belongs to", which states the existence of not just one Grothendieck universe, but an entire infinite hierarchy of Grothendieck universes.

In any case, once a Grothendieck universe is established, we may speak of *small* and *large* sets, which are sets that are and are not elements of the Grothendieck universe, respectively, instead of sets and (proper) classes.

Yet another approach is to abandon classical axiomatisations of set theory altogether, and formulate the foundations of mathematics in terms of category theory, or other theories that entirely bypass these set-theoretic issues. There are several such systems with very different approaches, the most popular of which include *Elementary Theory of the Category of Sets* (ETCS), *First Order Logic with Dependent Sorts* (FOLDS), and *type theory.* These topics are beyond the scope of this paper, but they make for very compelling motivations for the study of category theory. For the interested reader, [Awo11] and [LR03] are good references for these ideas.

6.2 Horizontal Composition

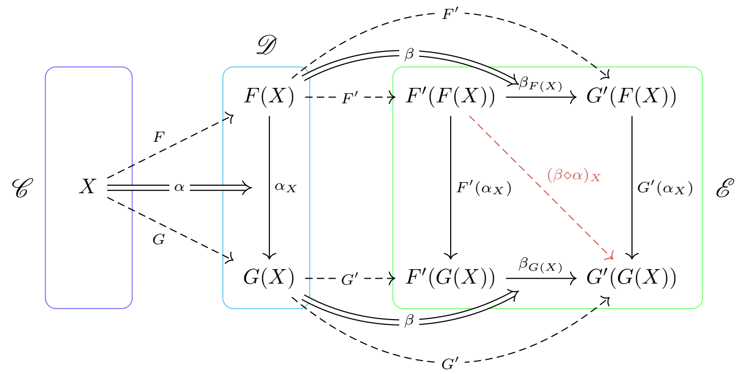
Given the name of vertical composition, it is unsurprising that we have a notion of *horizontal composition*, denoted by $\diamond$.

Fix categories $\mathscr{C}$, $\mathscr{D}$ and $\mathscr{E}$, and let F,$G : \mathscr{C} \to \mathscr{D}$ and F',$G' : \mathscr{D} \to \mathscr{E}$ be functors. Consider the natural transformations $\alpha : F \Rightarrow G$ and $\beta : F' \Rightarrow G'$.

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

Because functors compose, we also have functors $F' \circ F : \mathscr{C} \to \mathscr{E}$ and $G' \circ G : \mathscr{C} \to \mathscr{D}$. The horizontal composition $\beta \diamond \alpha$ then maps $F' \circ F$ to $G' \circ G$.

We again consider an object X in $\mathscr{C}$. F and G map X to a pair of objects in $\mathscr{D}$, and α gives the morphism between them. F', G' and β then map these objects and morphism to a square in $\mathscr{E}$.



The square of morphisms in $\mathscr{E}$ commutes as β is a natural transformation, so we can define $(\beta \diamond \alpha)_X = \beta_{G(X)} \circ F'(\alpha_X) = G'(\alpha_X) \circ \beta_{F(X)}$.

Next, we show naturality of this assignment.

First, consider the naturality diagram of α.

$$\begin{tikzcd}
A \arrow[d, "f"] & F(A) \arrow[r, "\alpha_A"] \arrow[d, "F(f)"] & G(A) \arrow[d, "G(f)"] \\
B & F(B) \arrow[r, "\alpha_B"] & G(B)
\end{tikzcd}$$

Then,

$$\begin{tikzcd}
A \arrow[d, "f"] & F'(F(A)) \arrow[r, "\alpha_A"] \arrow[d, "F'(F(f))"] & F'(G(A)) \arrow[d, "F'(G(f))"] \\
B & F'(F(B)) \arrow[r, "\alpha_B"] & F'(G(B))
\end{tikzcd} \tag{1}$$

also commutes for any choice of $A \xrightarrow{f} B$ in $\mathscr{C}$ as F' is a functor (Theorem 2.2).

