Part II

Abelian Categories

In this part we describe the abelian category which has all the properties needed to do homological algebra. Yoneda embedding and representable functors allow us to use the nice properties of the category of sets to study more complex categories that are not so nice.

1 | Yoneda Embedding

A set is a collection of 'elements'. A category $\mathcal{C}$ is more sophisticated, it possesses 'objects' similar to how sets posses elements, but for each pair of objects, X and Y in $\mathcal{C}$, there is a set of relations between X and Y, called morphisms, denoted by $\mathrm{Hom}_{\mathcal{C}}(X, Y)$. The Yoneda Lemma allows us to define an object by its relations to other objects.

A functor F between two categories $\mathcal{C}$ and $\mathcal{D}$ consists of a mapping of objects of $\mathcal{C}$ to objects of $\mathcal{D}$, $X \mapsto FX$ together with a map of the homomorphisms,

$$\mathrm{Hom}_{\mathcal{C}}(X, Y) \to \mathrm{Hom}_{\mathcal{D}}(FX, FY).$$

the image of $f \in \mathrm{Hom}_{\mathcal{C}}(X, Y)$ denoted by $F(f)$. That takes identity to identity and respects composition i.e.,

$$F(f \circ g) = F(f) \circ F(g)$$

They are called covariant functors. A contravariant functor is a functor from the opposite category, and hence should satisfy,

$$F(f \circ g) = F(g) \circ F(f).$$

Whenever we say functor, we assume it to be covariant functor. A contravariant functor from $\mathcal{C}$ to $\mathcal{D}$ can be thought of as a covariant functor from $\mathcal{C}^{\mathrm{op}}$ to $\mathcal{D}$. A functor F is faithful if the map $\mathrm{Hom}_{\mathcal{C}}(X, Y) \to \mathrm{Hom}_{\mathcal{D}}(FX, FY)$ is injective for all X, Y. It's full if the map is surjective. If it's a bijection the functor is called fully faithful

1.1 | Yoneda Lemma

We want to study the objects in terms of the maps to or from the object. This information is contained in the functor $\mathrm{Hom}_{\mathcal{C}}(X, -)$. Each $\mathrm{Hom}_{\mathcal{C}}(X, Y)$ tells us all the relations the object X has with other object Y. The thing we should be studying is the functor $h_X = \mathrm{Hom}_{\mathcal{C}}(X, -)$. These are called hom functors.

We will focus on the hom functors here.

$$\begin{aligned} h_X : \ & \mathcal{C} \to \mathbf{Sets} \\ & Y \mapsto \mathrm{Hom}_{\mathcal{C}}(Y, X). \end{aligned}$$

which maps a morphism $f : Y \to Z$ to a morphism $h_X(f) : \mathrm{Hom}_{\mathcal{C}}(X, Y) \to \mathrm{Hom}_{\mathcal{C}}(X, Z)$ given by, $g \mapsto f \circ g$, corresponding to the composition,

$$X \xrightarrow{g} Y \xrightarrow{f} Z$$

We will denote this by,

$$\begin{aligned} f\circ = \mathrm{Hom}_{\mathcal{C}}(X, f) :\ \mathrm{Hom}_{\mathcal{C}}(X, Y) &\to \mathrm{Hom}_{\mathcal{C}}(X, Z) \\ g &\mapsto f \circ g. \end{aligned}$$

similarly we can define the contravariant hom functor. Note that we are assuming here that $\mathrm{Hom}_{\mathcal{C}}(X, Y)$s are all sets. Such categories are called locally small categories.

A natural transformation between two functors F and G from category $\mathcal{C}$ to $\mathcal{D}$ is a mapping κ such that for every $X \in \mathcal{C}$, there exists a map κ_X such that for all $f : X \to Y$, the diagram,

$$\begin{tikzcd} X \arrow[d, "f"] & FX \arrow[r, "\kappa_X"] \arrow[d, "F(f)"'] & GX \arrow[d, "G(f)"] \\ Y & FY \arrow[r, "\kappa_Y"'] & GY \end{tikzcd} \tag{natural transformation}$$

commutes, i.e., it respects the new objects and morphisms and satisfies the composition law,

$$(\kappa \circ \varphi)_X = \kappa_X \circ \varphi_X$$

The collection of all natural transformation between two functors F and G is denoted by,

$$\mathrm{Nat}(F, G).$$

We say two functors F and G are isomorphic or naturally equivalent if the natural transformation between them is a natural isomorphism, denoted as, $F \cong G$. The collections of all functors from $\mathcal{C}$ to $\mathcal{D}$ together with the natural transformations as the morphisms between functors is a category, denoted by $\mathcal{D}^{\mathcal{C}}$.

