Data Visualization From a Category Theory Perspective

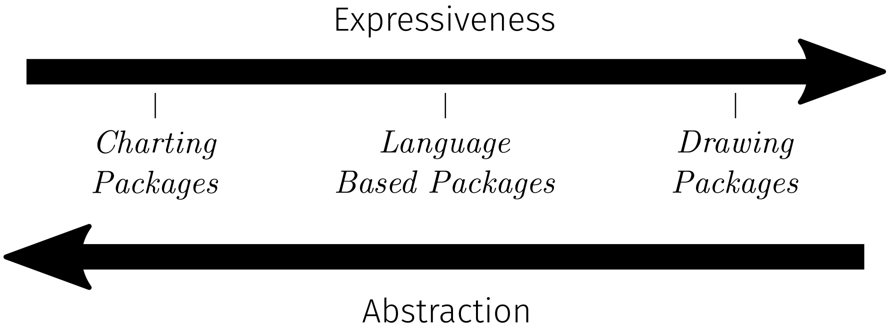
Davi Sales Barreira, Asla Medeiros e Sá

FGV - EMAp, IMPATech

February 5, 2025

Motivation

Balance expressiveness and abstraction in data visualization frameworks.



Motivation

How can we represent complex visualizations without resorting to low-level specifications?

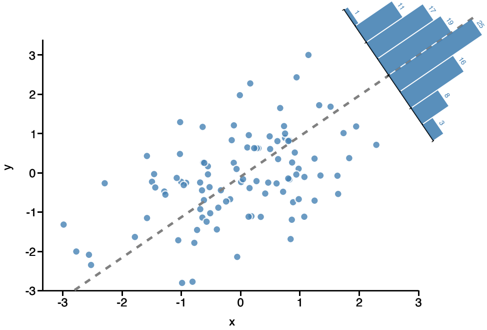


Figure: Rotated histogram aligned with second main PCA axis.

Motivation

How can we represent complex visualizations without resorting to low-level specifications?

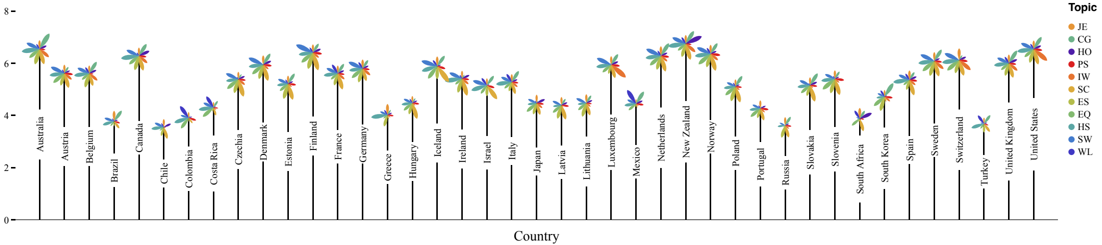


Figure: OECD Better Life Index visualization Stefaner and OECD [7].

Overview

- **Day 1**: Basics of Category Theory
- **Day 2**: Programming with Category Theory
- **Day 3**: Data Visualization Theory
- **Day 4**: Data Visualization + Categorical Programming

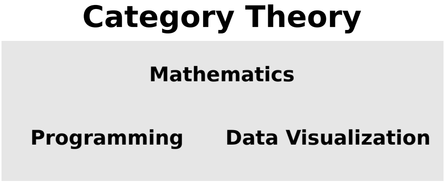
Table of contents

Why Category Theory?

Category Theory is a branch of mathematics that studies general abstract structures through their relationships.

Origin: Samuel Eilenberg e Saunders Mac Lane - 1940

As pointed by Fong and Spivak [2], Category Theory is unmatched in its ability to organize and relate abstractions.



What are Categories?

Category Theory is a branch of mathematics that studies general abstract structures through their relationships.

Definition (Category)

- A collection of objects $\mathrm{Ob}_{\mathcal{C}}$.
- A collection of morphisms $\mathrm{Hom}_{\mathcal{C}}$, where each morphism has a source object and a target object. $\mathrm{Hom}_{\mathcal{C}}(A, B)$ is the collection of morphisms going from object A to object B.
- A binary operation $\circ : \mathrm{Hom}_{\mathcal{C}}(A, B) \times \mathrm{Hom}_{\mathcal{C}}(B, C) \to \mathrm{Hom}_{\mathcal{C}}(A, C)$ such that:
    1. **Associative**: $(h \circ g) \circ f = h \circ (g \circ f)$.
    2. **Identity**: Every object has an identity morphism $1_A \in \mathrm{Hom}_{\mathcal{C}}(A, A)$.

What are Categories?

