Category theory basics

Alyson Mei

January 6, 2026

Contents

1 Categories

1.1 Basic definitions

Category theory extensively uses the language of commutative diagrams.

Definition 1.1 (Diagram)**.** A diagram consists of a directed graph whose vertices are objects and whose edges are maps between these objects.

Definition 1.2 (Commutative diagram)**.** A diagram is called commutative if for any two directed paths with the same starting object and the same ending object, the corresponding compositions of maps are equal.

Conventions:

– All diagrams shown are assumed to be commutative by default;

– By "satisfying the diagrams" we mean "such that the diagrams commute".

Example 1.3. Commutativity of the diagram below means that $g \circ f = h \circ p$:

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[d, "p"'] & B \arrow[d, "g"] \\ C \arrow[r, "h"'] & D \end{tikzcd} .$$

Definition 1.4 (Category)**.** A category $\mathcal{C}$ consists of:

– a collection $\mathrm{obj}(\mathcal{C})$ of objects,

– a collection $\mathrm{arr}(\mathcal{C})$ of morphisms (arrows) of the form $f : C \to D$, where $C, D : \mathrm{obj}(\mathcal{C})$,

satisfying the following diagrams for any morphisms f, g, h of compatible types:

- Composition:

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[dr, "g\circ f"'] & B \arrow[d, "g"] \\ & C \end{tikzcd} ,$$

- Identity:

$$\begin{tikzcd} A \arrow[r, "\mathrm{id}_A"] \arrow[dr, "f\circ \mathrm{id}_A = f"'] & A \arrow[d, "f"] \\ & B \end{tikzcd} \qquad \begin{tikzcd} C \arrow[d, "g"'] \arrow[dr, "\mathrm{id}_A \circ g = g"] & \\ A \arrow[r, "\mathrm{id}_A"'] & A \end{tikzcd} ,$$

- Associativity:

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[dr, "g\circ f"'] & B \arrow[d, "g"] \arrow[dr, "h\circ g"] & \\ & C \arrow[r, "h"'] & D \end{tikzcd} .$$

Remark 1.5.

- The class of morphisms between objects A and B is often denoted as

$$\mathcal{C}(A,B) \text{ or } \mathrm{Hom}_{\mathcal{C}}(A,B);$$

- In some contexts, it is convenient to represent objects by their identity morphisms. In such perspective, a category can be described in terms of arrows $f : A \to B$, where the objects are implicitly identified with their identities.

1.2 Additional definitions

Definition 1.6 ((Co)Domain)**.** Let $f : A \to B$ be a morphism in a category $\mathcal{C}$. The object A from which f originates is called the domain of f, and the object B at which f terminates is called the codomain of f.

Notation:

$$\mathrm{dom}(f),\ \mathrm{cod}(f).$$

Definition 1.7 (Mono, epi, iso)**.** Consider a category $\mathcal{C}$.

- A morphism $m : \mathrm{arr}(\mathcal{C})$ is called a monomorphism (mono) if it is left-cancellative:

$$\begin{tikzcd} A \arrow[r, shift left, "f_1"] \arrow[r, shift right, "f_2"'] & B \arrow[r, "m"] & C, \end{tikzcd} \qquad m \circ f_1 = m \circ f_2 \Rightarrow f_1 = f_2 \ ;$$

- A morphism $e : \mathrm{arr}(\mathcal{C})$ is called an epimorphism (epi) if it is right-cancellative:

$$\begin{tikzcd} A \arrow[r, "e"] & B \arrow[r, shift left, "g_1"] \arrow[r, shift right, "g_2"'] & C, \end{tikzcd} \qquad g_1 \circ e = g_2 \circ e \Rightarrow g_1 = g_2 \ ;$$

- A morphism $i : \mathrm{arr}(\mathcal{C})$ is called an isomorphism (iso) if has an inverse $i^{-1} : \mathrm{arr}(\mathcal{C})$:

$$\begin{tikzcd} A \arrow[r, "i"] \arrow[dr, "\mathrm{id}_A"'] & B \arrow[d, "i^{-1}"] \\ & A \end{tikzcd} \qquad \begin{tikzcd} B \arrow[d, "i^{-1}"'] \arrow[dr, "\mathrm{id}_B"] & \\ A \arrow[r, "i"'] & B \end{tikzcd} .$$

Remark 1.8.