A functor $F : \mathcal{C} \to \mathbf{Sets}$ is called representable if for some $X \in \mathcal{C}$,

$$F \cong h_X$$

in such a case, F is said to be represented by the object X. Where $\cong$ stands for natural isomorphism.

Theorem 1.1. (Yoneda Lemma) *For a functor $F : \mathcal{C} \to \mathbf{Sets}$ and any $A \in \mathcal{C}$, there is a natural bijection,*

$$\mathrm{Nat}(h_A, F) \cong F(A)$$

such that $\kappa \in \mathrm{Nat}(h_A, F) \leftrightarrow \kappa_A(\mathbb{1}_A) \in F(A)$.

Proof

In the natural transformation diagram, replace F by h_A, and G by F. For $A = X$, $\kappa_A : h_A A \to FA$. Now, $h_A A = \mathrm{Hom}_{\mathcal{C}}(A, A)$, which contains $\mathbb{1}_A$. Using this we construct a map,

$$\begin{aligned} \mu : \mathrm{Nat}(h_A, F) &\to FA \\ \kappa &\mapsto \kappa_A(\mathbb{1}_A). \end{aligned}$$

We have to now check that this is a bijection. We show this by showing κ is determined by $\mu(\kappa)$ for all $B \in \mathcal{C}$. For any $f : A \to B$, we have,

$$\begin{tikzcd} A \arrow[d, "f"] & h_A A \arrow[r, "\kappa_A"] \arrow[d, "h_A(f)"'] & FA \arrow[d, "F(f)"] & \mathbb{1}_A \arrow[r, mapsto, "\kappa_A"] \arrow[d, mapsto] & \mu(\kappa) \arrow[d, mapsto] \\ B & h_A B \arrow[r, "\kappa_B"'] & FB & f \arrow[r, mapsto, "\kappa_B"'] & \kappa_B(f) \end{tikzcd}$$

Hence $\kappa_B(f) = F(f)(\mu(\kappa))$, or the action of κ_B is determined by $\mu(\kappa)$. So, if $\mu(\kappa) = \mu(\varphi)$ then $\kappa_B(f) = \varphi_B(f)$ for all $B \in \mathcal{C}$, so it's injective.

For surjectivity we have to show that for all sets $u \in FA$, there exists a natural transformation φ such that $\varphi_A(\mathbb{1}_A) = u$. For $u \in FA$, and $f : A \to B$, construct the map,

$$\begin{aligned} \varphi : h_A &\to F \\ f &\mapsto F(f)(u). \end{aligned}$$

this satisfies the requirement that $\varphi_A(\mathbb{1}_A) = u$, because clearly, $\mathbb{1}_A \mapsto F(\mathbb{1}_A)(u) = \mathbb{1}_u(u) = u$. We must make sure it's indeed a natural transformation, i.e., check if the naturality diagram,

$$\begin{tikzcd} B \arrow[d, "g"] & h_A B \arrow[r, "\varphi_B"] \arrow[d, "h_A(g)"'] & FB \arrow[d, "F(g)"] \\ C & h_A C \arrow[r, "\varphi_C"'] & FC \end{tikzcd}$$

commutes for all $B, C \in \mathcal{C}$, $g \in \mathrm{Hom}_{\mathcal{C}}(B, C)$. For $f : A \to B$, by definition of φ,

$$F(g) \circ (\varphi_B(f)) = F(g) \circ F(f)(u)$$

which by functoriality of F is $= F(g \circ f)(u)$. On the other hand, by definition of the hom functor, we have,

$$\varphi_C \circ (h_A(g)(f)) = \varphi_C(h_A(g \circ f))$$

which again by definition of φ is $= F(g \circ f)(u)$.