Definition (Small and Locally Small Category)

A category $\mathcal{C}$ is *small* if $\mathrm{Ob}_{\mathcal{C}}$ and $\mathrm{Hom}_{\mathcal{C}}$ are sets. A category $\mathcal{C}$ is *locally small* if for any $A, B \in \mathrm{Ob}_{\mathcal{C}}$, then $\mathrm{Hom}_{\mathcal{C}}(A, B)$ is a set. Note that a small category is also locally small.

Note that when talking about $\mathrm{Ob}_{\mathcal{C}}$ and $\mathrm{Hom}_{\mathcal{C}}$, we didn't say that they were sets, instead we called them *classes*. The reason for this lies in the foundations of Set Theory. There are collections in mathematics that are "larger" than sets, e.g. the "set" of all sets, which itself cannot be a set, otherwise it would incur in a paradox (Russell's Paradox). A way to deal with this is making a distinction between classes and sets. This point is quite technical; readers interested in understanding this nuance can check books such as Borceux [1].

Examples of Categories

The category $\mathbf{1}$ consists of $\mathsf{Ob}_{\mathbf{1}} := \{A\}$ and $\mathrm{Hom}_{\mathbf{1}} = \mathrm{id}_A$.

$$\begin{tikzcd} A \arrow[loop above, "id_A"] \end{tikzcd}$$

Examples of Categories

The category **2** consists of $\mathrm{Ob}_{\mathbf{2}} := \{A, B\}$ and $\mathrm{Hom}_{\mathbf{1}} = \{id_A, id_B, f\}$, where $f : A \to B$. The diagram for such category is shown below.

$$\begin{tikzcd} A \arrow[loop above, "id_A"] \arrow[rr, "f"] & & B \arrow[loop above, "id_B"] \end{tikzcd}$$

Figure: Category **2**.

Examples of Categories

The category **3** has three morphisms besides the identities. The morphisms are f, g and their composition $g \circ f$. The figure below illustrates the category with all its morphisms.

$$
\begin{tikzcd}
A \arrow[loop above, "id_A"] \arrow[r, "f"] \arrow[rr, bend right, "g \circ f"'] & B \arrow[loop above, "id_B"] \arrow[r, "g"] & C \arrow[loop above, "id_C"]
\end{tikzcd}
$$

Figure: Category **3** showing all morphisms.

Examples of Categories

When drawing categories, it is common to omit the identity and/or composition morphism. From here on, we do the same, whenever the context is clear.

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

Figure: Category **3** omitting morphisms.

Examples of Categories

The discrete category $\underline{\mathbf{N}}$ is the category with N objects and $\mathrm{Hom}_{\underline{\mathbf{N}}} := \{id_1, ..., id_N\}$. An example of this category is illustrated below.

Figure: Category $\underline{\mathbf{N}}$.

Examples of Categories

Given a category $\mathcal{C}$ and an object S of this category, we can define a slice category $\mathcal{C}/S$, where:

- The objects are tuples (A, f) where A is an object in $\mathcal{C}$ and $f : A \to S$ is a morphism.
- A morphism $\varphi_{(A,B)} : (A, f) \to (B, g)$ is equivalent to a morphism $\varphi \in \mathrm{Hom}_{\mathcal{C}}(A, B)$ such that $f = g \circ \varphi$.

$$
\begin{array}{ccc}
\mathcal{C}/S & & \mathcal{C} \\
(A,\ f) \xrightarrow{\phi_{A,B}} (B,\ g\ ) & \cong &
\begin{tikzcd}
A \arrow[rr, "\phi"] \arrow[dr, "f"'] & & B \arrow[dl, "g"] \\
& S &
\end{tikzcd}
\end{array}
$$

Figure: Example of slice category.

Examples of Categories

Definition (Subcategory)

Let $\mathcal{C}$ be a category. A *subcategory* $\mathcal{S}$ of $\mathcal{C}$ is such that

(i) $\mathrm{Ob}_{\mathcal{S}} \subseteq \mathrm{Ob}_{\mathcal{C}}$;

(ii) For every $A, B \in \mathrm{Ob}_{\mathcal{S}}$, we have $\mathrm{Hom}_{\mathcal{S}}(A, B) \subseteq \mathrm{Ob}_{\mathcal{C}}(A, B)$;

(iii) Composition and identity in $\mathcal{S}$ are the same as in $\mathcal{C}$, restricted to morphisms and objects of $\mathcal{S}$.

A subcategory $\mathcal{S}$ is said to be *wide* if $\mathrm{Ob}_{\mathcal{S}} = \mathrm{Ob}_{\mathcal{C}}$, and it is said to be *full* if for every $A, B \in \mathrm{Ob}_{\mathcal{S}}$, then $\mathrm{Hom}_{\mathcal{S}}(A, B) = \mathrm{Hom}_{\mathcal{C}}(A, B)$.

