直積集合と「閉じている」

やっさん@物理のかぎプロジェクト

執筆中

代数の理論を追う上で **閉じている** と言う概念が出てきます．群，環，体の理論を追う上で必ず必要となる概念です．その **閉じている** と言う考え方を理解することがこの記事の目標です

1. 直積集合と写像

直積集合

「 **直積集合** 」とは同じ集合 S 同士の中から選んできた 2 つの元の組の集合のことを意味します．集合 S の直積集合は

$$S \times S$$

と表します．名前の中に "積" と言う言葉があるのですが直積集合とはただ **元の組み合わせのペアで作られる集合の事** を指します．

例えば炭水化物の集合

$$S = \{\mathrm{rice}, \mathrm{blead}, \mathrm{yakisoba}\}$$

を考えると，その直積集合 $S \times S$ の要素を具体的に書き出すと

$$\begin{aligned} S \times S = \{&(\mathrm{rice}, \mathrm{rice})(\mathrm{rice}, \mathrm{blead})(\mathrm{rice}, \mathrm{yakisoba}) \\ &(\mathrm{blead}, \mathrm{rice})(\mathrm{blead}, \mathrm{yakisoba})(\mathrm{blead}, \mathrm{yakisoba}) \\ &(\mathrm{yakisoba}, \mathrm{rice})(\mathrm{yakisoba}, \mathrm{blead})(\mathrm{yakisoba}, \mathrm{yakisoba})\} \end{aligned}$$

となります．今は集合の要素として **ペア** を考えているので別に blead と yakisoba だから焼きソバパンってわけではありません．また，$(\mathrm{rice}, \mathrm{blead})$ と $(\mathrm{blead}, \mathrm{rice})$ は違う要素とすることには注意です．ペアの順序も区別して考えます．なぜこれを区別するかは 次のセクション で触れます．

写像

「 **写像** 」とは集合の元を (一般には違う) 集合の元に対応させること，あるいは対応させるルールの事を言います．集合 T の元を集合 U の元に対応させる写像 f は

$$f : T \to U$$

と書きます．写像にも色々あって単写，全写，全単写と言う種類があるのですが，ここでは詳しくは触れません．気になる方は微積分，あるいは線形代数の教科書をひっくり返してみてください．

例えば， $T=\{1,2,3\}$ とし，写像 f を「自乗を計算すること」とすると集合 U は

$$\begin{aligned} U &= \{1^2, 2^2, 3^2\} \\ &= \{1, 4, 9\} \end{aligned}$$

となります．写像は計算だけでなく，複雑なことを考えることも出来ます．直積集合のところで出てきた炭水化物の直積集合 $S\times S$ について写像 g を「２つの炭水化物の中で最近食べてたもの」と考えて見るとします．

$$g: S\times S \to V$$

ここで，この写像の結果の集合 V について考えて見ると直積集合の要素のうち必ず片方の炭水化物が選ばれるはずなので

$$V=\{\mathrm{rice}, \mathrm{blead}, \mathrm{yakisoba}\}=S$$

となり，写像する前の集合が再び現れることがわかります．

この「直積集合を考えること」と「どんな写像が選ばれているのか」の二段階のステップを意識して例を確認していきましょう．

例１　四則演算

通常の数 (ベクトル等ではないという意味です) の四則演算ではまずその直積集合は複素数の集合を $\mathbf{C}$ として $\mathbf{C}\times\mathbf{C}$ と表せます．

写像は小中高で習ったとおりです．加法は直積集合の元の２つの数の和を計算し，減法は２つの数の差を，乗法，除法はそれぞれ２つの数の，積，商を計算します．集合 $\mathbf{C}$ (やその部分集合) で考えているときは特に断らず加法，減法，乗法，除法をそれぞれ

$$\begin{aligned} a+b&=c \\ d-e&=f \\ g\times h&=i \\ j\div k&=l \end{aligned}$$

と言う記号を使って表しました．四則演算はどれも結果は一般に $\mathbf{C}$ となっている[*1] ので，写像 f は直積集合の記号を使って

$$f: \mathbf{C}\times\mathbf{C}\to\mathbf{C}$$

と表します．

[*1] 厳密にはこの議論では不十分です．きっちりとやると演算のルールも定義しなければいけませんが話題がそれるのでここでは省きました．また，今は「数とは何ぞや」と言う根源的な問題は考えていないので「直積集合と写像」の一連の流れの例ということに納得してもらえれば十分です．