Hence the diagram commutes, and φ is a natural transformation. The map $\mu : \mathrm{Nat}(h_A, F) \to FA$ is a bijection. $\square$

So, the information about objects is contained in their associated hom functors, for locally small categories. The proof is same for the contravariant case. Immediate corollary is that if a functor $F : \mathcal{C}^{\mathrm{op}} \to \mathbf{Sets}$ is representable then it's unique upto isomorphism. The Yoneda embedding functor,

$$h_- : \ \mathcal{C} \to \mathbf{Sets}^{\mathcal{C}}$$

which sends an object $X \in \mathcal{C}$ to the sets of morphisms $\mathrm{Hom}_{\mathcal{C}}(-, X)$. Similarly we can define the contravariant Yoneda embedding,

$$h^- : \ \mathcal{C} \to \mathbf{Sets}^{\mathcal{C}^{\mathrm{op}}}$$

which sends an object $Y \in \mathcal{C}$ to the sets of morphisms $\mathrm{Hom}_{\mathcal{C}}(Y, -)$. These functors are fully faithful.

Let $F : \mathcal{C} \to \mathcal{D}$ and $G : \mathcal{D} \to \mathcal{C}$ be two functors, they are called an adjoint pair if

$$\mathrm{Hom}_{\mathcal{D}}(F(X), Y) = \mathrm{Hom}_{\mathcal{C}}(X, G(Y))$$

for all $X \in \mathcal{C}$ and $Y \in \mathcal{D}$. F is a left adjoint to G and G is a right adjoint to F. This is denoted by, $F \dashv G$. Adjoints are unique upto isomorphism and is the representative of the functor,

$$X \mapsto \mathrm{Hom}_{\mathcal{D}}(F(X), Y)$$

The isomorphism gives us,

$$\mathrm{Hom}_{\mathcal{C}}(G(X), G(Y)) \cong \mathrm{Hom}_{\mathcal{D}}(F \circ G(X), Y)$$

and similarly,

$$\mathrm{Hom}_{\mathcal{D}}(F(X), F(Y)) \cong \mathrm{Hom}_{\mathcal{C}}(X, G \circ F(Y))$$

2 | Abelian Categories

The suitable category for doing homological algebra is called an abelian category denoted by **Ab**. An abelian category is a category that has kernels, cokernels, quotients, direct sums, direct products, etc. i.e., if $\alpha : A \to B$ is a morphism in the category then $\ker \alpha$ is also an object in the category and similarly for image and quotient, direct sums and direct products. We want to define an appropriate category for sheaves where we can do homological algebra. Since all of these can be expressed as a representable functor or in terms of a universal property, it's possible to define them categorically. Representable functor definitions are simpler to study the consequences of the definitions and they inherit many interesting properties from the category of sets especially when we are dealing with locally small categories.

2.1 | Product & Coproduct

Let $\mathcal{C}$ be a category and consider a family $\{X_i\}_{i \in I}$ of objects of $\mathcal{C}$ indexed by a set I, then we can consider the contravariant functor,

$$G : Y \mapsto \prod_{i \in I} \mathrm{Hom}_{\mathcal{C}}(Y, X_i)$$

The product on the right side is the normal product in sets. Assuming the functor is representable, i.e., there exists an object P such that, $G(Y) = \mathrm{Hom}_{\mathcal{C}}(Y, P)$. This is called the product, denoted by, $\prod_{i \in I} X_i$. So by definition we have,

$$\mathrm{Hom}_{\mathcal{C}}(Y, \prod_{i \in I} X_i) = \prod_{i \in I} \mathrm{Hom}_{\mathcal{C}}(Y, X_i)$$

This isomorphism can be translated into the universal property definition as follows, given an object Y and a family of morphisms $f_i : Y \to X_i$ this family factorizes uniquely through $\prod_{i \in I} X_i$, visualized by the diagram,

$$\begin{tikzcd}
 & Y \arrow[dl, "f_i"'] \arrow[d, dashed, "\exists! h"] \arrow[dr, "f_j"] & \\
X_i & \prod_{i \in I} X_i \arrow[l, "\pi_i"] \arrow[r, "\pi_j"'] & X_j
\end{tikzcd}$$

The order of I is unimportant as composition with a permutation of I also belongs to the same hom set. If all $X_i = X$ then this is denoted by X^I.