Next, we observe the naturality diagram of β.

$$
\begin{tikzcd}
X \arrow[d, "g"] & F'(X) \arrow[r, "\beta_X"] \arrow[d, "F'(g)"] & G'(X) \arrow[d, "G'(g)"] \\
Y & F'(Y) \arrow[r, "\beta_Y"] & G'(Y)
\end{tikzcd}
$$

This diagram commutes for choice of objects and morphism $X \xrightarrow{g} Y$ in $\mathscr{D}$, so, picking $X = G(A)$, $Y = G(B)$, and $g = G(f)$, we have that

$$
\begin{tikzcd}
G(A) \arrow[d, "G(f)"] & F'(G(A)) \arrow[r, "\beta_{G(A)}"] \arrow[d, "F'(G(f))"] & G'(G(A)) \arrow[d, "G'(G(f))"] \\
G(B) & F'(G(B)) \arrow[r, "\beta_{G(B)}"] & G'(G(B))
\end{tikzcd}
\tag{2}
$$

commutes (again, for any choice of $A \xrightarrow{f} B$ in $\mathscr{C}$).

Pasting diagrams (1) and (2) together, we have,

$$
\begin{tikzcd}
A \arrow[d, "f"] & F'(F(A)) \arrow[r, "F'(\alpha_A)"] \arrow[d, "F'(F(f))"] & F'(G(A)) \arrow[r, "\beta_{G(A)}"] \arrow[d, "F'(G(f))"] & G'(G(A)) \arrow[d, "G'(G(f))"] \\
B & F'(F(B)) \arrow[r, "F'(\alpha_B)"] & F'(G(B)) \arrow[r, "\beta_{G(B)}"] & G'(G(B))
\end{tikzcd}
$$

We have just shown that the left and right squares commute, and hence the outer square also commutes.

This gives,

$$
\begin{aligned}
(G' \circ G)(f) \circ (\beta \circ \alpha)_A &= G'(G(f)) \circ \beta_{G(A)} \circ F'(\alpha_A) \\
&= B_{G(B)} \circ F'(\alpha_B) \circ F'(F(f)) \\
&= (\beta \circ \alpha)_A \circ (F' \circ F)(f)
\end{aligned}
$$

which is exactly the naturality condition.

Vertical and horizontal composition are related by the *interchange law*: given categories, functors, and natural transformations,

$$
\begin{tikzcd}
\mathscr{C} \arrow[r, bend left=60, "F"] \arrow[r, "G" description] \arrow[r, bend right=60, "H"'] \arrow[r, phantom, shift left=4, "\Downarrow\alpha"] \arrow[r, phantom, shift right=4, "\Downarrow\beta"] & \mathscr{D} \arrow[r, bend left=60, "F'"] \arrow[r, "G'" description] \arrow[r, bend right=60, "H'"'] \arrow[r, phantom, shift left=4, "\Downarrow\alpha"] \arrow[r, phantom, shift right=4, "\Downarrow\beta"] & \mathscr{E}
\end{tikzcd}
$$

we have,

$$
(\beta' \circ \alpha') \diamond (\beta \circ \alpha) = (\beta' \diamond \beta) \circ (\alpha' \diamond \alpha)
$$

In these situations, not only do we have objects and morphisms in the form of categories and functors, but we also have morphisms between morphisms in the form of natural transformations between those functors.

What we have really been examining is an example of a *2-category*, which is a generalisation of a category to include morphisms between morphisms. But of course, there are 3-categories, and now we've started counting. This line of inquiry quickly leads to ∞-categories, which are some of the objects of study in a generalisation of category theory called *higher category theory*.

6.3 Adjoint Functors

Fix categories $\mathscr{C}$ and $\mathscr{D}$, and let $\mathscr{C} \underset{G}{\overset{F}{\rightleftarrows}} \mathscr{D}$ be functors. F is *left adjoint* to G, and G is *right adjoint* to F, if,

$$\hom_{\mathscr{C}}(F(A),B) \cong \hom_{\mathscr{D}}(A,G(B))$$

naturally in $A \in \mathrm{ob}(\mathscr{C})$ and $B \in \mathrm{ob}(\mathscr{D})$, and we write $F \dashv G$ to denote this relationship.

Often, forgetful functors from a category $\mathscr{C}$ of algebraic objects to **Set** admit a left adjoint which is often given by the free functor that constructs the associated free algebraic object on any set.

