Category Theory Cheat Sheet

Category

A *category* consists of the following,

- Objects: A,B,C,...
- Arrows/Morphisms: f,g,h,...
- For each f there exists, $\text{dom}(f), \text{cod}(f)$ called domain and codomain of f. We write $f : A \to B$ to indicate $A = \text{dom}(f)$ and $B = \text{cod}(f)$.
- Given $f : A \to B$ and $g : B \to C$ there exists, $g \circ f : A \to C$ called the *composite* of f and g.
- For each A, there exists $1_A : A \to A$ called the *identity arrow* of A.
- Arrows should also satisfy the following,
  - Associativity: $h \circ (g \circ f) = (h \circ g) \circ f$, for all $f : A \to B, g : B \to C, h : C \to D$.
  - Unit: $f \circ 1_A = f = 1_B \circ f$, for all $f : A \to B$.

$$\begin{tikzcd}
A \arrow[loop above, "1_A"] \arrow[r, "f"] \arrow[dr, "g\circ f"'] & B \arrow[loop above, "1_B"] \arrow[d, "g"] \arrow[dr, "h\circ g"] & \\
 & C \arrow[loop above, "1_C"] \arrow[r, "h"'] & D \arrow[loop above, "1_D"]
\end{tikzcd}$$

Functor

A *functor* $F : C \to D$ between C and D is a mapping of objects and arrows to arrows, such that

- $F(f : A \to B) = F(f) : F(A) \to F(B)$
- $F(1_a) = 1_{F(A)}$
- $F(g \circ f) = F(g) \circ F(f)$.

Isomorphism

In any category $\mathbf{C}$, an arrow $f : A \to B$ is called an *isomorphism* if there exists an arrow $g : B \to C$ s.t. $g \circ f = 1_A$ and $f \circ g = 1_B$. We say, $g = f^{-1}$. And that $A \cong B$, i.e., A is isomorphic to B.

Monoid

A set M with binary operation $\cdot$ is called a *monoid* if it is associative and has an identity $u \in M$, i.e., for $x, y, z \in M$

- $x \cdot (y \cdot z) = (x \cdot y) \cdot z$
- $u \cdot x = x = x \cdot u$.

A monoid with an inverse for each element is called a *group*.

- *Cayley's theorem:* Every group G is isomorphic to a group of permutations.

Constructions on categories

- **Product category:** The product of two categories $\mathbf{C}$ and $\mathbf{D}$ written as $\mathbf{C} \times \mathbf{D}$ has objects of the form (C, D) for $C \in \mathbf{C}$ and $D \in \mathbf{D}$, and arrows of the form $(f, g) : (C, D) \to (C', D')$ for $f : C \to C' \in \mathbf{C}$ and $g : D \to D' \in \mathbf{D}$.
  Composition and units are defined componentwise, i.e. $(f', g') \circ (f, g) = (f' \circ f, g' \circ g)$ and $1_{(C,D)} = (1_C, 1_D)$.
- **Opposite/Dual category:** For category $\mathbf{C}$ its opposite category $\mathbf{C}^{op}$ has the same objects as $\mathbf{C}$ but an arrow $f : C \to D$ in $\mathbf{C}^{op}$ is an arrow $f : D \to C$ in $\mathbf{C}$.
  For notational simplicity we say $f^* : D^* \to C^*$ in $\mathbf{C}^{op}$ for $f : C \to D$ in $\mathbf{C}$. Composition and units are therefore defined as follows, $f^* \circ g* = (g \circ f)^*$ and $(1_{C^*} = 1_C)^*$

Constructions on categories contd.