Similarly we can consider the functor,

$$Y \mapsto \prod_{i \in I} \mathrm{Hom}_{\mathcal{C}}(X_i, Y)$$

This is a covariant functor. Assuming it's representable there exists an object C such that, $F(Y) = \mathrm{Hom}_{\mathcal{C}}(C, Y)$. The representative C is denoted by $\coprod_{i \in I} X_i$ and by definition we have,

$$\mathrm{Hom}_{\mathcal{C}}(\coprod_{i \in I} X_i, Y) = \prod_{i \in I} \mathrm{Hom}_{\mathcal{C}}(X_i, Y).$$

This isomorphism can be translated as follows, given an object Y and a family of morphisms $f_i : X_i \to Y$ this family factorizes uniquely through $\coprod_{i \in I} X_i$, visualized by the diagram,

$$\begin{tikzcd}
X_j \arrow[r, "\epsilon_j"] \arrow[dr, "f_j"'] & \coprod_{i \in I} X_i \arrow[d, dashed, "h"'] & X_i \arrow[l, "\epsilon_i"'] \arrow[dl, "f_i"] \\
& Y &
\end{tikzcd}$$

In algebra, for modules, etc. the coproduct is denoted by $\oplus$, and is called direct sum. It follows directly from definition that,

$$\mathrm{Hom}_{\mathcal{C}}(\coprod_{i \in I} X_i, Y) = \mathrm{Hom}_{\mathcal{C}}(Y, \prod_{i \in I} X_i)$$

2.2 | Exponentiation

The categorical notions of product and coproduct correpsond to the arithematic operations such as multiplication and addition. We can similarly talk about exponentiation. In the category of sets, **Sets**, for $X, Z \in \mathcal{C}$, Z^X is the function set consisting of all functions $h : X \to Z$. Here we have the bijection,

$$\mathrm{Hom}_{\mathbf{Sets}}(Y \times X, Z) \to \mathrm{Hom}_{\mathbf{Sets}}(Y, Z^X).$$

for a function, $f : Y \times X \to Z$, this map sends each $y \in Y$ to the function $f(y, -) \in Z^X$. Conversely given a function $f' : Y \to Z^X$, we can define a map $f(y, x) = f'(y)(x)$. So,

$$\mathrm{Hom}(Y \times X, Z) \cong \mathrm{Hom}(Y, Z^X)$$

or equivalently, $(-)^X$ is the right adjoint of $(-) \times X$. By setting $Y = 1$, we obtain,

$$Z^X \cong \mathrm{Hom}(1, Z^X) \cong \mathrm{Hom}(X, Z).$$

2.3 | Kernel & Cokernel

For sets, the kernel of two maps s, t is defined as the set $\ker(s, t) = \{x \in S \mid s(x) = t(x)\}$. Using this, for any two maps $f, g : Y \rightrightarrows Z$, we have set maps,

$$\mathrm{Hom}_{\mathcal{C}}(X, Y) \to \mathrm{Hom}_{\mathcal{C}}(X, Z)$$

given by the action, $h \mapsto f \circ h$. Using these set maps we can define the functor,

$$Y \mapsto \ker\big(\mathrm{Hom}_{\mathcal{C}}(X, Y) \rightrightarrows \mathrm{Hom}_{\mathcal{C}}(X, Z)\big).$$

This is a covariant functor from the category $\mathcal{C}$ to **Sets**. Assuming this functor is representable, the representative denoted by $\ker(f, g)$ is called the equalizer of f, g.

This isomorphism can be translated as follows, given an object X and morphisms $i : X \to Y$ and $j : X \to Z$ such that $i \circ f = j \circ g$, uniquely factors through $\ker(f, g)$, visualized by the diagram,

$$\begin{tikzcd}
X \arrow[d, dashed, "\exists!"] \arrow[dr, "i"] \arrow[drr, bend left, "j"] & & \\
\ker(f,g) \arrow[r, "e"'] & Y \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & Z
\end{tikzcd}$$

