カテゴリー論基礎

Agenda Part 1 CategoryとFunctor

- Part 1-1 Category
  - Categoryとは何か？
  - Categoryの例
- Part 1-2 Functor
- Part 1-3 Natural Transformation

Agenda Part 2 Limit

- Part 2-1 Limit
  - Limit と Colimit とは何か？
  - Product
  - Pullback

- Part 2-2 その他のLimit
  - Equalizer
  - Inverse Limit
  - Terminal Object


カテゴリー論基礎
Part 1
Category と Functor

Agenda Part 1 CategoryとFunctor

- Part 1-1 Category
  - Categoryとは何か？
  - Categoryの例
- Part 1-2 Functor
- Part 1-3 Natural Transformation

Part 1-1

Category

Categoryとは何か？

カテゴリー理論は数学を鳥の目のように俯瞰する。

上空から見ると細部は見えなくなるが、地上からは発見できなかったパターンを見つけることができる。

二つの数の最小公倍数は、２つのベクトル空間の直和とどのように似ているのか？　離散位相空間、自由群、有理数体の共通点とは？

これらや多くの類似の疑問に対する答えを発見し、今まで見たこともないような数学のパターンを見ることができるだろう。

“Basic Category Theory” *Tom Leinster*

カテゴリー理論の特徴のひとつは、多くの細部を取り除いてしまうことだ。集合の要素や、群が可解かどうか、位相空間が可算基底を持つかどうかには、あまり関心がない。

だから、あなたは不思議に思うかもしれない -- そして、それは当然だ – それは、どうすれば役に立つのだろうか？

“It's all about relationships!” *Tai-Danae Bradley*

数学的対象は、その種の他のすべての対象との関係のネットワークによって決定され、それによって理解されるのである。

When is one thing equal to some other thing? *Barry Mazur*

カテゴリーとは、関連するオブジェクトのシステムである。オブジェクトは孤立して生きているのではなく、オブジェクトの間に何らかの写像の概念があり、それらを結びつけている。

「オブジェクト」が意味する典型的な例は「群」と「位相空間」であり、「写像」が意味する典型的な例はそれぞれ「準同型写像」と「連続写像」である。

私たちは多くの例を見て、いくつかのカテゴリーが今挙げた2つとは全く異なる趣を持つことも学ぶだろう。実際、カテゴリー理論の「写像」は、皆さんがよく知っているような意味での写像である必要はない。

“*Basic Category Theory*” *Tom Leinster*

categoryとは何か？
categoryを構成するもの

category Cは、次のものからできています。

- オブジェクト (object): Cを構成する要素
- 射 (morphism): Cの二つのオブジェクトx, yを結ぶもの。

この射fを $f: x \to y$ と表します。この時、xをfのdomain、yをfのcodomainと呼びます。

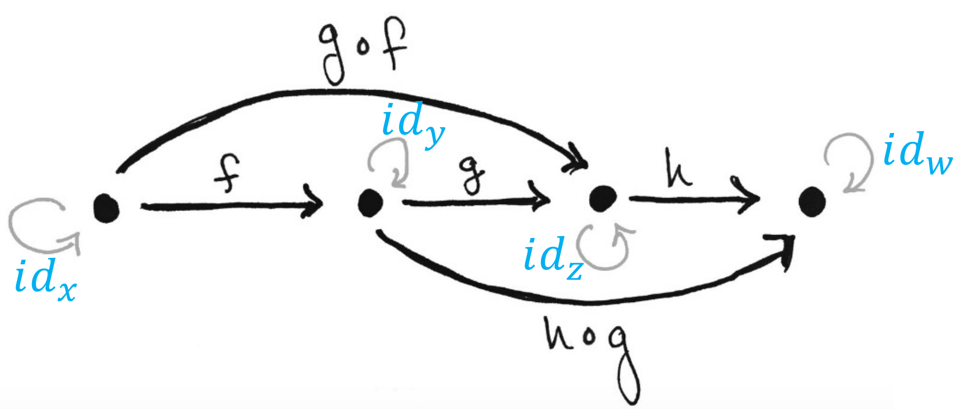
$f: x \to y,\ g: y \to z$ なる 射f, gに対して、$g \circ f: x \to z$ なる射が存在します。これを射 f, gの合成 (composition) といいます。

射 fのcodomainと射 gのdomainが一致する時、射の合成$g \circ f$ が定義されるということです。

categoryとは何か？
categoryが満たすべき性質

category C は、次の性質を満たさなければなりません。

- **同一射の存在:**
  Cのすべてのオブジェクト x について、xを同じxと結ぶ射 $id_x : x \to x$ が存在する。
- **射の合成の結合性:**
  $f : x \to y, g : y \to z, h : z \to w$の時、$(h \circ g) \circ f = h \circ (g \circ f)$が成り立つ



categoryとしての言語

大規模言語モデルの入力に与えられるテキストデータを、「語の並び」からなる「表現」の集まりと考えることにしましょう。

二つの表現SとTがあるとき、文字列の包含関係に基づいて二つの表現SとTとの間の順序 $\le$ を、次のように定義できます。

- SがTの部分文字列である時、S $\le$ T
- そうでない時、SとTの間には、順序関係は存在しない。

この順序 $\le$ は、反射律と推移律を満たしますので、preorder（前順序）です。

「表現」の集まりとしての言語は、preorderの構造を持ちます。

preorderとしての言語はcategoryである

先に見た、preorderとしての言語 P がcategoryとしても解釈できることを見ていきましょう。

このcategoryとしての言語ををLとします。

Lのオブジェクトを、Pの要素と同じものとします。

Pの要素をS,Tとすると、S,Tは、Lのオブジェクトでもあります。それは言語の語の並びからなる表現です。

Lの射 $f: S \rightarrow T$ を、Pで S ≤ T である時かつその時に限り、定義されるものとします。

Lのオブジェクトと射が定義できたので、このLがcategoryの要件を満たすことを見ていきましょう。

まずは、射の合成についてです。

Pはpreorderで推移律を満たしますので、Pの要素 S,T,Uについて、 S ≤ T かつ T ≤ U なら S ≤ U が成り立ちます。
S,T,Uは、Lのオブジェクトですので、このことは、Lにおいて
f:S→T かつ g:T→U なら h:S→U が成り立つことを意味します。このhを射 f,g の合成$g \circ f$ と解釈することができます。

次は、同一射の存在についてです。

Pは反射律を満たしますので、すべてのPの要素Sについて、
S ≤ S が成り立ちます。このことは、すべてのLのオブジェクトSについて、同一射$id_S: S \rightarrow S$ が存在することを意味します。

最後は、射の合成の結合性についてです。
次の前順序 Pとcategory Lの対応を見ればわかると思います。

P

$$\begin{tikzcd} S \arrow[r, phantom, "\leq"] \arrow[rr, bend left, phantom, "\leq"] & T \arrow[r, phantom, "\leq"] \arrow[rr, bend right, phantom, "\leq"'] & U \arrow[r, phantom, "\leq"] & V \end{tikzcd}$$

$$S \leq U \leq V \Rightarrow S \leq V$$

$$S \leq T \leq V \Rightarrow S \leq V$$

L

$$\begin{tikzcd} S \arrow[r, "f"'] \arrow[rr, bend left, "f \circ g"] & T \arrow[r, "g"'] \arrow[rr, bend right, "g \circ h"'] & U \arrow[r, "h"'] & V \end{tikzcd}$$

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

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

こうして、「表現」の包含関係で定義されるpreorderの順序構造を持つ「言語」は、categoryであることがわかります。

順序が定義された集合は、preorder, partial order, total order そして pregroup といろいろあるのですが、それらはいずれも、categoryの言葉に翻訳できます。

preorder は、categoryを構成する順序集合の中で、もっとも仮定が少ないものになっています。

Categoryの例

それぞれの新しい概念は、惜しみなく提供される例で説明されているが、そのすべてを理解する必要はない。

このテキストの以前のバージョンに基づいて私が教えたコースでは、おそらくすべての例を理解できる背景を持った学生はいなかっただろう。

重要なのは、新しい概念をすでに知っている数学と結びつけられる例だけをを理解することである。

“[Basic Category Theory]”*Tom Leinster*

ノテーション

Categoryの例を紹介する上で、次の二つのノテーションを導入しておきましょう。

- $\boldsymbol{Obj(C)}$:
  $obj(C)$は、category Cのオブジェクト cを表します。
  ただ、コンテキストから明らかな時は、$c \in C$と表すこともあります。

- $\boldsymbol{Hom_C(A,B)}$ あるいは $\boldsymbol{C(A,B)}$:
  $Hom_C(A,B)$ あるいは $C(A,B)$は、$A,B \in obj(C)$の時、すなわち、A,Bがcategory CのオブジェクトCである時、Cの$A \rightarrow B$である射を表します。

数学的構造のcategory

categoryの例として、わかりやすいのは多くの数学的構造が、それ自体categoryとして解釈できることです。先のセッションで見たpreorderがcategory だというのも、その例の一つです。

ここでは、そうした例を見ていきましょう。

Set -- Category としての集合

category **Set**のオブジェクトは、集合です。
$s \in obj(\boldsymbol{Set})$なら sは、集合だということです。

また、category **Set**の射は、普通の関数と考えれば、それが「射の合成」と「射の合成の結合性」を満たすのは明らかです。

「同一射」の存在に関しても、

$$f: A \to A \text{ で}$$
$$x \in A \longmapsto x \in A$$

という関数 f が、category **Set** の同一射になるのも明らかです。

Vect -- Category としてのベクトル空間

category **Vect**のオブジェクトは、ベクトル空間です。

ベクトル空間 Vは、一般には、ある集合Xから体kへの関数の集まりです。あるベクトル空間が、体kに値を取る関数であることを明確に表すために、$\boldsymbol{Vect_k}$ **という表記を用いることがあります。**

$V, W \in obj(Vect_k)$の時（つまり、V,Wが共に体kに値を取るベクトル空間である時）、$f \in Hom_{Vect_k}(V, W)$ (すなわち、category $Vect_k$の射 f)は、ベクトル空間Vからベクトル空間Wへの線型写像として定義されます。

ベクトル空間Vからベクトル空間Wへの写像fは、次の条件を満たす時、線型写像と呼ばれます。

$$f(u+v) = f(u) + f(v)$$
$$\lambda f(u) = f(\lambda u)$$

ここで、$u. v \in V$で、λはスカラー（体kの要素）です。

ベクトル空間と線型写像については、マルレク「密度行列 ρ で理解する確率の世界 – 意味の分散表現の数理」
https://www.marulabo.net/docs/density2/の
「Part 2 Bra-Ket記法とテンソル・ネットワーク-2」をご覧ください。

pdf資料は、こちらです。
https://drive.google.com/file/d/1Fo808lNltFsc9TUVzHlWuMrNOnfFi-N7/view

Top -- Category としての位相空間

category **Top**のオブジェクトは、位相空間です。
また、category **Top**の射は、連続写像です。

位相空間(Topological Space)へのカテゴリー論的アプローチについては、Tai-Danaeらの次の本が、おすすめです。

Tai-Danae Bradley, Tyler Bryson, and John Terilla
Topology A Categorical Approach
https://topology.mitpress.mit.edu/

数学的構造としてのcategory

これまでに挙げた「数学的構造のcategory」の例は重要なものなのですが、それらは、いずれも、すでに数学的理論の対象となっている数学的構造をcategoryとして捉え返すものでした。

categoryのオブジェクトは構造を持つ集合で、射はその構造を保存する関数でした。また、与えられた対象の要素には、明確な意味がありました。

ただ、categoryは、一般には、そうしたものだけではありません。

categoryのオブジェクトは、「集合の要素」とは違うものかもしれませんし、categoryの射も、「集合上で定義された関数」とは、違うものかもしれません。

ここではそうしたcategoryの例を、先に見た「数学的構造のcategory」とは区別して、「数学的構造としてのcategory」として、紹介したいと思います。

数学的構造としてのcategoryは、直接、オブジェクトと射、射の合成と同一射を与えることで構成できます。

category ϕ とcategory $\mathbf{1}$

- category ϕ ： まったくオブジェクトも射もないcategoryを考えます。それを、category ϕ としましょう。

- category $\mathbf{1}$ ： 一つのオブジェクトと、同一射しか持たないcategoryを考えて、それをcategory $\mathbf{1}$ とすることができます。

category $\mathbf{1}$を図で表すと、次のようになります。

すべてのオブジェクトは、同一射を持つのは明らかなので、図では、それを省略しています。

二つのオブジェクトからなるcategory

今度は、二つのオブジェクトと、一つの同一射ではない射を持つcategory を考えます。

このcategoryは、次のように図で表すことができます。

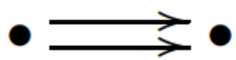
$$\bullet \rightarrow \bullet$$

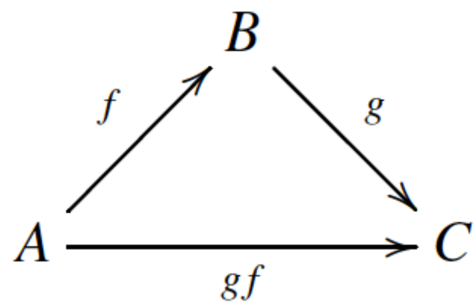
このcategoryを、次のように表すこともできます。

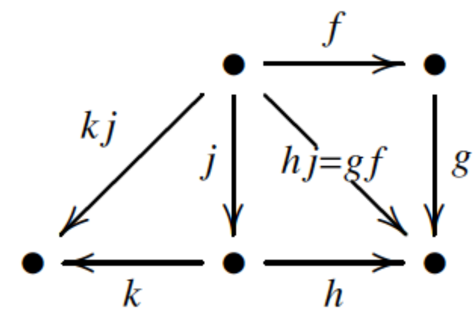
$$A \xrightarrow{f} B$$

ただ、オブジェクトに名前をどうつけてもcategoryの構造は変わらないし、射fを「関数」として理解するのは、難しいと思います。

次のcategoryを、どう解釈するか？







上のような図式は、いずれもcategoryの要件（「射の合成」「合成の結合律」）を満たしています。

ただ、これらのcategoryのオブジェクトが、特定の集合の「要素」であるという解釈も、その射が特定の「関数」であるという解釈も必要としません。

でも、これらは、数学的に定義された、categoryです。

Discrete category

オブジェクトの数に関係なく、同一射以外の射を全く持たない category を考えることも可能です。

そうしたcategoryを、**Discrete** categoryと呼びます。

Discrete categoryは、**それぞれのオブジェクトが**category内の他のオブジェクトから全く切り離されて孤立しているcategoryです。

Opposite category

category C が与えられた時、Cから新しいcategoryを作る方法があります。それは、Cのオブジェクトはそのままにして、Cのすべての射の矢印の向きを逆にすることです。

こうして得られるcategoryを、Cの**Opposite** category（あるいは、**Dual** category)と呼んで、C^{op} と表します。

射の矢印の向きが逆になりますので、category C の射 $A \rightarrow B$ は、category C^{op}では、射$A \leftarrow B$になります。

また、category C の射の合成 $A \rightarrow B \rightarrow C$ は、category C^{op} では、射の合成 $A \leftarrow B \leftarrow C$になります。