- **Arrow category:** For category $\mathbf{C}$ its arrow category $\mathbf{C}^{\to}$ has the arrows of $\mathbf{C}$ as objects and an arrow g from $f : A \to B$ to $f' : A' \to B'$ in $\mathbf{C}^{\to}$ is the following commutative square

$$\begin{tikzcd}
A \arrow[r, "g_1"] \arrow[d, "f"'] & A' \arrow[d, "f'"] \\
B \arrow[r, "g_2"'] & B'
\end{tikzcd}$$

  where g_1, g_2 are arrows in $\mathbf{C}$, i.e. an arrow is a pair of arrows $g = (g_1, g_2)$ s.t. $g_2 \circ f = f' \circ g_1$. The identity of an object $f : A \to B$ is the pair $(1_A, 1_B)$ and composition is componentwise.
- **Slice category:** For category $\mathbf{C}$ its slice category over $C \in \mathbf{C}$ is $\mathbf{C}/C$ where its objects are all arrows $f \in \mathbf{C}$ s.t. $\text{cod}(f) = C$. An arrow a from $f : X \to C$ to $f' : X' \to C$ is an arrow $a : X \to X'$ in $\mathbf{C}$ s.t. $f' \circ a = f$, i.e.

$$\begin{tikzcd}
X \arrow[rr, "a"] \arrow[dr, "f"'] & & X' \\
 & C \arrow[ur, "f'"'] &
\end{tikzcd}$$

  If $g : C \to D$ is any arrow then there exists a composition functor $g_* : \mathbf{C}/C \to \mathbf{C}/D$ defined as $g_*(f) = g \circ f$. Therefore the slice category of $\mathbf{C}$ with any of its objects is a functor from $\mathbf{C} \to \mathbf{Cat}$. This is called the *forgetful* functor as the base object is "forgetten".

Free monoid

For a set A a *word* over A is any finite sequence of its elements. The *Kleene closure* of A is defined to be the set of all words over A. Define the binary operation of concatenation on $A*$. Since it is associative, A along with $*$ with the empty word "$-$" is a monoid called the **free monoid** on the set A.

Universal mapping property (UMP) of free monoid: Let $M(A)$ be the free monoid on a set A. There is a function $i : A \to |M(A)|$, and given any monoid N and any function $f : A \to |N|$, there is a unique monoid homomorphism $\overline{f} : M(A) \to N$ s.t. $|\overline{f}| \circ i = f$.

$$\begin{tikzcd}
M(A) \arrow[d, dashed, "\overline{f}"] \\
N
\end{tikzcd}
\qquad
\begin{tikzcd}
{|M(A)|} \arrow[r, "|\overline{f}|"] & {|N|} \\
A \arrow[u, "i"] \arrow[ur, "f"'] &
\end{tikzcd}$$

$A*$ has the UMP of the free monoid on A.

Free category

A directed graph G "generates" a free category $\mathbf{C}(G)$ whose objects are the vertices of the graph and its arrows are paths. Composition of arrows is defined as concatenation of paths.

UMP of $\mathbf{C}(G)$ There is a graphic homomorphism $i : G \to |\mathbf{C}(G)|$, and given any category $\mathbf{D}$ and any graph homomorphism $h : G \to |\mathbf{D}|$, there is a unique functor $\overline{h} : \mathbf{C}(G) \to \mathbf{D}$ with $\overline{h} \circ i = h$.

$$\begin{tikzcd}
\mathbf{C}(G) \arrow[d, dashed, "\overline{h}"] \\
\mathbf{D}
\end{tikzcd}
\qquad
\begin{tikzcd}
{|\mathbf{C}(G)|} \arrow[r, "|\overline{h}|"] & {|\mathbf{D}|} \\
G \arrow[u, "i"] \arrow[ur, "h"'] &
\end{tikzcd}$$

Small categories

A category is called *small* if it has a small set of objects and arrows. (i.e., not classes). It is called large otherwise.

A category $\mathbf{C}$ is *locally small* if for all objects $X, Y \in \mathbf{C}$, the collection $\text{Hom}_{\mathbf{C}}(X, Y) = \{f \in \mathbf{C}_1 \mid f : X \to Y\}$ is a small set.

Types of morphisms

Monomoprhism