To be able to describe kernel and cokernel we have to first have a zero object, i.e,. an object that's both initial and terminal. An object Z is called a zero object if for any object A, there exists a unique morphism $Z \to A$ and a unique morphism $A \to Z$. It's unique upto isomorphism and denoted by 0. Between any two objects $A, B \in \mathcal{C}$, there exists a unique morphism $0_{A,B}$ given by the composition,

$$A \to 0 \to B$$

In this case, the kernel of a map f is defined as the equalizer of the maps $f, 0 : \mathcal{C} \to \mathcal{C}$, $\ker(f) = \ker(f, 0)$. The kernel of a map $f : Y \to Z$ is a morphism $\iota : \ker(f) \to A$ such that $f \circ \iota = 0_{\ker(f),B}$ and any other morphism $i : X \to Y$ with $f \circ i = 0_{K,B}$ uniquely factors through $\ker(f)$, visualized by the diagram,

$$\begin{tikzcd}
X \arrow[d, dashed, "e"] \arrow[dr, bend left, "i"] & & \\
\ker(f) \arrow[r, "\iota"'] & Y \arrow[r, "f"'] & Z
\end{tikzcd}$$

Here we have not written the zero morphism from X to Z. Similarly we can define coequalizer and cokernel. Given two maps $f, g : Y \rightrightarrows Z$, we have set maps, $\mathrm{Hom}_{\mathcal{C}}(Y, X) \to \mathrm{Hom}_{\mathcal{C}}(Z, X)$ given by the action, $h \mapsto h \circ f$. Coequalizer is the representative of the functor,

$$Y \mapsto \ker\big(\mathrm{Hom}_{\mathcal{C}}(Y, X) \rightrightarrows \mathrm{Hom}_{\mathcal{C}}(Z, X)\big).$$

This can be visualized by the diagram,

$$\begin{tikzcd}
Y \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] \arrow[drr, bend right, "l"'] & Z \arrow[r, "\iota"] \arrow[dr, "k"] & \operatorname{coker}(f,g) \arrow[d, dashed, "e"] \\
& & X
\end{tikzcd}$$

The cokernel of a morphism f is a morphism $\iota : X \to \operatorname{coker}(f)$ with $\iota \circ f = 0_{A,\operatorname{coker}(f)}$, and for any morphism $k : B \to L$ with $k \circ f = 0_{A,L}$ will factor uniquely through $\operatorname{coker}(f)$.

$$\begin{tikzcd}
Y \arrow[r, "f"] & Z \arrow[r, "\iota"] \arrow[dr, bend right, "k"] & \operatorname{coker}(f) \arrow[d, dashed, "e"] \\
& & X
\end{tikzcd}$$

2.4 | Limits & Colimits

To motivate what we are trying to do, consider first the case of a topological space. A limit of a sequence of points in the topological space is the best approximation of the sequence by a single point i.e., it's an element, to which the sequence gets closer to. The relation between these points comes from topology. We want something similar in the general context of categories, with the notion of closeness coming from the relation between the objects, i.e., the morphisms. The indexing set is replaced by a category and the points are replaced objects in some other category. An inductive system in $\mathcal{C}$ indexed by a category $\mathcal{I}$ is a functor,

$$F : \mathcal{I} \to \mathcal{C}.$$

Similarly a projective system in $\mathcal{C}$ indexed by $\mathcal{I}$ is a functor,

$$G : \mathcal{I}^{\mathrm{op}} \to \mathcal{C}.$$

We would like to take limits of these systems. We interchangeably use limits and projective limits and similarly colimits and inductive limits.

The limit of a system is best approximation of the system by a single object. Suppose we have relations between an object Y in $\mathcal{C}$ and each object in $F(\mathcal{I})$, i.e., a collection of morphisms

$$f_X : L \to F(X)$$

then we should expect this to factor through the limit. This amounts to saying the limit is 'closer' to the system $F : \mathcal{I} \to \mathcal{C}$ than the object L. To formalize this, we can think of the object Y itself as a functor $\widehat{Y}$ from $\mathcal{I}$ to $\mathcal{C}$, that sends every object in $\mathcal{I}$ to Y and every morphism to $\mathbb{1}_Y$. The collection of morphisms $\{f_X\}$ is then a natural transformation from the constant functor $\widehat{Y}$ to F.

A cone over a system F is a natural transformation $f : \widehat{Y} \to F$, and, dually, a cocone is a natural transformation $g : F \to \widehat{Y}$. Y is called the summit of the cone. This gives us to each object $Y \in \mathcal{C}$ a set of cones.

$$\mathrm{Cone}(-, F) : \mathcal{C}^{\mathrm{op}} \to \mathbf{Sets}$$

We say the limit of F exists if the functor $\mathrm{Cone}(-, F)$ is representable. If it's representable, the representative is denoted by $\varprojlim_{\mathcal{I}} F$ or $\lim F$ when there is no confusion, and by definition,

$$\mathrm{Cone}(-, F) = \mathrm{Hom}_{\mathcal{C}}(-, \varprojlim_{\mathcal{I}} F).$$

So, knowing the object $\varprojlim_{\mathcal{I}} F$ is sufficient to construct all cones. Natural transformation from $\widehat{Y}$ to F is obtained by composition with a choice of morphism from Y to $\varprojlim_{\mathcal{I}} F$. In this sense, $\varprojlim_{\mathcal{I}} F$ is a terminal object in the category of cones over F.

