

13 Euler-Maclaurinの和公式

黒木玄

2018-07-04~2019-04-03, 2023-06-22

- Copyright 2018,2019,2023 Gen Kuroki
- License: MIT <https://opensource.org/licenses/MIT> (<https://opensource.org/licenses/MIT>)
- Repository: <https://github.com/genkuroki/Calculus> (<https://github.com/genkuroki/Calculus>)

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/13%20Euler-Maclaurin%20summation%20formula.ipynb> (<http://nbviewer.jupyter.org/github/genkuroki/Calculus/blob/master/13%20Euler-Maclaurin%20summation%20formula.ipynb>)
- <https://genkuroki.github.io/documents/Calculus/13%20Euler-Maclaurin%20summation%20formula.pdf> (<https://genkuroki.github.io/documents/Calculus/13%20Euler-Maclaurin%20summation%20formula.pdf>)

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://julia-lang.org) (<https://julia-lang.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>)
- [Julia v1.1.0 の Windows 8.1 へのインストール](https://nbviewer.jupyter.org/github/genkuroki/msfd28/blob/master/install.ipynb) (<https://nbviewer.jupyter.org/github/genkuroki/msfd28/blob/master/install.ipynb>)

を参照せよ. 前者は古く, 後者の方が新しい.

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 Bernoulli多項式](#)
 - [1.1 Bernoulli多項式の定義](#)
 - [1.2 Bernoulli多項式の基本性質](#)
 - [1.3 べき乗和](#)
 - [1.4 Bernoulli数の計算法](#)
 - [1.5 周期的Bernoulli多項式のFourier級数展開](#)
- ▼ [2 Euler-Maclaurinの和公式](#)
 - [2.1 Euler-Maclaurinの和公式の導出](#)
 - [2.2 Euler-Maclaurinの和公式の2つの解釈](#)
 - [2.3 Euler-Maclaurinの和公式の形式的導出](#)
 - [2.4 べき乗和\(再\)](#)
 - [2.5 Stirlingの近似公式](#)
 - [2.6 Poissonの和公式とEuler-Maclaurinの和公式の関係](#)
 - [2.7 台形公式とPoissonの和公式の関係](#)
- ▼ [3 ゼータ関数への応用](#)
 - [3.1 解析接続](#)
 - [3.2 \$\zeta\(2\)\$ の近似計算](#)
 - [3.3 \$s = 1\$ での \$\zeta\(s\)\$ の定数項がEuler定数になること](#)
 - [3.4 負の整数におけるゼータ関数の特殊値の計算](#)
 - [3.5 発散級数の有限部分と \$\zeta\(-r\)\$ の関係](#)

```
In [1]: ▶ 1 using Base.MathConstants
2 using Base64
3 using Printf
4 using Statistics
5 const e = e
6 endof(a) = lastindex(a)
7 linspace(start, stop, length) = range(start, stop, length=length)
8
9 using Plots
10 #gr(); ENV["PLOTS_TEST"] = "true"
11 #clibrary(:colorcet)
12 #clibrary(:misc)
13 default(fmt=:png)
14
15 function pngplot(P...; kwargs...)
16     sleep(0.1)
17     pngfile = tempname() * ".png"
18     savefig(plot(P...; kwargs...), pngfile)
19     showing("image/png", pngfile)
20 end
21 pngplot(; kwargs...) = pngplot(plot!(; kwargs...))
22
23 showing(mime, fn) = open(fn) do f
24     base64 = base64encode(f)
25     display("text/html", """""")
26 end
27
28 using SymPy
29 #sympy.init_printing(order="lex") # default
30 #sympy.init_printing(order="rev-lex")
31
32 using SpecialFunctions
33 using QuadGK
```

```
In [2]: ▶ 1 # Override the Base.show definition of SymPy.jl:
2 # https://github.com/JuliaPy/SymPy.jl/blob/29c5bfd1d10ac53014fa7fef468bc8deccadc2fc/src/
3
4 @eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::SymbolicObject)
5     print(io, as_markdown("\displaystyle " * sympy.latex(x, mode="plain", fold_short_fr
6 end
7 @eval SymPy function Base.show(io::IO, ::MIME"text/latex", x::AbstractArray{Sym})
8     function toeqnarray(x::Vector{Sym})
9         a = join(["\displaystyle " * sympy.latex(x[i]) for i in 1:length(x)], "\\\")
10        """\left[ \begin{array}{r}$a\end{array} \right]"""
11    end
12    function toeqnarray(x::AbstractArray{Sym,2})
13        sz = size(x)
14        a = join([join("\displaystyle " .* map(sympy.latex, x[i,:]), "&") for i in 1:sz
15        "\left[ \begin{array}{r} * repeat("r",sz[2]) * "]" * a * "\end{array}\right]"
16    end
17    print(io, as_markdown(toeqnarray(x)))
18 end
```

1 Bernoulli多項式

1.1 Bernoulli多項式の定義

定義(Bernoulli多項式): Bernoulli多項式 $B_n(x)$ ($n = 0, 1, 2, \dots$)を

$$\frac{ze^{zx}}{e^z - 1} = \sum_{n=0}^{\infty} \frac{B_n(x)}{n!} z^n$$

によって定義する。□

1.2 Bernoulli多項式の基本性質

一般化Bernoulli多項式の基本性質: Bernoulli多項式 $B_n(x)$ は以下の性質を満たしている:

(1) $B_0(x) = 1$.

