

## 02 級数

黒木玄

2018-04-18

- Copyright 2018 Gen Kuroki
- License: MIT <https://opensource.org/licenses/MIT> (<https://opensource.org/licenses/MIT>)

このファイルは次の場所できれいに閲覧できる:

- <http://nbviewer.jupyter.org/github/genkuroki/Calculus/blob/master/02%20series.ipynb>  
(<http://nbviewer.jupyter.org/github/genkuroki/Calculus/blob/master/02%20series.ipynb>)

このファイルは [Julia Box](https://juliabox.com) (<https://juliabox.com>) で利用できる.

自分のパソコンに [Julia 言語](https://julialang.org/) (<https://julialang.org/>) をインストールしたい場合には

- [WindowsへのJulia言語のインストール](http://nbviewer.jupyter.org/gist/genkuroki/81de23edcae631a995e19a2ecf946a4f) (<http://nbviewer.jupyter.org/gist/genkuroki/81de23edcae631a995e19a2ecf946a4f>)

を参照せよ.

論理的に完璧な説明をするつもりはない. 細部のいい加減な部分は自分で訂正・修正せよ.

## 目次

### 1 級数

#### 1.1 絶対収束と条件収束

- 1.1.1 絶対収束と条件収束の定義
- 1.1.2 絶対収束級数の収束先は和を取る順番によらないこと

#### 1.2 絶対収束級数の例

- 1.2.1 絶対収束級数の例1 (Riemannのゼータ函数)
- 1.2.2 絶対収束級数の例2

#### 1.3 条件収束級数の例

- 1.3.1 条件収束級数の例1
- 1.3.2 Euler変換による交代級数の数値計算
- 1.3.3 無限和と積分の交換に関する注意
- 1.3.4 条件収束級数の例2

### 2 複素数の級数

- 2.1 複素数の絶対収束級数
- 2.2 条件収束する複素級数の例
- 2.3 多重対数函数
- 2.4 二重対数函数の $\zeta(2)$ の近似計算への応用
- 2.5 二重対数函数の高木貞治『解析概論』§35の例2(p.113)への応用
- 2.6 二重対数函数の五項関係式

### 3 無限積

- 3.1 無限積の絶対収束
- 3.2 絶対収束する無限積の収束先が積の順序によらないこと
- 3.3 量子二重対数函数
  - 3.3.1 量子二重対数函数と量子指数函数との関係
  - 3.3.2  $q$ 二項定理
  - 3.3.3  $q$ 指数函数の加法公式
  - 3.3.4 量子二重対数函数の五角形恒等式

```
In [1]: 1 using Plots
2 gr(); ENV["PLOTS_TEST"] = "true"
3 #clibrary(:colorcet)
4 clibrary(:misc)
5
6 ▼ function pngplot(P...; kwargs...)
7     sleep(0.1)
8     pngfile = tempname() * ".png"
9     savefig(plot(P...; kwargs...), pngfile)
10    showing("image/png", pngfile)
11 end
12 pngplot(; kwargs...) = pngplot(plot!(); kwargs...)
13
14 ▼ showing(mime, fn) = open(fn) do f
15     base64 = base64encode(f)
16     display("text/html", """""")
17 end
18
19 using SymPy
20 #sympy[:init_printing](order="lex") # default
21 #sympy[:init_printing](order="rev-lex")
22
23 using SpecialFunctions
24 using QuadGK
25 import PyPlot
```

# 1 級数

## 1.1 絶対収束と条件収束

### 1.1.1 絶対収束と条件収束の定義

**定理:**  $a_n \geq 0$  であると仮定する. そのとき, ある正の定数  $M$  が存在して, すべての番号  $n$  について  $\sum_{k=1}^n a_k \leq M$  が成立している

ならば, 無限級数  $\sum_{n=1}^{\infty} a_n$  は有限の値に収束する.

**証明:**  $s_n = \sum_{k=1}^n a_n$  とおくと,  $s_n$  は上に有界な単調増加実数列になる. ゆえに実数の連続性より,  $s_n$  は有限の値に収束する.  $\square$

**定理:**  $\sum_{n=1}^{\infty} |a_n| < \infty$  ならば級数  $\sum_{n=1}^{\infty} a_n$  は収束する. そのとき, 級数  $\sum_{n=1}^{\infty} a_n$  は**絶対収束する**という.

**証明:**  $s_n = \sum_{k=1}^n a_k$  とおく. 任意に  $\varepsilon > 0$  を取る.  $\sum_{n=1}^{\infty} |a_n| < \infty$  より, ある番号  $N$  が存在して,  $n \geq N$  ならば

$$\sum_{k=n+1}^{\infty} |a_k| < \varepsilon$$

となる. ゆえに  $m \geq n \geq N$  のとき,

$$|s_m - s_n| = \left| \sum_{k=n+1}^m a_k \right| \leq \sum_{k=n+1}^m |a_k| \leq \sum_{k=n+1}^{\infty} |a_k| < \varepsilon.$$

ゆえに  $s_n$  はCauchy列になる. ゆえに実数の完備性より,  $s_n$  は収束する.  $\square$

**注意:** 上の定理の逆は成立しないことに注意に注意せよ. 級数  $\sum_{n=1}^{\infty} a_n$  が収束していても  $\sum_{n=1}^{\infty} |a_n| = \infty$  となる場合がある. そのと

き, 級数  $\sum_{n=1}^{\infty} a_n$  は**条件収束する**という.  $\square$

**注意:** 上の定理は  $a_n$  が実数列でも複素数列でも成立している. さらに  $a_n$  が  $\mathbb{R}^n$  や  $\mathbb{C}^n$  の点列でかつ絶対値をノルムで置き換えた場合にも成立している. 直観的には「長さが有限の紐の先端に対応する点が収束先として存在する」というような発想で考えればよい.

より一般にはノルム空間においては上の定理が成立することと完備性の条件は同値になっている. 実数列の場合には**実数の連続性**と同値になっている.

**実数の連続性**について理解していなくても, 絶対収束の概念を理解していれば足りることは多い.  $\square$

### 1.1.2 絶対収束級数の収束先は和を取る順番によらないこと

**定理:** 絶対収束する無限級数の収束先は足し上げる順序をどのように変えても同じになる.  $\square$

このノートでは, 絶対収束する無限級数  $\sum_{n=1}^{\infty} a_n$  に関する以下の2つの結果を証明しておくことにしよう.

(1)  $k \mapsto n_k$  は正の整数全体と正の整数全体を一对一に対応させる写像であるならば,  $\sum_{k=1}^{\infty} a_{n_k}$  も絶対収束して,  $\sum_{k=1}^{\infty} a_{n_k} = \sum_{n=1}^{\infty} a_n$  が成立する.

(2)  $(k, l) \mapsto n_{k,l}$  が正の整数の対の全体と正の整数全体を一对一に対応させる写像であるならば, すべての  $k$  について  $\sum_{l=1}^{\infty} a_{n_{k,l}}$  は絶対収束し, さらに  $\sum_{k=1}^{\infty} \sum_{l=1}^{\infty} a_{n_{k,l}}$  も絶対収束して,  $\sum_{k=1}^{\infty} \sum_{l=1}^{\infty} a_{n_{k,l}} = \sum_{n=1}^{\infty} a_n$  が成立する.

**証明:**  $M = \sum_{n=1}^{\infty} |a_n|$  とおく.

(1)  $n_1, n_2, \dots, n_K$  の最大値を  $N$  と書くと,

$$\sum_{k=1}^K |a_{n_k}| \leq \sum_{n=1}^N |a_n| \leq M < \infty$$

なので  $\sum_{k=1}^{\infty} |a_{n_k}|$  は収束する. つまり  $\sum_{k=1}^{\infty} a_{n_k}$  は絶対収束する.

$n_k = 1, 2, \dots, N$  となる  $k$  の最大値を  $K_N$  書くと,  $n_1, n_2, \dots, n_{K_N}$  は  $1, 2, \dots, N$  をすべて含み,  $N \rightarrow \infty$  のとき  $K_N \rightarrow \infty$  となり,

$$\left| \sum_{k=1}^{K_N} a_{n_k} - \sum_{n=1}^N a_n \right| \leq \sum_{n=N+1}^{\infty} |a_n| = M - \sum_{n=1}^N |a_n| \rightarrow 0 \quad (N \rightarrow \infty)$$

となるので,  $\sum_{k=1}^{\infty} a_{n_k} = \sum_{n=1}^{\infty} a_n$  となることがわかる.

(2) 各  $n_{k,1}, n_{k,2}, \dots, n_{k,L}$  の最大値を  $N$  と書くと,

$$\sum_{l=1}^L |a_{n_{k,l}}| \leq \sum_{n=1}^N |a_n| \leq M < \infty$$

なので,  $\sum_{l=1}^{\infty} a_{n_{k,l}}$  は絶対収束する.

任意に  $\varepsilon > 0$  を取る.  $k = 1, 2, \dots, K$  ごとに  $L_k$  を十分に大きくすると,  $\sum_{l=1}^{\infty} |a_{k,l}| \leq \sum_{l=1}^{L_k} |a_{k,l}| + \varepsilon$  となるので,  $n_{k,l}$  ( $k = 1, 2, \dots, K, l = 1, 2, \dots, L_k$ ) の最大値を  $N$  と書くと,

$$\sum_{k=1}^K \left| \sum_{l=1}^{\infty} a_{k,l} \right| \leq \sum_{k=1}^K \sum_{l=1}^{\infty} |a_{n_{k,l}}| \leq \sum_{k=1}^K \sum_{l=1}^{L_k} |a_{n_{k,l}}| + K\varepsilon \leq \sum_{n=1}^N |a_n| + K\varepsilon \leq M + K\varepsilon.$$

$\varepsilon > 0$  はいくらでも小さくできるので,

$$\sum_{k=1}^K \left| \sum_{l=1}^{\infty} a_{k,l} \right| \leq \sum_{k=1}^K \sum_{l=1}^{\infty} |a_{k,l}| \leq M < \infty.$$

ゆえに  $\sum_{k=1}^{\infty} \sum_{l=1}^{\infty} a_{n_{k,l}}$  も絶対収束する.

任意に  $\varepsilon > 0$  を取る.  $\{n_{k,l} \mid k = 1, 2, \dots, K, l = 1, 2, 3, \dots\} \supset \{1, 2, \dots, N\}$  となるような最小の  $K$  を  $K_N$  と書くことにする. このとき,  $N \rightarrow \infty$  ならば  $K_N \rightarrow \infty$  となる. 各  $k = 1, 2, \dots, K_N$  に対して, ある  $L_k$  が存在して,

$$\{n_{k,l} \mid k = 1, 2, \dots, K, l = 1, 2, \dots, L_k\} \supset \{1, 2, \dots, N\}$$

かつ

$$\left| \sum_{l=1}^{\infty} a_{n_{k,l}} - \sum_{l=1}^{L_k} a_{n_{k,l}} \right| < \frac{\varepsilon}{K_N}$$

となる. ゆえに,

$$\begin{aligned} \left| \sum_{k=1}^{K_N} \sum_{l=1}^{\infty} a_{n_{k,l}} - \sum_{n=1}^N a_n \right| &\leq \left| \sum_{k=1}^{K_N} \sum_{l=1}^{\infty} a_{n_{k,l}} - \sum_{k=1}^{K_N} \sum_{l=1}^{L_k} a_{n_{k,l}} \right| + \left| \sum_{k=1}^{K_N} \sum_{l=1}^{L_k} a_{n_{k,l}} - \sum_{n=1}^N a_n \right| \\ &\leq \sum_{k=1}^{K_N} \left| \sum_{l=1}^{\infty} a_{n_{k,l}} - \sum_{l=1}^{L_k} a_{n_{k,l}} \right| + \sum_{n=N+1}^{\infty} |a_n| \\ &< \varepsilon + \sum_{n=N+1}^{\infty} |a_n|. \end{aligned}$$

$\varepsilon > 0$  はいくらでも小さくできるので,

$$\left| \sum_{k=1}^{K_N} \sum_{l=1}^{\infty} a_{n_{k,l}} - \sum_{n=1}^N a_n \right| \leq \sum_{n=N+1}^{\infty} |a_n|.$$

右辺は  $N \rightarrow \infty$  のとき 0 に収束するので,  $\sum_{k=1}^{\infty} \sum_{l=1}^{\infty} a_{n_{k,l}} = \sum_{n=1}^{\infty} a_n$  が成立することがわかる.  $\square$

上の定理から次の系(corollary, 当然の結果)がただちに導かれる.