Similarly,

$$\mathrm{Cone}(F, -) : \mathcal{C} \to \mathbf{Sets}$$

If it's represenatable, the representative is called a colimit, denoted by $\varprojlim_{\mathcal{I}} F$, and by definition,

$$\mathrm{Cone}(F, -) = \mathrm{Hom}_{\mathcal{C}}(\varinjlim_{\mathcal{I}} F, -).$$

A category is said to be complete if it admits all limits for all systems with a small indexing category $\mathcal{I}$. It's said to have finite limits if limits exist whenever $\mathcal{I}$ is a finite set.

In case of the category of sets, **Sets**, we can define the limit in terms of the initial/terminal object, which is the set with one element.

$$\varprojlim_{\mathcal{I}} F := \mathrm{Cone}(1, F) = \mathrm{Nat}(1, F)$$

This is a set, as we assumed the indexing category is small. Since we work with locally small categories, we could use this as definition for limit in the category of sets, then use this to define inductive and projective limits representably using hom sets of categories.

Let $\mathcal{C}$ be a locally small category, let $F : \mathcal{I} \to \mathcal{C}$ be a inductive system. For any object $X \in \mathcal{C}$ we can construct the composite functor,

$$\mathcal{I} \xrightarrow{F} \mathcal{C} \xrightarrow{\mathrm{Hom}_{\mathcal{C}}(-,X)} \mathbf{Sets}.$$

This is a inductive system in the category of sets, $\mathrm{Hom}_{\mathcal{C}}(F-, X) : \mathcal{I} \to \mathbf{Sets}$ and the inductive limit exists. The limit of this inductive system F, denoted by $\varprojlim_{\mathcal{I}} F$, can be defined as the representative of the functor,

$$Y \mapsto \varprojlim_{\mathcal{I}} \mathrm{Hom}_{\mathcal{C}}(F, Y).$$

So, we have,

$$\mathrm{Hom}_{\mathcal{C}}(\varprojlim_{\mathcal{I}} F, Y) \cong \varprojlim_{\mathcal{I}}(\mathrm{Hom}_{\mathcal{C}}(F, Y)).$$

This can be translated as follows, for all objects $C \in \mathcal{C}$ and all family of morphisms $f_X : C \to FX$, in $\mathcal{C}$ such that for all $s \in \mathrm{Hom}_{\mathcal{I}}(X, Y)$, with $f_Y = f_X \circ F(s)$ factors uniquely through $\varprojlim_{\mathcal{I}} F$.

$$\begin{tikzcd}
 & C \arrow[ddl, "f_X"'] \arrow[d, dashed] \arrow[ddr, "f_Y"] & \\
 & \varprojlim_{\mathcal{I}} F \arrow[dl, "\varphi_X"] \arrow[dr, "\varphi_Y"'] & \\
FX \arrow[rr, "F(s)"'] & & FY
\end{tikzcd}$$

This might be the reason for naming it cones. Similarly, projective limits can be written in terms of universal property as,

$$\begin{tikzcd}
FX \arrow[rr, "F(s)"] \arrow[dr, "\varphi_X"] \arrow[ddr, "f_X"'] & & FY \arrow[dl, "\varphi_Y"'] \arrow[ddl, "f_Y"] \\
 & \varinjlim_{\mathcal{I}} G \arrow[d, dashed] & \\
 & L &
\end{tikzcd}$$

Note that if $\mathcal{I}$ admits terminal object t, then the limit $\varprojlim_{\mathcal{I}} F$ corresponds to the object $F(t)$. If the category $\mathcal{C}$ has limits, then the limit is the functor,

$$\varprojlim_{\mathcal{I}} : \mathcal{C}^{\mathcal{I}} \to \mathcal{C}$$

When the indexing category is discrete, i.e., the only morphisms are the identity morphisms, the limit and colimit correpsonds to products and coproducts.