Examples of Categories

Definition (Dual Category)

Given a category $\mathcal{C}$, the dual (opposite) category $\mathcal{C}^{op}$ is defined as:

- $\text{Ob}_{\mathcal{C}^{op}} = \text{Op}_{\mathcal{C}}$;
- For every morphism $f : A \to B$ in $\mathcal{C}$, have an equivalent morphism $f^{op} : B \to A$ in $\mathcal{C}^{op}$;
- The composition in $\mathcal{C}^{op}$ satisfies $f^{op} \circ g^{op} = (g \circ f)^{op}$.

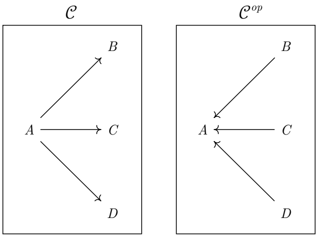


Figure: Example of dual category.

Examples of Categories

Here are some more interesting categories:

1. **Set** which is the category of sets, where the objects are sets and the morphisms are functions between sets.
2. **Top** is the category where topological spaces are the objects and continuous functions are the morphisms.
3. $\mathbf{Vec}_{\mathbb{F}}$ is the category where vector spaces over field $\mathbb{F}$ are the objects, and linear transformations are the morphisms.
4. **Mon** is the category of monoids, where morphisms are monoid homormophisms.

Isomorphisms, monomorphism and epimorphism

A very important definition in Category Theory is the notion of isomorphism. In Set Theory, we say that two sets are isomorphic if there is a bijective function between them. Yet, this concept is not restricted to Set Theory and can be generalized in Category Theory as follows:

Definition (Categorical Isomorphism)

Let $\mathcal{C}$ be a category with $X, Y \in Ob_{\mathcal{C}}$ and $f \in \mathrm{Hom}_{\mathcal{C}}(X, Y)$.

(i) We say that f is *left invertible* if there exists $f_l \in \mathrm{Hom}_{\mathcal{C}}(Y, X)$ such that $f_l \circ f = id_X$;

(ii) We say that f is *right invertible* if there exists $f_r \in \mathrm{Hom}_{\mathcal{C}}(Y, X)$ such that $f \circ f_r = id_Y$;

(iii) We say that f is invertible if it's both left and right invertible.

Exercise: Prove that in the category **Set**, the categorical isomorphism is equivalent to the set theoretic isomorphism.

Isomorphisms, monomorphism and epimorphism

Definition (Monomorphism)

In a given category $\mathcal{C}$, a morphism $m : X \to Y$ is said to be a monomorphism (monic), if for every $A \in \mathrm{Ob}_{\mathcal{C}}$ and $f, g : A \to X$ we have that:

$$m \circ f = m \circ g \implies f = g.$$

Definition (Epimorphism)

In a given category $\mathcal{C}$, a morphism $e : X \to Y$ is said to be an epimorphism (epic), if for every $A \in \mathrm{Ob}_{\mathcal{C}}$ and $f, g : A \to X$ we have that:

$$f \circ e = g \circ e \implies f = g.$$

Isomorphisms, monomorphism and epimorphism

Proposition

The following properties on monomorphism and epimorphism are true:

1. f left-invertible $\implies$ f is monic. The converse is not true.
2. f right-invertible $\implies$ f is epic. The converse is not true.
3. f invertible $\implies$ f is monic and epic. The converse is not true.
4. f monic and right-invertible $\implies$ f is isomorphism.
5. f epic and left-invertible $\implies$ f is isomorphism.

Proof.

Left as exercise. $\square$

Universal Constructions

Objects defined in terms of existence and uniqueness of morphisms are known as **universal constructions**.

Definition (Zero, Initial and Terminal)

Let $\mathcal{C}$ be a category.

1. An object $I \in \mathrm{Ob}_{\mathcal{C}}$ is *initial* if for every $A \in \mathrm{Ob}_{\mathcal{C}}$, there is exactly one morphism from I to A. Thus, from I to I there is only the identity.
2. An object $T \in \mathrm{Ob}_{\mathcal{C}}$ is *terminal* if for every $A \in \mathrm{Ob}_{\mathcal{C}}$, there is exactly one morphism from A to T. Thus, from I to I there is only the identity.
3. An object is *zero* if it is both terminal and initial.

Universal Constructions

Theorem

Every initial object is unique up to an isomorphism, i.e. if in a category there are two initial objects, then they are isomorphic. Similarly, terminal objects are unique up to an isomorphism. Moreover, the isomorphism is unique between initial object, and between terminal objects.

Proof.