形式的には、category C^{op}のオブジェクトと射は、次のように定義されます。

$$obj(C^{op}) = obj(C)$$
$$Hom_{C^{op}}(B, A) = Hom_C(A, B)$$

最初に見たように、$Hom_C(A, B)$ は、category Cの$A \to B$である射を表しています。これを Homを省略して、$C(A, B)$と書くこともあります。こちらの表記を使えば、上の二番目の式は次のようになります。

$$C^{op}(B, A) = C(A, B)$$

この式で、category C^{op}とcategory Cの射の向きが逆であることを表現できます。

Part 1-2

Functor

カテゴリーはそれ自体が数学的な対象であり、それを考えれば、「カテゴリー間の写像」という良い概念があるのは当然である。このような写像はFunctorと呼ばれる。

また、Natural Transformation(自然変換)と呼ばれるFunctor間の写像について語ることができる。

実際、Natural Transformationの概念を形式化したいという願望が、カテゴリー理論の誕生につながった。

1940年代初頭までに、代数学・位相幾何学の研究者たちは「Natural Transformation」という言葉を使い始めていたが、それはあくまでも非公式なものであった。

Samuel EilenbergとSaunders MacLaneという2人の数学者は、正確な定義が必要だと考えた。

しかし、Natural Transformationを定義する前にFunctorを定義しなければならず、Functorを定義する前にCategoryを定義しなければならなかった。

こうして、このテーマが誕生した。

Tom Leinster

実際には、Functor $F: C \to D$を、（CとDを結びつける）ある種の不変なものを表していると考えると役に立つかもしれない。

というのも、（Functor Fのもとでは）、C で二つのオブジェクトxとyが「同じ」ものであるなら、Dで$F(x)$と$F(y)$は「同じ」ものであるからである。

Tai-Danae Bradley

十分に優れたアナロジーは皆、Functorになることを切に望んでいる。

John Baez

functorとはなにか？

カテゴリー論の教訓のひとつは、新しいタイプの数学的対象に出会うたびに、そのような対象の間に「写像」という概念があるかどうかを常に問うべきだということです。

categoryという数学的対象についても、この問いを適用することができます。

category間の写像はFunctorと呼ばれます。

CAT -- categoryのcategory

先のことは、次のように言い換えることができます。Functorを射とするcategoryのcategoryを考えてみようと。このcategoryをcategory **CAT**と呼ぶことにしましょう。

category **CAT**のオブジェクトはcategoryで、その射はFunctorです。

ただ、これがcategoryとなるためには、category **CAT**の射であるFunctorが、「射の合成」と「射の合成の結合性」を満たす必要があります。また、「同一射」の存在も保証されねばなりません。

こうした要請を満たすように、Functorは定義されることになります。

functorの定義

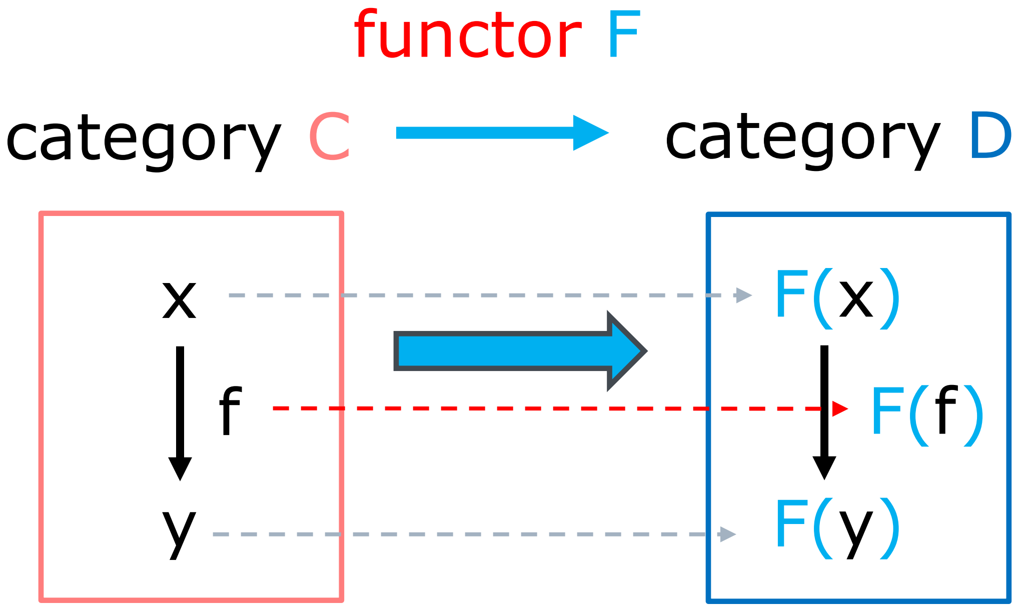
category C から category D へのfunctor $F\colon C \rightarrow D$ は、次のように定義されます。

まず、Cのオブジェクトと射について、functor Fがどう作用するかです。

- Cのすべてのオブジェクトx について、F(x)はDのオブジェクトである。
- Cのすべての射 $f : x \rightarrow y$ について、F(f)はDの射 $F(f) : F(x) \rightarrow F(y)$ である。

要するに、functor Fによって、CのオブジェクトはDのオブジェクト F(x)にうつされ、Cの射 fはDの射 F(f)にうつるということです。

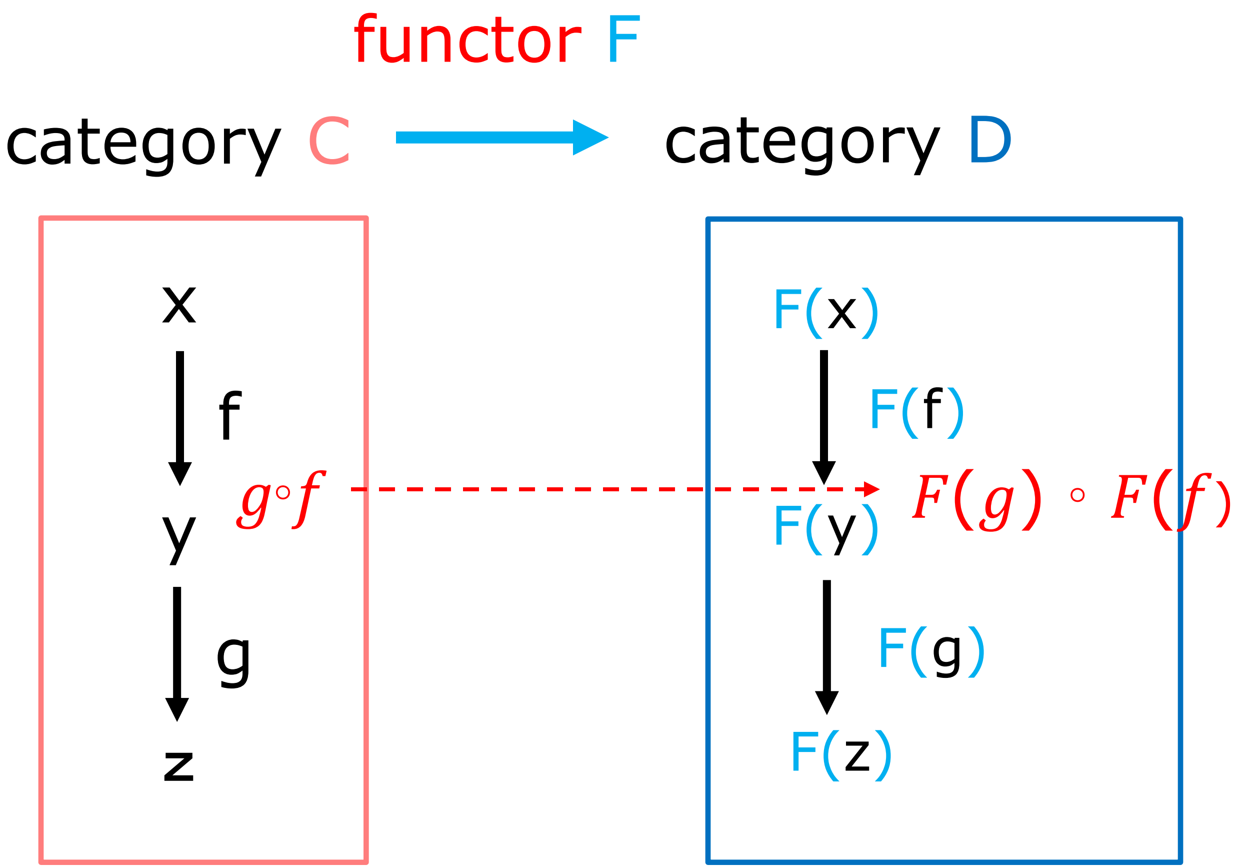
図で表すと、次のようになります。



functorは、次の条件も満たす必要があります。

- 射 g, f がCで合成可能な時、すなわち、射 $g \circ f$ が C に存在する時、$F(g \circ f) = F(g) \circ F(f)$
- FはCのすべてのオブジェクトxについて、Cの同一射をDの同一射にうつす。$F(id_x) = id_{\mathrm{F(x)}}$

射の合成は図で表すと、次のようになります。



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

Functorの例
$\pi_1 : Top \to Group$

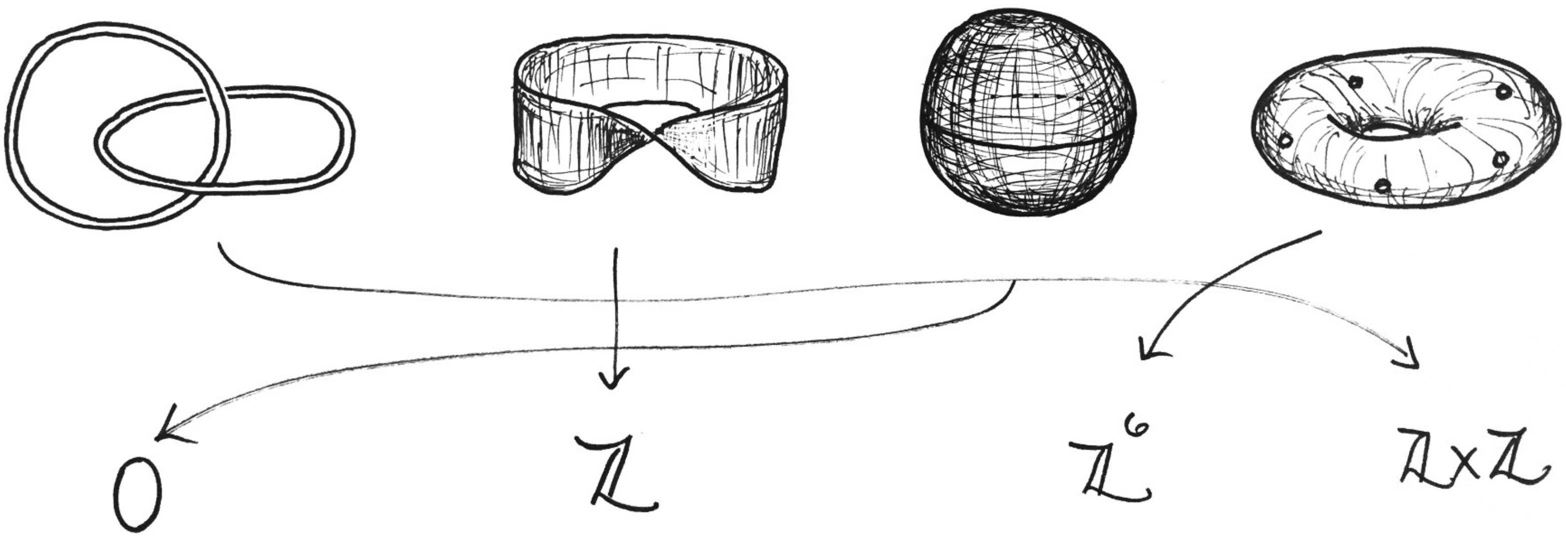
Functorの例としてよくあげられるのは、位相空間のcategory Top から群のcategory GroupへのFunctor π_1です。

$$\pi_1 : Top \to Group$$

Functor π_1は、category Topの射であるすべての連続写像 $f: X \to Y$を、category $Group$の射である群の準同型写像 $\pi_1(f): \pi_1(X) \to \pi_1(Y)$にうつします。

この$\pi_1(X)$は、Xの「基本群」と言われるものを与えます。

Ta-Danaeは、次のような例を挙げています。



基本群に興味がある方は、次のblogがわかりやすいと思います。

https://www.jeremykun.com/2013/01/12/the-fundamental-group-a-primer/

ただ、このことをFunctorの初等的な説明として利用するのは、難しいと思います。

DiagramとDiagramの間のFunctor

Tai-Danaeは、次のblogで、Functorのとてもわかりやすい例を紹介しています。

A Diagram is a Functor
https://www.math3ma.com/blog/a-diagram-is-a-functor

なんのラベルも付いていないドットとドットを結ぶ矢印だけからなる図形を考えます。それを Indexing-category と呼んで、I で表します。

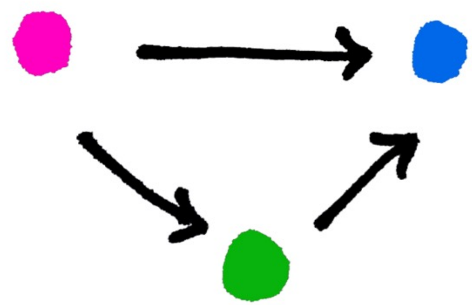
前回のセッションで出てきた、 $\bullet$ や $\bullet \to \bullet$ や、$\bullet \rightrightarrows \bullet$
は、Indexing category です。

Indexing category I のdiagram のドットをラベルで置き換え、diagramの矢印にラベルをつけたdiagramを、I-shaped diagramと呼びます。

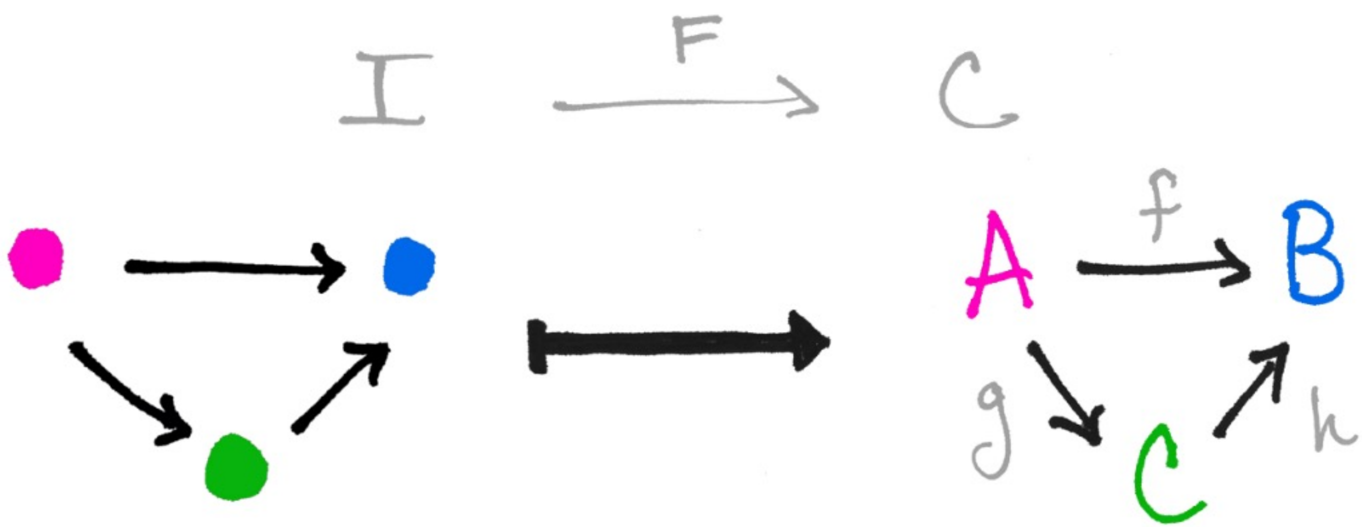
Indexing category I と I-shaped diagram Cが与えられた時、Functor $F: I \to C$ を考えます。