**系:**  $a_n \geq 0$  のとき  $a_1, a_2, \dots$  の無限和は足し上げる順序をどのように変えても同じになる.  $\square$

**注意:**  $a_n \geq 0$  のとき  $a_1, a_2, \dots$  の無限和の値として  $\infty$  も許せば, その無限和は常に値を持っているとみなせる. 非負の実数の無限和を扱うときには収束先として  $\infty$  も許したくなる場合が多いので, その場合には収束先が有限の値になるか否かを明瞭に述べておいた方が誤解が少なくなる.  $\square$

**問題:**  $\sum_{m=1}^{\infty} a_m$  と  $\sum_{n=1}^{\infty} b_n$  が絶対収束しているならば,  $\sum_{m,n \geq 1, m+n \leq N+1} |a_m b_n|$  も  $N \rightarrow \infty$  で収束しており,

$$\lim_{N \rightarrow \infty} \sum_{m,n \geq 1, m+n \leq N+1} a_m b_n = \sum_{m=1}^{\infty} a_m \sum_{n=1}^{\infty} b_n \quad (*)$$

が成立していることを示せ.

**解答例:**  $L = \sum_{m=1}^{\infty} |a_m|, M = \sum_{n=1}^{\infty} |b_n|$  とおく. このとき,

$$\sum_{m,n \geq 1, m+n \leq N+1} |a_m b_n| \leq \sum_{m=1}^N |a_m| \sum_{n=1}^N |b_n| \leq LM < \infty.$$

これより,  $\sum_{m,n \geq 1, m+n \leq N+1} |a_m b_n|$  も  $N \rightarrow \infty$  で収束することがわかる. これは  $\lim_{N \rightarrow \infty} \sum_{m,n \geq 1, m+n \leq N+1} a_m b_n$  が絶対収束することを意味している. したがって, 上の方の定理の(2)より(\*)が成立することもわかる.  $\square$

**注意:** 条件収束級数の収束先は足し上げる順序によって変わるので扱いには注意が必要になる.  $\square$

**問題:** 実数  $a_n$  の級数  $\sum_{n=1}^{\infty} a_n$  は条件収束していると仮定する. このとき, 任意の実数  $\alpha$  に対して, その級数の足し上げる順序をうまく変えることによって, 収束先を  $\alpha$  にできることを示せ.

**解答例:**  $a_n$  うち非負のものを  $b_1 \geq b_2 \geq \dots$  と書き, 負のものを  $c_1 \geq c_2 \geq \dots$  と書く. このとき,  $b_n \rightarrow 0, c_n \rightarrow 0$ ,  $\sum_{n=1}^{\infty} b_n = \infty$  かつ  $\sum_{n=1}^{\infty} c_n = -\infty$  が成立している. まず,  $b_1 + \dots + b_k$  が最初に  $\alpha$  以上になる  $k$  を  $k_1$  と定める. 次に  $b_1 + \dots + b_{k_1} + c_1 + \dots + c_l$  が最初に  $\alpha$  以下になる  $l$  を  $l_1$  と定める. これを同様に繰り返すと, 級数  $\sum_{n=1}^{\infty} a_n$  の項の順序を並び換えた級数

$$b_1 + \dots + b_{k_1} + c_1 + \dots + c_{l_1} + b_{k_1+1} + \dots + b_{k_2} + c_{l_1+1} + \dots + c_{l_2} + \dots$$

がで上がる.  $b_{k_i}$  までの和は和の途中段階も含めてすべて  $\alpha + b_{k_i}$  未満であり,  $c_{l_j}$  までの和は和の途中段階も含めてすべて  $\alpha + c_{l_j}$  より大きくなる ( $c_l < 0$  に注意せよ). したがって,  $b_n \rightarrow 0$  と  $c_n \rightarrow 0$  より, 上のように並び換えた級数が  $\alpha$  に収束することがわかる.  $\square$

**注意:** 上の解答例の最後の部分の議論は  $a_n \geq 0, a_n \geq a_{n+1}, a_n \rightarrow 0$  のとき, 交代級数  $\sum_{n=1}^{\infty} (-1)^{n-1} a_n$  が常に収束することを示すためにも使える.  $\square$

## 1.2 絶対収束級数の例

### 1.2.1 絶対収束級数の例1 (Riemannのゼータ函数)

$s > 1$  と仮定する. このとき  $\sum_{n=1}^{\infty} \frac{1}{n^s} < \infty$  である. なぜならば  $n \rightarrow \infty$  のとき

$$\sum_{k=1}^n \frac{1}{k^s} \leq 1 + \int_1^n \frac{dx}{x^s} = 1 + \frac{1 - n^{1-s}}{s-1} \rightarrow 1 + \frac{1}{s-1}.$$

これより  $s > 1$  のとき

$$\sum_{n=1}^{\infty} \frac{1}{n^s} \leq 1 + \frac{1}{s-1}$$

となることがわかった.  $s > 1$  の函数  $\zeta(s)$  を

$$\zeta(s) = \sum_{n=1}^{\infty} \frac{1}{n^s}$$

と定める. これを**Riemannのゼータ函数**と呼ぶ. Riemannのゼータ函数の定義域は  $s \neq 1$  の複素数全体に自然に拡張される(解析接続される). Riemannのゼータ函数は整数論における最も基本的で重要な函数である.

$s > 1$  のとき,  $\sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n^s}$  は絶対収束級数である. この級数は後で示すように  $s = 1$  でも収束している. (さらに複素平面全体に解析接続されることも示せる.)

**問題:**  $s > 1$  と仮定する. この仮定のもとで次の公式を証明せよ:

$$(1 - 2^{1-s})\zeta(s) = \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n^s}.$$

**証明:** 絶対収束級数を扱っているので, 以下のように形式的に計算した結果がそのまま証明になる.

$$\begin{aligned} (1 - 2^{1-s})\zeta(s) &= \left(1 - \frac{2}{2^s}\right) \sum_{n=1}^{\infty} \frac{1}{n^s} = \sum_{n=1}^{\infty} \frac{1}{n^s} - \sum_{n=1}^{\infty} \frac{2}{(2n)^s} \\ &= \left(\frac{1}{1^s} + \frac{1}{2^s} + \frac{1}{3^s} + \frac{1}{4^s} + \dots\right) - \left(\frac{2}{2^s} + \frac{2}{4^s} + \dots\right) \\ &= \frac{1}{1^s} - \frac{1}{2^s} + \frac{1}{3^s} - \frac{1}{4^s} + \dots = \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n^s}. \quad \square \end{aligned}$$

**問題:**  $\zeta(2)$  を数値計算して,  $\pi^2/6$  に近いことを確認せよ.

```
In [2]: 1 N = 10^8
2 SimpleSum = sum(n->1/n^2, 1:N)
3 TrueValue = pi^2/6
4 @show SimpleSum
5 @show TrueValue
6 @show SimpleSum - TrueValue;
```

```
SimpleSum = 1.6449340568482278
TrueValue = 1.6449340668482264
SimpleSum - TrueValue = -9.99999860695766e-9
```

**問題:**  $\zeta(2) = \pi^2/6$  の証明について調べよ.

**ヒント:** この問題は**Basel問題**などと呼ばれているようである. インターネットで検索してみよ.  $\square$

正の偶数  $2m$  について  $\zeta(2m)$  はすべて計算されている. しかし, 3以上の奇数  $2m+1$  に対する  $\zeta(2m+1)$  の数学的な素性はまだよくわかっていない.

### 1.2.2 絶対収束級数の例2

**問題:**  $\sum_{n=2}^{\infty} (\zeta(n) - 1) = 1$  を示せ.

**証明:**  $k \geq 2$  のとき

$$\sum_{n=2}^{\infty} \frac{1}{k^n} = \frac{1/k^2}{1 - 1/k} = \frac{1}{k(k-1)} = \frac{1}{k-1} - \frac{1}{k}.$$

そして

$$\zeta(n) - 1 = \frac{1}{2^n} + \frac{1}{3^n} + \frac{1}{4^n} + \cdots = \sum_{k=2}^{\infty} \frac{1}{k^n}$$

なので,

$$\begin{aligned} \sum_{n=2}^{\infty} (\zeta(n) - 1) &= \sum_{n=2}^{\infty} \sum_{k=2}^{\infty} \frac{1}{k^n} = \sum_{k=2}^{\infty} \sum_{n=2}^{\infty} \frac{1}{k^n} = \sum_{k=2}^{\infty} \left( \frac{1}{k-1} - \frac{1}{k} \right) \\ &= \left( \frac{1}{1} - \frac{1}{2} \right) + \left( \frac{1}{2} - \frac{1}{3} \right) + \left( \frac{1}{3} - \frac{1}{4} \right) + \cdots = 1. \end{aligned}$$

2つ目の等号で  $1/k^n \geq 0$  なので足し上げる順序を変えても総和は等しいことを使った.  $\square$

以下はコンピューターによる計算.

```
In [3]: 1 N = 50
2 sum(n->zeta(n)-1, 2:N)
```

```
Out[3]: 1.00000000000000044
```

```
In [4]: 1 l, n = symbols("l n", integer=true, positive=true) # l = k-1
2 s = simplify(doit(Sum(1/(l+1)^n, (n,2,oo))))
3 S = doit(Sum(s, (l,1,oo)))
```

```
Out[4]: 1
```

**問題:** 上の問題の結果より,  $\sum_{n=2}^{\infty} (-1)^n (\zeta(n) - 1)$  は絶対収束する. その収束先を求めよ.

**証明:**  $k \geq 2$  のとき

$$\sum_{n=2}^{\infty} \frac{(-1)^n}{k^n} = \frac{1/k^2}{1+1/k} = \frac{1}{k(k+1)} = \frac{1}{k} - \frac{1}{k+1}.$$

そして

$$(-1)^n(\zeta(n) - 1) = \frac{(-1)^n}{2^n} + \frac{(-1)^n}{3^n} + \frac{(-1)^n}{4^n} + \cdots = \sum_{k=2}^{\infty} \frac{(-1)^n}{k^n}$$

なので,

$$\begin{aligned} \sum_{n=2}^{\infty} (\zeta(n) - 1) &= \sum_{n=2}^{\infty} \sum_{k=2}^{\infty} \frac{(-1)^n}{k^n} = \sum_{k=2}^{\infty} \sum_{n=2}^{\infty} \frac{(-1)^n}{k^n} = \sum_{k=2}^{\infty} \left( \frac{1}{k} - \frac{1}{k+1} \right) \\ &= \left( \frac{1}{2} - \frac{1}{3} \right) + \left( \frac{1}{3} - \frac{1}{4} \right) + \left( \frac{1}{4} - \frac{1}{5} \right) + \cdots = \frac{1}{2}. \end{aligned}$$

2つ目の等号で絶対収束級数なので足し上げる順序を変えても総和は等しいことを使った。□

以下はコンピューターによる計算.

```
In [5]: 1 N = 50
        2 sum(n->(-1)^n*(zeta(n)-1), 2:N)
```

```
Out[5]: 0.50000000000000018
```

```
In [6]: 1 l, n = symbols("l n", integer=True, positive=True) # l = k-1
        2 s = simplify(doit(Sum((-1)^n/(l+1)^n, (n,2,oo))))
        3 S = doit(Sum(s, (l,1,oo)))
```

```
Out[6]: 1/2
```

問題: 以上の例以外の絶対収束級数の例について調べ, ノートにまとめよ.

### 1.3 条件収束級数の例

**定理:** 一般に  $a_1 \geq a_2 \geq a_3 \geq \cdots$  かつ  $a_n \geq 0$  かつ  $a_n \rightarrow 0$  のとき, **交代級数**

$$\sum_{n=1}^{\infty} (-1)^{n-1} a_n = a_1 - a_2 + a_3 - a_4 + \cdots$$

は常に収束する.

**注意:** この定理中の  $a_n$  について  $\sum_{n=1}^{\infty} a_n = \infty$  が成立しているとき, 交代級数は条件収束級数になる.

**証明:**  $c_n = \sum_{k=1}^n (-1)^{k-1}$  とおくと,  $n$  が奇数のとき  $c_n = 1$  となり,  $n$  が偶数のとき  $c_n = 0$  となる.  $c_n - c_{n-1} = (-1)^{n-1}$  なので,

$$\sum_{k=1}^n (-1)^{k-1} a_k = \sum_{k=1}^n (c_k - c_{k-1}) a_k = \sum_{k=1}^n c_k a_k - \sum_{k=0}^{n-1} c_k a_{k+1} = c_n a_n - c_0 a_1 + \sum_{k=1}^{n-1} c_k (a_k - a_{k+1})$$

であり,  $|c_k| \leq 1$  かつ  $a_n$  が単調減少することより  $a_k - a_{k+1} \geq 0$  であり,  $a_n \rightarrow 0$  より  $a_n \geq 0$  なので,

$$\sum_{k=1}^{n-1} |c_k| (a_k - a_{k+1}) \leq \sum_{k=1}^{n-1} (a_k - a_{k+1}) = a_1 - a_n \leq a_1.$$

ゆえに  $\sum_{k=1}^{\infty} c_k (a_k - a_{k+1})$  は絶対収束する. そして,  $a_n \rightarrow 0$  かつ  $|c_n| \leq 1$  より  $c_n a_n \rightarrow 0$  となるので,  $\sum_{k=1}^n (-1)^{k-1} a_k$  が  $n \rightarrow \infty$  で収束することがわかる. □

**注意:** 上の証明は部分積分と類似の計算を和の場合に行っている. 部分積分やそれと同様の和の変形は収束し難い積分や無限和を収束し易くするために役に立つ. □

注意: 上の証明の議論は  $c_n$  が有界で  $a_n$  が単調減少しながら 0 に収束する場合に一般化される.  $\square$

### 1.3.1 条件収束級数の例1

$\sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n}$  は収束するが,  $\sum_{n=1}^{\infty} \frac{1}{n} = \infty$  となる.