Left as exercise. □

Universal Constructions

In Set Theory, we are used to the notion of a Cartesian product. Similarly to how we did for isomorphism, the idea of a product can be generalized via Category Theory

Definition (Span)

Let A, B be objects in a category $\mathcal{C}$. A span on A and B is a triple (Z, f, g) where $f : Z \to A$ and $g : Z \to B$ are morphisms in $\mathcal{C}$.

$$\begin{tikzcd} & Z \arrow[dl, "f"'] \arrow[dr, "g"] & \\ A & & B \end{tikzcd}$$

Universal Constructions

Definition (Categorical Product)

Let A, B be objects in a category $\mathcal{C}$. A span $(A \times B, \pi_1, \pi_2)$ is called a product between A and B if for every span (Z, f, g) of A and B, there exists a unique morphism $h_{f,g} : Z \to A \times B$ such that $\pi_1 \circ h_{f,g} = f$ and $\pi_2 \circ h_{f,g} = g$.

$$
\begin{tikzcd}
& Z \arrow[ld, "f"'] \arrow[rd, "g"] \arrow[dd, dashed, "h_{f,g}"] & \\
A & & B \\
& A \times B \arrow[lu, "\pi_1"] \arrow[ru, "\pi_2"'] &
\end{tikzcd}
$$

Exercise: Prove that the categorical product is equivalent to Cartesian product in **Set**.

Universal Constructions

Given the Categorical Product, we can think of a dual concept, called coproduct.

Definition (Cospan)

Let A, B be objects in a category $\mathcal{C}$. A span on A and B is a triple (Z, f, g) where $f : A \to Z$ and $g : B \to Z$ are morphisms in $\mathcal{C}$.

$$\begin{tikzcd} & Z & \\ A \arrow[ur, "f"] & & B \arrow[ul, "g"'] \end{tikzcd}$$

Universal Constructions

Definition (Categorical Coproduct)

Let A, B be objects in a category $\mathcal{C}$. A cospan $(A + B, \pi_1, \pi_2)$ is called a coproduct between A and B if for every cospan (Z, f, g) of A and B, there exists a unique morphism $l_{f,g} : Z \to A + B$ such that $h_{f,g} \circ i_1 = f$ and $h_{f,g} \circ i_2 = g$.

$$\begin{tikzcd}
 & Z \arrow[dd, dashed, "h_{f,g}"] & \\
A \arrow[ur, "f"] \arrow[dr, "i_1"'] & & B \arrow[ul, "g"'] \arrow[dl, "i_2"] \\
 & A + B &
\end{tikzcd}$$

Set Category

Here are some more properties related to **Set**:

- Initial object: $\varnothing$;
- Terminal object: any singleton set up to isomorphism;
- For any two objects A and B, the $\mathrm{Hom}(A, B)$ is also an object (called exponential object B^A);
- Monomorphisms are equivalent to injective functions;
- Epimorphisms are equivalent to surjective functions.
- Categorical products are equivalent to Cartesian products;
- Categorical coproducts are equivalent to the disjoint unions.

Functors

Definition (Functor)

A functor $F : \mathcal{C} \to \mathcal{D}$ consists of:

- A mapping between objects: $F : \mathrm{Ob}_{\mathcal{C}} \to \mathrm{Ob}_{\mathcal{D}}$.
- A mapping between morphisms:
  - Covariant: $F(f) : F(A) \to F(B)$, for $f : A \to B$.
  - Contravariant: $F(f) : F(B) \to F(A)$, for $f : A \to B$.
- Identity preservation: $F(id_A) = id_{F(A)}$.
- Composition preservation:
  - Covariant: $F(g \circ f) = F(g) \circ F(f)$.
  - Contravariant: $F(g \circ f) = F(f) \circ F(g)$.

Functors

Definition (Functor)

Let $\mathcal{C}$ and $\mathcal{D}$ be two categories. A functor $F : \mathcal{C} \to \mathcal{D}$ is a pair of mappings with the following properties:

Covariant Functor

$$
\begin{tikzcd}
A \arrow[loop above, "id_A"] \arrow[r, Rightarrow, "F"] & FA \arrow[loop above, "F(id_A) = id_{FA}"]
\end{tikzcd}
\qquad
\begin{tikzcd}
A \arrow[r, "f"] \arrow[dr, "g \circ f"'] & B \arrow[d, "g"] \\
 & C
\end{tikzcd}
\overset{F}{\Longrightarrow}
\begin{tikzcd}
FA \arrow[r, "Ff"] \arrow[dr, "Fg \circ Ff"'] & FB \arrow[d, "Fg"] \\
 & FC
\end{tikzcd}
$$

Contravariant Functor

$$
\begin{tikzcd}
A \arrow[loop above, "id_A"] \arrow[r, Rightarrow, "F"] & FA \arrow[loop above, "F(id_A) = id_{FA}"]
\end{tikzcd}
\qquad
\begin{tikzcd}
A \arrow[r, "f"] \arrow[dr, "g \circ f"'] & B \arrow[d, "g"] \\
 & C
\end{tikzcd}
\overset{F}{\Longrightarrow}
\begin{tikzcd}
FA & FB \arrow[l, "Ff"'] \\
 & FC \arrow[ul, "Ff \circ Fg"] \arrow[u, "Fg"']
\end{tikzcd}
$$

Figure: Diagrams showcasing the properties of functors.