文字で書くより、図で書く方がわかりやすいので、図の例で説明します。

次のような Index-category I が与えられたとします。



この時、Functor Fは、次のように作用します。



DiagramはFunctorである！

この時、Functor Fの出力である diagram Cは、Fの働きそのものを表しています。

もちろん、それはIndex category Iに対する働きなのですが、Cの形をみれば、そのことは自明です。

ですから、FとCを区別する必要はありません。そのことを、彼女は、Diagram Cは、Functorであると言います。

彼女のあげている例を見てみましょう。

このDiagramは	こうしたFunctorである
$A \quad B \quad C$	$\bullet \quad \bullet \quad \bullet \quad \longmapsto \quad A \quad B \quad C$
$\cdots \to C \to B \to A$	$\cdots \to \bullet \to \bullet \to \bullet \quad \longmapsto \quad \cdots \to C \to B \to A$
$A \to B \to C \to \cdots$	$\bullet \to \bullet \to \bullet \to \cdots \quad \longmapsto \quad A \to B \to C \to \cdots$

$$\begin{tikzcd} & B \arrow[d] \\ A \arrow[r] & C \end{tikzcd} \qquad \begin{tikzcd} & \bullet \arrow[d] \\ \bullet \arrow[r] & \bullet \end{tikzcd} \quad \longmapsto \quad \begin{tikzcd} & B \arrow[d] \\ A \arrow[r] & C \end{tikzcd}$$

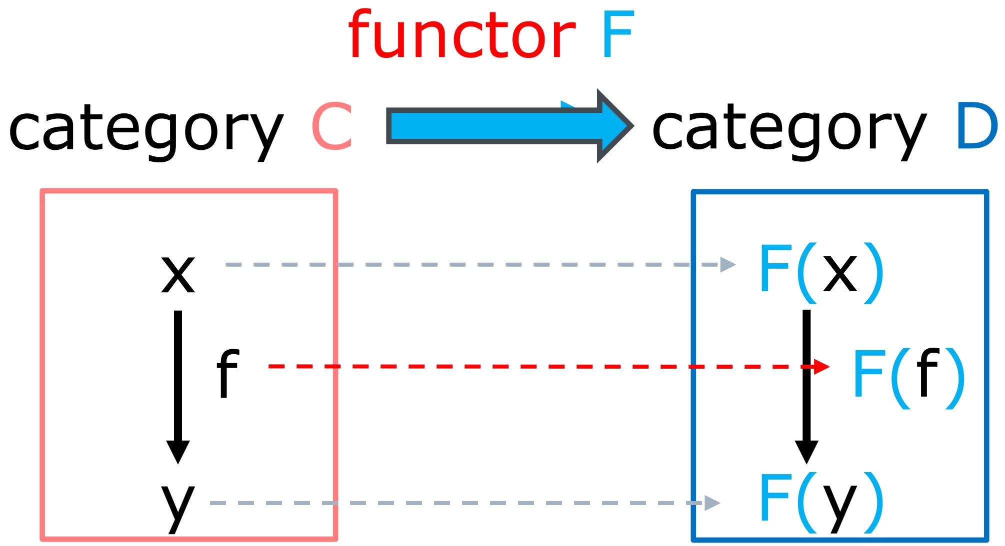
$$\begin{tikzcd} C \arrow[r] \arrow[d] & B \\ A & \end{tikzcd} \qquad \begin{tikzcd} \bullet \arrow[r] \arrow[d] & \bullet \\ \bullet & \end{tikzcd} \quad \longmapsto \quad \begin{tikzcd} C \arrow[r] \arrow[d] & B \\ A & \end{tikzcd}$$

$$A \rightrightarrows B \qquad \bullet \rightrightarrows \bullet \quad \longmapsto \quad A \rightrightarrows B$$

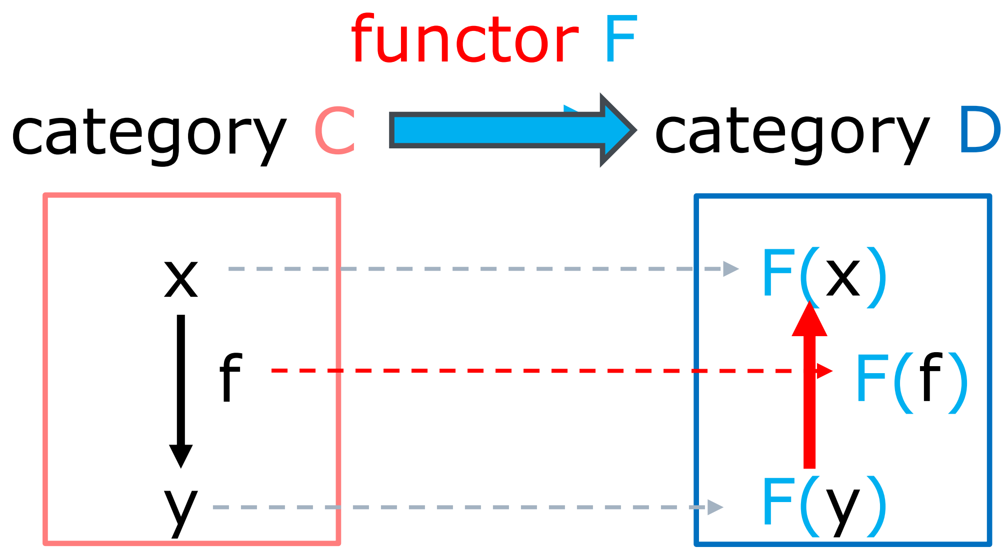
covariant Functorとcontravariant Functor

これまで、Functorを右のような図式で表してきました。

ただ、次のようなFunctorも存在します。



オブジェクトの対応は同じですが、category D での射の向きが反対になっています。



この時、上の形のFunctorをcovariant Functor といい、

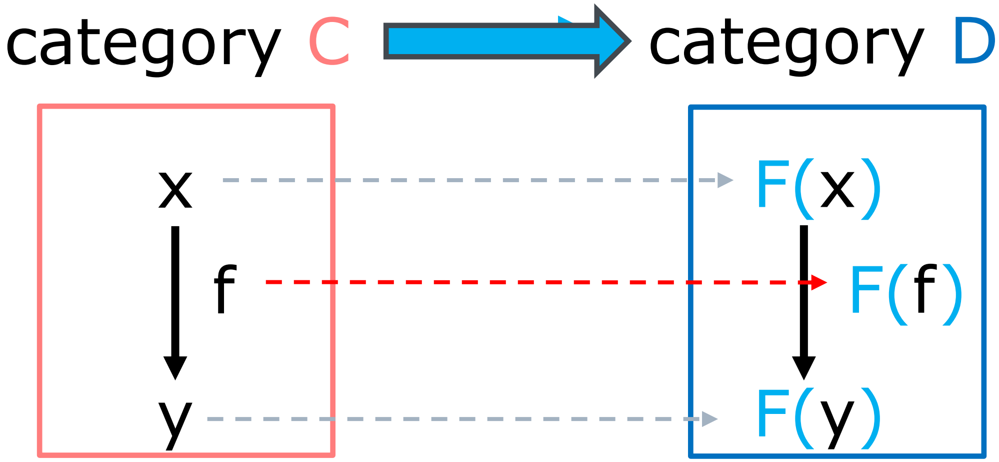
下の形のFunctorをcontravariant Functorといいます。

category Cから category Dへの contravariant Functor を、

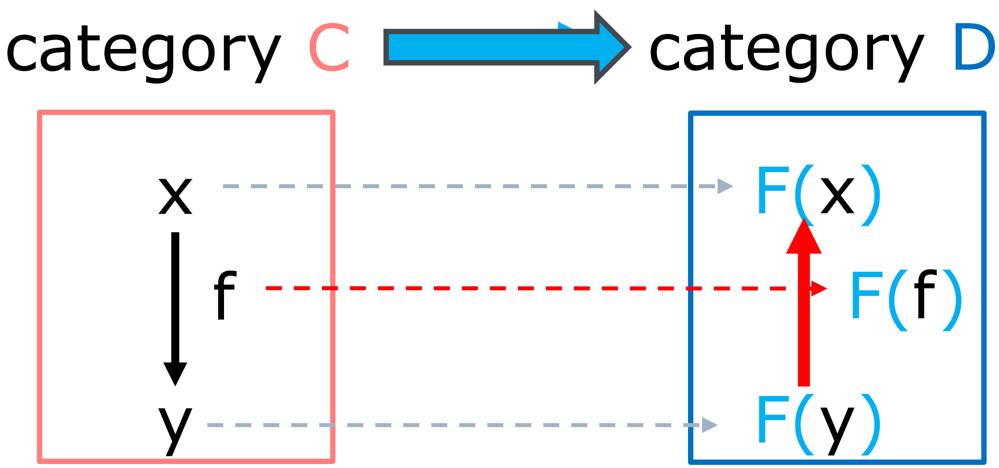
$$F: C^{op} \rightarrow D$$

で表します。

covariant functor F



contravariant functor F



Forgetfull Functor

単純ですが、重要なFunctorにForgetfull Functor というのがあります。

例えば、Forgetfull Functor $U: Vect_k \to Set$ は、ベクトル空間の性質をすべて忘れて、そのオブジェクトだけを返します。

Forgetfull Functor $U: Preorder \to Set$ は、Preorderとしての性質を全て忘れてしまいます。

そんなFunctor、なんの役に立つのかと思われるかもしれませんが、それについては、後のセミナーのAdjointのところで、説明したいと思います。

Part 1-3

Natural Transformation

categoryという概念は、functorという概念を持つために必要なものとして理解するのが一番である。

しかし、進歩はここで止まるわけではない。functorの間には写像があり、それらはnatural transformationと呼ばれる。

そして、EilenbergとSaunders Laneがfunctorを最初に定義したのは、これらを定義するためであった。

Peter J. Freyd

我々は今や、categoryについて知っている。また、category間の写像であるfunctorについても知っている。

驚くべきことに、「functor間の写像」という概念もある。
このような写像はnatural transformationと呼ばれる。

この概念は、functorが同じdomainとcodomainを持つ場合にのみ適用される。

Tom Leinster

natural transformation

二つのfunctor F,Gがあって、F,Gともに、category Cから category Dへのfunctorだとします。

この時、次の条件を満たすη(エータ)を、FからGへのnatural transformation と言います。

- Cのすべてのオブジェクトx について、η_xは$F(x)$から$G(x)$の射である。

$$\eta_x : F(x) \rightarrow G(x)$$

- $f : x \rightarrow y$ がCの射である時、次の式が成り立つ。

$$G(f) \circ \eta_x = \eta_y \circ F(f)$$

natural transformationの働きを
可換図式で表すと、次のようになります。

functor F, G はともに、
category Cからcategory Dへのfunctor

category C

$$
\begin{tikzcd}
x \arrow[d, "f"] \\
y
\end{tikzcd}
$$

functor F

functor G

category D

$$
\begin{tikzcd}
F(x) \arrow[r, "\eta_x"] \arrow[d, "F(f)"'] & G(x) \arrow[d, "G(f)"] \\
F(y) \arrow[r, "\eta_y"'] & G(y)
\end{tikzcd}
$$

$$G(f) \circ \eta_x = \eta_y \circ F(f)$$

natural transformationのノテーション

$functor\ F$から$functor$ Gへのnatural transformation η を、二重の矢印で、次のように表します。

$$\eta\colon F \Longrightarrow G$$

次のようなノテーションもあります。

$$\begin{tikzcd} \mathscr{A} \arrow[r, bend left, "F"{name=U}] \arrow[r, bend right, "G"'{name=D}] & \mathscr{B} \arrow[from=U, to=D, Rightarrow, "\alpha"] \end{tikzcd}$$

このノテーションでも、natural transformation α は、二重の矢印で、次のように表されていますね

$$\alpha\colon F \Longrightarrow G$$

natural transformationの全体を考える

ただ、先に見た次のDiagramは、$\mathrm{F}, \mathrm{G} : \mathrm{C} \to D$というcategory Cから category Dへの二つのfunctor F, Gの間の $\eta\colon F \Rightarrow G$ という natural transformation η の全体を捉えたものではありません。

$$
\begin{tikzcd}
F(x) \arrow[r, "\eta_x"] \arrow[d, "F(f)"'] & G(x) \arrow[d, "G(f)"] \\
F(y) \arrow[r, "\eta_y"'] & G(y)
\end{tikzcd}
$$

$$G(f) \circ \eta_x = \eta_y \circ F(f)$$

それは、この図式に、x, y 以外のCのオブジェクト zの
functor F, Gによるイメージを書き込んでみるとわかります。

$$
\begin{tikzcd}
F(x) \arrow[r, "\eta_x"] \arrow[d] & G(x) \arrow[d] \\
F(y) \arrow[r, "\eta_y"'] & G(y)
\end{tikzcd}
$$

$$\Downarrow$$

$$
\begin{tikzcd}
F(x) \arrow[rrr, "\eta_x"] \arrow[dr] \arrow[dd] & & & G(x) \arrow[dr] \arrow[dd] & \\
& F(z) \arrow[rrr, "\eta_z"'] \arrow[dl] & & & G(z) \arrow[dl] \\
F(y) \arrow[rrr, "\eta_y"'] & & & G(y) &
\end{tikzcd}
$$

natural transformation η には、以前のdiagramに登場していた η_x, η_y 以外にもたくさんのnatural transformation があることがわかります。

それは、category Cのオブジェクト $x, y, z, \ldots$に対応しています。

一般の場合、natural transformation η の全体は、次のように表すことができます。

$$\eta = \{\eta_x | x \in obj(C)\}$$

個々のη_xを、natural transformation η のcomponent と呼びます。

natural transformationの例

natural transformation $\eta: F \Longrightarrow G$ のイメージをもう少し具体的に持つ為に、ここでは、functor F, Gが単純なものである場合について、natural transformation が、どのようなものになるかを見てみましょう。