(2) $\int_0^1 B_n(x) dx = \delta_{n,0}$.

$$(3) B_n(x+h) = \sum_{k=0}^n \binom{n}{k} B_{n-k}(x) h^k = \sum_{k=0}^n \binom{n}{k} B_k(x) h^{n-k}.$$

$$(4) B'_n(x) = nB_{n-1}(x).$$

$$(5) B_n(x+1) = B_n(x) + nx^{n-1}.$$

$$(6) B_n(1-x) = (-1)^n B_n(x).$$

$$(7) B_n(1) = B_n(0) + \delta_{n,1} \text{ となる.}$$

$$(8) B_n(0) = 1, B_n(0) = -\frac{1}{2} \text{ とな, } n \text{ が3以上の奇数ならば } B_n(0) = 0 \text{ となる.}$$

証明: (1) $e^{zx} = 1 + O(z)$, $\frac{e^z - 1}{z} = 1 + O(z)$ より, $\frac{ze^{zx}}{e^z - 1} = 1 + O(z)$ なので $B_0(x) = 1$.

(2)を示そう.

$$\int_0^1 \frac{ze^{zx}}{e^z - 1} dx = \frac{z}{e^z - 1} \int_0^1 e^{zx} dx = \frac{z}{e^z - 1} \frac{e^z - 1}{z} = 1,$$

$$\int_0^1 \frac{ze^{zx}}{e^z - 1} dx = \sum_{n=0}^{\infty} \frac{z^n}{n!} \int_0^1 B_n(x) dx$$

なので, これらを比較して $\int_0^1 B_n(x) dx = \delta_{n,0}$.

(3) 二項定理より,

$$\int_0^1 (x+y)^n dy = \sum_{k=0}^n \binom{n}{k} x^{n-k} \int_0^1 y^k dy.$$

ゆえに, x の関数を x の関数に移す線形写像(前方移動平均)

$$f(x) \mapsto \int_0^1 f(x+y) dy$$

は多項式を多項式に移し, 最高次の係数が1の多項式を最高次の係数が1の同次の多項式に移す. これより, 線形写像

$f(x) \mapsto \int_0^1 f(x+y) dy$ は多項式どうしの一対一対応を与える線形写像になっていることがわかる. そして,

$$\int_0^1 \frac{ze^{z(x+y)}}{e^z - 1} dx = \sum_{n=0}^{\infty} \frac{\int_0^1 B_n(x+y) dy}{n!} z^n,$$

$$\int_0^1 \frac{ze^{z(x+y)}}{e^z - 1} dx = \frac{ze^{zx}}{e^z - 1} \int_0^1 e^{zy} dy = \frac{ze^{zx}}{e^z - 1} \frac{e^z - 1}{z} = e^{zx} = \sum_{n=0}^{\infty} \frac{x^n}{n!} z^n$$

なので, これらを比較して,

$$\int_0^1 B_n(x+y) dy = x^n$$

が成立することがわかる. ゆえに,

$$\int_0^1 B_n(x+h+y) dy = (x+h)^n = \sum_{k=0}^n \binom{n}{k} x^{n-k} h^k = \int_0^1 \sum_{k=0}^n \binom{n}{k} B_{n-k}(x+y) h^k dy$$

より

$$B_n(x+h) = \sum_{k=0}^n \binom{n}{k} B_{n-k}(x) h^k.$$

(4) すぐ上の等式の右辺の h の係数を見ることによって,

$$B'_n(x) = nB_{n-1}(x).$$

(5) Bernoulli多項式の母函数の x に $x+1$ を代入すると,

$$\frac{ze^{z(x+1)}}{e^z - 1} = \frac{ze^z e^{zx}}{e^z - 1} = \frac{z(1 + (e^z - 1))e^{zx}}{e^z - 1} = \frac{ze^{zx}}{e^z - 1} + ze^{zx}$$

なので両辺を z について展開して比較すれば(5)が得られる.

(6) Bernoulli多項式の母関数の x に $1-x$ を代入すると,

$$\frac{ze^{z(1-x)}}{e^z - 1} = \frac{ze^z e^{-zx}}{e^z - 1} = \frac{ze^{-zx}}{1 - e^{-z}} = \frac{-ze^{-zx}}{e^{-z} - 1}$$

とBernoulli多項式の母関数の z に $-z$ を代入したものになるので, 両辺を z について展開して比較すれば(5)が得られる.