Examples of Functors

Definition (Power set functor)

The power-set functor $\mathcal{P} : \mathbf{Set} \to \mathbf{Set}$ sends a set A to its power set $\mathcal{P}(A)$, and sends $f : A \to B$ to $\mathcal{P}f : \mathcal{P}(A) \to \mathcal{P}(B)$, where given $X \in \mathcal{P}(A)$:

$$\mathcal{P}f(X) = \{f(x) : x \in X\}.$$

Exercise: Show that this is indeed a (covariant) functor.

Examples of Functors

Definition (Inverse Image Functor)

An example of **contravariant** functor is the inverse image functor $Im : \mathbf{Set} \to \mathbf{Set}$, which sends a set A to its power-set $\mathcal{P}(A)$, but sends $f : A \to B$ to the inverse image of f, i.e. for any $Y \subset B$ we have:

$$Im\ f(Y) = f^{-1}(Y) := \{x \in A : f(x) \in Y\}.$$

Note that the inverse image satisfy the contravariant property

$$Im(f \circ g) = (f \circ g)^{-1} = g^{-1} \circ f^{-1} = Im\ g \circ Im\ f.$$

Also, for $id_A : A \to A$, we have

$$Im(id_A) = id_{\mathcal{P}(A)}.$$

Examples of Functors

Definition (Group Homormophism as Functors)

In abstract algebra, a group is a triple $(G, \cdot, e)$, where G is a set, $\cdot : G \times G \to G$ is the product mapping which is associative and has an inverse, and $e \in G$ is the identity element.

We can *categorify* groups (also called delooping [5]), i.e. we can interpret them as categories. Define a category $\mathbf{B}G$ as containing a single object G, the elements of G are the morphisms, i.e. for $g \in G$ we have $g : G \to G$. Morphism composition is given by $\cdot$, hence $g_1 \cdot g_2 \equiv g_1 \circ g_2$.

Let $(G, \cdot_G, e_G)$ and $(H, \cdot_H, e_H)$ be two groups. A function $f : G \to H$ is a homomorphism between G and H if for every $g_1, g_2 \in G$ we have:

$$f(g_1 \cdot_G g_2) = f(g_1) \cdot_H f(g_2),$$

Note that this is exactly the definition of a functor $F : \mathbf{B}G \to \mathbf{B}H$. This is actually by design, as functors are effectively *homormophisms between categories* [4].

Examples of Functors

Definition (Functor Composition and Identity)

For two functors $F : \mathcal{C} \to \mathcal{D}$ and $G : \mathcal{D} \to \mathcal{E}$, then $G \circ F$ is a functor from $\mathcal{C}$ to $\mathcal{E}$ where

(i) For any $A \in Ob_{\mathcal{C}}$, $G \circ F(A) = G(F(A))$,

(ii) For any $f \in \text{Hom}_{\mathcal{C}}$, $G \circ F(f) = G(F(f))$.

We also have identity functors, denoted by $1_{\mathcal{C}} : \mathcal{C} \to \mathcal{C}$, where:

(i) $1_{\mathcal{C}}(A) = A \in \text{Ob}_{\mathcal{C}}$;

(ii) $1_{\mathcal{C}}(f) = f \in \text{Hom}_{\mathcal{C}}$.

Natural Transformations

Definition (Natural Transformations)

Let $\mathcal{C}$ and $\mathcal{D}$ be categories, and let $F, G : \mathcal{C} \to \mathcal{D}$ be functors. A natural transformation $\alpha : F \to G$ is such that the following diagram commutes:

$$
\begin{tikzcd}
FA \arrow[r, "\alpha_A"] \arrow[d, "Ff"'] & GA \arrow[d, "Gf"] \\
FB \arrow[r, "\alpha_B"'] & GB
\end{tikzcd}
$$

$$Gf \circ \alpha_A = \alpha_B \circ Ff$$

Figure: Commutative diagram of a natural transformation highlighting the commutative property of the definition.

Natural Transformations

Definition (Vertical Composition)

Let $\alpha : F \to G$ and $\beta : G \to H$ be natural transformations where F, G and H are functors from $\mathcal{C}$ to $\mathcal{D}$. We can define a vertical composition between them by making $(\beta \circ \alpha)_A = \beta_A \circ \alpha_A$ for every object $A \in \mathrm{Ob}_{\mathcal{C}}$. One can check that this composition is associative. The vertical composition is shown below.

$$\begin{tikzcd}[column sep=huge] \mathcal{C} \arrow[r, bend left=50, "F", ""{name=U, below}] \arrow[r, "G" description, ""{name=M}] \arrow[r, bend right=50, "H"', ""{name=L, above}] & \mathcal{D} \arrow[Rightarrow, from=U, to=M, "\alpha"] \arrow[Rightarrow, from=M, to=L, "\beta"] \end{tikzcd}$$

Figure: Vertical composition of natural transformations.