例２　ベクトルのスカラー倍

ベクトルのスカラー倍はスカラーの集合とベクトルの集合の直積集合に定義されます．ベクトルが n 次元複素数ベクトル空間 $\mathbf{C}^n$ (やその部分集合) の元とし，スカラーが複素数の集合を $\mathbf{C}$ とすれば直積集合は

$$\mathbf{C}^n \times \mathbf{C}$$

と表せます．

ベクトルのスカラー倍という写像は数ベクトルの各成分全てがスカラー倍されると言う写像なので，その結果は再び n 次元複素数ベクトル空間 $\mathbf{C}^n$ の元になるので，写像 f は

$$f : \mathbf{C}^n \times \mathbf{C} \to \mathbf{C}^n$$

となります．

例３多項式同士の積と和

この例では分けずに一気に書きます．多項式同士の積という写像 f は多項式の集合を P として

$$f : P \times P \to P$$

と書きます．そろそろ考え方には慣れてきたでしょうか?

同様の集合の記号を用いると多項式同士の和という写像 g は

$$g : P \times P \to P$$

と書きます．写像の表記が見た目上は多項式の積と一致します．

異なる写像なのに見分けがつかないのは困ります．そこで具体的に写像を明示する場合には演算のルールより細かく書きます．その書き方については この次のセクション で触れます．

例４ベクトルの内積，外積

n 次元複素数ベクトル空間 $\mathbf{C}^n$ の元同士の内積という写像 f は

$$f : \mathbf{C}^n \times \mathbf{C}^n \to \mathbf{C}$$

となります．(内積の結果はスカラー)

一方外積という演算 g は特に 3 次元複素数ベクトル空間 $\mathbf{C}^3$ の元について

$$g : \mathbf{C}^3 \times \mathbf{C}^3 \to \mathbf{C}^3$$

とかけます．(外積の結果もやっぱり 3 次元ベクトル)

例５複雑なもの (関数の内積)

ある，複素数値関数の集合の元を $\psi(x),\phi(x)$ とするとき，その関数同士の内積を

$$<\psi,\phi>\stackrel{\text{def}}{=}\int_{-\infty}^{\infty}\mathrm{d}x\psi^*\phi$$

と定義[*2] します．関数同士に定義される内積はその結果が一般には複素数 $\mathbf{C}$ となるので関数の集合を S として内積という写像 f は

$$f:S\times S\to\mathbf{C}$$

と表します．

2．写像に関する注意

写像の具体的な表記

写像を集合の記号だけを使って書くと非常に抽象的です．それに，写像としては全然違うのに一見見分けがつかない事にもなります．そこで写像をより具体的に明示したい場合には要素を明示して書く方法があります．例えば整数同士に定義された和であればその写像を f ，要素を $a,b\in\mathbf{Z}$ として

$$f:(a,b)\mapsto a+b$$

と書きます．集合で表現とするときとは違って飾りの付いた矢印 $\mapsto$ で表すのが主流です．これとは別に整数同士に定義された積はその写像を g ，要素 $a,b\in\mathbf{Z}$ とし

$$g:(a,b)\mapsto a\times b$$

と表せます．要素を明示すると区別がつくのですが集合を用いて表すと上の二つの写像は

$$f:\mathbf{Z}\times\mathbf{Z}\to\mathbf{Z}$$
$$g:\mathbf{Z}\times\mathbf{Z}\to\mathbf{Z}$$

となり，見た目上一致してしまいます．

写像の行われている集合について

写像は「 **どの集合の元で定義されているか** 」だけではなく，「 **写像の結果，出て来るものがどんな集合の元になっているか** 」にも気を使わないといけません．どういうことかというと，もし写像を考える集合が同じでも写像後の結果の集合が違えばそれは **違う写像** と考えるからです．例えば，乗法に関しても (実数) × (実数) → (実数) と (複素数) × (複素数) → (複素数) とは区別して扱わなければいけません．極端な例として