2.4.1 | Pullback or Fibered Product

Let $\mathcal{I}$ be the indexing category with three objects X, Y, Z and two morphisms, $Y \leftarrow X \to Z$ then for functors $F : \mathcal{I} \to \mathcal{C}$, pullback $FY \times_{FX} FZ$ is defined to be the limit of this functor. In terms of universal property, a pullback for a diagram $FY \xrightarrow{f} FX \xleftarrow{g} FZ$ in a category $\mathcal{C}$ is the commutative square with vertex $FY \times_{FX} FZ$ such that any other commutative square factors through it, i.e.,

$$\begin{tikzcd}
P \arrow[dr, dashed, "\exists!"] \arrow[drr, bend left, "f_P"] \arrow[ddr, bend right, "g_P"'] & & \\
& FY \times_{FX} FZ \arrow[r, dashed] \arrow[d, dashed] & FZ \arrow[d, "g"] \\
& FY \arrow[r, "f"'] & FX
\end{tikzcd}$$

The limit is called the fibered product. In case of **Sets** the pullback always exist because limits exist and consists of all elements (x, y) such that $f(x) = g(y)$.

Similarly, a pushforward corresponds to the limit of the functor $G : \mathcal{I}^{\mathrm{op}} \to \mathcal{C}$ as above,

$$\begin{tikzcd}
FX \arrow[r, "f"] \arrow[d, "g"'] & FZ \arrow[d, dashed] \arrow[ddr, bend left, "f_P"] & \\
FY \arrow[r, dashed] \arrow[drr, bend right, "g_P"'] & FY \coprod_{FX} FZ \arrow[dr, dashed, "\exists!"] & \\
& & C
\end{tikzcd}$$

Theorem 2.1. *$\mathcal{C}$ admits finite limits iff it admits pullbacks and terminal object.*

Proof

If $\mathcal{C}$ admits finite limits, then it admits pullbacks because pullbacks are limits when $\mathcal{I}$ has only 3 objects. Conversely, suppose $\mathcal{C}$ has a terminal object T and admits pullbacks, then it has the object $X \times_T Y$

Limits and colimits exist in the category of sets, so the main goal of doing all this Yoneda and representable functors stuff is to use this structure of the category of sets to study more general categories which do not have limits and colimits. This allows us to evade the problem by going to the functor category.

Theorem 2.2. *In a functor category* $\boldsymbol{Sets}^{\mathcal{C}^{\mathrm{op}}}$*, every object* P *is the colimit of a diagram of representable objects in a canonical way.*

This asserts that given a functor $P : \mathcal{C}^{\mathrm{op}} \to \mathbf{Sets}$, there is a canonical way of constructing a small indexing category J and a diagram $A : J \to \mathcal{C}$ such that P is the colimit of the diagram,

$$J \xrightarrow{A} \mathcal{C} \xrightarrow{h^{-}} \mathbf{Sets}^{\mathcal{C}^{\mathrm{op}}}$$

and hence,

$$P \cong \varinjlim_{J}((h^{-}) \circ A)$$

Sketch of Proof

So, the goal is to construct a small indexing category using the functor $P : \mathcal{C}^{\mathrm{op}} \to \mathbf{Sets}$ which we will denote by $\int_{\mathcal{C}} P$ following [1].

Since $P : \mathcal{C}^{\mathrm{op}} \to \mathbf{Sets}$ for each $C \in \mathcal{C}$, $P(C)$ is a set. Let the pairs (C, p) be objects of $\int_{\mathcal{C}} P$ where $C \in \mathcal{C}$ and $p \in P(C)$. Let morphisms between these objects be those maps,

$$u : (C, p) \to (C', p')$$

such that $u : C \to C'$ and the map $P(u)$ should take p to p'.

$$\begin{tikzcd} C \arrow[r, "P"] & PC \arrow[d, "Pu"] \\ C' \arrow[u, "u"] \arrow[r, "P"] & PC' \end{tikzcd} \qquad \begin{tikzcd} p \arrow[d, "Pu"] \\ p' \end{tikzcd} \qquad \begin{tikzcd} (C, p) \arrow[d, "u"] \\ (C', p') \end{tikzcd}$$

Composition of morphisms is defined by compositions of the underlying arrows.