Functor F, Gがconstantな場合

一番単純な想定は、Functor F, Gが、共にconstant functorである場合です。

すなわち、Functor Fは、Cのオブジェクトを全てDの一つのオブジェクト dにうつし、Cの射を全てDの一つの射 Id_d(dの同一射です)に写すとします。

同様に、Functor Gは、Cのオブジェクトを全てDの一つのオブジェクト d′にうつし、Cの射を全てDの一つの射 $Id_{d'}$(d′の同一射です)に写すとします。

この時、$\eta: F \Longrightarrow G$は、普通の射 $\eta: d \rightarrow d'$ になります。

Functor Fがconstantな場合

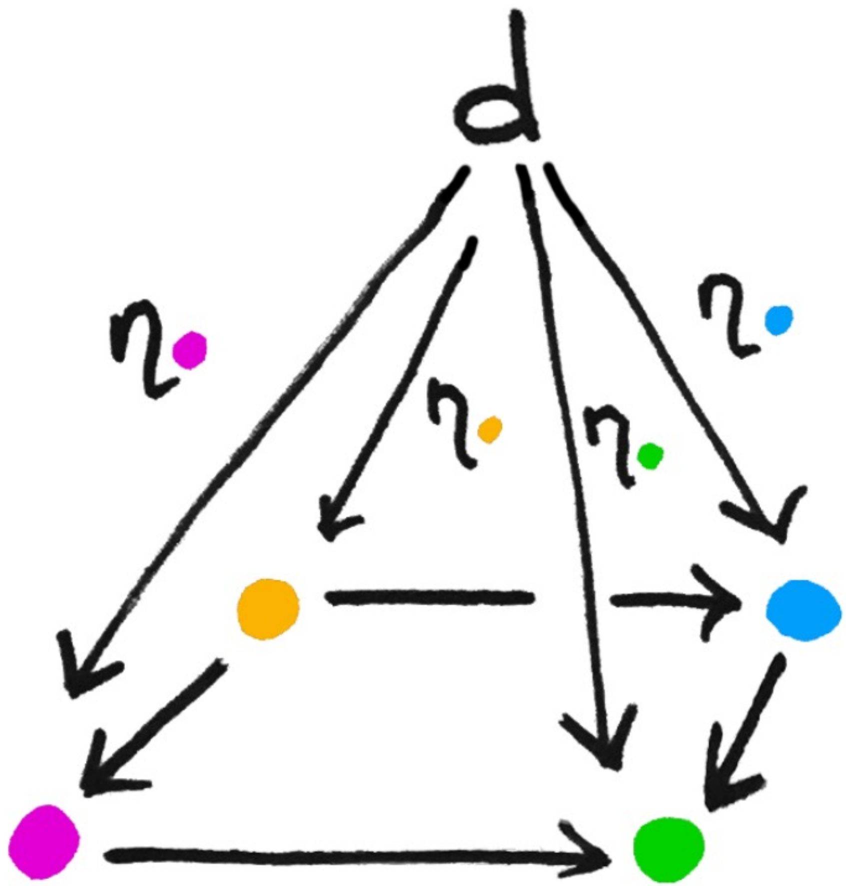
今度は、functor Fが前述のconstant functorで、functor Gが一般のfunctorである場合を考えてみましょう。

$\eta: F \Longrightarrow G$ は、Cのすべてのオブジェクトxに対応するηのcomponent $\eta_x: d \to G(x)$から構成されています。(functor Fはconstantなので、$F(x) = d$ です)

η_xは、Cの射$f: x \to y$について ηが natural transformationであるための条件 $G(f) \circ \eta_x = \eta_y \circ F(f)$ (これをnaturalityの条件と言います)を満たさなければなりません。

ただし、functor Fはconstantで、$F(f) = Id_d$ですので、この条件は、$G(f) \circ \eta_x = \eta_y$ になります。

Tai-Danaeは、(category Dが四つのオブジェクトからなる場合ですが)次のようなdiagramで、Fがconstatな場合の natural transformation η: $F \Rightarrow G$ 表して見せました。



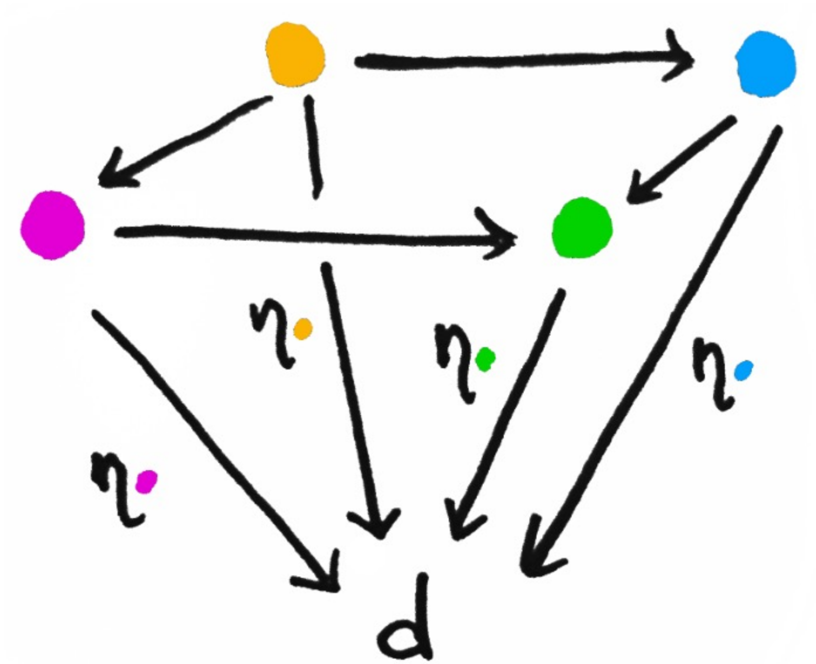
このdiagramの頂点のdは、constant functor Fによって与えられ、底辺の四角形は、functor Gによって与えられたものです。

頂点から、底辺の四つのノード（ピンク、緑、青、オレンジ）に向かう矢印は、natural transformation η のcomponentを表しています。

η のnaturalityの条件 $G(f) \circ \eta_x = \eta_y$ は、この四角錐の側面を構成する三つの三角形のdiagramが、可換であることを主張しています。

こうしたdiagramで表現されるη を、Gの上のcone (cone over G)と呼びます。

Functor Gがconstantな場合



functor Fが一般のfunctorで、functor Gがconstant functorである場合には、次のようなdiagramが、natural transformationを表現します。

ここでも、側面の三角形のdiagramの可換性が、naturalityを表現しています。

こうしたdiagramで表現されるηを、Gの下のcone (cone under G)と呼びます。

natural transformationの合成

natural transformationは写像なので、その合成を考えることができます。

二つのnatural transformation

$$\alpha : F \Longrightarrow G$$
$$\beta : G \Longrightarrow H$$

が与えられた時、αとβの合成

$$\beta \circ \alpha : F \Longrightarrow H$$

を考えようということです。

図で表すと、次のようになります。

$$
\begin{tikzcd}[column sep=huge]
\mathscr{A} \arrow[r, bend left=60, "F"] \arrow[r, "G" description] \arrow[r, bend right=60, "H"'] \arrow[r, phantom, shift left=4, "\Downarrow\alpha"] \arrow[r, phantom, shift right=4, "\Downarrow\beta"] & \mathscr{B}
\end{tikzcd}
,
$$

$\alpha: F \Longrightarrow G$

$\beta: G \Longrightarrow H$

$$\Downarrow$$

$$
\begin{tikzcd}[column sep=huge]
\mathscr{A} \arrow[r, bend left=30, "F"] \arrow[r, bend right=30, "H"'] \arrow[r, phantom, "\Downarrow\beta\circ\alpha"] & \mathscr{B}
\end{tikzcd}
$$

$\beta \circ \alpha: F \Longrightarrow H$

$A \in \mathcal{A}$であるすべてのAについて $(\beta \circ \alpha)_A = \beta_A \circ \alpha_A$と定義すればいいです。

また、任意のfunctor Fについて、$(1_F)_A = 1_{F(A)}$と定義すれば、同一natural transformation 1_Fが得られます。

$$
\begin{tikzcd}
\mathscr{A} \arrow[r, bend left=50, "F"{name=U}] \arrow[r, bend right=50, "F"'{name=D}] & \mathscr{B}
\arrow[from=U, to=D, phantom, "\Downarrow 1_F"]
\end{tikzcd}
$$

functor category

任意の二つのcategory A, Bについて、

- そのオブジェクトは、AからBへのfunctor
- その射は、それらの間のnatural transformation

であるようなcategoryが存在します。

こうしたcategoryを、functor category と呼んで、
$[A, B]$ あるいは B^Aで表します。

カテゴリー論基礎

Part 2

Limit

Agenda Part 2 Limit

- Part 2-1 Limit
  - Limit と Colimit とは何か？
  - Product
  - Pullback
- Part 2-2 その他のLimit
  - Equalizer
  - Inverse Limit
  - Terminal Object

Part 2-1

Limit

Limit / Colimit とは何か？

Limitという概念は、数学でよく現れる多くの構成を統一するものである。

あるcategoryでいくつかのオブジェクトや射を取り出し、それらから新しい対象を構成する方法に出会うときはいつも、limitか(その双対の概念である)colimitのどちらかを見ている可能性が高い。

例えば群論では、２つの群の間の準同型写像をとってそのkernelを作ることができる。この構成は群のcategoryにおけるlimitの例である。

また、２つの自然数をとり、その最小公倍数を作ることもできる。これは可約性で順序づけた自然数の半順序集合におけるcolimitの例である。

Tom Leinster

数学的な構成には、２つの類型があることにお気づきだろうか。

類型１:部分的なものを取る構成

例えば、要素を一つだけ持つ集合、共通部分、前像、積。

これらはすべて、与えられた集合から、ある条件をみたす構成要素の部分集合を選び出すことで構成される。

これらはlimitの例である。

類型 2: 物事を接着する構成

例えば、要素のない集合、disjoint union、集合の和、商。

これらは、物事を組み立てるか「接着」することによって構成される。

これらはcolimitの例である。

Tai-Danae Bradley

natural transformationのdiagramで Limit / Colimit を特徴づける

先のセッションで、natural transformationの「全体」を表すdiagramを見てきたので、このセッションではそれを利用して、Limit / Colimit の特徴づけをしてみたいと思います。

ここでの展開は、Tai-Danaeの次のblogに基づいています。

Limits and Colimits, Part 2 (Definitions)

https://www.math3ma.com/blog/limits-and-colimits-part-2

二つのfunctor F と X を考える

あるcategory Cが与えられているとしましょう。

Cのdiagramは、Indexing category I から I-shaped category C へのfunctor だと考えることができます。このfunctorを、$F: I \rightarrow C$ とします。Fは、Cのdiagramだと考えて構いません。

次に、Cのオブジェクトを一つ選んで、Iのすべてのオブジェクトを CのオブジェクトXにうつし、Iの射をすべてXの同一射 id_x にうつす constant functor を考えます。このfunctorを、再びXと呼ぶことにしましょう。$X: I \rightarrow C$ です。

XからFへ、あるいは、FからXへの natural transformation を考える

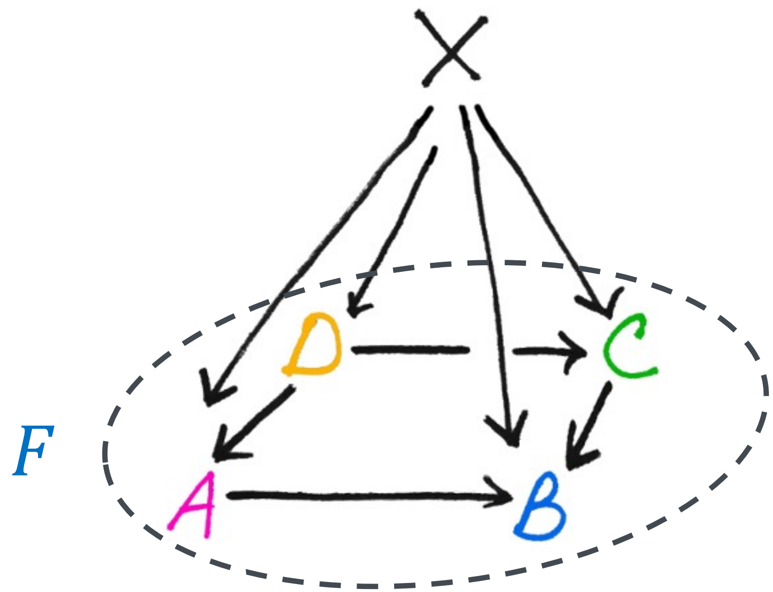
X, Fは共に$X, F: I \to C$なるfunctor ですので、XからFへ、あるいは、FからXへのnatural transformation を考えることができます。

XからFへのnatural transformation $X \Longrightarrow F$ を $Nat(X, F)$ と表すことがあります。natural transformation $F \Longrightarrow X$ は $Nat(F, X)$ と表されます

Xはconstant functor ですので、先のセッションで見たように、そのnatural transformationは、次のようなdiagramで表されることになります。