*2 この積分値が収束しなければ当然定義できないのでこの関数にある制約を加えなければいけません．ですが議論の方向がそれてしまうので，その “ある制約” は課されているものとしてください．

1. 多項式同士の積
2. 多項式同士の積の **次数**

と言う二つの写像を考えるとおいては実際に手で追う計算は一緒であっても全く違うことをしていると言うことが納得していただけると思います．前者の演算の結果の集合は “関数” ですが，後者は “自然数 ($\mathbf{N}$)” です．集合を用いて書けば関数の集合を S として

$$
\begin{aligned}
1.&: S \times S \to S \\
2.&: S \times S \to \mathbf{N}
\end{aligned}
$$

となります．[*3]

また，直積集合においては元のペアは順序付で区別しておかないといけません．これを具体的な写像を例に確認してみます．実数 $\mathbf{R}$ 同士の直積集合 $\mathbf{R} \times \mathbf{R}$ から実数 $\mathbf{R}$ への差 $d : \mathbf{R} \times \mathbf{R} \to \mathbf{R}$ について考えることにすると，この写像は $\mathbf{R} \times \mathbf{R}$ の要素 (a, b) を用いて

$$
(a, b) \mapsto a - b \tag{1}
$$

で定義するとします．このように差を定義すれば (a, b) とは異なる要素 (b, a) で定義される差は $(b, a) \mapsto b - a$ となります．(1) の意味は「直積集合の最初の値から，二番目の値を引く」というように考えるので，要素のペアの順番を入れ替えてしまうとその差は変わってしまいます．一般に差は可換ではありませんから ($a - b \neq b - a$) 一般には要素のペアの順序は区別しなければいけません．

3．「写像が閉じている」

写像が閉じている とは,

1. 同じ集合同士で定義される写像であること
2. 写像の結果が再びもとの集合と一致していること

と言う写像の事を指します．すぐ上の直積集合の記法を用いれば閉じている写像 f とは

$$
f : S \times S \to S
$$

とすっきり表すことができます．特にこのように閉じている写像の事を **二項演算**[*4] とも言います．

例で確認してみましょう．

[*3] よく用いる集合として実数 $\mathbf{R}$ ，複素数 $\mathbf{C}$ 有理数 $\mathbf{Q}$ ，無理数 $\mathbf{R}\backslash\mathbf{Q}$ ，整数 $\mathbf{Z}$ ，自然数 $\mathbf{N}$ があります．特に断りなくこれらの文字が出てきたらそれぞれの集合のことを表しています．また，好みによっては $\mathbb{R}, \mathbb{C}, \mathbb{Q}, \mathbb{R}\backslash\mathbb{Q}, \mathbb{Z}, \mathbb{N}$ 等を用いることもあります．

[*4] 演算とほぼ同義の言葉として「算法」と言う言葉も使われます．四則演算の「加法」「減法」「乗法」「除法」はこの名残かもしれません．また，この「演算」と言う言葉は物理のそれとは意味が異なるのでちょっと注意です．ベクトル解析で出てくる $\boldsymbol{\nabla}$ や，量子力学ででてくる一般の演算子 $\hat{\Omega}$ 等はより純粋な数学では「作用素」といいます．でも英語では同じ “operator” だったりします．

例１　数の加法

複素数同士の加法

複素数同士の加法は **閉じています．** 確かに複素数同士を足してもその結果が複素数になることは直感的にわかると思います．(この場合，たまたま実数になったとしてもそれは複素数の中に含まれると考えます)

複素数以外の集合同士の加法

集合を複素数に限らず，実数，有理数，無理数，整数，自然数同士として考えてみても全ての集合について **閉じています．** 実数同士を足して複素数になることはありませんし，有理数同士を足して無理数を作ることも出来ませんから，閉じていることが確認できます．(他の集合が閉じていることも確認してみてください)