なぜならば,  $\sum_{k=1}^n \frac{1}{k} \geq \int_1^{n+1} \frac{dx}{x} = \log(n+1) \rightarrow \infty (n \rightarrow \infty)$  が成立するからである.

問題:  $\sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n}$  を求めよ.

解答:  $\log 2$ .

証明: 一般に

$$\int_1^x \frac{dt}{t} = \log x.$$

ゆえに,

$$\int_0^1 \frac{dx}{1+x} = \int_1^2 \frac{dy}{y} = \log 2.$$

一方,  $|x| < 1$  のとき

$$\frac{1}{1+x} = 1 - x + x^2 - x^3 + \dots$$

なので

$$\int_0^1 \frac{dx}{1+x} = \int_0^1 (1 - x + x^2 - x^3 + \dots) = 1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \dots = \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n}.$$

したがって,

$$\int_0^1 \frac{dx}{1+x} = \log 2 = \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n}. \quad \square$$

以下はコンピューターによる計算.

```
In [7]: 1 n = symbols("n", integer=True)
        2 S = doit(Sum((-1)^(n-1)/n, (n, 1, oo)))
```

Out[7]:  $\log(2)$

```
In [8]: 1 N = 1000
        2 SimpleSum = sum(n->(-1)^(n-1)/n, 1:N)
        3 TrueValue = log(2)
        4 @show SimpleSum
        5 @show TrueValue
        6 @show SimpleSum - TrueValue;
```

```
SimpleSum = 0.6926474305598223
TrueValue = 0.6931471805599453
SimpleSum - TrueValue = -0.0004997500001230337
```

### 1.3.2 Euler変換による交代級数の数値計算

$\sum_{n=1}^N \frac{(-1)^{n-1}}{n}$  で  $N = 1000$  としても収束先の値を小数点以下第2桁までしか再現してくれない. 一般に条件収束する交代級数を定義通り足し上げると収束は遅い.

収束を加速するための様々な方法が知られているが, 交代級数の場合にはEuler変換が有名である. 詳しくは



- 交代ゼータ関数のオイラー変換 (<http://nbviewer.jupyter.org/gist/genkuroki/4a8fea5e2ed1b4e3b737c99acf237042>)

を参照せよ. 交代級数  $\sum_{k=0}^{\infty} (-1)^k a_k$  のEuler変換の結果は

$$\lim_{L \rightarrow \infty} \sum_{k=0}^{L-1} w_k^{(L)} (-1)^k a_k, \quad w_k^{(L)} = \sum_{n=k}^{L-1} 2^{-n-1} \binom{n}{k}.$$

すなわち交代級数のEuler変換は交代級数の各項にウェイト  $w_k^{(L)}$  をかけてから  $L \rightarrow \infty$  の極限を取る形になる.

以下のようにEuler変換を使えば数十項で精密な計算が可能になる.

```
In [9]: 1 binom(n,k) = exp(lgamma(n+1) - lgamma(k+1) - lgamma(n-k+1))
2
3 struct EulerWeight{T} W::T end
4 ▼ function EulerWeight(; Lmax=2^6)
5     W = zeros(Lmax,Lmax)
6 ▼     for L in 1:Lmax
7 ▼         for k in 0:L-1
8             W[L,k+1] = sum(n->2.0^(-n-1)*binom(n,k), k:L-1)
9         end
10     end
11     EulerWeight(W)
12 end
13 (eulerweight::EulerWeight)(L,k) = eulerweight.W[L,k+1]
14 eulerweight = EulerWeight()
```

```
Out[9]: EulerWeight{Array{Float64,2}}{[0.5 0.0 ... 0.0 0.0; 0.75 0.25 ... 0.0 0.0; ... ; 1.0 1.0 ... 1.0842e-19 0.0; 1.0 1.0 ... 3.52366e-18 5.42101e-20]}
```

```
In [10]: 1 L = 2^6
2 EulerTran = sum(k->eulerweight(L,k)*(-1)^k/(k+1), 0:L-1)
3 TrueValue = log(2)
4 @show EulerTran
5 @show TrueValue
6 @show EulerTran - TrueValue
```

```
EulerTran = 0.6931471805599451
TrueValue = 0.6931471805599453
EulerTran - TrueValue = -2.220446049250313e-16
```

```
Out[10]: -2.220446049250313e-16
```

Euler変換を使うと, 次のセルのように  $\zeta(s)$  の非自明な零点をプロットすることができる.

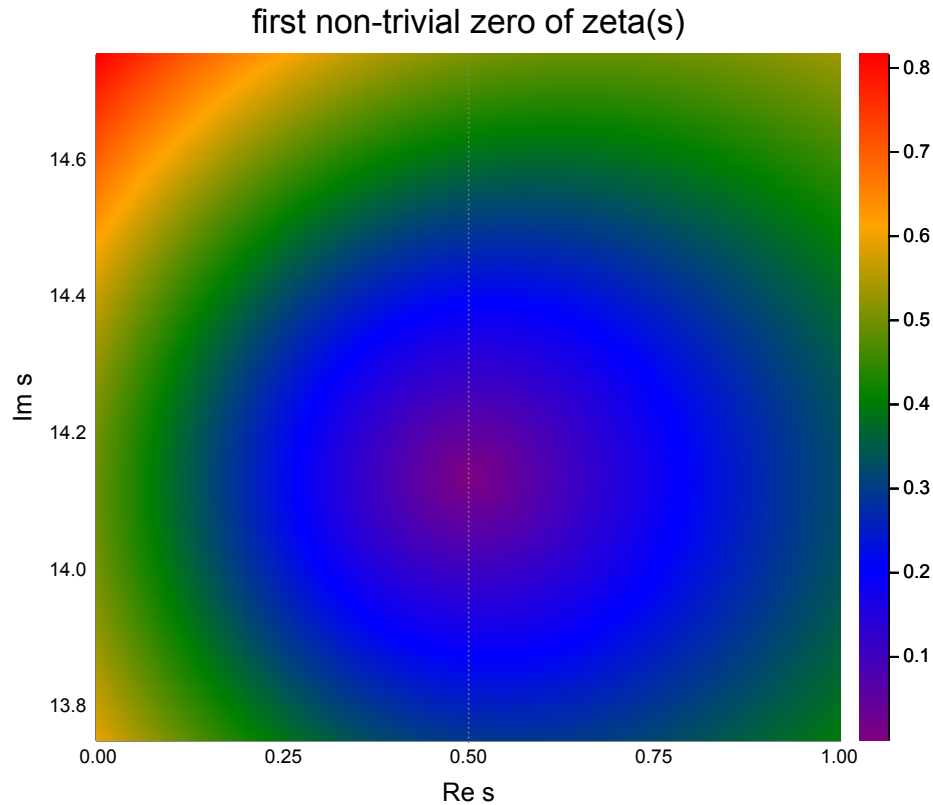
```

In [11]: 1 f(s) = sum(k -> eulerweight(L,k)*(-1)^k/(1+k)^s, 0:L-1)/(1-2^(1-s))
2 x = 0.0:0.005:1.0
3 y = 13.75:0.005:14.75
4 z = x' .+ im.*y
5 @time w = f.(z)
6 plot(size=(600, 500), title="first non-trivial zero of zeta(s)")
7 heatmap!(x, y, abs.(w), color=:rainbow, xlabel="Re s", ylabel="Im s")
8 vline!([0.5], color="gray", ls=:dot, label="")

```

2.117314 seconds (13.50 M allocations: 339.953 MiB, 6.87% gc time)

Out[11]:



複素平面における実部が 0 から 1 の縦に無限に長い帯状の領域を  $\zeta(s)$  の critical strip と呼ぶ.  $\zeta(s)$  の critical strip における零点を  $\zeta(s)$  の非自明な零点と呼ぶ.

**Riemann予想:**  $\zeta(s)$  の非自明な零点の実部はすべて  $1/2$  である.  $\square$

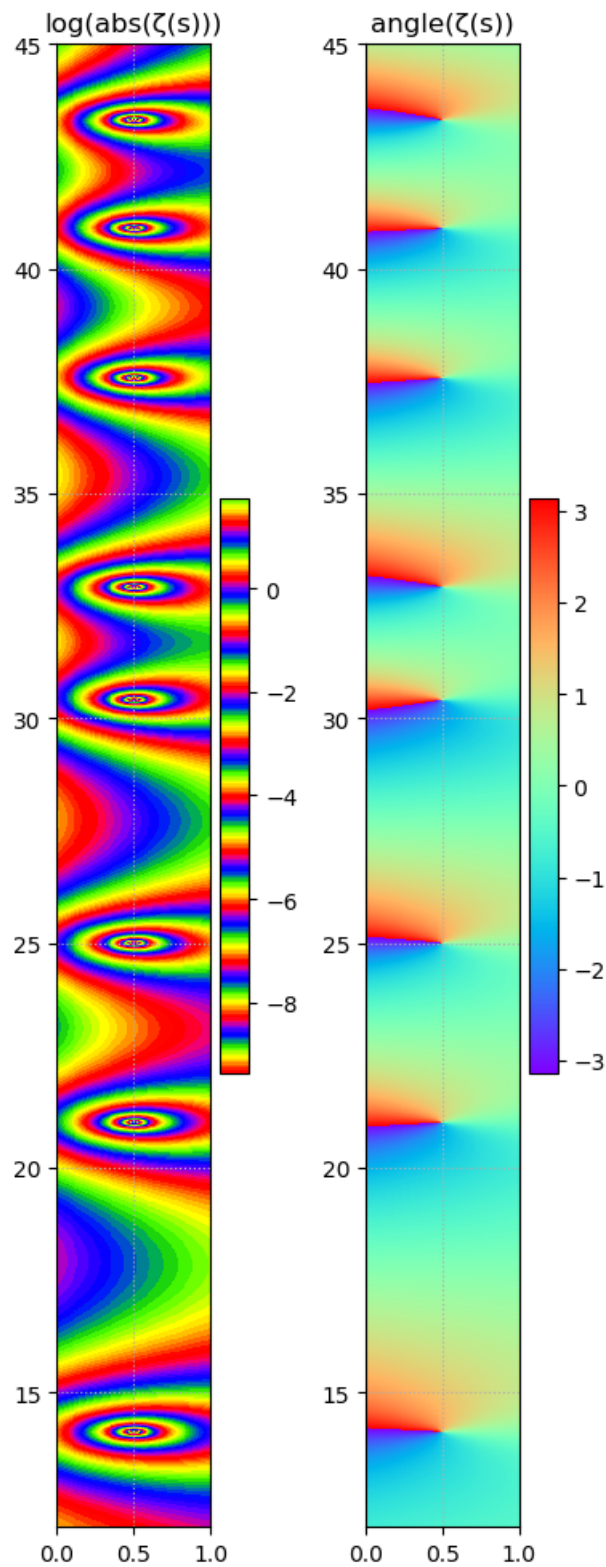
この予想は非常に有名であり, もしも証明されれば素数の分布に関する精密な評価が得られる.

**問題:**  $\zeta(s)$  の critical strip における様子をプロットせよ.  $\square$

次のセルを見よ. 虚部が12から45までのcritical stripでの  $\zeta(s)$  の様子がプロットされている. 少なくともその範囲内で  $\zeta(s)$  の非自明な零点の実部はどれも  $1/2$  になっていそうなのがわかる.

In [12]:

```
1  x = 0:0.01:1
2  y = 12:0.005:45
3  s = x' .+ y*im
4  @time w = zeta.(s)
5
6  PyPlot.figure(figsize=(4,10))
7
8  PyPlot.subplot(121)
9  PyPlot.pcolormesh(x, y, log.(abs.(w)), cmap="prism")
10 PyPlot.colorbar()
11 PyPlot.grid(ls=":")
12 PyPlot.title("log(abs( $\zeta(s)$ ))")
13
14 PyPlot.subplot(122)
15 PyPlot.pcolormesh(x, y, angle.(w), cmap="rainbow")
16 PyPlot.colorbar()
17 PyPlot.grid(ls=":")
18 PyPlot.title("angle( $\zeta(s)$ )")
19
20 PyPlot.tight_layout()
```



3.747516 seconds (152.42 k allocations: 16.502 MiB, 0.20% gc time)

### 1.3.3 無限和と積分の交換に関する注意

上では  $\int_0^1 x^n dx = 1/(n+1)$  となることと、無限和  $1 - x + x^2 - x^3 + \dots$  と積分  $\int_0^1 dx$  の順序を交換できることを用いた。

無限和と積分の順序交換が可能なのは以下のようにして確認できる。まず  $|x| < 1$  で

$$\frac{1}{1+x} = 1 - x + x^2 - x^3 + \dots + (-1)^{N-1} x^{N-1} + (-1)^N \frac{x^N}{1+x}$$

となることを確認せよ。この公式を使うと、

$$\int_0^1 \frac{dx}{1+x} = 1 - \frac{1}{2} + \frac{1}{3} + (-1)^{N-1} \frac{1}{N} + (-1)^N \int_0^1 \frac{x^N}{1+x} dx$$

さらに  $0 \leq x \leq 1$  のとき、 $1 \leq 1+x$  なので

$$0 \leq \frac{x^N}{1+x} \leq x^N$$

となるから,

$$0 \leq \int_0^1 \frac{x^N}{1+x} dx \leq \int_0^1 x^N dx = \frac{1}{N+1}$$

なので,  $N \rightarrow \infty$  で

$$\int_0^1 \frac{x^N}{1+x} dx \rightarrow 0.$$

以上を合わせると,

$$\int_0^1 \frac{dx}{1+x} = \lim_{N \rightarrow \infty} \sum_{n=1}^N \frac{(-1)^{n-1}}{n} = \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n}$$

となることがわかる.