- Mono, epi and iso are the generalizations of injection, surjection and bijection respectively;
- Every isomorphism is both mono and epi, but the converse is not generally true.

Definition 1.9 (Initial, terminal and zero objects)**.** Consider a category $\mathcal{C}$.

- $I : \mathrm{obj}(\mathcal{C})$ is called initial object if for every $C : \mathrm{obj}(\mathcal{C})$ there exists a unique morphism $I \to C$;
- $T : \mathrm{obj}(\mathcal{C})$ is called terminal object if for every $C : \mathrm{obj}(\mathcal{C})$ there exists a unique morphism $C \to T$;
- $0 : \mathrm{obj}(\mathcal{C})$ is called zero object if it is both initial and terminal.

An object is called universal if it is initial or terminal.

Remark 1.10.

- Every two initial objects / every two terminal objects are isomorphic to each other;
- The concept of universal object combined with comma categories allows to gracefully define the variety of categorical structures.

Definition 1.11 (Small and locally small categories)**.** The category is called:

- Small, if both objects and morphisms form a set;
- Locally small, if morphisms between any two objects form a set.

Definition 1.12 (Subcategory, full subcategory)**.** Consider the category $\mathcal{C}$. A category $\mathcal{D}$ is called a subcategory of $\mathcal{C}$ if $\mathrm{arr}(\mathcal{D}) \subseteq \mathrm{arr}(\mathcal{C})$. It is called a full subcategory if, for any $A, B \in \mathrm{obj}(\mathcal{D})$, every morphism $A \to B$ in $\mathcal{C}$ is also in $\mathrm{arr}(\mathcal{D})$.

1.3 List of examples

This list presents examples of categories from different branches of mathematics and introduces notation for some frequently used categories.

Example 1.13.

- Trivial and finite categories:
    - The empty category 0;
    - The terminal category 1: $\mathrm{obj}(1) = \{*\}$, $\mathrm{arr}(1) = \{\mathrm{id}_*\}$;
    - A monoid: a category with a single object;
    - The category 2 with two objects and a single non-identity arrow;
    - The discrete category $\mathcal{C}_X$ on a set X: $\mathrm{obj}(\mathcal{C}_X) = X$, $\mathrm{arr}(\mathcal{C}_X) = \{\mathrm{id}_x \mid x \in X\}$;
    - The diagram category $\mathcal{C}_G$ associated to a directed graph G: $\mathrm{obj}(\mathcal{C}_G) = \mathrm{vertices}(G)$, $\mathrm{arr}(\mathcal{C}_G) = \mathrm{edges}(G)$, with all compatible paths commuting;

- Order-theoretic categories:
  - Preorders as categories: objects are elements of preorder P, and there is a unique arrow $x \to y$ whenever $x \leq y$;
  - Preorders / Lattices / Boolean algebras: same as above, only the arrows satisfy additional axioms;
- Basic concrete categories:
  - $\mathcal{S}et$: objects are sets, morphisms are functions;
  - $\mathcal{F}in\mathcal{S}et$: objects are finite sets, morphisms are functions;
  - $\mathcal{R}el$: objects are sets, morphisms are binary relations;
  - $\mathcal{P}art$: objects are sets, morphisms are partial functions;
  - $\mathcal{P}oint$: objects are pointed sets (sets with a distinguished element), morphisms are functions preserving the base point;
- Logical / type-theoretic categories:
  - Boolean algebras / Heyting algebras: objects are Boolean or Heyting algebras, morphisms are structure-preserving functions;
  - Cartesian closed categories: categories with finite products and exponentials;
  - Syntactic categories: constructed from logical theories or type theories; objects are formulas or types, morphisms are proofs or terms;
  - Toposes: categories with certain logical and categorical structure supporting an internal logic;
- Algebraic categories:
  - $\mathcal{M}on$: monoids and monoid homomorphisms;
  - $\mathcal{G}rp$: groups and group homomorphisms;
  - $\mathcal{A}b$: abelian groups;
  - $(\mathcal{C})\mathcal{R}ing$: (commutative) rings and ring homomorphisms;
  - $\mathcal{V}ect_{\mathbf{k}}$: vector spaces over a field k and linear maps;
  - $\mathcal{A}lg_{\mathbf{k}}$: algebras over a field or ring $\mathbf{k}$;
  - $\mathcal{M}od_R$: modules over a ring R;
  - $\mathcal{L}ie\mathcal{A}lg$ and $\mathcal{H}opf\mathcal{A}lg$: Lie and Hopf algebras and structure preserving maps;
- Topological / geometric categories:
  - $\mathcal{T}op$: topological spaces with continuous maps;
  - $\mathcal{T}op_*$: pointed spaces with basepoint-preserving maps;