2.5 | Additive Category

So, limits and colimits can be used to characterize categories that have the necessary properties that have the appropriate structure to do homological algebra.

2.5.1 | Properties of Limits

The category of modules has the property that the hom-sets are abelian groups. A linear category is a category $\mathcal{C}$ whose hom-sets, $\mathrm{Hom}_{\mathcal{C}}(A,B)$ are abelian groups for all $A, B \in \mathcal{C}$ and composition is bilinear, i.e., if $f, f' \in \mathrm{Hom}_{\mathcal{C}}(A,B)$ and $g, g' \in \mathrm{Hom}_{\mathcal{C}}(B,C)$ we must have,

$$(g+g') \circ f = g \circ f + g' \circ f \text{ and } g \circ (f + f') = g \circ f + g' \circ f.$$

A linear category with zero object and direct sums is called an additive category. A functor $F : \mathcal{C} \to \mathcal{D}$ between additive categories is additive if each $F_{A,B} : \mathrm{Hom}_{\mathcal{C}}(A,B) \to \mathrm{Hom}_{\mathcal{D}}(F(A), F(B))$ is a group homomorphism.

$$\begin{tikzcd}
A \arrow[d, "e"'] \arrow[dr, tail, bend left, "f"] \\
\ker(p) \arrow[r, "I"'] & B \arrow[r, "p"'] & \mathrm{coker(p)}
\end{tikzcd}$$

$$\begin{tikzcd}
\ker(f) \arrow[r, "I"] & A \arrow[r, "p"] \arrow[dr, two heads, bend right, "f"'] & \mathrm{coker}(i) \arrow[d, dashed, "e"] \\
& & \bullet
\end{tikzcd}$$

$$\begin{tikzcd}
A \arrow[r, two heads, "p"] \arrow[dr, bend right, "f"'] & \mathrm{im}(p) \arrow[d, tail, "I"] \\
& \bullet
\end{tikzcd}$$

$$\begin{tikzcd}
A \arrow[d, dashed, "e"'] \arrow[dr, bend left, "f"] \\
\ker(i) \arrow[r, "i"'] & B \arrow[r, "p"'] & \mathrm{coker}(f)
\end{tikzcd}$$

$$\begin{tikzcd}
A \arrow[r, "g"] & I \arrow[r, "m"] & B \arrow[r, "\varphi"] \arrow[d, "p"] & C \\
& ker \arrow[u, dashed, "v"] \arrow[ur, bend right, "I"'] & coker \arrow[ur, dashed, bend right, "\psi"']
\end{tikzcd}$$

$$\begin{tikzcd}
& ker \arrow[d, "k"] \\
A \arrow[ur, dashed, bend left, "w"] \arrow[r, "e"'] & ker \arrow[r, "I"] \arrow[d, "g"] & B \arrow[r, "p"] & coker \\
& T
\end{tikzcd}$$

$$\begin{tikzcd}
& ker \arrow[d, dashed, "\delta"] \arrow[dr, bend left, "I"] \\
A \arrow[ur, bend left, "e"] \arrow[r, "e'"'] & I' \arrow[r, "i'"'] & B
\end{tikzcd}$$

$$\begin{tikzcd}
& ker \arrow[d, "I"] \\
A \arrow[ur, bend left, "e"] \arrow[r, "f"'] & I' \arrow[r, "g"] \arrow[d, "p"'] & B \arrow[dl, dashed, bend left, "t"] \\
& coker
\end{tikzcd}$$

Now consider the functor category $\mathbf{Sets}^{\mathcal{C}^{\mathrm{op}}}$, with X, Y, B being functors to $\mathbf{Sets}$.

2.6 | Categories of Complexes in $\mathcal{A}$

3 | Sheaves on Sites

4 | TLDR: Category of Sheaves

The take away from this section is that abelian categories have all the neccessary structure needed to do homological algebra, i.e., it has kernels, cokernels, quotients, direct sums, direct products, etc.

The category of sheaves of abelian groups forms an abelian category, and hence we can do homological algebra on the category of sheaves.

References

[1] S Mac Lane, L Moerdijk, Sheaves in Geometry and Logic: A First Introduction to Topos Theory, Springer, 1992

[2] S Ramanan, Global Calculus, Graduate Studies in Mathematics, AMS, 2004