natural transformation

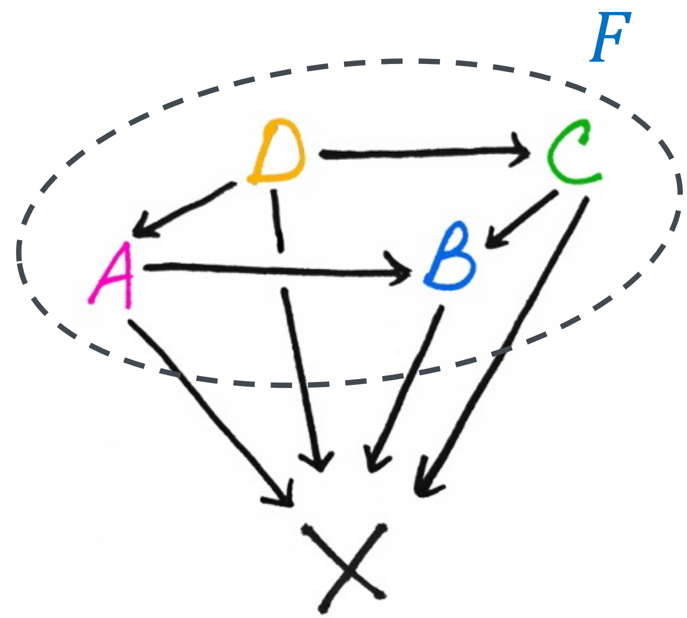
$X \Longrightarrow F$



cone over F

natural transformation

$F \Longrightarrow X$



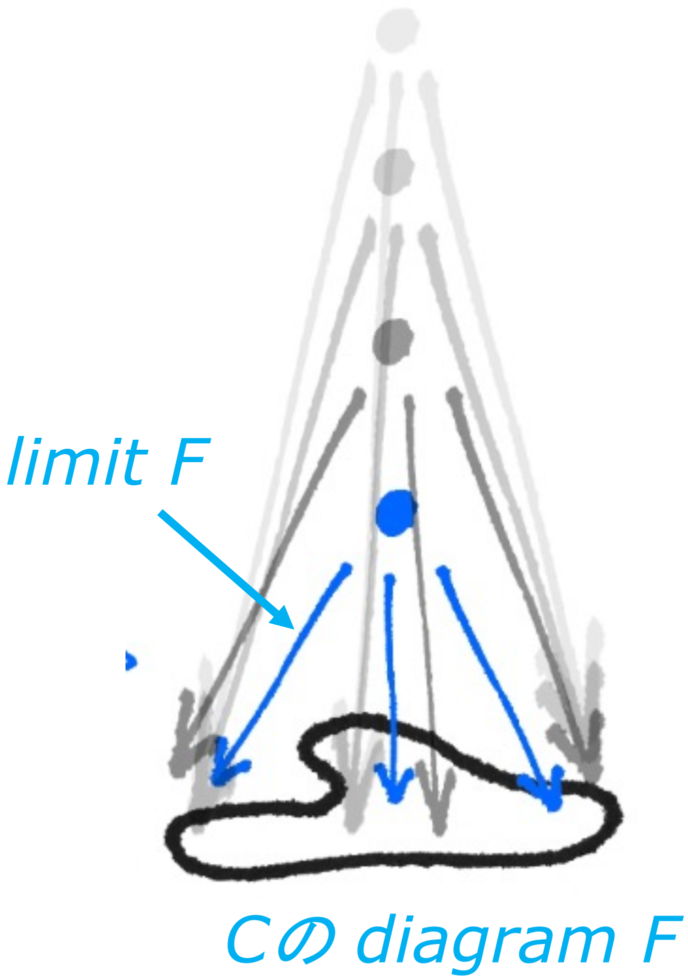
cone under F

limit のイメージ

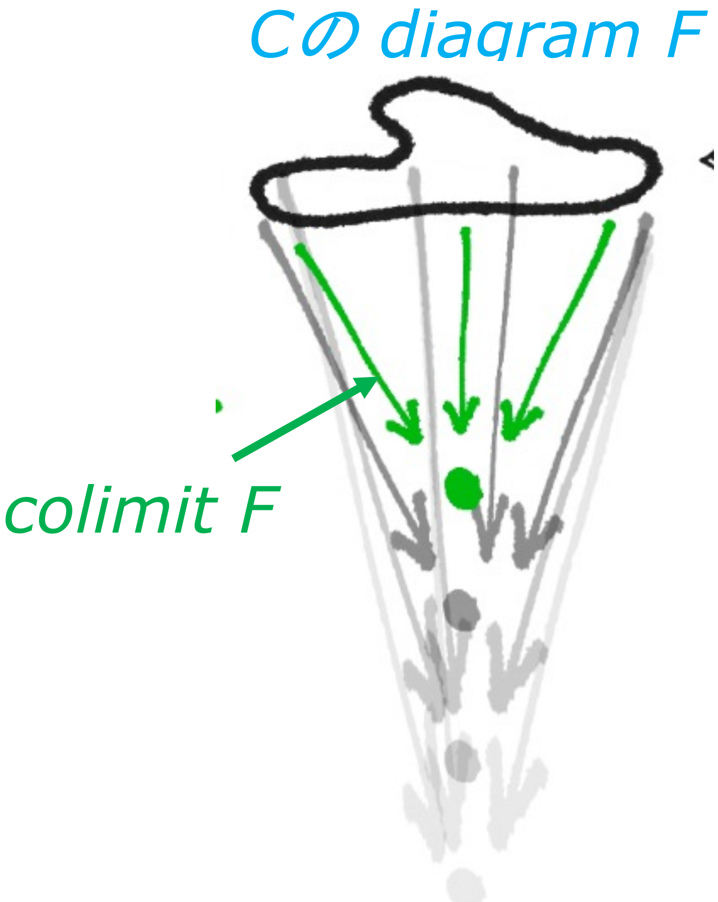
limitの直感的なイメージです。

CのオブジェクトXの選択を、いろいろ変化させるとnatural transformation $X \Longrightarrow F$ を表す corn over Fのcorn の形は変化します。

Fのlimitは、これらのcornのうちで、一番 F に「近い」ものだとイメージすることができます。



colimit のイメージ



colimitの直感的なイメージです。

CのオブジェクトXの選択を、いろいろ変化させるとnatural transformation $\mathrm{F} \Longrightarrow X$ を表すcorn under Fのcorn の形は変化します。

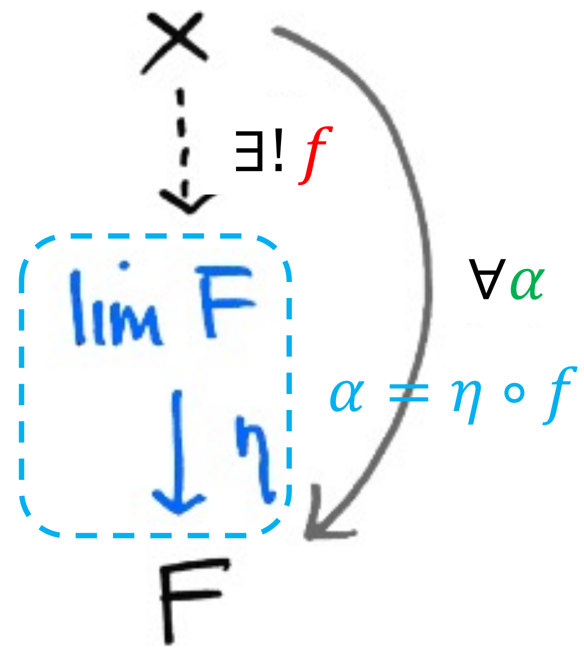
Fのcolimitは、これらのcornのうちで、一番 F に「近い」ものだとイメージすることができます。

limit の定義
universal corn over F

diagram $F: I \to C$ のlimitは、natural transformation $\eta: \lim F \Longrightarrow F$ を伴ったCのオブジェクト $\lim F$ で、次の性質を持ったcorn over Fです。

すべてのオブジェクトXと、すべてのnatural transformation $\alpha: X \Longrightarrow F$ について、射 $f: X \to \lim F$ で $\alpha = \eta \circ f$ が成り立つようなfがただ一つ存在する。

こうした性質を持つcornを、universal corn over Fと呼びます。

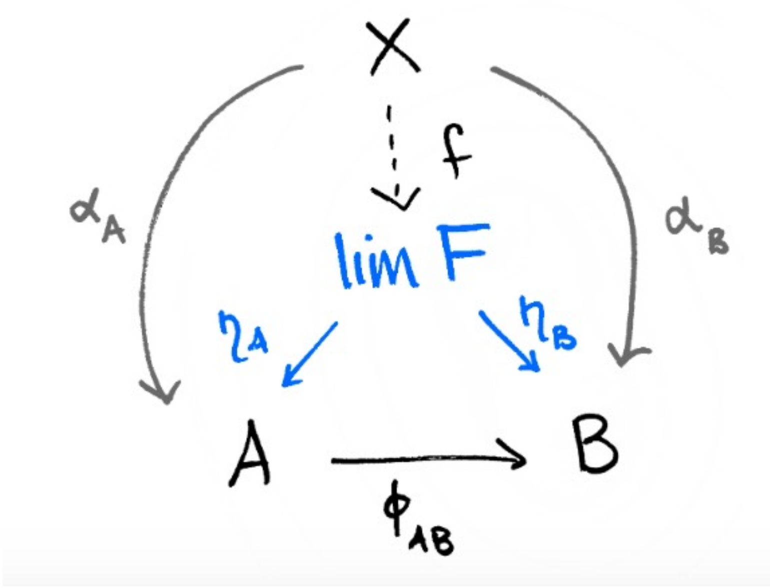


この定義は、次のように言い換えることができます。

diagram $F: I \to C$のlimitは、Cのオブジェクト lim Fと、Fのdiagram中のすべてのAと射ϕ_{AB}とに対して、$\eta_B = \phi_{AB} \circ \eta_A$を満たすような射$\eta_A: \mathrm{lim\ F} \to A$ とを一緒に考えものである。

これらの写像は、次のような性質を持っています。

すべてのオブジェクトXと$\alpha_\beta = \phi_{AB} \circ \alpha_A$を満たすすべての射 $\alpha_A: X \to A$ の集まりに対して、diagram中のすべてのオブジェクトAについて$\alpha_A = \eta_A \circ f$ を満たすユニークな射 $f: X \to \lim\ F$ が存在する



$$\hom(X, \lim F) \cong Nat(X, F)$$

colimit の定義
universal corn under F

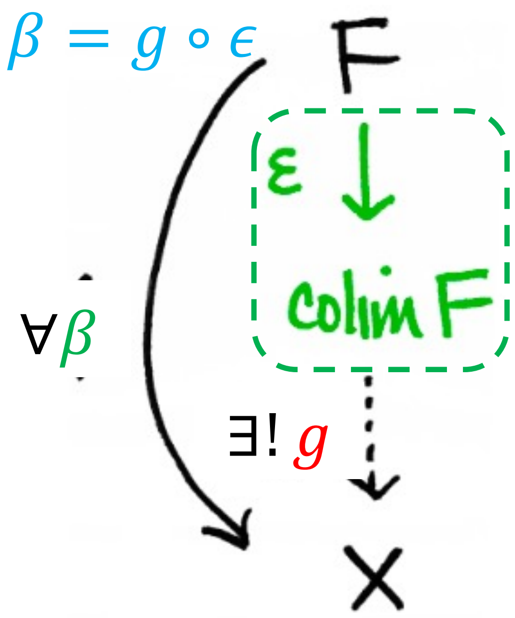
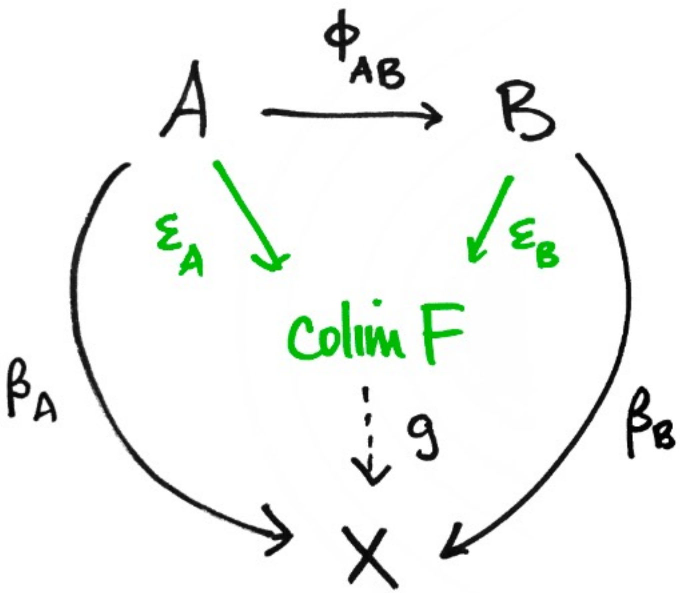


diagram $F: I \to C$のcolimitは、natural transformation $\epsilon: F \Longrightarrow colim\ F$ を伴った Cのオブジェクト $\mathrm{colim}\ F$ で、次の性質を持ったcorn under Fです。

すべてのオブジェクトXと、すべてのnatural transformation $\beta: F \Longrightarrow X$について、射 $\mathrm{g: colim\ F \to X}$ で $\beta = g \circ \epsilon$ が成り立つようなgがただ一つ存在する。

こうした性質を持つcornを、universal corn under F と呼びます。



この定義は、次のように言い換えることができます。

diagram $F: I \to C$の colimitは、Cのオブジェクト colim Fと、Fのdiagram中のすべてのAと射ϕ_{AB}とに対して、$\epsilon_A = \epsilon_B \circ \phi_{AB}$を満たすような射$\epsilon_A$: F $\to$ $colim\ F$とを一緒に考えものである。

これらの写像は、次のような性質を持っています。

すべてのオブジェクトXと$\beta_A = \beta_B \circ \phi_{AB}$を満たすすべての射 $\beta_A: A \to X$ の集まりに対して、diagram中のすべてのオブジェクトAについて$\beta_A = g \circ \epsilon_B$を満たすユニークな射 $f: F \to X$ が存在する

$$\mathrm{hom}(colim\ F, X) \cong Nat(F, X)$$

Product

数学において、より具体的にはカテゴリー理論において、universal propertyとは、ある構成法の結果を同型性まで(up to an isomorphism)特徴づける性質のことである。

したがって、universal propertyは、それらを構成するために選択された方法から独立して、いくつかのオブジェクトを定義するために使用することができる。

例えば、自然数から整数を、整数から有理数を、有理数から実数を、係数の体から多項式環を定義することは、すべてuniversal propertyの観点から行うことができる。

wikipedia

limitとcolimitの正式な定義は先で述べた。そこでは「[何か]の(co)limit」について述べた。

前回見たように、「何か」とは、indexing categoryから目的のcategoryへのfunctorであるdiagramのことである。

さらに、そのindexing categoryの形によって、(co)limitの名前が決まる:product、coproduct、pullback、pushoutプッシなど。

Tai-Danae Bradley

そろそろ、冗長になりつつあることにお気づきだろう。

*category **Set**, **Top**, **Group**, **Ring**, **Fvect** のproduct はすべて同じように感じられる。いずれの場合も、discrete diagramのlimitは、必要なときに余分な構造を与えられたデカルト積にすぎない。*

トポロジカル空間、群、環、ベクトル空間は、結局のところ、余分なものを持った集合にすぎないからだ。位相空間はトポロジーを持つ集合である。群とは、結合的な2項演算を持つ集合である。環は2つの結合的2項演算を持つ集合で、そのうちの1つは可換でもある。ベクトル空間は、結合的2項演算を持ち、体からの作用を持つ集合である。

Tai-Danae Bradley

Universal Property
前のセッションのまとめ

すべてのdiagram F 上のconeに対して、limitの役割を果たすcone $(\lim F, \eta)$が、ただ一つ存在します。このconeは、Cのオブジェクト $\lim F$ とnatural transformation $\eta\colon \lim F \Longrightarrow F$ のペアとして記述されています。

確かに、diagram F 上のcone には似たような役割を果たすcone (X, α) も存在します。αはおそらくηとよく似ているでしょう。

ただ、この類似は、偶然ではないのです。重要なことは、αがηと似た振る舞いをするのは、αがηから構成されているからだということです。もっと正確にいうと、あるユニークな射fが存在して、αはηを因子として、$\alpha = \eta \circ f$ のように分解されるからです。

Product の定義 1

ここでは、まず、indexing categoryによらない定義を、まず与えることにします。

Aをcategoryとし、X,YをAのオブジェクトとする時、XとYのproduct P は、ある性質を満たすPと射 p_1, p_2 で定義される。

$$
\begin{tikzcd}
 & P \arrow[dl, "p_1"'] \arrow[dr, "p_2"] & \\
X & & Y
\end{tikzcd}
$$

射 p_1, p_2 を、projection と呼ぶ。

Pと射 p_1, p_2 が満たすべき性質は、次のdiagramを満たすAのすべてのオブジェクトと射に対して、

$$\begin{tikzcd}
 & A \arrow[dl, "f_1"'] \arrow[dr, "f_2"] & \\
X & & Y
\end{tikzcd}$$