- $h\mathcal{T}op$: homotopy category; morphisms are homotopy classes of continuous maps;
- $\mathcal{M}an$: smooth manifolds with smooth maps;
- $\mathcal{D}iff$: diffeological spaces with smooth maps;
- $\mathcal{S}ch$: schemes and morphisms of schemes;
- $\mathcal{V}ar_{\mathbf{k}}$: algebraic varieties over a field $\mathbf{k}$;
- $\mathcal{C}ob$: cobordisms; objects are manifolds, morphisms are cobordism classes;

– Higher level constructions:

- $\mathcal{C}at$: small categories and functors between them;
- $\mathcal{CAT}$: (large) categories and functors — "set of all categories" is only valid if we allow a universe distinction to avoid size issues;
- $\mathcal{G}rpd$: groupoids — categories in which all morphisms are invertible;
- $\mathcal{M}on\mathcal{C}at$: monoidal categories and monoidal functors;
- $\mathcal{A}b\mathcal{C}at$: abelian categories and additive functors;
- $[\mathcal{C}, \mathcal{D}]$: functor category; objects are functors $\mathcal{C} \to \mathcal{D}$, morphisms are natural transformations;
- Comma categories $(F \downarrow G)$: objects are triples $(C, D, f : F(C) \to G(D))$, morphisms are pairs making the corresponding squares commute.

2 Functors

Definition 2.1 (Functor). Consider categories $\mathcal{C}$ and $\mathcal{D}$. A functor $F : \mathcal{C} \to \mathcal{D}$ is a map from $\mathcal{C}$ to $\mathcal{D}$ that preserves identities and composition:

$$\begin{tikzcd} A \arrow[d, "\mathrm{id}_A"'] & FA \arrow[d, "F\mathrm{id}_A = \mathrm{id}_{FA}"'] \\ A & FA \end{tikzcd} ;$$

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[dr, "g \circ f"'] & B \arrow[d, "g"] & FA \arrow[r, "Ff"] \arrow[dr, "F(g\circ f) = Fg \circ Ff"'] & FB \arrow[d, "Fg"] \\ & C & & FC \end{tikzcd} .$$

Remark 2.2. The image of a functor is not necessarily a category.

Definition 2.3 (Full, faithful and fully faithful functors). The functor is called:

– Faithful, if it is injective on morphisms;

– Full, if it is surjective on morphisms;

– Fully faithful, if it is both full and faithful.

3 Natural transformations

Definition 3.1 (Natural transformation)**.** Let $F, G : \mathcal{C} \to \mathcal{D}$. A natural transformation α between F and G is a collection of morphisms $(\alpha_c \mid c \in \mathrm{obj}(\mathcal{C}))$ satisfying the diagrams

$$\begin{tikzcd} c \arrow[d, "f"'] \\ c' \end{tikzcd} \qquad\qquad \begin{tikzcd} Fc \arrow[r, "\alpha_c"] \arrow[d, "Ff"'] & Gc \arrow[d, "Gf"] \\ Fc' \arrow[r, "\alpha_{c'}"'] & Gc' \end{tikzcd} .$$

Notation:

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