Natural Transformations

Definition (Horizontal Composition)

Let $\alpha : F \to G$ and $\beta : H \to K$ be natural transformations where $F, G : \mathcal{C} \to \mathcal{D}$ and $H, K : \mathcal{D} \to \mathcal{F}$. We can define a horizontal composition between them by making

$$(\beta * \alpha)_A = \beta_{GA} \circ H(\alpha_A) = K(\alpha_A) \circ \beta_{FA}.$$

Hence, this defines a natural transformation

$$\beta * \alpha : H \circ F \to K \circ G$$

The horizontal composition is illustrated below.

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

Figure: Horizontal composition of natural transformations.

Natural Transformations

Definition (Category of Functors)

For a small category $\mathcal{C}$ and a category $\mathcal{D}$, denote $\mathcal{D}^{\mathcal{C}}$ as the category of functors from $\mathcal{C}$ to $\mathcal{D}$ where objects are functors, morphisms are natural transformations, i.e.

$$\mathrm{Ob}_{\mathcal{D}^{\mathcal{C}}} := \text{Functors from } \mathcal{C} \text{ to } \mathcal{D}$$

$$\mathrm{Mor}_{\mathcal{D}^{\mathcal{C}}}(F, G) := \text{Natural Transformations from } F \text{ to } G.$$

The identity morphism for a functor F is the identity natural transformation id_F, and the composition is the vertical composition of natural transformations.

An endofunctor is a functor $F : \mathcal{C} \to \mathcal{C}$, i.e. a functor that has the same category as domain and codomain. We call $\mathbf{End}_{\mathcal{C}}$ the category of endofunctors in $\mathcal{C}$, i.e. $\mathcal{C}^{\mathcal{C}}$.

Monoids and Monads

Monoids and **Monads** are two ubiquitous constructions both in Category Theory and Functional Programming. These two concepts will be used when talking about data visualization. Therefore, it is required of us to introduce these constructions.
Let's start with the definition of a monoid in the context of Set Theory.

Definition (Monoid - Set Theory)

A monoid is a triple $(M, \otimes, e_M)$ where M is a set, $\otimes : M \times M \to M$ is a binary operation and e_M the neutral element, such that:

1. $a \otimes (b \otimes c) = (a \otimes b) \otimes c$
2. $a \otimes e_M = e_M \otimes a = a$.

An example of a monoid is $(\mathbb{N} \cup \{0\}, +, 0)$. It is easy to check that the summation operator satisfies the associativity neutrality properties.

Monoids and Monads

Definition (Monoid in the category **Set**)

A monoid in **Set** is a triple (M, μ, η), where $M \in \mathrm{Ob}_{\mathbf{Set}}$, $\mu : M \times M \to M$ and $\eta : 1 \to M$ are two morphisms in **Set** satisfying the commutative diagrams below. Note that 1 is the terminal object in **Set**, i.e. the singleton set (which is unique up to an isomorphism).

$$\begin{tikzcd}
M \times M \times M \arrow[r, "id_M \times \mu"] \arrow[d, "\mu \times id_M"'] & M \times M \arrow[d, "\mu"] \\
M \times M \arrow[r, "\mu"] & M
\end{tikzcd}
\qquad
\begin{tikzcd}
M \arrow[r, "\eta \times id_M"] \arrow[dr, "id_M"'] & M \times M \arrow[d, "\mu"] & M \arrow[l, "id_M \times \eta"'] \arrow[dl, "id_M"] \\
& M &
\end{tikzcd}$$

Figure: Commutative diagram for monoid.

Monoids and Monads

Definition (Monad)

A monad is a monoid in $\mathbf{End}_{\mathcal{C}}$, which is the triple (T, μ, η), where $T : \mathcal{C} \to \mathcal{C}$ is a functor, $\mu : T \circ T \to T$ and $\eta : 1 \to T$ are natural transformations in $\mathbf{End}_{\mathcal{C}}$ satisfying the commutative diagrams below. Note that 1 is the identity functor in $\mathcal{C}$.

$$
\begin{tikzcd}
T \circ T \circ T \arrow[r, "id_T * \mu"] \arrow[d, "\mu * id_T"'] & T \circ T \arrow[d, "\mu"] \\
T \circ T \arrow[r, "\mu"] & T
\end{tikzcd}
\qquad
\begin{tikzcd}
T \arrow[r, "\eta * id_T"] \arrow[dr, "id_T"'] & T \circ T \arrow[d, "\mu"] & T \arrow[l, "id_T * \eta"'] \arrow[dl, "id_T"] \\
& T &
\end{tikzcd}
$$

Figure: Commutative diagram for monad.

Monoids and Monads

Monads induce a category called *Kleisli category*. Such category is of special interest in Functional Programming, where it is used to model computational effects.