次のdiagramが可換となるような、ユニークな射 $f: A \to P$ が存在することである。

$$\begin{tikzcd}
 & A \arrow[d, dotted, "\bar{f}"] \arrow[ddl, "f_1"'] \arrow[ddr, "f_2"] & \\
 & P \arrow[dl, "p_1"] \arrow[dr, "p_2"'] & \\
X & & Y
\end{tikzcd}$$

Product の定義 2

二つのdiscreteなノードからなるindexing category I

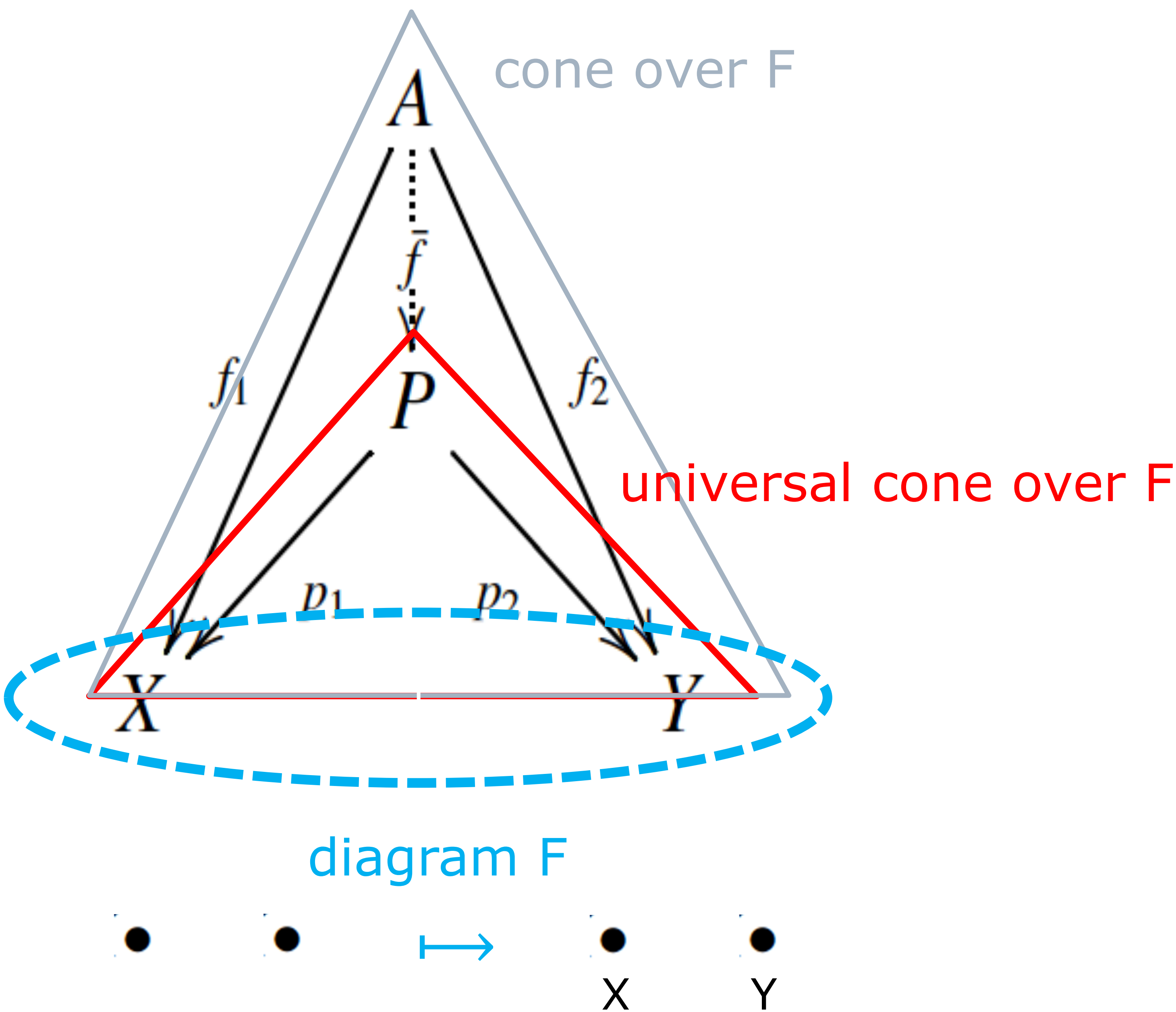
●　　　●

から、functor $F\colon I \to A$ で構成されるAのdiagram F

●　　　●

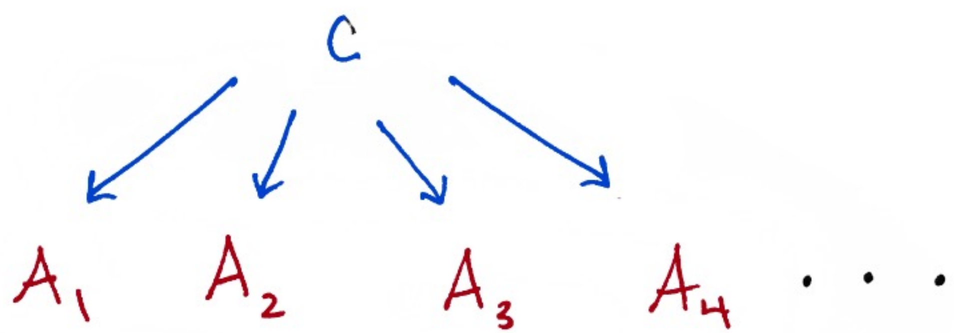
X　　　Y

のlimitが、XとYのproductである。


cone over F
A
$\bar{f}$
f_1
f_2
P
universal cone over F
p_1
p_2
X
Y
diagram F
X
Y

category Cでは、 ● ● ● ··· のようなdiscreteなdiagramのlimitは、product と呼ばれる。

productは、オブジェクトc と、cからdiagram上のオブジェクトへのuniversal propertyを満たす射で構成される。



集合のcategory **Set**の場合

$A_1, A_2, \ldots$ が 集合の時、そのproductは次のものからなる。

1. $A_1, A_2, \ldots$の直積(Cartesian product) $\prod_i A_i$
2. 射影(projection) $\prod_i A_i \to A_i$

a $\in A, b \in B$である
ペア(a, b)の集合

$$
\begin{tikzcd}
 & A \times B \arrow[dl] \arrow[dr] & \\
A & & B
\end{tikzcd}
$$

ペア(a, b)の一番目の
要素への射影

ペア(a, b)の二番目の
要素への射影

Coproduct の定義

二つのdiscreteなノードからなるindexing category Ｉ

から、functor $F: I \to A$ で構成されるAのdiagram F

のcolimitが、XとYのcoproductである。

Pullback

Pullback の定義 1

Aをcategoryとし、次のようなAのオブジェクトと射を考えます。

$$\begin{tikzcd} & Y \arrow[d, "t"] \\ X \arrow[r, "s"'] & Z \end{tikzcd}$$

この図式のPullbackは、射$p_1 : P \to X$ と射$p_2 : P \to Y$ を伴った、次に述べる性質を持った、AのオブジェクトPです。

$$\begin{tikzcd} P \arrow[r, "p_2"] \arrow[d, "p_1"'] & Y \arrow[d, "t"] \\ X \arrow[r, "s"'] & Z \end{tikzcd}$$

Aの、次のようなすべての可換図式について

$$
\begin{tikzcd}
A \arrow[r, "f_2"] \arrow[d, "f_1"'] & Y \arrow[d, "t"] \\
X \arrow[r, "s"'] & Z
\end{tikzcd}
$$

次の図式が可換となるような、すなわち、$p_1 \circ f = f_1, p_2 \circ f = f_2$
となるようなユニークな射 $f: A \to P$ が存在する。

$$
\begin{tikzcd}
A \arrow[dr, dotted, "\bar{f}"] \arrow[drr, bend left, "f_2"] \arrow[ddr, bend right, "f_1"'] & & \\
& P \arrow[r, "p_2"] \arrow[d, "p_1"'] & Y \arrow[d, "t"] \\
& X \arrow[r, "s"'] & Z
\end{tikzcd}
$$

Pullback の定義 2

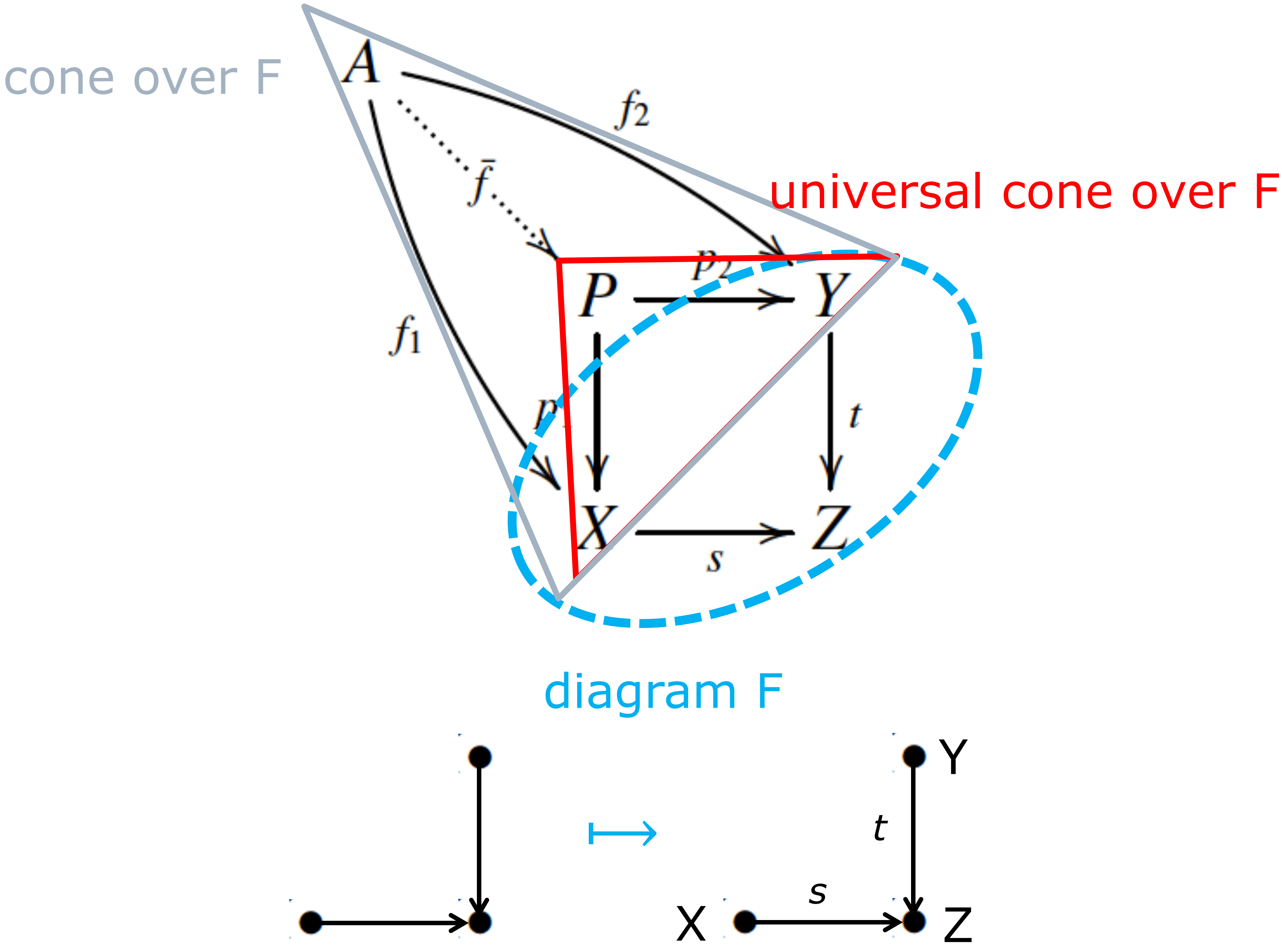
次のような indexing category I

$$
\begin{tikzcd}
 & \bullet \arrow[d] \\
\bullet \arrow[r] & \bullet
\end{tikzcd}
$$

から、functor $F: I \to A$ で構成されるAのdiagram F

$$
\begin{tikzcd}
 & Y \arrow[d, "t"'] \\
X \arrow[r, "s"] & Z
\end{tikzcd}
$$

のlimitが、Fのpullbackである。



cone over F
A
f_2
$\bar{f}$
f_1
universal cone over F
P
Y
X
Z
t
s
diagram F
Y
t
s
X
Z

pullbackの例

pullbackには、いろんな例があります。
そのいくつかを見ておきましょう。

集合でのpullback $X \times_Z Y$

$$\begin{tikzcd} X \times_Z Y \arrow[r, "p_2"] \arrow[d, "p_1"'] & Y \arrow[d, "t"] \\ X \arrow[r, "s"'] & Z \end{tikzcd}$$

p_1, p_2を $p_1(x, y) = x, p_2(x, y) = y$ を満たす射影と考えると、

$$X \times_Z Y = \{(x, y) \in X \times Y \mid s(x) = t(y)\}$$

は、pullback図式を満たすことがわかります。

それは、Z上で $s(x) = t(y)$という条件を満たす(x,y)のペアで、
$X \times Y$の部分集合です。これを $X \times_Z Y$と表すことにしましょう。

$Z = *$ の場合　$X \times Y$

$Z = *$ の場合、すなわちZが一つの要素しか持たない集合の場合のpullbackを考えてみましょう。

$$\begin{tikzcd}
X \times_* Y \arrow[r, "p_2"] \arrow[d, "p_1"'] & Y \arrow[d, "t"] \\
X \arrow[r, "s"'] & *
\end{tikzcd}$$

$$X \times_* Y = \{(x, y) \in X \times Y \mid s(x) = t(y) = *\}$$

この条件部分は、すべてのx,yについて成り立ちますので、

$Z = *$ の場合、$X \times_* Y$ は、すべての(x,y)の組、すなわち$X \times Y$に一致します。

Y ＝ ∗ の場合

Y ＝ ∗ としましょう。

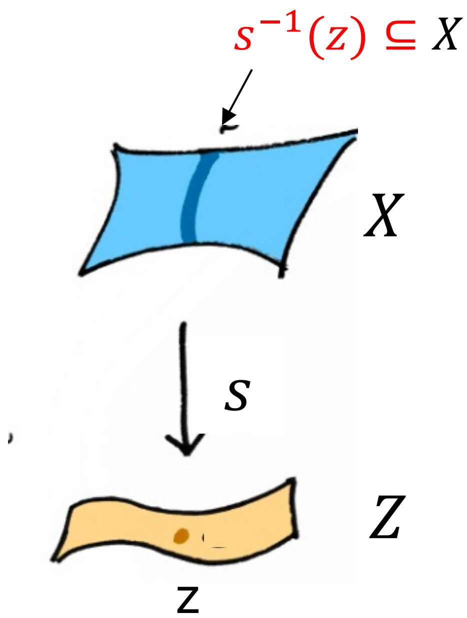
$$
\begin{tikzcd}
X \times_Z * \arrow[r, "p_2"] \arrow[d, "p_1"'] & * \arrow[d, "t"] \\
X \arrow[r, "s"'] & Z
\end{tikzcd}
$$

