Category Theory

Paolo Bettelini

Contents

1 Category

A category consists of *objects* and *morphism* or *arrows*.
An arrow has a beginning and an ending, and it goes from one object to another.
Objects serve the purpose of marking the beginning and ending of a morphism.

$$\begin{tikzcd}
a \arrow[loop left] \arrow[r, bend left] \arrow[r] & b \arrow[l, bend left] \arrow[loop right]
\end{tikzcd}$$

An example of
objects and morphisms

The set of morphisms of a category C is denoted $\hom(C)$ and the set of objects $\mathrm{ob}(C)$.

1.1 Composition or Transitivity

Composition is a property that says that if there is an arrow from a to b, and an arrow from b to c, there must exist an arrow from a to c.

$$\begin{tikzcd}
a \arrow[r, "f"] \arrow[rr, bend left, "f\circ g"] & b \arrow[r, "g"] & c
\end{tikzcd}$$

1.2 Identity or Reflexivity

For every object there is an identity arrow.

$$\begin{tikzcd}
a \arrow[loop right, "\mathrm{id}_a"]
\end{tikzcd}$$

The composition of an arrow with an identity is the arrow itself

$$\begin{tikzcd}
a \arrow[r, "f"] & b \arrow[loop right, "\mathrm{id}_b"]
\end{tikzcd}$$

$$f \circ \mathrm{id}_b = f$$

and also vice versa

$$\mathrm{id}_b \circ f = f$$

1.3 Associativity

Compositions have the associative property

$$\begin{tikzcd}
a \arrow[r, "f"] \arrow[rr, bend left, "g\circ f"] \arrow[rrr, bend left=50, "h\circ(g\circ f)"] \arrow[rrr, bend right=50, "(h\circ g)\circ f"'] & b \arrow[r, "g"] \arrow[rr, bend right, "h\circ g"'] & c \arrow[r, "h"] & d
\end{tikzcd}$$

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

2 Homomorphisms

A homomorphism is a map between two structures of the same type.

2.1 Epimorphisms

An epimorphism is a **surjective** morphism.

We can define surjectivity only in terms of morphisms.
Consider a morphism $f : a \to b$ which maps elements of a onto b. Let's also define the morphisms g_1 and g_2 which map elements from b to c. The domain of g_1 and g_2 is the codomain of f. These two functions act as f for object in the image of f, but may map objects differently for objects in the codomain of f but outside the image of f. If the morphism is surjective, hence if the codomain and the image of f are the same, then g_1 and g_2 will always act as f.

$$\begin{tikzcd} a \arrow[r, "f"] & b \arrow[r, shift left, "g_1"] \arrow[r, shift right, "g_2"'] & c \end{tikzcd}$$

formally,

$$\forall c \forall g_1, g_2 : b \to c, g_1 \circ f = g_2 \circ f \Rightarrow g_1 = g_2$$

An epimorphism is labelled with $\twoheadrightarrow$.

2.2 Monomorphisms

An epimorphism is an **injective** morphism.

$$\begin{tikzcd} c \arrow[r, shift left, "g_1"] \arrow[r, shift right, "g_2"'] & a \arrow[r, "f"] & b \end{tikzcd}$$

A morphism $f : a \to b$ is a monomorphism if

$$\forall c \forall g_1, g_2 : c \to a, f \circ g_1 = f \circ g_2 \Rightarrow g_1 = g_2$$

A monomorphism is labelled with $\hookrightarrow$.

2.3 Isomorphisms

An isomorphism is a **bijective** morphism (mono and epic, but not every mono and epic is an isomorphism).

A morphism $f : a \to b$ is invertible if there is a function g that goes from b to a

$$b : \quad b \to a$$

such that

$$g \circ f = \mathrm{id}_b$$
$$f \circ g = \mathrm{id}_a$$

$$\begin{tikzcd} a \arrow[r, bend left, "f"] & b \arrow[l, bend left, "g"] \end{tikzcd}$$

An isomorphism is labelled with $\xrightarrow{\sim}$.

2.4 Homomorphism sets

A hom-set is a set of all morphisms from an object to another of a category C. It is denoted as

$$C(a,b)$$
$$\mathrm{Hom}_C(a,b)$$
$$\mathrm{Hom}(a,b)$$

Note

$$\mathrm{Hom}(a,b) \neq \mathrm{Hom}(b,a)$$

3 Types of elements

3.1 Void

The void element is equivalent to the logical **false**. It is impossible to construct. This, functions that take void as an argument are impossible to call.

3.2 Singleton

A singleton is a single empty tuple element and is equivalent to the logical **true**. It can be constructed from nothing.

4 Types of categories

4.1 Thin categories

A thin category is a category in which each pair of objects has either 0 or 1 morphism. Every hom-set of a thin category has either 1 or 0 elements.

4.1.1 Order categories

An order category is a thin category where morphisms represent relationships.

For example, here we have an equality relationship

$$a \xrightarrow{\leq} b$$

The relationship must be reflexive since there must exist an identity morphsim.

$$\begin{tikzcd} a \arrow[loop left, "\leq"] \end{tikzcd}$$

It must also be composable and associative. The relations may for example be preorders, partial orders or total orders.

4.2 Monoid

A monoid is a category with a single object. It is equivalent to a set closed under an associative binary operation.

4.3 Kleisli category