少し複雑な集合同士の加法

この集合をさらに条件をつけて，集合 S を「3 で割ると 1 余る整数」という集合とすると，この集合同士の加法は **閉じていません．** なぜなら適当な整数 n, m を用いて 3 で割ると 1 余る数はそれぞれ $3n+1, 3m+1$ とかけますから，その和は

$$\begin{aligned}(3n+1)+(3m+1) &= 3n+1+3m+1 \\ &= 3(n+m)+2\end{aligned}$$

となり，「3 で割って 2 あまる整数」となったのでもとの集合からはみ出してしまっているからです．

例２　数の減法

複素数，実数，有理数，無理数，整数同士の減法

複素数，実数，有理数，無理数，整数同士の減法は **閉じています．** 加法と同様演算の結果，自分自身の集合からはみ出してしまうことはありません．

自然数同士の減法

自然数同士の減法は **閉じていません．** 例えば， $2-3$ と言う演算を行うと結果は -1 となり，これは自然数の集合からはみ出しています．写像を集合を用いて表すと

$$f : \mathbf{N} \times \mathbf{N} \to \mathbf{Z}$$

です．

例３　数の乗法

複素数，実数，有理数，無理数，整数，自然数同士の乗法

複素数，実数，有理数，無理数，整数，自然数同士の乗法は **閉じています．**

例４　数の除法

複素数，実数，有理数，無理数同士の除法

複素数，実数，有理数，無理数同士の除法は集合から **0を除いた集合で 閉じています．** 集合に0を含めると除法そのものが定義できなくなってしまうので通常は集合から0を除いて考えます.

例５　ベクトルの内積

n次元数ベクトルの内積は **閉じていません．** 演算の結果がスカラーになってしまうからです．演算を表すと

$$f : \mathbf{C}^n \times \mathbf{C}^n \to \mathbf{C}$$

です.

例７　多項式同士の和・積

多項式同士の和と積は **閉じています．** 多項式同士の和もやっぱり多項式になりますし，積も多項式となります.

また，集合に制限を加えると容易に閉じていない例を考えることが出来ます．n次の多項式の集合を P_n とすると和 f は

$$f_{n,m} : P_n \times P_m \to P_{\max\{m,n\}}$$

となり，積 g は

$$g_{n,m} : P_n \times P_m \to P_{n+m}$$

となるので，同じ集合同士で考えて見ると和では

$$f_{n,n} : P_n \times P_n \to P_n$$

となり， **閉じている** のですが積の方はというと

$$g_{n,n} : P_n \times P_n \to P_{2n}$$

となり， **閉じていません．**

例題

以下の例が閉じているかを頭の中で考えてみてください．閉じていなければ反例を探してみてください.

例題１

偶数同士の加法と，奇数同士の加法

例題２

偶数同士の減法と，奇数同士の減法

例題３

「３で割ったあまりが１となる数」数同士の乗法

例題４整数，自然数の除法

整数，自然数の除法

例題５ベクトルの外積

ベクトルの外積

４．閉じていたらどうなのか

二項演算が閉じていると，議論の対象を限ることが出来ます．

ひとつは演算の結果が演算する元と同じ集合になっているので複数回の演算が可能になることが挙げられます．例えば，実数は三項の演算が出来ますがベクトルの内積はそういうわけには行きません．

また，一旦集合を限ると議論を一般化しなくてよいこともあります．例えば議論を始める際に，集合を実数に限っておけば閉じた演算を繰り返す限りは複素数の存在を考えなくてもよくなります．線形代数をやるときによく，「複素数または実数」の集合を $\mathbf{K}$ とします．なぜなら個々の成分などの演算はいずれも閉じているので最初に実数とすれば最後まで実数に限った議論が複素数とした時と同様に行えるからです．[*5] 同じ議論が出来れば二回やる必要が無いので使っていると言うのが理由のひとつです．ですから，この場合はひとつの議論をしているときに実数と複素数をほいほい変えていいというわけではありません．

代数の分野で閉じていることを前提に議論をするのはは特に一つ目に挙げた理由が大きいようです．演算の結果が閉じていれば複数回の演算が可能になりより多様な議論につなげられるんですね．

[*5] 実は議論の抽象度によっては実数 or 複素数である必要すらなく，“体” でさえあれば左記の二つでなくても構いません．体については 体 を参照してください．