この時、pullbackは次の集合になります。

$$
\begin{aligned}
X \times_Z * &= \{(x, *) \in X \times * \mid s(x) = z\} \\
&\cong \{x \in X \mid s(x) = z\}
\end{aligned}
$$

$$X \times_Z * \equiv \{x \in X \mid s(x) = z\}$$

この集合は、sによって、Zのある要素zに写像されるXのすべての要素を含んでいます。それは、sのもとでのzのpreimage です。そうしたものをfiberと言います。



$\mathrm{Y} = *$ の場合、pullbackは、fiber $s^{-1}z \subseteq X$ になります。

X, YがZの部分集合の場合

X, YがZの部分集合の場合を考えましょう。

$$
\begin{tikzcd}
X\times_Z Y \arrow[r, "p_2"] \arrow[d, "p_1"'] & Y \arrow[d, "include"', "t"] \\
X \arrow[r, "include", "s"'] & Z
\end{tikzcd}
$$

$$X\times_Z Y = \{(x, y) \in X\times Y \mid include(x) = include(y)\}$$
$$= \{(x, y) \in X\times Y \mid x = y \ in\ C\}$$

$X\times_Z Y$は、$x = y$であるすべての(x, y)を含んでいます。そのことは、$x = y$がXの要素であると同時にYの要素でもあることを意味します。pullbackは、$X \cap Y$ に等しくなります。

Part 2-2

その他のLimit

Equalizer

Atiyahは数学を「アナロジーの科学」と表現した。この流れの中でいうと、カテゴリー理論の視点は数学的アナロジーである。

具体的には、カテゴリー理論は、あらゆる数学的文脈における現象を記述するために展開できる統一的な言語を提供する。

その一般性のレベルを考えると驚くべきことに思えるが、これらの概念は無意味なものではないのだが、多くの場合、その出現に先立っては、それほど明確に目に見えるものでもないのだ。

これは部分的には、視点の微妙なシフトによって達成される。

カテゴリー論的アプローチは、数学的オブジェクトを直接特徴付けるのではなく、同じ型のオブジェクト間の比較を与える射を重視する。

特定のオブジェクトに関連する構造は、そのUniversal Property、すなわち、類似の形を持つ他のオブジェクトに対するある種の標準的射の存在によって特徴づけられることが多い。

"A Survey of Categorical Concepts"　Emily Riehl

Equalizer の定義

次のような Indexing category I から category C への functor $F\colon I \to C$ で作られた次のような図形Fを考えます。

$$
\begin{tikzcd}
\bullet \arrow[r, shift left] \arrow[r, shift right] & \bullet
\end{tikzcd}
\quad \longmapsto \quad
\begin{tikzcd}
A \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & B
\end{tikzcd}
$$

この図形F上のconeは、Cのオブジェクトcと、$i_2 = f \circ i_1$ かつ $\mathrm{i}_2 = \mathrm{g} \circ i_1$を満たすCの射で構成されています。

$$
\begin{tikzcd}
 & c \arrow[dl, "i_1"'] \arrow[dr, "i_2"] & \\
A \arrow[rr, shift left, "f"] \arrow[rr, shift right, "g"'] & & B
\end{tikzcd}
$$

ただし、f, g が与えられた時、この条件で、i_2は i_1によって一意に決まりますので($i_2 = f \circ i_1$)、図形F上のcone は、次の形をしていると考えて構いません。

$$\begin{tikzcd} & c \arrow[dl, "i_1"'] & \\ A \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & B & \end{tikzcd}$$

$$\begin{tikzcd} c \arrow[r, "i_1"] & A \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & B \end{tikzcd}$$

この図形F上のconeのlimit を f と gの equalizer と言います。

ここでは、それを$eq(f, g)$と表すことにしましょう。

Equalizerの例

$$eq(f,g) \xrightarrow{i_1} A \underset{g}{\overset{f}{\rightrightarrows}} B$$

$$eq(f,g) = \{a \in A \mid f(a) = g(a)\}$$

$eq(f,g)$は、fとgを通過すると「等しくなる」Aの要素の集まり。

Cが、category $\boldsymbol{Set}, \boldsymbol{Top}, \boldsymbol{Group}, \boldsymbol{Ring}, \boldsymbol{FVect}$である場合。

- equalizerは、オブジェクトAのサブオブジェクト（先のcategoryに対応して、subset / subspace / subgroup / subring / subspace ）となり、
- 射i_1は inclusion map $\hookrightarrow$ で与えられる。

A, Bが集合で、gが定数関数の場合

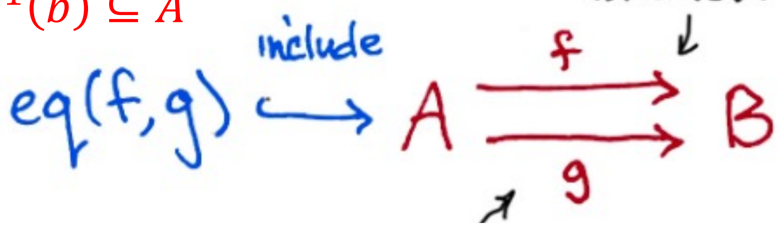
$b \in B$を一つ固定して、gをすべての$a \in A$について$g(a) = b$となる定数関数としましょう。この時、

$$eq(f, g) = \{a \in A \mid f(a) = g(a) = b\}$$

これは、fのもとで値bをとるAの要素の集まりなので、
equalizer は、b上のfiber $f^{-1}(b) \subseteq A$になります。

fは、任意の関数

$f^{-1}(b) \subseteq A$



gは、全てを $b \in B$に送る

A, Bが群で、gが単位元eを返す定数関数の場合

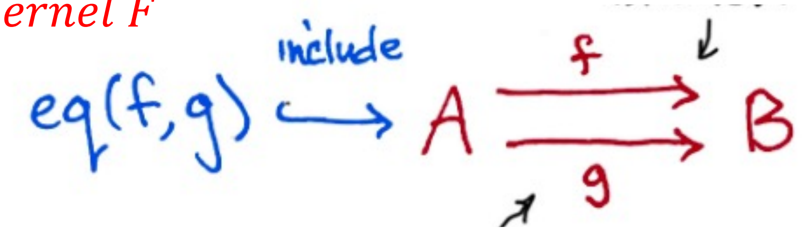
すべての$a \in A$について、$g(a) = e$とします。eは単位元です。
fを任意の群の準同型写像とすると、

$$eq(f, g) = \{a \in A \mid f(a) = g(a) = e\}$$

これは、準同型fのもとで単位元eに値をとる群Aの要素の集まりなので、equalizer は、fの$Kernel$ になります。

fは、任意の準同型

$Kernel\ F$



gは、全てを e∈ Bに送る

Inverse Limit

ここまで、product、equalizer、pullbackという3つの構成を見てきたが、これらには明らかに共通点がある。

それぞれは、いくつかのオブジェクトと、それらの間のいくつかの写像から始まる。それぞれに、我々はuniversal propertyを持つ新しいオブジェクトを、そのオブジェクトから元のオブジェクトへの写像とともに構成することを目指している。

Tom　Leinster

カテゴリー論の主要なテーマの1つは、集合論的に定義されることが多い数学的構造は、特定のカテゴリーのオブジェクトとしての構造からの射の記述、または、その構造への射の記述によって、完全に特徴付けることができるということである。

このような特徴付けは、そのように記述されたオブジェクトのuniversal property（普遍的性質）と呼ばれ、集合から他のカテゴリーに直ちに一般化することができる。

Emily Riehl

universal propertyによって定義されるオブジェクトは、適切なカテゴリーにおいて、limitまたはcolimitのいずれかに分類することができる。

limitとcolimitの概念はdualである。つまり、limitは、付加的な一貫した条件を課すことによって、他のオブジェクトから構築され、colimitはオブジェクト同士を接着することによって形成される。

Emily Riehl

Limitの定義 -- 振り返り

ここでは、diagram, cone に基づいたLimitの定義を改めてみておきましょう。

Diagram

$\mathcal{A}$ をcategory とし、I を small category とする。

functor $D: I \to \mathcal{A}$

を、$\mathcal{A}$のIの形のdiagram という。

diagram Dの例

$D: I \to \mathcal{A}$

product

$$\bullet \qquad \bullet \quad \longmapsto \quad X \qquad Y$$

pullback

$$\begin{tikzcd} & \bullet \arrow[d] \\ \bullet \arrow[r] & \bullet \end{tikzcd} \quad \longmapsto \quad \begin{tikzcd} & Y \arrow[d, "t"'] \\ X \arrow[r, "s"] & Z \end{tikzcd}$$

equalizer

$$\bullet \rightrightarrows \bullet \quad \longmapsto \quad \begin{tikzcd} A \arrow[r, shift left, "f"] \arrow[r, shift right, "g"'] & B \end{tikzcd}$$

Cone

Dの上のconeとは、$\mathcal{A}$のオブジェクトA（coneの頂点）と、次の条件を満たす写像の族

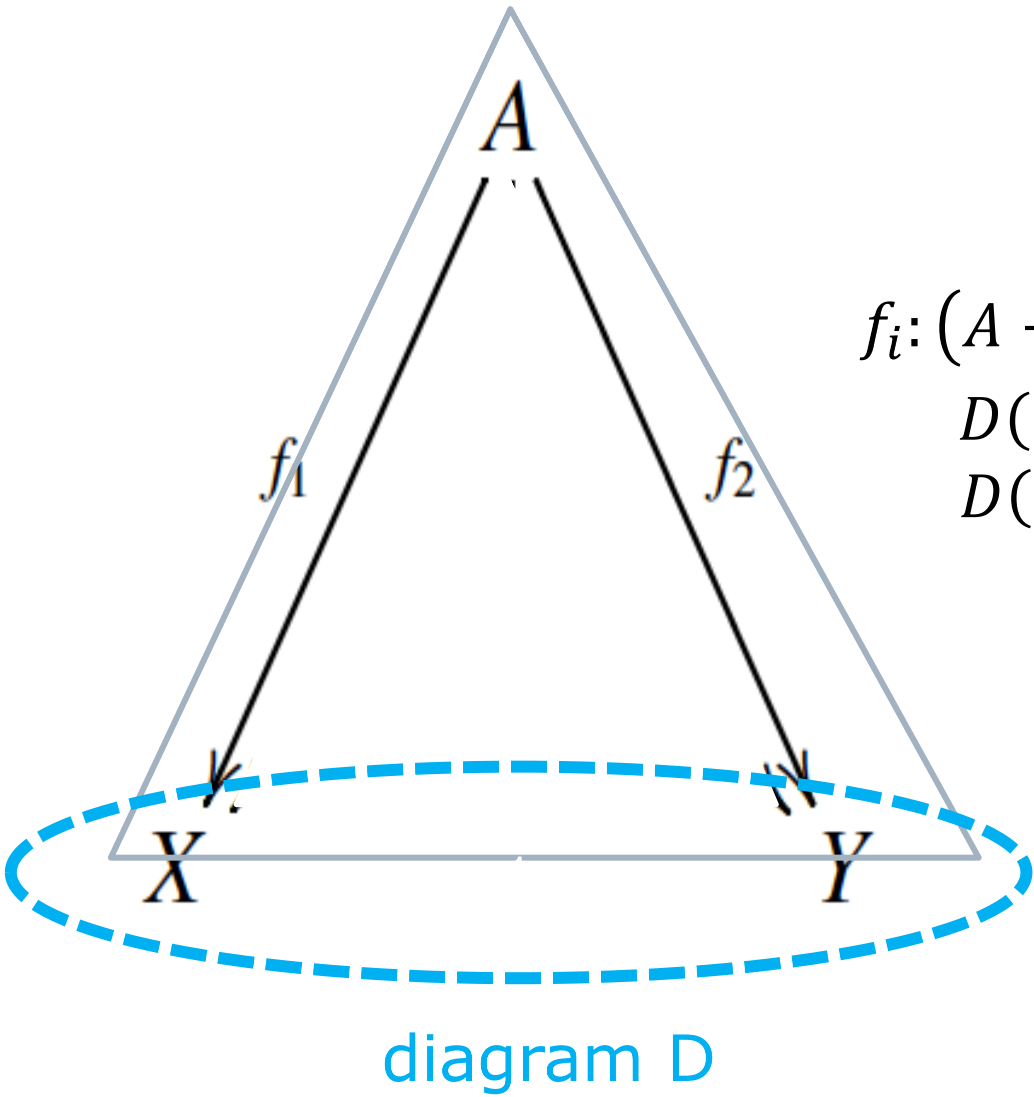
$$f_i\colon \left(A \to D(I)\right)_{i\in I}$$

のペア $(A, \{f_i\}_{i\in I})$ である。

Iのすべての写像 $u\colon I \to J$ と$\mathcal{A}$ のすべての写像について、次の図式は可換である。

$$\begin{tikzcd}
 & D(I) \arrow[d, "Du"] \\
A \arrow[ur, "f_I"] \arrow[dr, "f_J"'] & \\
 & D(J)
\end{tikzcd}$$

cone $(A, \{f_i\}_{i \in I})$ over D の例



$$f_i : \big(A \to D(I)\big)_{i \in I}$$
$$D(1) = X$$
$$D(2) = Y$$

Limit cone

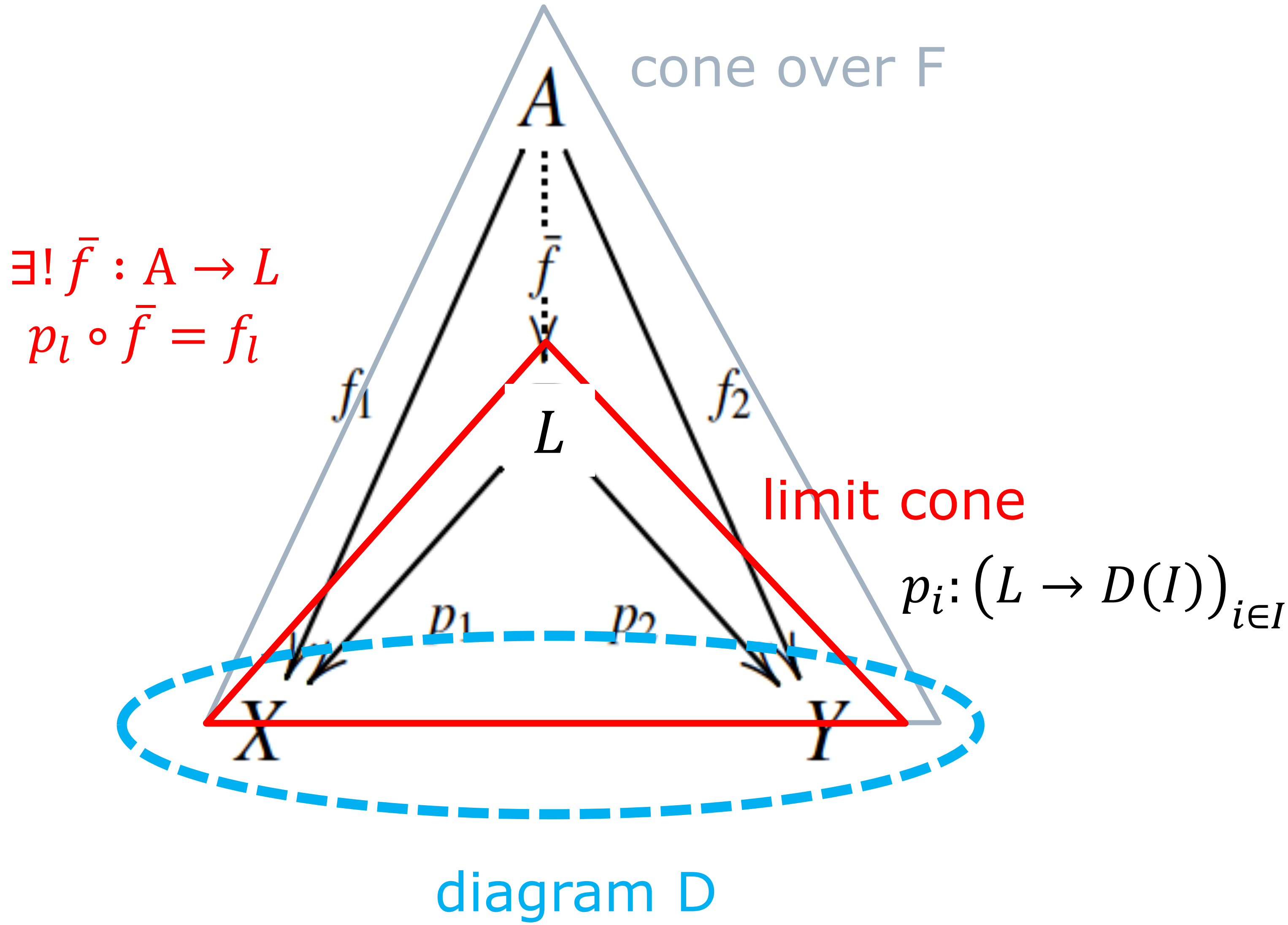
diagram Dのlimit coneは、次のような性質を持った Lを頂点とするcone $(L, \{p_i\}_{i \in I})$ である。ただし、