Definition (Kleisli Category)

Let (T, μ, η) be a monad over a category $\mathcal{C}$. The Kleisli category is the category $\mathcal{C}_T$, where:

(i) $\mathrm{Ob}_{\mathcal{C}_T} = \mathrm{Ob}_{\mathcal{C}}$;

(ii) $\mathrm{Mor}_{\mathcal{C}_T}(A, B) = \mathrm{Mor}_{\mathcal{C}}(A, TB)$;

(iii) the composition of morphisms in $\mathcal{C}_T$ is given by:

$$g \circ_T f := \mu_C \circ Tg \circ f,$$

with $f : A \to TB$ and $g : B \to TC$ as morphisms in $\mathcal{C}$;

(iv) the identity morphism of an object A in $\mathcal{C}_T$ is $\eta : A \to TA$.

In summary, the Kleisli category is defined in such a way that two morphisms $f : A \to TB$ and $g : B \to TC$ can be composed using $g \circ_T f : A \to TC$.

F-Algebras and F-Coalgebras

The informal idea of "doing algebra" is that we have an expression, such as $1 + 1$, and we want to evaluate this expression so that we get a single number, e.g. $1 + 1 = 2$. This concept of defining generic expressions and evaluating them can be expressed in Category Theory via the F-algebras:

Definition (F-algebra)

Let $F : \mathcal{C} \to \mathcal{C}$ be an endofunctor. An F-algebra is a tuple (A, ϕ), where:

- An object $A \in \mathcal{C}$, called *carrier*;
- A morphism $\phi : FA \to A$, called *structure map*.

Note that for a fixed endofunctor F, we can define different algebras by picking different carriers or different structure maps.

F-Algebras and F-Coalgebras

Consider an endofunctor F as:

$$(\cdot \times \cdot + 1) : \mathbf{Set} \to \mathbf{Set},$$

where $\times$ is the Cartesian product, $+$ is the disjoint union operator and 1 is the terminal object in **Set**, i.e. a generic singleton set $\{e\}$.
We can then define an F-algebra $(\mathbb{Z}, \phi)$, where:

$$\phi : (\mathbb{Z} \times \mathbb{Z} + \{e\}) \to \mathbb{Z}, \text{ such that}$$
$$\phi((a, b)) \mapsto a + b$$
$$\phi(e) \mapsto 0.$$

What this means is that, for our specific F-algebra, an expression is a value of $F\mathbb{Z} = \mathbb{Z} \times \mathbb{Z} + \{e\}$, which is either a tuple of integers, or e. Our map ϕ evaluates expressions by either summing the tuple of integers or returning 0 for e.

F-Algebras and F-Coalgebras

For a fixed endofunctor F, we can define the category of F-algebras:

Definition (Category of F-algebras)

Let $F : \mathcal{C} \to \mathcal{C}$ be an endofunctor. The category of F-algebras, denoted by $\mathcal{A}lg(F)$, has F-algebras as objects, and algebra homomorphisms as morphisms. Given two F-algebras (ϕ, A) and (ψ, B), a morphism $f : A \to B$ is an *algebra homomorphism* between (ϕ, A) and (ψ, B) if the diagram below commutes.

$$\begin{tikzcd}
FA \arrow[r, "Ff"] \arrow[d, "\phi"'] & FB \arrow[d, "\psi"] \\
A \arrow[r, "f"'] & B
\end{tikzcd}$$

$$\psi \circ Ff = f \circ \phi$$

Figure: Commutative diagram for F-algebra morphism.

F-Algebras and F-Coalgebras

Definition (Initial F-algebras)

An initial F-algebra (if it exists) is an initial object in $\mathcal{A}lg(F)$. In other words, an initial F-algebra is a tuple (I, in) with $in : FI \to I$, such that for any other F-algebra (A, ϕ), there exists a unique algebra homomorphism $cata_\phi : I \to A$, commonly called *catamorphism* [3].

$$\begin{tikzcd} FI \arrow[r, "Fcata_\phi"] \arrow[d, "in"'] & FA \arrow[d, "\phi"] \\ I \arrow[r, "cata_\phi"'] & A \end{tikzcd}$$

$$\phi \circ Fcata_\phi = cata_\phi \circ in$$

Figure: Commutative diagram for F-algebra morphism.

F-Algebras and F-Coalgebras

Definition (Initial F-algebras)

An initial F-algebra (if it exists) is an initial object in $\mathcal{A}lg(F)$. In other words, an initial F-algebra is a tuple (I, in) with $in : FI \to I$, such that for any other F-algebra (A, ϕ), there exists a unique algebra homomorphism $cata_\phi : I \to A$, commonly called *catamorphism* [3].

$$
\begin{tikzcd}
FI \arrow[r, "Fcata_\phi"] \arrow[d, "in"'] & FA \arrow[d, "\phi"] \\
I \arrow[r, "cata_\phi"'] & A
\end{tikzcd}
$$

$$\phi \circ Fcata_\phi = cata_\phi \circ in$$

Figure: Commutative diagram for F-algebra morphism.