**注意:** 上の計算における  $(-1)^N \int_0^1 \frac{x^N}{1+x} dx$  のように, 極限で 0 になって欲しい項が積分表示を持つことは多い. 極限で 0 になって欲しい部分を「+ ...」のように書くと正体不明で不安になるのだが, 積分表示を持つ場合にはそのような不安は完全に解消してしまう.  $\square$

### 1.3.4 条件収束級数の例2

**問題:**  $\sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{2n-1}$  を求めよ.

**答え:**  $\pi/4$ .

**証明:**  $a = \tan \theta$  の  $-\pi/2 < \theta < \pi/2$  における逆関数を  $\theta = \arctan a$  と書くと,

$$\int_0^a \frac{dt}{1+t^2} = \arctan a.$$

この公式は左辺で  $t = \tan \theta$  と置換すれば確認できる.

ゆえに,

$$\int_0^1 \frac{dx}{1+x^2} = \arctan 1 = \frac{\pi}{4}$$

一方,  $|x| < 1$  のとき

$$\frac{1}{1+x^2} = 1 - x^2 + x^4 - x^6 + \dots$$

なので

$$\int_0^1 \frac{dx}{1+x^2} = \int_0^1 (1 - x^2 + x^4 - x^6 + \dots) = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots = \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{2n-1}.$$

したがって,

$$\int_0^1 \frac{dx}{1+x^2} = \arctan 1 = \frac{\pi}{4} = \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{2n-1}. \quad \square$$

以下はコンピューターによる計算.

```
In [13]: 1 n = symbols("n", integer=True)
          2 S = doit(Sum((-1)^(n-1)/(2n-1), (n, 1, oo)))
```

```
Out[13]:  $\frac{\pi}{4}$ 
```

```
In [14]: 1 N = 10^8
2 SimpleSum = sum(n->(-1)^(n-1)/(2n-1), 1:N)
3 TrueValue = pi/4
4 @show SimpleSum
5 @show TrueValue
6 @show SimpleSum - TrueValue;
```

```
SimpleSum = 0.7853981608974486
TrueValue = 0.7853981633974483
SimpleSum - TrueValue = -2.499999651739415e-9
```

```
In [15]: 1 L = 2^6
2 EulerTran = sum(k->eulerweight(L,k)*(-1)^k/(2k+1), 0:L-1)
3 TrueValue = pi/4
4 @show EulerTran
5 @show TrueValue
6 @show EulerTran - TrueValue
```

```
EulerTran = 0.7853981633974486
TrueValue = 0.7853981633974483
EulerTran - TrueValue = 3.3306690738754696e-16
```

```
Out[15]: 3.3306690738754696e-16
```

**問題:** 以上の計算における無限和と積分の正当化を自分で行ってみよ. ☐

**問題:** 以上の例以外の条件収束級数の例について調べ, ノートにまとめよ. ☐

以下はコンピューターによる  $\int_0^1 \frac{dx}{1+x^3}$  や  $\int_0^1 \frac{dx}{1+x^4}$  などの計算.

```
In [16]: 1 x = symbols("x", real=true, positive=true)
2 integrate(1/(1+x^3), (x,0,1))
```

```
Out[16]:  $\frac{1}{3}\log(2) + \frac{\sqrt{3}\pi}{9}$ 
```

```
In [17]: 1 n = symbols("n", integer=true, positive=true)
2 S = doit(Sum((-1)^(n-1)/(3n-2), (n,1,oo)))
```

```
Out[17]:  $\frac{1}{3}\log(2) - \frac{1}{3}e^{-\frac{5\pi}{3}i}\log\left(1 + e^{\frac{2\pi}{3}i}\right) - \frac{1}{3}e^{-\frac{i\pi}{3}}\log\left(1 + e^{-\frac{2\pi}{3}i}\right)$ 
```

```
In [18]: 1 x = symbols("x", real=true, positive=true)
2 integrate(x/(1+x^3), (x,0,1))
```

```
Out[18]:  $-\frac{1}{3}\log(2) + \frac{\sqrt{3}\pi}{9}$ 
```

```
In [19]: 1 n = symbols("n", integer=true, positive=true)
2 S = doit(Sum((-1)^(n-1)/(3n-1), (n,1,oo)))
```

```
Out[19]:  $-\frac{1}{3}\log(2) - \frac{1}{3}e^{-\frac{2\pi}{3}i}\log\left(1 + e^{-\frac{2\pi}{3}i}\right) - \frac{1}{3}e^{-\frac{10\pi}{3}i}\log\left(1 + e^{\frac{2\pi}{3}i}\right)$ 
```

```
In [20]: 1 x = symbols("x", real=true, positive=true)
2 integrate(1/(1+x^4), (x,0,1))
```

```
Out[20]:  $-\frac{\sqrt{2}}{8}\log(-\sqrt{2}+2) + \frac{\sqrt{2}}{8}\log(\sqrt{2}+2) + \frac{\sqrt{2}\pi}{8}$ 
```

```
In [21]: 1 n = symbols("n", integer=True, positive=True)
2 s = doit(Sum((-1)^(n-1)/(4n-3), (n,1,oo)))
```

```
Out[21]: 
$$-\frac{i}{4}e^{-\frac{i\pi}{4}}\log\left(1+e^{\frac{3\pi i}{4}}\right)+\frac{1}{4}e^{-\frac{i\pi}{4}}\log\left(1+e^{\frac{i\pi}{4}}\right)+\frac{i}{4}e^{-\frac{i\pi}{4}}\log\left(1+e^{-\frac{i\pi}{4}}\right)-\frac{1}{4}e^{-\frac{i\pi}{4}}\log\left(1+e^{-\frac{3\pi i}{4}}\right)$$

```

```
In [22]: 1 x = symbols("x", real=True, positive=True)
2 s = symbols("s", real=True, positive=True)
3 J = simplify(integrate(1/(1+x^(1/s)), (x,0,1)))
```

```
Out[22]:  $s\Phi\left(e^{i\pi}, 1, s\right)$ 
```

## 2 複素数の級数

この節では複素数  $a_n$  の級数  $\sum_{n=1}^{\infty} a_n$  を扱う。

### 2.1 複素数の絶対収束級数

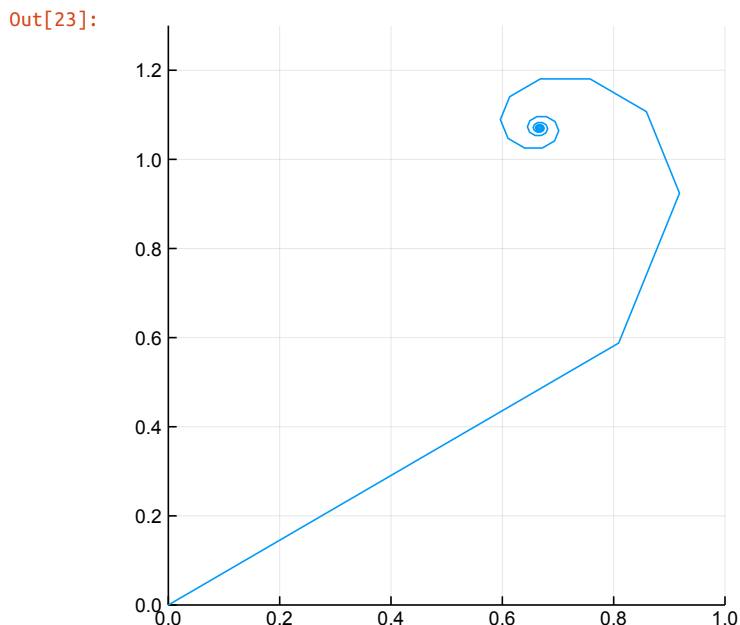
**定理:** 複素数  $a_1, a_2, \dots$  について, 級数  $\sum_{n=1}^{\infty} |a_n|$  が有限の値に収束するならば, 級数  $\sum_{n=1}^{\infty} a_n$  も収束する. このとき, 級数  $\sum_{n=1}^{\infty} a_n$  は **絶対収束**するという.  $\square$

絶対収束していないが収束する級数は**条件収束**するという。

上の定理の証明は略す. しかし, 下の方で, 級数  $\sum_{n=1}^{\infty} |a_n|$  が有限の値に収束するならば, 級数  $\sum_{n=1}^{\infty} a_n$  も収束する直観的な理由を説明する。

**例:**  $\theta$  は実数であるとする. このとき, 複素級数  $\sum_{n=1}^{\infty} \frac{e^{in\theta}}{n^{3/2}}$  は絶対収束する. 以下のセルにおける計算とプロットを参照せよ。

```
In [23]: 1 theta = 2pi/10
2 a(n) = exp(im*n*theta)/n^1.5
3 sum_a(n) = iszero(n) ? Complex(0,0) : sum(k->a(k), 1:n)
4 N = 70
5 n = 0:N
6 z = sum_a.(n)
7 plot(size=(400, 400), xlims=(0,1.0), ylim=(0,1.3))
8 plot!(real.(z), imag.(z), legend=false)#, line=:arrow)
```



In [24]:

```

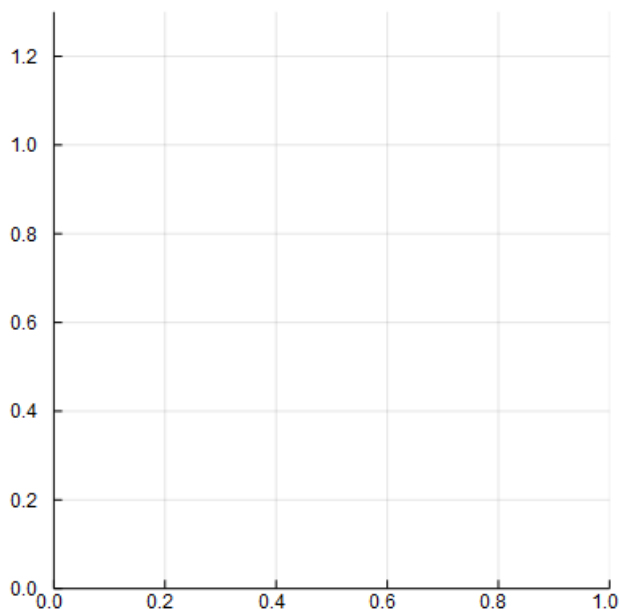
1  θ = 2π/10
2  a(n) = exp(im*n*θ)/n^1.5
3  sum_a(n) = iszero(n) ? Complex{0.0} : sum(k->a(k), 1:n)
4  ▼ @time anim = @animate for N in 0:70
5      n = 0:N
6      z = sum_a.(n)
7      plot(size=(400, 400), xlims=(0,1.0), ylim=(0,1.3))
8      plot!(real.(z), imag.(z), legend=false, line=:arrow)
9  end
10
11 gifname = "sum_a.gif"
12 @time gif(anim, gifname; fps=10)
13 sleep(0.1)
14 showing("image/gif", gifname)

```

2.816637 seconds (19.47 M allocations: 502.176 MiB, 3.88% gc time)

0.831693 seconds (91.73 k allocations: 4.913 MiB, 0.69% gc time)

INFO: Saved animation to C:\Users\genkuroki\OneDrive\Calculus\sum\_a.gif

**絶対収束の概念の直観的な説明:**

各複素数  $a_n$  は平面上の点と同一視できる(複素平面の考え方).

平面上の点と原点からその点への矢線(ベクトル)は同一視できる(ベクトルの考え方).

複素数の和はベクトルの和に対応している.

ベクトルの和は矢線としてのベクトルを連結することを意味する(ベクトルの和の幾何学的意味).

ゆえに,  $s_N = \sum_{n=1}^N a_n$  は各  $a_1, a_2, \dots, a_N$  の各々に対応する矢線を連結して得られる折れ線の先端の点に対応する複素数に等しい.

$L = \sum_{n=1}^{\infty} |a_n|$  は各  $a_n$  に対応する矢線の長さの総和を意味する.

ゆえに,  $L$  が有限の値になることは, 各  $a_n$  に対応する矢線をすべて連結して得られる折れ線の長さが有限になることを意味する.