$$p_i : \left(L \to D(I)\right)_{i \in I}$$

である。

D上の任意のconeに対して、すべての$l \in I$について $p_l \circ \bar{f} = f_l$ となるようなユニークな写像 $\bar{f} : \mathrm{A} \to L$が存在する。

Limit cone $(L, \{p_i\}_{i \in I})$ の例



Inverse Limit

Inverse Limitは、次のようなfunctor Dで定義されるdiagram D のlimitとして定義されます。

$$D: I \to C$$

$$\bullet \leftarrow \bullet \leftarrow \bullet \leftarrow \cdots \quad \longmapsto \quad A_1 \xleftarrow{f_1} A_2 \xleftarrow{f_2} A_3 \xleftarrow{f_3} \cdots$$

この図形の inverse limit を次のように表します。

$$\varprojlim A_i$$

Inverse Limit $\varprojlim A_i$ を構成するもの

$$A_1 \xleftarrow{f_1} A_2 \xleftarrow{f_2} A_3 \xleftarrow{f_3} \cdots$$

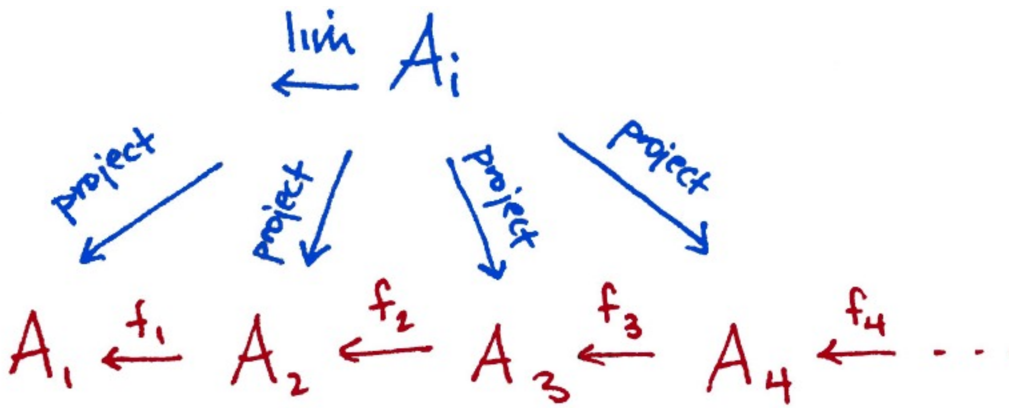
Inverse Limit $\varprojlim A_i$ は、次のものから構成されています。

- 直積 $\prod_i A_i$ （$A_1 \times A_2 \times A_3 \times \cdots$のこと）のサブ・オブジェクト$\varprojlim A_i$

  ($A_i \in \mathcal{A}$ である category $\mathcal{A}$に応じて、例えば $\mathcal{A}$ が、**Set**, **Top**, **Group**, **Ring**, **Vect** なら、sub(set/space/group/ring/space)になります)
  直積 $\prod_i A_i$は、列$(a_1, a_2, \cdots)$をすべて含んでいます。ここで、i番目の因子はi番目の写像f_iの像になります。すなわち、$f_i(a_{i+1}) = a_i$ です。

- 列$(a_1, a_2, \cdots)$からi番目の因子a_iへの射影 $\varprojlim A_i \to A_i$

Inverse Limit $\varprojlim A_i$ のイメージ

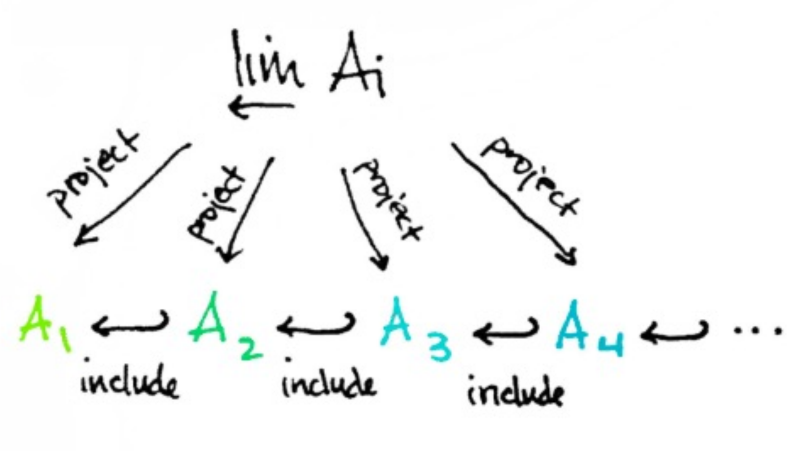


ここで写像f_iを消してしまうと（図形がdiscreteだとすると）、$\varprojlim A_i$は、直積 $A_1 \times A_2 \times A_3 \times \cdots$になります。

逆にいうと、写像f_iが与えられている図形の$\varprojlim A_i$は、直積 $A_1 \times A_2 \times A_3 \times \cdots$ではなく、次の条件を満たすその部分集合になります。

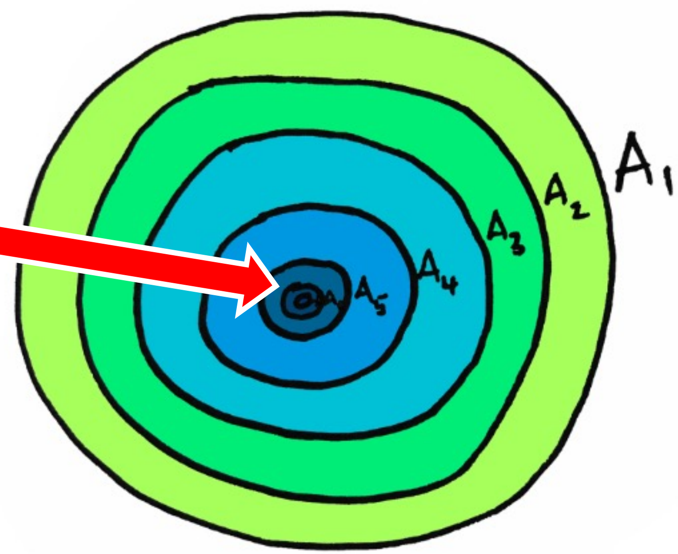
$$(a_1, a_2, a_3, \ldots) = (fa_2, fa_3, fa_4, \ldots)$$

写像 f_iがすべてinclusion map の場合



$$A_1 \supset A_2 \supset A_3 \supset \cdots$$

$\varprojlim A_i$ *は*
intersection
$\bigcap_i A_i$ *になる*



Terminal Object

先生: ここで、terminalオブジェクトとは何かをおさらいしておこう。Tがカテゴリ―Cにおけるterminalオブジェクトであるというのは

ということは.... どういうことだっけ？

生徒: 写像は1つだけということ

先生: 一つだけの写像？ それは、何から何への写像？

生徒: 他のオブジェクトからTへの写像

先生: 他のオブジェクトとは？

生徒: 他のどんなオブジェクトでも

先生: そうだね。他のどんなオブジェクトからでも、terminal オブジェクト Tへの写像は一つだけだ。

先生: さて、initial オブジェとは何だろうか？

生徒: initialオブジェクトとは、他のオブジェクトに正確に1つだけ対応するオブジェクトのことです。

先生: そうだね。しかし、その他のオブジェクトから始めることに慣れるべきだね。でも、それは不思議な定義だね。それが universal property による定義の特徴なんだ。集合の category でのterminal オブジェクトとは何だったけ？

生徒: 単一の要素。

先生: そのとおり。ちょうど1つの要素を持つ集合だ。集合のカテゴリのterminalオブジェクトは、自分自身の要素への同一性写像とちょうど1つの要素を持つ、任意の集合のことなんだ。

Conceptual Mathematics Lawvere

Terminal Object の定義

カテゴリー Cで、空のdiagramのlimitをterminal objectと呼びます。

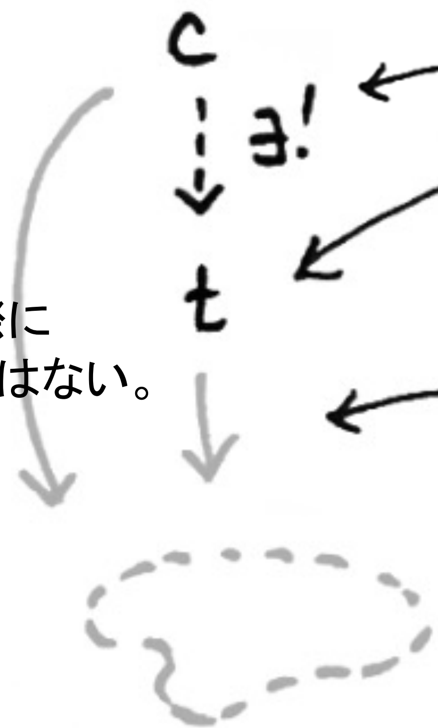
空のdiagramというのは、オブジェクトも射もないdiagram のことです。

limitの図式は、次のようになります。

空のdiagramのlimit

任意のCのオブジェクト

$c \in C$



universal propertyから
$c \to t$ であるただ一つの写像
が存在する

terminal object

この写像は、実際に
ここにあるわけではない。

この写像は、実際に
ここにあるわけではない。

というのも、この図形が
実際にここにあるわけ
ではないから

空のdiagram

Terminal Objectの性質

limitとuniversal propertyの定義から、terminal object t は、次の性質を持っていることがわかります。

Cのすべてのオブジェクトcに対して、$c \to t$ である射が、ただ一つ存在する。

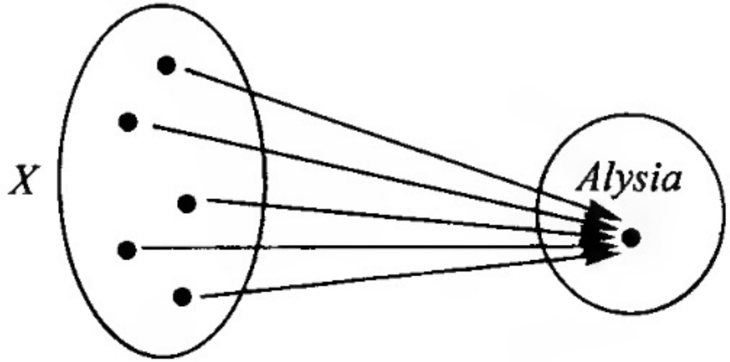
terminal オブジェクトは、どんなオブジェクトであっても、それへの写像は、一通りしか存在しないオブジェクトです。

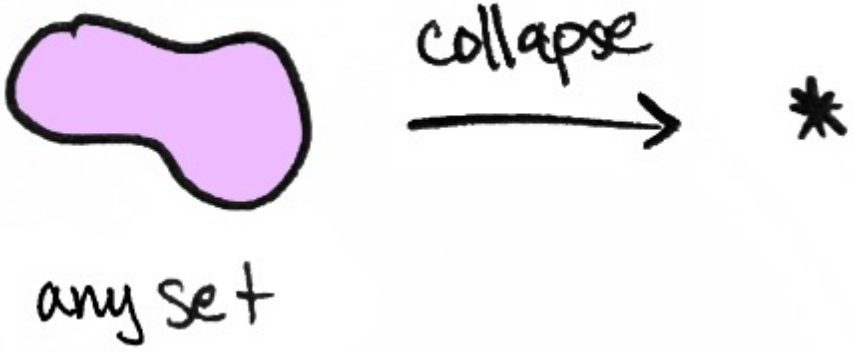
terminal object は、**1** や ***** で表されます。

集合でのterminal object

集合では、terminal object は、ただ一つの要素からなる集合です。ここではそれを * で表すことにしましょう。

どんな集合Sについても、$S \to *$ なる写像は一つしか存在しません。それはSのすべての要素を*に写す写像です。

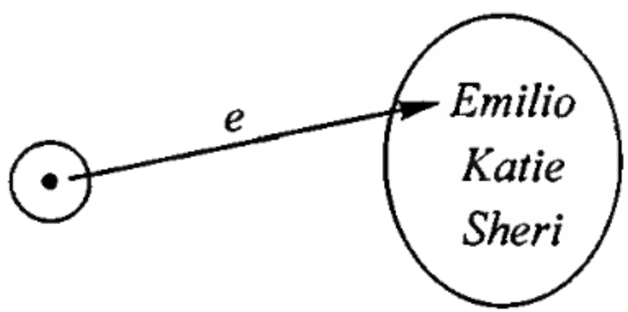




terminal objectから集合への写像

次のような三つの要素を持つ集合Sを考えましょう。

この時、たとえば、Sの要素の Emilioは、次のようなterminal object * からSへの写像 e と考えることができます。



同様に、Sの要素である Kate, Sherlもterminal object * からSへの写像と考えることができます。

$$* \rightarrow \ S$$

には、次のような写像が可能です。

$$e: * \mapsto Emilio$$
$$k: * \mapsto Kate$$
$$s: * \mapsto Sherl$$

$* \rightarrow \ S$ という形の写像を、Sの「点」と呼ぶことにすると、次のことがわかります。

写像であるSの「点」は、集合Sの「要素」に対応する