Recall that a free group F_S on a set S consists of all words whose letters are either elements $s \in S$, or their formal inverses s^{-1}, modulo the equivalence relation that identifies xx^{-1} and $x^{-1}x$ with the empty string, ε. The group operation is then given by concatenation of words, and the identity element is given by ε. Note that the free group on a single generator is isomorphic to $(\mathbb{Z},+)$, with the isomorphism $\phi : F(\mathbf{1}) \to \mathbb{Z}$ given by mapping each word to its length.

As you'd might expect, this assignment is a functorial: there is a functor $\mathcal{F} : \mathbf{Set} \to \mathbf{Grp}$ called the *free group functor* that sends every set S to the free group F_S.

Let X be a set, Y a group, and $U : \mathbf{Grp} \to \mathbf{Set}$ the forgetful functor. Every group homomorphism $\phi : \mathcal{F}(X) \to Y$ is determined uniquely by the image of the generators of $\mathcal{F}(X)$, which are exactly the elements of the set underlying Y, or, $U(Y)$. That is, every group homomorphism $\phi \in \hom_{\mathbf{Grp}}(\mathcal{F}(X),Y)$ corresponds uniquely to a function $X \to U(Y)$, which is exactly the statement,

$$\hom_{\mathbf{Grp}}(\mathcal{F}(X),Y) \cong \hom_{\mathbf{Set}}(X,U(Y))$$

Through some tedious algebra, naturality can also be verified, and the free group functor $\mathcal{F} : \mathbf{Set} \to \mathbf{Grp}$ is left adjoint to the forgetful functor $U : \mathbf{Grp} \to \mathbf{Set}$. Again, similar arguments show that the forgetful and free functors for other algebraic structures like rings and monoids are all adjoint pairs.

Now, suppose a functor $F : \mathbf{Set} \to \mathscr{C}$ is left adjoint to a functor $G : \mathscr{C} \to \mathbf{Set}$, so we have,

$$\hom_{\mathscr{C}}(F(A),X) \cong \hom_{\mathbf{Set}}(A,G(X))$$

Note that the hom-set on the right is in **Set**, and we know that the set functions $\mathbf{1} \to X$ are in bijection with elements of X for any set X, so we have,

$$\hom_{\mathbf{Set}}(\mathbf{1},G(X)) \cong G(X)$$

so G is representable! Moreover, we have,

$$\hom_{\mathscr{C}}(F(\mathbf{1}),X) \cong G(X)$$

so G is specifically represented by $F(\mathbf{1})$. Because F and G were arbitrary, this shows that any such right adjoint is representable. In the case where F and G are a free and forgetful adjoint functor pair, this also shows that forgetful functors are always represented by free objects on single generators.

References

[Lei14] Leinster, T. *Basic Category Theory*. Cambridge University Press, 2014. ISBN: 9781107044241.

[Lan13] Lane, S.M. *Categories for the Working Mathematician*. Springer New York, 2013. ISBN: 9781475747218.

[Bor+94] Borceux, F. et al. *Handbook of Categorical Algebra: Volume 1, Basic Category Theory*. Cambridge University Press, 1994. ISBN: 9780521441780.

[nLa23] nLab authors. *Yoneda embedding*. Revision 39. 2023. URL: `https://ncatlab.org/nlab/show/Yoneda+embedding`.

[Rie17] Riehl, E. *Category Theory in Context*. Dover Publications, 2017. ISBN: 9780486820804.

[Dot23] Dotto, E. *Personal communications*. 2023.

[Hal20] Halpern-Leistner, Daniel. *Moduli theory*. 2020. URL: `http://pi.math.cornell.edu/~danielhl/modern_moduli_theory.pdf`.

[Bor21] Borcherds, R.E. *Categories for the Idle Mathematician*. 2021. URL: `https://www.youtube.com/watch?v=JOp7mH72Jlg`.

[Shu08] Shulman, Michael A. *Set Theory for Category Theory*. 2008. arXiv: `0810.1279 [math.CT]`.

[AHS90] Adamek, J., Herrlich, H., and Strecker, G.E. *Abstract and Concrete Categories: The Joy of Cats*. Wiley, 1990. ISBN: 9780471609223.

[Awo11] Awodey, Steve. *From Sets to Types to Categories to Sets*. 2011. ISBN: 9789400704305.

[LR03] Lawvere, F.W. and Rosebrugh, R. *Sets for Mathematics*. Cambridge University Press, 2003. ISBN: 9780521010603.

All diagrams were written in LaTeX with the `tikz` package.