F-Algebras and F-Coalgebras

Theorem (Lambek's Lemma)

Let $F : \mathcal{C} \to \mathcal{C}$ be an endofunctor, and let (I, in) be the initial F-algebra. Then, $in : FI \to I$ is an isomorphism, meaning that there exists an inverse morphism $in^{-1} : I \to FI$.

$$
\begin{tikzcd}
FI \arrow[r, "Fcata_\phi"] \arrow[d, "in"'] & FA \arrow[d, "\phi"] \\
I \arrow[r, "cata_\phi"'] & A
\end{tikzcd}
\qquad\qquad
\begin{tikzcd}
FI \arrow[r, "Fcata_\phi"] & FA \arrow[d, "\phi"] \\
I \arrow[u, "in^{-1}"] \arrow[r, "cata_\phi"'] & A
\end{tikzcd}
$$

$$\phi \circ Fcata_\phi = cata_\phi \circ in \qquad\qquad cata_\phi = \phi \circ Fcata_\phi \circ in^{-1}$$

Figure: Commutative diagram for F-algebra morphism.

F-Algebras and F-Coalgebras

Theorem (Lambek's Lemma)

Let $F : \mathcal{C} \to \mathcal{C}$ be an endofunctor, and let (I, in) be the initial F-algebra. Then, $in : FI \to I$ is an isomorphism, meaning that there exists an inverse morphism $in^{-1} : I \to FI$.

Proof.

Left as exercise.
Step 1. Prove that for the initial F-algebra (I, in), for any other F-algebra (I, a), we have $a = id_I$.
Step 2. Since (I, in) is initial, given the F-algebra (FI, Fin), there exists a unique algebra homomorphism $h : I \to FI$. Prove that in is the inverse of h. □

F-Algebras and F-Coalgebras

In programming, the usefulness of this whole discussion on F-algebras and initial algebras is that they help us formalize the idea of recursive data structures. For example, consider an endofunctor $F_{\mathbb{Z}} : \mathbf{Set} \to \mathbf{Set}$, where:

$$F_{\mathbb{Z}}(X) := 1 + \mathbb{Z} \times X.$$

The data type representing a list of integers is equal to the carrier object for the initial algebra for this functor:

$$\text{List}_{\mathbb{Z}} \cong 1 + \mathbb{Z} \times \text{List}_{\mathbb{Z}} = 1 + \mathbb{Z} + (\mathbb{Z} \times \mathbb{Z}) + (\mathbb{Z} \times \mathbb{Z} \times \mathbb{Z}) \ldots$$

In this context, the catamorphism is the function that generalizes how to apply an $F_{\mathbb{Z}}$-algebra on $\text{List}_{\mathbb{Z}}$.

F-Algebras and F-Coalgebras

If algebras evaluate expressions, then coalgebras produce expressions. In categorical terms, the F-coalgebra is simply the dual of an F-algebra:

Definition (F-coalgebra)

Let $F : \mathcal{C} \to \mathcal{C}$ be an endofunctor. An F-coalgebra is a tuple (U, φ), where:

- An object $U \in \mathcal{C}$;
- A morphism $\varphi : U \to FU$.

One can also construct a category of F-coalgebras and consider its terminal object as (T, out). The dual notion of the catamorphism is called anamorphism, and is given by:

$$ana_\varphi = out^{-1} \circ F ana_\varphi \circ \varphi.$$

F-Algebras and F-Coalgebras

We can then think of a map that combines both operations together, i.e. it uses the coalgebra to produce expressions, and then consumes it using the algebra. The name of such map is hylomorphism, defined as $hylo : (A, alg) \times (B, coalg) \times B \rightarrow A$. An example of this is the factorial function, as discussed by Slodičák and Macko [6], where the coalgebra takes an integers and produces a list of integers, which is then consumed by multiplying each value. The hylomorphism is computed by:

$$hylo(alg, coalg) = alg \circ Fhylo \circ coalg.$$

References

[1] Borceux, F. (1994). *Handbook of categorical algebra: volume 1, Basic category theory*, volume 1. Cambridge University Press.

[2] Fong, B. and Spivak, D. I. (2019). *An invitation to applied category theory: seven sketches in compositionality*. Cambridge University Press.

[3] Milewski, B. (2018). *Category theory for programmers*. Blurb.

[4] nLab contributors (n.d.). Functor. Accessed: 2025-02-05.

[5] Perrone, P. (2024). *Starting Category Theory*. World Scientific.

[6] Slodičák, V. and Macko, P. (2011). Some new approaches in functional programming using algebras and coalgebras. *Electronic Notes in Theoretical Computer Science*, 279(3):41–62.

[7] Stefaner, M. and OECD (2012). Oecd better life index. `http://www.oecdbetterlifeindex.org/`. Accessed on 14 Oct. 2012.