そのとき,  $a_n$  に対応する矢線をすべて連結して得られる折れ線の「先端」の点に対応する複素数が級数  $\sum_{n=1}^{\infty} a_n$  の収束先になる.

1つ前のセルのアニメーションはそのような様子の一例になっている.

**注意:** 上の説明を理解できれば, 級数  $\sum_{n=1}^{\infty} |a_n|$  が有限の値に収束するならば, 級数  $\sum_{n=1}^{\infty} a_n$  も収束することも十分に納得できと思う. その直観は数学的には**完備性**(<https://www.google.co.jp/search?>



q=%E5%AE%8C%E5%82%99%E6%80%A7+%E7%B5%B6%E5%AF%BE%E5%8F%8E%E6%9D%9F)の概念によって正当化される.  $\square$

## 2.2 条件収束する複素級数の例

例:  $\theta$  は実数であり,  $e^{i\theta} \neq 1$  であると仮定する. このとき  $\sum_{n=1}^{\infty} \frac{e^{in\theta}}{n}$  は条件収束する.  $\square$

例の解説のための一般論: 一般に  $S_n = \sum_{k=1}^n a_k b_k$  のとき,  $A_k = \sum_{j=1}^k a_j$ ,  $A_0 = 0$  とおくと,  $a_k = A_k - A_{k-1}$  なので

$$\begin{aligned} S_n &= \sum_{k=1}^n (A_k - A_{k-1}) b_k = \sum_{k=1}^n A_k b_k - \sum_{k=0}^{n-1} A_k b_{k+1} \\ &= A_n b_n - A_0 b_1 + \sum_{k=1}^{n-1} A_k (b_k - b_{k+1}). \end{aligned}$$

この公式は部分積分の公式の和に関するバージョンであることに注意せよ. (「部分積分」のアイデアは普遍的によく現われる.)

仮定の追加1: 数列  $b_k$  は単調減少実数列でかつ 0 に収束すると仮定する.

このとき,  $b_k - b_{k+1} \geq 0$  かつ

$$\sum_{k=1}^{n-1} (b_k - b_{k+1}) = b_1 - b_n \rightarrow b_1 \quad (n \rightarrow \infty).$$

仮定の追加2:  $A_k$  は有界数列であると仮定する. すなわちある正の実数で  $|A_k| \leq M$  ( $k = 1, 2, \dots$ ) を満たすものが存在すると仮定する.

このとき,

$$\sum_{k=1}^{\infty} |A_k| |b_k - b_{k+1}| \leq M \sum_{k=1}^{\infty} (b_k - b_{k+1}) = M b_1 < \infty$$

なので級数  $\sum_{k=1}^{\infty} A_k (b_{k+1} - b_k)$  は絶対収束する. さらに,  $n \rightarrow \infty$  で  $b_n \rightarrow 0$  なので  $A_n b_n \rightarrow 0$  となる. ゆえに級数  $\sum_{k=1}^{\infty} a_k b_k$  は収束する. (積分の場合にも, 絶対収束しない積分を部分積分によって絶対収束する積分に変形できる場合がある. 各自自分で研究してみよ.)  $\square$

例の解説: 以上の一般論を上例に適用しよう. すなわち  $a_n = e^{in\theta}$ ,  $b_n = 1/n$  の場合を考える. このとき

$$A_k = \sum_{j=1}^k a_j = \frac{e^{i\theta} - e^{i(n+1)\theta}}{1 - e^{i\theta}}$$

であるから,  $|A_k| \leq 2/|1 - e^{i\theta}|$  なので,  $A_k$  は有界である. 実数列  $b_k$  は単調減少しながら 0 に収束する. したがって上の一般論より, 級数  $\sum_{n=1}^{\infty} \frac{e^{in\theta}}{n}$  は条件収束する.  $\square$

注意:  $-\log(1 - z)$  の収束半径 1 のMaclaurin展開は

$$-\log(1 - z) = \sum_{n=1}^{\infty} \frac{z^n}{n}$$

の形をしている.  $\theta \in \mathbb{R}$ ,  $e^{i\theta} \neq 1$  のとき,  $\sum_{n=1}^{\infty} \frac{e^{in\theta}}{n}$  の収束先は  $-\log(1 - e^{i\theta})$  になる.  $\square$

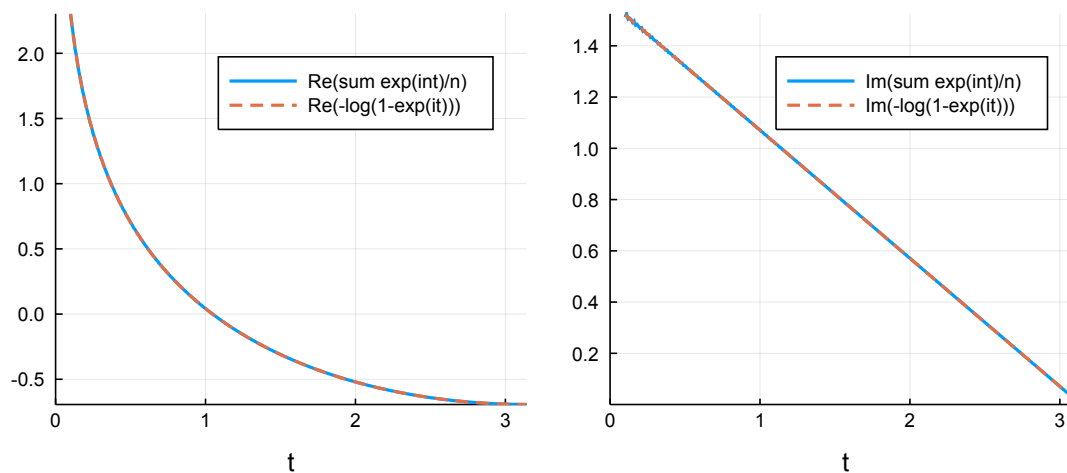
In [25]:

```

1  f(z) = -log(1-z)
2  g(z; N=1000) = sum(n->z^n/n, 1:N)
3
4  θ = 0.1:0.01:π
5  v = @.(f(exp(im*θ)))
6  w = @.(g(exp(im*θ)))
7
8  P1 = plot(xlims=(0,π), xlabel="t")
9  plot!(θ, real.(w), label="Re(sum exp(int)/n)", lw=2)
10 plot!(θ, real.(v), label="Re(-log(1-exp(it)))", lw=2, ls=:dash)
11
12 P2 = plot(xlims=(0,π), xlabel="t")
13 plot!(θ, imag.(w), label="Im(sum exp(int)/n)", lw=2)
14 plot!(θ, imag.(v), label="Im(-log(1-exp(it)))", lw=2, ls=:dash)
15
16 plot(P1, P2, size=(700,300))

```

Out[25]:



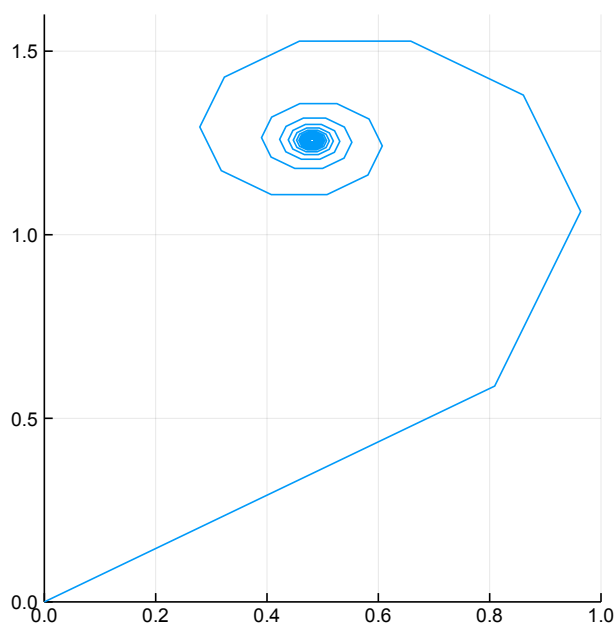
In [26]:

```

1  θ = 2π/10
2  a(n) = exp(im*n*θ)/n
3  sum_a(n) = iszero(n) ? Complex{0.0} : sum(k->a(k), 1:n)
4  N = 400
5  n = 0:N
6  z = sum_a.(n)
7  plot(size=(400, 400), xlims=(0,1.0), ylim=(0,1.6))
8  plot!(real.(z), imag.(z), legend=false)#, line=:arrow)

```

Out[26]:



In [27]:

```

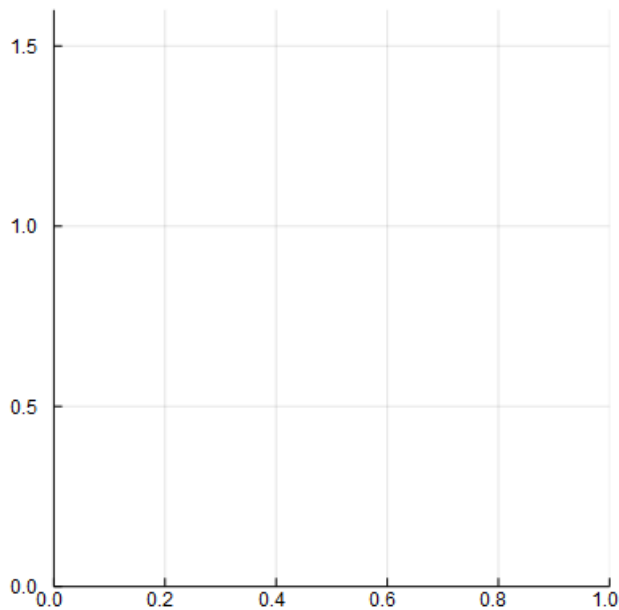
1  θ = 2π/10
2  a(n) = exp(im*n*θ)/n
3  sum_a(n) = iszero(n) ? Complex{0.0} : sum(k->a(k), 1:n)
4  ▼ @time anim = @animate for N in 0:4:400
5      n = 0:N
6      z = sum_a.(n)
7      plot(size=(400, 400), xlims=(0,1.0), ylim=(0,1.6))
8      plot!(real.(z), imag.(z), legend=false, line=:arrow)
9  end
10
11 gifname = "sum_a.gif"
12 @time gif(anim, gifname; fps=10)
13 sleep(0.1)
14 showing("image/gif", gifname)

```

4.377847 seconds (40.28 M allocations: 1.076 GiB, 8.45% gc time)

0.683254 seconds (1.05 k allocations: 69.125 KiB)

INFO: Saved animation to C:\Users\genkuroki\OneDrive\Calculus\sum\_a.gif



上のセルのアニメーションの折れ線の長さは無限大である。無限に長くなる折れ線の先端が無限に回転しながら収束している。

## 2.3 多重対数函数

複素数  $z$  に関するべき級数

$$\mathrm{Li}_r(z) = \sum_{n=1}^{\infty} \frac{z^n}{n^r}$$

は  $r > 1$ ,  $|z| \leq 1$  および  $r = 1$ ,  $|z| < 1$  のとき絶対収束する。  $r = 1$ ,  $|z| = 1$  の場合には  $z \neq 1$  なら条件収束し,  $z = 1$  ならば  $\infty$  に発散する。このように定義される函数  $\mathrm{Li}_r(z)$  を  $r$  重の**多重対数函数**(polylogarithm)と呼ぶ。  $r = 2$  の場合はdilogarithmと呼ばれ,  $r = 3$  の場合にはtrilogarithmと呼ばれる。このとき,

$$\begin{aligned} \mathrm{Li}_1(z) &= \sum_{n=1}^{\infty} \frac{z^n}{n} = -\log(1-z) = \int_0^z \frac{dt}{1-t}, \\ \mathrm{Li}_2(z) &= \sum_{n=1}^{\infty} \frac{z^n}{n^2} = \int_0^z \frac{\mathrm{Li}_1(t)}{t} dt = \int_0^z dt_2 \int_0^{t_2} dt_1 \frac{1}{(1-t_1)t_2}, \\ \mathrm{Li}_3(z) &= \sum_{n=1}^{\infty} \frac{z^n}{n^3} = \int_0^z \frac{\mathrm{Li}_2(t)}{t} dt = \int_0^z dt_3 \int_0^{t_3} dt_2 \int_0^{t_2} dt_1 \frac{1}{(1-t_1)t_2 t_3}, \\ &\dots \dots \dots \\ \mathrm{Li}_r(z) &= \sum_{n=1}^{\infty} \frac{z^n}{n^r} = \int_0^z \frac{\mathrm{Li}_{r-1}(t)}{t} dt = \int_0^z dt_r \int_0^{t_r} dt_{r-1} \dots \int_0^{t_2} dt_1 \frac{1}{(1-t_1)t_2 \dots t_r}, \end{aligned}$$

$r > 1$  のとき,

$$\text{Li}_r(1) = \sum_{n=1}^{\infty} \frac{1}{n^r} = \zeta(r).$$

したがって, 2以上の整数  $m$  に対して,

$$\zeta(m) = \text{Li}_m(1) = \int_0^1 dt_m \int_0^{t_m} dt_{m-1} \cdots \int_0^{t_2} dt_1 \frac{1}{(1-t_1)t_2 \cdots t_m}.$$

$t_1 = x_1 x_2 \cdots x_m, t_2 = x_2 \cdots x_m, \dots, t_m = x_m$  と置換すると,

$$\zeta(m) = \text{Li}_m(1) = \int_0^1 dx_m \int_0^1 dx_{m-1} \cdots \int_0^1 dx_1 \frac{1}{1-x_1 \cdots x_m}.$$

これの右辺が  $\zeta(m)$  に一致することは

$$\frac{1}{1-x_1 \cdots x_m} = \sum_{n=1}^{\infty} x_1^{n-1} \cdots x_m^{n-1},$$

$$\int_0^1 dx_m \int_0^1 dx_{m-1} \cdots \int_0^1 dx_1 x_1^n \cdots x_m^n = \left( \int_0^1 x^{n-1} dx \right)^m = \frac{1}{n^m}$$

に注意すれば明らかであろう.

## 2.4 二重対数関数の $\zeta(2)$ の近似計算への応用

$\zeta(2) = \sum_{n=1}^{\infty} \frac{1}{n^2}$  を計算せよという問題は**Basel問題**と呼ばれているらしい. Basel問題はEulerによって1743年ごろに解かれたらしい. Eulerがどのように考えたかについては次の文献を参照せよ.

- 杉本敏夫, [バーゼル問題とオイラー](http://www.kurims.kyoto-u.ac.jp/~kyodo/kokyuroku/contents/pdf/1583-12.pdf) (<http://www.kurims.kyoto-u.ac.jp/~kyodo/kokyuroku/contents/pdf/1583-12.pdf>), 2007年8月23日, 数理解析研究所講究録, 第1583巻, 2008年, pp.159-167

その第6節によれば, Eulerは, 積分  $\int \frac{-\log(1-x)}{x} dx$  を2通りに計算して比較することによって,  $\zeta(2)$  を定義に基いて直接計算するよりもずっと少ない項数の和で同精度で  $\zeta(2)$  の計算する方法を発見している.

現代的には二重対数関数(dilogarithm, 以下ダイログと呼ぶ)を用いてその近似計算法を見付けたことになる. ダイログ  $\text{Li}_2(z)$  の定義は

$$\text{Li}_2(z) = \int_0^z \frac{-\log(1-x)}{x} dx = \sum_{n=1}^{\infty} \frac{z^n}{n^2}$$

であった.  $x = 1-y$  と置換すると,

$$\text{Li}(z) = \int_1^{1-z} \frac{\log y}{1-y} dy$$

となり,

$$\frac{d}{dy}(-\text{Li}_2(y) - \log(1-y) \log y) = \frac{\log y}{1-y}$$

なので,  $\text{Li}_2(1) = \zeta(2)$  と  $y \rightarrow 1$  で  $\log(1-y) \log y \rightarrow 0$  となることより,

$$\text{Li}_2(z) = [-\text{Li}_2(y) - \log(1-y) \log y]_1^{1-z} = -\text{Li}_2(1-z) + \zeta(2) - \log z \log(1-z).$$

すなわち,

$$\zeta(2) = \text{Li}_2(z) + \text{Li}_2(1-z) + \log z \log(1-z).$$

特に  $z = 1/2$  のとき,

$$\zeta(2) = 2 \text{Li}_2(1/2) + (\log 2)^2 = \sum_{n=1}^{\infty} \frac{1}{2^{n-1} n^2} + (\log 2)^2.$$

$\zeta(2)$  の定義に基いて最初の10項を足した結果

$$\sum_{n=1}^{10} \frac{1}{n^2} = 1.54976\ 77311\ 665 \dots$$

は正確な値  $\zeta(2) = \frac{\pi^2}{6} = 1.64493\ 40668\ 482 \dots$  と整数部分しか一致しないが,

$$\sum_{n=1}^{10} \frac{1}{2^{n-1} n^2} + (\log 2)^2 = 1.64492\ 00516\ 736 \dots$$

は正確な値に小数点以下第4桁まで一致している. 最初の30項の和では, 前者は 1.61215 01176 015 ... で正確な値と小数点以下第1桁までしか一致しないが, 後者は 1.64493 40668 463 ... は小数点以下第11桁まで一致している.

In [28]:

```
1 Q(N) = sum(n->1/n^2, 1:N)
2 R(N) = sum(n->1/(2.0^(n-1)*n^2), 1:N) + log(2)^2
3 @show Q(10), Q(10)-pi^2/6, R(10), R(10)-pi^2/6
4 @show Q(30), Q(30)-pi^2/6, R(30), R(30)-pi^2/6
5 [(N, Q(N), Q(N)-pi^2/6, R(N), R(N)-pi^2/6) for N in 10:10:50]
```

```
(Q(10), Q(10) - pi ^ 2 / 6, R(10), R(10) - pi ^ 2 / 6) = (1.5497677311665408, -0.09516633568168564, 1.6449200516736
968, -1.4015174529591334e-5)
(Q(30), Q(30) - pi ^ 2 / 6, R(30), R(30) - pi ^ 2 / 6) = (1.6121501176015975, -0.03278394924662886, 1.6449340668463
974, -1.828981410767483e-12)
```

Out[28]:

```
5-element Array{Tuple{Int64,Float64,Float64,Float64,Float64},1}:
 (10, 1.54977, -0.0951663, 1.64492, -1.40152e-5)
 (20, 1.59616, -0.0487708, 1.64493, -3.98311e-9)
 (30, 1.61215, -0.0327839, 1.64493, -1.82898e-12)
 (40, 1.62024, -0.0246901, 1.64493, -1.11022e-15)
 (50, 1.62513, -0.0198013, 1.64493, -2.22045e-16)
```

## 2.5 二重対数関数の高木貞治『解析概論』§35の例2(p.113)への応用

問題: 次を示せ:

$$I := \int_0^{\pi/2} \log(\sin x) dx = -\frac{\pi}{2} \log 2.$$

解答例(高木貞治『解析概論』§35の例2(p.113)):  $\sin(\pi/2 - x) = \cos x$  より,  $I = \int_0^{\pi/2} \log(\cos x) dx$ .  $\sin(\pi - x) = \sin x$  より,

$I = \int_{\pi/2}^{\pi} \log(\sin x) dx$  なので  $\int_0^{\pi} \log(\sin x) dx = 2I$ . そこで,  $x = 2t$  と置換して,  $\sin(2t) = 2 \cos t \sin t$  を使うと,

$$I = \frac{1}{2} \int_0^{\pi} \log(\sin x) dx = \int_0^{\pi/2} \log(2 \cos t \sin t) dt = \frac{\pi}{2} \log 2 + I + I.$$

ゆえに  $I = -\frac{\pi}{2} \log 2$ .  $\square$

注意: 上の問題の定積分は, 高木貞治『解析概論』§35の例2(p.113)だけではなく, 同書§62の例3(p.225)でも扱われている. そこで

$$\frac{\log(1-z)}{z} = -1 - z - \frac{z^2}{2} - \dots$$

の原点を中心とする周回積分を用いて,

$$\int_0^{\pi} \log(1 - 2r \cos \theta + r^2) \theta = 2\pi \log r \quad (r > 0)$$

を得ている. そしてこれの  $r = 1$  の場合から上の問題の結果を得ている. 二重対数関数の定義は  $-\frac{\log(1-z)}{z}$  の不定積分だったので, 上の問題は二重対数関数と関係していることになる. 以下の2つの問題も見よ.  $\square$

問題(WolframAlpha): 次の  $F(x)$  について  $F'(x) = \log(\sin x)$  となることを示せ.

$$F(x) = \frac{i}{2} x^2 + \frac{i}{2} \text{Li}_2(e^{2ix}) - x \log(1 - e^{2ix}) + x \log(\sin x).$$

解答例:  $\text{Li}_2'(x) = -\frac{\log(1-x)}{x}$ ,  $(-\log(1-x))' = \frac{1}{1-x}$  より,

$$F'(x) = ix + \frac{i}{2} 2ie^{2ix} \frac{-\log(1-e^{2ix})}{e^{2ix}} - \log(1-e^{2ix}) + x \frac{2ie^{2ix}}{1-e^{2ix}} + x \frac{\cos x}{\sin x} + \log(\sin x).$$

第2項と第3項がキャンセルして消える. さらに,

$$x \frac{\cos x}{\sin x} = ix \frac{e^{ix} + e^{-ix}}{e^{ix} - e^{-ix}} = ix \frac{e^{2ix} + 1}{e^{2ix} - 1}$$

を代入すると, 第1,4,5項の和を  $ix$  で割ったものは

$$1 + \frac{-2e^{2ix}}{e^{2ix} - 1} + \frac{e^{2ix} + 1}{e^{2ix} - 1} = 0$$

となるので,  $F'(x) = \log(\sin x)$  となることがわかる.  $\square$

筆者はこの問題の  $F(x)$  の式は [WolframAlphaによる計算 \(http://www.wolframalpha.com/input/?i=%5Cint+log\(sin+x\)+dx\)](http://www.wolframalpha.com/input/?i=%5Cint+log(sin+x)+dx) で知った.

問題: すぐ上の問題の結果を用いて, 次を示せ:

$$\int_0^{\pi/2} \log(\sin x) dx = -\frac{\pi}{2} \log 2, \quad \zeta(2) = \text{Li}_2(1) = \frac{\pi^2}{6}.$$

解答例:  $\text{Li}_2(1)$  と  $\text{Li}(-1)$  は

$$\text{Li}_2(1) = \sum_{n=1}^{\infty} \frac{1}{n^2}, \quad \text{Li}_2(-1) = \sum_{n=1}^{\infty} \frac{(-1)^n}{n^2}$$

と表わされるので,

$$\text{Li}_2(1) + \text{Li}_2(-1) = \sum_{k=1}^{\infty} \frac{2}{(2k)^2} = \frac{1}{2} \text{Li}_2(1).$$

ゆえに

$$\text{Li}_2(-1) = -\frac{1}{2} \text{Li}_2(1). \quad (1)$$

そして, 上の問題の  $F(x)$  について

$$F\left(\frac{\pi}{2}\right) = \frac{\pi^2}{8}i + \frac{i}{2} \text{Li}_2(-1) - \frac{\pi}{2} \log 2, \quad F(0) = \frac{i}{2} \text{Li}_2(1)$$

なので, 上の問題の結論より,

$$\mathbb{R} \ni \int_0^{\pi/2} \log(\sin x) dx = [F(x)]_0^{\pi/2} = \frac{\pi^2}{8}i + \frac{i}{2} \text{Li}_2(-1) - \frac{\pi}{2} \log 2 - \frac{i}{2} \text{Li}_2(1).$$

両辺の実部と虚部(の2倍)を比較して,  $\int_0^{\pi/2} \log(\sin x) dx = -\frac{\pi}{2} \log 2$  と

$$0 = \frac{\pi^2}{4} + \text{Li}_2(-1) - \text{Li}_2(1) \quad (2)$$

を得る. 以上の(1),(2)より,  $\zeta(2) = \text{Li}_2(1) = \frac{\pi^2}{6}$  が得られる.  $\square$

注意: 以上の計算に関連する話題については, [筆者によるツイッターにおける一連の発言 \(https://twitter.com/genkuroki/status/830016144933167104\)](https://twitter.com/genkuroki/status/830016144933167104) も参照せよ.  $\square$

## 2.6 二重対数関数の五項関係式

[Rogersの二重対数関数 \(https://www.google.co.jp/search?q=Rogers+dilogarithm\)](https://www.google.co.jp/search?q=Rogers+dilogarithm)  $L(z)$  を次のように定義する:

$$L(z) = \frac{1}{2} \int_0^z \left( \frac{-\log(1-x)}{x} + \frac{-\log x}{1-x} \right) dx = \text{Li}_2(z) + \frac{1}{2} \log z \log(1-z).$$

1つ目の等号は  $L(z)$  の定義であり, 2つ目の等号は次を使えば得られる:

$$\frac{d}{dx}(\text{Li}_2(x) + \log x \log(1-x)) = \frac{-\log x}{1-x}.$$

$L(z)$  と  $\text{Li}_2(z)$  の違いは初等函数  $\frac{1}{2} \log z \log(1-z)$  の分だけである.

$$\text{Li}_2(0) = 0, \text{Li}_2(1) = \zeta(2) = \frac{\pi^2}{6} \text{ より,}$$

$$L(0) = 0, \quad L(1) = \frac{\pi^2}{6}.$$

この節では次の**五項関係式** (five-term relation)

$$L(x) + L(1-xy) + L(y) + L\left(\frac{1-y}{1-xy}\right) + L\left(\frac{1-x}{1-xy}\right) = \frac{\pi^2}{2}. \quad (\text{FTR})$$

を示そう. 二重対数函数の五項関係式については19世紀にすでに先行する研究結果があったが, この形式に整理された五項関係式は

- R. Rogers, On Function Sum Theorems Connected with the Series  $\sum_{n=1}^{\infty} \frac{z^n}{n^2}$ , Proc. London Math. Soc. 4 (1907), 169-189, [DOI](https://doi.org/10.1112/plms/s2-4.1.169) (<https://doi.org/10.1112/plms/s2-4.1.169>)

で証明された.

**補題:** 函数  $f_i(x)$  達について

$$F(x, X) = \sum_{i=1}^N \log(f_i(x)) \log(1 - f_i(X))$$

は  $F(x, X) = F(X, x)$  を満たしていると仮定する. このとき,

$$f(x) = \sum_{i=1}^N L(f_i(x))$$

は定数函数になる.

**証明:**  $df = 0$  を示せばよい.

$$-2 dL(x) = \frac{\log(1-x) dx}{x} + \frac{\log(x) dx}{1-x} = d \log(x) \log(1-x) - \log(x) d \log(1-x)$$

なので,

$$-2 df(x) = \sum_{i=1}^N d \log(f_i(x)) \log(1 - f_i(X)) - \sum_{i=1}^N \log(f_i(x)) d \log(1 - f_i(x)).$$

$F(x, X)$  と  $F(X, x)$  の  $x$  に関する外微分  $d$  を取ると,

$$\begin{aligned} dF(x, X) &= \sum_{i=1}^N d \log(f_i(x)) \log(1 - f_i(X)), \\ dF(X, x) &= \sum_{i=1}^N \log(f_i(X)) d \log(1 - f_i(x)). \end{aligned}$$

$F(x, X) = F(X, x)$  より, これらは等しい:

$$\sum_{i=1}^N d \log(f_i(x)) \log(1 - f_i(X)) = \sum_{i=1}^N \log(f_i(X)) d \log(1 - f_i(x))$$

$X = x$  とおくと,

$$\sum_{i=1}^N d \log(f_i(x)) \log(1 - f_i(x)) = \sum_{i=1}^N \log(f_i(x)) d \log(1 - f_i(x)).$$

以上を合わせると,  $-2df = 0$  が得られる.  $\square$

**五項関係式の証明:**  $f_i(x, y)$  ( $i = 1, 2, 3, 4, 5$ ) をそれぞれ  $x, 1 - xy, y, \frac{1-y}{1-xy}, \frac{1-x}{1-xy}$  と定め,  $f(x, y) = \sum_{i=1}^5 L(f_i(x, y))$  とおく.

\$\lrcorner\$

$f(x, y)$  が定数関数になることを示そう. 上の補題より, そのためには

$$F(x, y; X, Y) = \sum_{i=1}^5 \log(f_i(x, y)) \log(1 - f_i(X, Y))$$

が  $F(x, y; X, Y) = F(X, Y; x, y)$  を満たしていることを示せば十分である. 具体的に計算すると,

$$1 - (1 - XY) = XY, \quad 1 - \frac{1 - Y}{1 - XY} = \frac{Y(1 - X)}{1 - XY}, \quad 1 - \frac{1 - X}{1 - XY} = \frac{X(1 - Y)}{1 - XY}$$

より

$$\begin{aligned} F(x, y; X, Y) &= \log(x) \log(1 - X) + \log(1 - xy) \log(1 - XY) + \log(y) \log(1 - Y) \\ &\quad + (\log(1 - y) - \log(1 - xy))(\log(Y) + \log(1 - X) - \log(1 - XY)) \\ &\quad + (\log(1 - x) - \log(1 - xy))(\log(X) + \log(1 - Y) - \log(1 - XY)) \\ &= \log(x) \log(1 - X) + \log(1 - x) \log(X) \\ &\quad + \log(y) \log(1 - Y) + \log(1 - y) \log(Y) \\ &\quad + \log(1 - x) \log(1 - Y) + \log(1 - y) \log(1 - X) \\ &\quad - \log(1 - x) \log(1 - XY) - \log(1 - xy) \log(1 - X) \\ &\quad - \log(1 - y) \log(1 - XY) - \log(1 - xy) \log(1 - Y) \\ &\quad + 2 \log(1 - xy) \log(1 - XY) \\ &\quad + \log(1 - xy) \log(XY) - \log(1 - xy) \log(X) - \log(1 - xy) \log(Y). \end{aligned}$$

最下段の3つの項がキャンセルして消えるので,  $F(x, y; X, Y) = F(X, Y; x, y)$  となることがわかる. これで  $f(x, y)$  が定数関数であることがわかった.

そして,  $L(0) = 0, L(1) = \frac{\pi^2}{6}$  より,

$$f(0, 0) = L(0) + L(0) + L(1) + L(1) + L(1) = \frac{\pi^2}{2}.$$

以上によって  $f(x, y) = \frac{\pi^2}{2}$  となることが示された.  $\square$

**注意:** 上の補題を使えば,  $L(x) + L(1 - x)$  も定数関数になることがわかり,  $L(0) = 0, L(1) = \frac{\pi^2}{6}$  より,

$$L(x) + L(1 - x) = \frac{\pi^2}{6}$$

となることも示される. これを使って, 五項関係式は次のように書き直される:

$$L(x) + L(y) = L(xy) + L\left(\frac{x(1-y)}{1-xy}\right) + L\left(\frac{y(1-x)}{1-xy}\right).$$

この形の公式も各種文献でよく見る.  $\square$

**注意:**  $x_1 = x, x_2 = 1 - xy$  とおき, 帰納的に  $x_n$  を

$$x_{n-1}x_{n+1} = 1 - x_n$$

という関係式で定めると,

$$x_3 = y, \quad x_4 = \frac{1-y}{1-xy}, \quad x_5 = \frac{1-x}{1-xy}, \quad x_6 = x, \quad x_7 = 1-xy, \quad \dots$$

$x_{n+5} = x_n$  が成立しており, 出て来た  $x, y$  の式がちやうど五項関係式の5つの  $f_i(x, y)$  に一致していることがわかる. この事実は [クラスター代数](https://www.google.co.jp/search?q=%E3%82%AF%E3%83%A9%E3%82%B9%E3%82%BF%E3%83%BC%E4%BB%A3%E6%95%B0) (<https://www.google.co.jp/search?q=%E3%82%AF%E3%83%A9%E3%82%B9%E3%82%BF%E3%83%BC%E4%BB%A3%E6%95%B0>) と [dilogarithm identities](https://www.google.co.jp/search?q=dilogarithm+identities) (<https://www.google.co.jp/search?q=dilogarithm+identities>) の理論として大幅に一般化されている. クラスター代数の入門的解説については

- 黒木玄, [リンク2のクラスター代数](https://genkuroki.github.io/documents/cluster_algebra_rank2.pdf) ([https://genkuroki.github.io/documents/cluster\\_algebra\\_rank2.pdf](https://genkuroki.github.io/documents/cluster_algebra_rank2.pdf)), 2010年



- 黒木玄, [フリーズパターン — 数の繰返し模様の不思議](https://genkuroki.github.io/documents/20120810FriezePattern.pdf) (<https://genkuroki.github.io/documents/20120810FriezePattern.pdf>), 2013年
- 黒木玄, [フリーズパターンの不思議](https://genkuroki.github.io/documents/20121115Akita.pdf) (<https://genkuroki.github.io/documents/20121115Akita.pdf>), 2013年

などを見よ. 検索すれば他にもたくさん見付かる. □

**注意:** このノートの下の方で二重対数関数の量子版と五項関係式の量子化とみあされる五角形恒等式について説明する. □

### 3 無限積

多項式の計算で因数分解が極めて有用であるのと同様に, より一般の関数の無限積への因数分解も極めて有用である. そのためには無限積を正しく扱う方法を知っておかなければいけない.

数列  $a_n$  から作られる無限積  $\prod_{n=1}^{\infty} a_n$  が収束するとは, 数列  $p_n = \prod_{k=1}^n a_k = a_1 a_2 \cdots a_n$  が収束することであると定める. そのとき, その無限積の収束先が数列  $p_n$  の収束先で定義される.

無限積については高木貞治『解析概論』の§51(pp.178-181)を参照せよ.

**例:** 以下では  $p_i$  は  $i$  番目の素数であるとする.  $p_1 = 2, p_2 = 3, p_3 = 5, \dots$  である. さらに  $s > 1$  であると仮定する. このとき, 素数全体に関する無限積  $\prod_{i=1}^{\infty} \frac{1}{1 - p_i^{-s}}$  がどうなるかを調べてみよう.  $Z_N = \prod_{i=1}^N \frac{1}{1 - p_i^{-s}}$  とおく. このとき,

$$Z_N = \prod_{i=1}^N (1 + p_i^{-s} + p_i^{-2s} + \cdots)$$

なので,  $Z_N$  は非負の整数たち  $k_1, \dots, k_N$  に対する

$$p_1^{-k_1 s} \cdots p_N^{-k_N s} = \frac{1}{(p_1^{k_1} \cdots p_N^{k_N})^s}$$

達すべての和に等しい. 正の整数は一意的に素因数分解できるので,  $N \rightarrow \infty$  で  $Z_N$  は正の整数  $n$  に対する  $\frac{1}{n^s}$  達全体の和に等しくなる:

$$\prod_{i=1}^{\infty} \frac{1}{1 - p_i^{-s}} = \sum_{n=1}^{\infty} \frac{1}{n^s} = \zeta(s).$$

左辺をゼータ関数  $\zeta(s)$  の**Euler積**と呼ぶ. □

#### 3.1 無限積の絶対収束

**定理:** 級数  $\sum_{n=1}^{\infty} a_n$  が絶対収束していれば, 無限積  $\prod_{n=1}^{\infty} (1 + a_n)$  も収束する. このとき, その無限積は**絶対収束**するという.

**証明:** 級数  $\sum_{n=1}^{\infty} a_n$  は絶対収束していると仮定し,

$$M = \sum_{n=1}^{\infty} |a_n|, \quad p_n = \prod_{k=1}^n (1 + a_k), \quad b_n = p_n - p_{n-1} = p_{n-1} a_n$$

とおく. このとき,  $p_n = \sum_{k=1}^n b_k$  であり,  $x \geq 0$  のとき  $1 + x \leq e^x$  であることより,

$$|p_n| \leq \prod_{k=1}^n (1 + |a_k|) \leq \prod_{k=1}^n e^{|a_k|} = e^M$$

となり,  $|b_k| = |p_{k-1}| |a_k| \leq e^M |a_k|$  となるので,

$$\sum_{n=1}^{\infty} |b_n| \leq e^M \sum_{n=1}^{\infty} |a_n| = e^M M < \infty.$$

よって、級数  $\sum_{n=1}^{\infty} b_n$  は絶対収束する。  $p_n = \sum_{k=1}^n b_k$  なので無限積  $\prod_{n=1}^{\infty} (1 + a_n) = \lim_{n \rightarrow \infty} p_n$  が収束することがわかった。□

例:  $n \rightarrow \infty$  のとき  $a_n = O(1/n^2)$  であるとする。このとき、ある定数  $C \geq 0$  が存在して、 $|a_n| \leq C/n^2$  となる。このとき、

$$\sum_{n=1}^{\infty} |a_n| \leq C \sum_{n=1}^{\infty} \frac{1}{n^2} = C\zeta(2) < \infty$$

より  $\sum_{n=1}^{\infty} a_n$  は絶対収束するので、無限積  $\prod_{k=1}^{\infty} (1 + a_n)$  は絶対収束する。例えば、

$$f(s) = s \prod_{n=1}^{\infty} \left(1 - \frac{s^2}{n^2}\right)$$

の右辺の無限積は絶対収束している。このとき、すべての整数  $n \in \mathbb{Z}$  について  $f(n) = 0$  となる。実はこの  $f(s)$  は  $f(s) = \frac{\sin(\pi s)}{\pi}$  となっていることを示せる。この一連のノート群の後の方で複数の方法で証明することになるだろう。□

注意:  $\prod_{n=1}^{\infty} (1 + a_n)$  は絶対収束していると仮定する。そのとき、 $\prod_{n=1}^{\infty} \frac{1}{1 - a_n} = \prod_{n=1}^{\infty} \left(1 + \frac{a_n}{1 - a_n}\right)$  も絶対収束している。なぜならば、 $a_n$  は 0 に収束しているの、ある番号  $N$  が存在して、 $n \geq N$  ならば  $|1 - a_n| \geq \frac{1}{2}$  となる。絶対収束性を論実場合には  $N = 1$  の場合だけを考えればよいので、 $N = 1$  であるとする。そのとき、

$$\sum_{n=1}^{\infty} \left| \frac{a_n}{1 - a_n} \right| \leq 2 \sum_{n=1}^{\infty} |a_n| < \infty.$$

これで、 $\prod_{n=1}^{\infty} \frac{1}{1 - a_n}$  も絶対収束していることが示された。□

例:  $|q| < 1$  のとき、 $\sum_{n=1}^{\infty} q^n$  は絶対収束しているの、無限積  $f(q) = \prod_{n=1}^{\infty} (1 - q^n)$  も絶対収束しており、 $\frac{1}{f(q)} = \prod_{n=1}^{\infty} \frac{1}{1 - q^n}$  も絶対収束している。そして、

$$\frac{1}{f(q)} = \prod_{m=1}^{\infty} \sum_{k=0}^{\infty} q^{mk} = \sum_{k_1, k_2, \dots = 0}^{\infty} q^{k_1 + 2k_2 + 3k_3 + \dots} = \sum_{n=0}^{\infty} p(n)q^n.$$

ただし、有限個を除いて 0 であるような非負の整数たちの組全体を  $(k_1, k_2, \dots)$  は動くとする。その約束のもとで

$$p(n) = \#\{(k_1, k_2, k_3, \dots) \mid k_1 + 2k_2 + 3k_3 + \dots = n\}.$$

ここで  $\#X$  は集合  $X$  の元の個数を表す。すなわち、 $p(n)$  は  $n$  を正の整数の和に分解する方法の個数に等しい。 $p(n)$  は  $n$  の**分割数**と呼ばれている。例えば、 $p(1) = 1, p(2) = 2, p(3) = 3, p(4) = 5$  であり、

$$5 = 4 + 1 = 3 + 2 = 3 + 1 + 1 = 2 + 2 + 1 = 2 + 1 + 1 + 1 = 1 + 1 + 1 + 1 + 1$$

なので  $p(5) = 7$  である。 $\eta(q) = q^{1/24}f(q)$  は**Dedekindのη函数**と呼ばれるモジュラー変換性を持つ重要な特殊函数である。分割数のような組み合わせ論的な数とモジュラー変換性を持つ特殊函数が関係しており、豊かな数学の世界を作っている。□

```
In [29]: 1 partition_number(n) = sympy[:functions][:combinatorial][:numbers][:nT](n)
         2 [(n, partition_number(n)) for n in 1:10]
```

```
Out[29]: 10-element Array{Tuple{Int64,Int64},1}:
 (1, 1)
 (2, 2)
 (3, 3)
 (4, 5)
 (5, 7)
 (6, 11)
 (7, 15)
 (8, 22)
 (9, 30)
 (10, 42)
```

```
In [30]: 1 q = symbols("q")
2 N = 10
3 series(prod(1/(1-q^n) for n in 1:N), n=N+1)
```

Out[30]:  $1 + q + 2q^2 + 3q^3 + 5q^4 + 7q^5 + 11q^6 + 15q^7 + 22q^8 + 30q^9 + 42q^{10} + \mathcal{O}(q^{11})$

注意:  $\prod_{n=1}^{\infty} (1 - q^n)$  について

$$\prod_{n=1}^{\infty} (1 - q^n) = \sum_{n=-\infty}^{\infty} (-1)^n q^{n(3n-1)/2} = \sum_{n=-\infty}^{\infty} (-1)^n q^{((6n-1)^2-1)/24}$$

となることが知られている. 次のセルを見よ.  $\square$

```
In [31]: 1 q = symbols("q")
2 N = 5
3 M = N*(3N-1)/2
4 @time series(sum((-1)^n*q^(n*(3n-1)/2) for n in -M:M), q, n=M+1) !>display
5 @time series(prod(1-q^n for n in 1:M), q, n=M+1) !> display
```

$$1 - q - q^2 + q^5 + q^7 - q^{12} - q^{15} + q^{22} + q^{26} - q^{35} + \mathcal{O}(q^{36})$$

$$1 - q - q^2 + q^5 + q^7 - q^{12} - q^{15} + q^{22} + q^{26} - q^{35} + \mathcal{O}(q^{36})$$

0.353393 seconds (60.13 k allocations: 2.974 MiB)  
5.947022 seconds (24.66 k allocations: 1.159 MiB)

### 3.2 絶対収束する無限積の収束先が積の順序によらないこと

定理: 絶対収束する無限積の収束先は積の順序によらない.

証明: 絶対収束する級数の収束先が和の順序によらないことに帰着する. 無限積  $\prod_{n=1}^{\infty} (1 + a_n)$  は絶対収束していると仮定し,

$M = \sum_{n=1}^{\infty} |a_n|$  とおく. その無限積の収束先は

$$p_n = \prod_{k=1}^n (1 + a_k) = \sum_{I \subset \{1, 2, \dots, n\}} \prod_{i \in I} a_i$$

の収束先に等しい. さらに,

$$\sum_{I \subset \{1, 2, \dots, n\}} \left| \prod_{i \in I} a_i \right| = \prod_{k=1}^n (1 + |a_k|) \leq \prod_{k=1}^n e^{|a_k|} \leq e^M < \infty.$$

ゆえに, 正の整数全体の集合の有限部分集合  $I$  に対する積  $\prod_{i \in I} a_i$  達すべての無限和は絶対収束している. したがって, その無限和の収束先は和の順序によらない. 無限積の積の順序を変えることは, その無限和の和の順序を変える場合の特別な場合になっているので, 無限積の収束先も積の順序によらない.  $\square$

### 3.3 量子二重対数関数

以下,  $|q| < 1$  であると仮定する.

このノートでは量子二重対数関数(quantum dilogarithm)  $E_q(x)$  を

$$E_q(x) = \left( \prod_{k=0}^{\infty} (1 + q^{k+1/2}x) \right)^{-1} = \frac{1}{(1 + q^{1/2}x)(1 + q^{1+1/2}x)(1 + q^{2+1/2}x) \dots}.$$

と定める. (注意:  $E_q(-x)$  の方を量子二重対数と呼んでいる文献が多い.) この  $E_q(x)$  は

$$E_q(qx) = (1 + q^{1/2}x)E_q(x), \quad E_q(0) = 1$$

という条件で一意に特徴付けられる.

上の  $E_q(x)$  が量子二重対数函数と呼ばれる理由はその対数が次の表示を持つからである:

$$\begin{aligned}\log E_q(-x) &= -\sum_{k=0}^{\infty} \log(1 - q^{k+1/2}x) = \sum_{k=0}^{\infty} \sum_{n=1}^{\infty} \frac{(q^{k+1/2}x)^n}{n} \\ &= \sum_{n=1}^{\infty} \left( \sum_{k=0}^{\infty} q^{nk} \right) (q^{1/2}x)^n = \sum_{n=1}^{\infty} \frac{(q^{1/2}x)^n}{n(1-q^n)}.\end{aligned}$$

ゆえに,

$$(1-q) \log E_q(-x) = \sum_{n=1}^{\infty} \frac{(q^{1/2}x)^n}{n(n)_q}, \quad (n)_q = \frac{1-q^n}{1-q}.$$

ここで,  $(n)_q$  は **q数** (q-number) と呼ばれ,  $q \rightarrow 1$  で  $n$  に収束する. ゆえに,  $q \rightarrow 1$  のとき,

$$(1-q) \log E_q(-x) \rightarrow \sum_{n=1}^{\infty} \frac{x^n}{n^2} = \text{Li}_2(x).$$

$E_q(x)$  の  $q \rightarrow 1$  での漸近挙動は(古典)二重対数函数  $\text{Li}_2(x)$  を含んでいる.

### 3.3.1 量子二重対数函数と量子指数函数との関係

**量子指数函数**  $e_q(x)$  を

$$e_q(x) = \sum_{n=0}^{\infty} \frac{x^n}{(n)_q!}, \quad (n)_q! = (1)_q(2)_q \cdots (n)_q$$

と定める.  $(n)_q!$  は**q階乗 (q-factorial)** と呼ばれている. このとき,

$$\frac{e_q(x) - e_q(qx)}{(1-q)x} = \sum_{n=1}^{\infty} \frac{(n)_q x^{n-1}}{(n)_q!} = \sum_{m=0}^{\infty} \frac{x^m}{(m)_q!} = e_q(x).$$

ゆえに,  $n \rightarrow \infty$  のとき,  $e_q(q^n x) \rightarrow e_q(0) = 1$  より,

$$\begin{aligned}e_q(x) &= \frac{e_q(qx)}{1 - (1-q)x} = \frac{e_q(q^2x)}{(1 - (1-q)x)(1 - (1-q)xq)} \\ &= \frac{e_q(q^3x)}{(1 - (1-q)x)(1 - (1-q)xq)(1 - (1-q)xq^2)} \\ &= \cdots \cdots \cdots \\ &= \frac{1}{(1 - (1-q)x)(1 - (1-q)xq)(1 - (1-q)xq^2) \cdots} \\ &= \left( \prod_{k=0}^{\infty} (1 - (1-q)xq^k) \right)^{-1}.\end{aligned}$$

これより,

$$E_q(x) = e_q \left( -\frac{q^{1/2}x}{1-q} \right) = \sum_{n=0}^{\infty} \frac{q^{n/2}(-x)^n}{(1-q)(1-q^2) \cdots (1-q^n)}.$$

### 3.3.2 q二項定理

**q二項係数** (q-binomial coefficients) を

$$\binom{n}{k}_q = \frac{(n)_q!}{(k)_q!(n-k)_q!} = \frac{(n)_q(n-1)_q \cdots (n-k+1)_q}{(k)_q!}$$

と定める. このとき,

$$(k)_q + q^k(n-k+1)_q = \frac{1 - q^k + q^k(1 - q^{n-k+1})}{1-q} = \frac{1 - q^{n+1}}{1-q} = (n+1)_q$$

より,

$$\binom{n}{k-1}_q + q^k \binom{n}{k}_q = \binom{n}{k-1}_q \frac{(k)_q + q^k(n-k+1)_q}{(k)_q} = \binom{n+1}{k}_q. \quad (1)$$

文字  $x, y$  は関係式  $yx = qxy$  を満たしていると仮定する. このとき, (1)を使うと, 通常の二項定理の場合と同様に  $n$  に関して帰納的に次の公式が成立することを示せる(実際に示してみよ):

$$(x+y)^n = \sum_{k=0}^n \binom{n}{k}_q x^k y^{n-k} \quad (yx = qxy).$$

これを **q二項定理** と呼ぶ.

### 3.3.3 q指数関数の加法公式

$yx = qxy$  のとき,

$$e_q(x+y) = e_q(x)e_q(y)$$

となることを示そう.

$$\begin{aligned} e_q(x+y) &= \sum_{n=0}^{\infty} \frac{(x+y)^n}{(n)_q!} = \sum_{n=0}^{\infty} \frac{1}{(n)_q!} \sum_{k=0}^n \binom{n}{k}_q x^k y^{n-k} \\ &= \sum_{n=0}^{\infty} \sum_{k=0}^n \frac{x^k}{(k)_q!} \frac{y^{n-k}}{(n-k)_q!} = \sum_{k,l=0}^{\infty} \frac{x^k}{(k)_q!} \frac{y^l}{(l)_q!} = e_q(x)e_q(y). \end{aligned}$$

$$E_q(x) = e_q\left(-\frac{q^{1/2}x}{1-q}\right) \text{ より,}$$

$$E_q(x+y) = E_q(x)E_q(y) \quad (yx = qxy).$$

### 3.3.4 量子二重対数関数の五角形恒等式

$yx = qxy$  と仮定し,

$$:xy := q^{1/2}xy = q^{-1/2}yx$$

とおく. そのとき,  $yE_q(x) = E_q(qx)y = E_q(x)(1+q^{1/2}x)y = E_q(x)(:xy: + y)$  より,

$$y^n E_q(x) = E_q(x)(:xy: + y)^n.$$

この等式の両辺に適当な係数をかけて  $n$  について足し上げると,

$$E_q(y)E_q(x) = E_q(x)E_q(:xy: + y).$$

$y :xy: = q :xy: y$  なので, 前節の結果より,

$$E_q(:xy: + y) = E_q(:xy:)E_q(y).$$

したがって,

$$E_q(y)E_q(x) = E_q(x)E_q(:xy:)E_q(y).$$

これを量子二重対数関数の**五角形恒等式** (pentagon identity)と呼ぶ.

五角形恒等式は量子二重対数関数  $E_q(x)$  の 0 でない形式べき級数として一意に特徴付ける. なぜならば, 五角形恒等式の  $x, y$  に 0 を代入すると  $E_q(0) = 1$  であることがわかり, 五角形恒等式の両辺の  $y$  に関する1次の項を比較すると,

$$E_q(qx)y = yE_q(x) = E_q(x)(:xy: + y) = E_q(x)(1 + q^{1/2}x)y$$

となるので  $E_q(qx) = E_q(1 + q^{1/2}x)$  が得られ, 以上で得られた2つの条件で  $E_q(x)$  が一意に特徴付けられるからである.

**注意:** 量子二重対数関数の五角形恒等式は  $A_2$  型の量子クラスター代数から得られる quantum dilogarithm identity とみなされる. さらにアフィン型の場合の無限積版の quantum dilogarithm identities も存在する. それらの恒等式の背景には量子クラスター代数や量子群の普遍  $R$  行列の理論が隠れている. 有限積の量子二重対数恒等式の作り方については

- 黒木玄, [量子 dilogarithm 恒等式の作り方](https://genkuroki.github.io/documents/20101228HowToMakeQDIs.pdf) (https://genkuroki.github.io/documents/20101228HowToMakeQDIs.pdf), 2010年12月

にも解説がある. □

In [ ]:

1