(7) 上の(2)と(4)より, n が2以上のとき,

$$B_n(1) - B_n(0) = \int_0^1 B_n'(x) dx = n \int_0^1 B_{n-1}(x) dx = n\delta_{n-1,0} = \delta_{n,1}$$

ゆえに n が2以上のとき $B_n(1) = B_n(0) + \delta_{n,1}$.

(8) 次の関数が z の偶関数で $z \rightarrow 0$ で 1 になることから, (6)が得られる:

$$\frac{z}{e^z - 1} + \frac{z}{2} = \frac{z}{2} \frac{e^{z/2} + e^{-z/2}}{e^{z/2} - e^{-z/2}}. \quad \square$$

注意: $B_n = B_n(0)$ は**Bernoulli数**と呼ばれている. (3)で (x, h) を $(0, x)$ で置き換えると, Bernoulli多項式がBernoulli数で表わされることがわかる:

$$B_n(x) = \sum_{k=0}^n \binom{n}{k} B_k x^{n-k}.$$

上の定理の条件(1)(2)(4)によってBernoulli多項式 $B_n(x)$ が n について帰納的に一意に決まる. \square

例:

$$B_0 = 1, \quad B_1 = -\frac{1}{2}, \quad B_2 = \frac{1}{6}, \quad B_3 = 0, \quad B_4 = -\frac{1}{30}$$

なので

$$\begin{aligned} B_0(x) &= 1, & B_1(x) &= x - \frac{1}{2}, & B_2(x) &= x^2 - x + \frac{1}{6}, \\ B_3(x) &= x^3 - \frac{3}{2}x^2 + \frac{1}{2}x, & B_4(x) &= x^4 - 2x^3 + x^2 - \frac{1}{30}. \end{aligned} \quad \square$$

```
In [3]: ► 1 BernoulliPolynomial(n,x) = sympy.bernoulli(n,x)
        2 x = symbols("x", real=True)
        3 [BernoulliPolynomial(n,x) for n in 0:10]
```

```
Out[3]: [ 1
          x - 1/2
          x^2 - x + 1/6
          x^3 - 3x^2/2 + x/2
          x^4 - 2x^3 + x^2 - 1/30
          x^5 - 5x^4/2 + 5x^3/3 - x/6
          x^6 - 3x^5 + 5x^4/2 - x^2/2 + 1/42
          x^7 - 7x^6/2 + 7x^5/2 - 7x^3/6 + x/6
          x^8 - 4x^7 + 14x^6/3 - 7x^4/3 + 2x^2/3 - 1/30
          x^9 - 9x^8/2 + 6x^7 - 21x^5/5 + 2x^3 - 3x/10
          x^10 - 5x^9 + 15x^8/2 - 7x^6 + 5x^4 - 3x^2/2 + 5/66]
```

```

In [4]: ▶ 1 # (2)  $\int_0^1 B_n(x) dx = \delta_{n0}$ 
2
3 BernoulliPolynomial(n,x) = sympy.bernoulli(n,x)
4 x = symbols("x", real=true)
5 [integrate(BernoulliPolynomial(n,x), (x,0,1)) for n = 0:10]'

Out[4]: [1 0 0 0 0 0 0 0 0 0 0]

In [5]: ▶ 1 # (3)  $B_n(x+h) = \sum_{k=0}^n \binom{n}{k} B_{n-k}(x) h^k$ 
2
3 BernoulliNumber(n) = sympy.bernoulli(n)
4 BernoulliPolynomial(n,x) = sympy.bernoulli(n,x)
5 BinomCoeff(n,k) = sympy.binomial_coefficients_list(n)[k+1]
6 x, h = symbols("x h", real=true)
7 [BernoulliPolynomial(n,x) == sum(k->BinomCoeff(n,k)*BernoulliNumber(k)*x^(n-k), 0:n) for

Out[5]: 1x11 adjoint(::Vector{Bool}) with eltype Bool:
1 1 1 1 1 1 1 1 1 1 1

In [6]: ▶ 1 # (4)  $B_n'(x) = n B_{n-1}(x)$ 
2
3 BernoulliPolynomial(n,x) = sympy.bernoulli(n,x)
4 x = symbols("x", real=true)
5 [diff(BernoulliPolynomial(n,x), x) == n*BernoulliPolynomial(n-1,x) for n = 1:10]'

Out[6]: 1x10 adjoint(::Vector{Bool}) with eltype Bool:
1 1 1 1 1 1 1 1 1 1

In [7]: ▶ 1 # (5)  $B_n(x+1) = B_n(x) + n x^{n-1}$ 
2
3 BernoulliPolynomial(n,x) = sympy.bernoulli(n,x)
4 x = symbols("x", real=true)
5 [simplify(BernoulliPolynomial(n,x+1) - BernoulliPolynomial(n,x)) for n in 0:10]

Out[7]: [ 0
1
2x
3x^2
4x^3
5x^4
6x^5
7x^6
8x^7
9x^8
10x^9]

In [8]: ▶ 1 # (6)  $B_n(1-x) = (-1)^n B_n(x)$ 
2
3 BernoulliPolynomial(n,x) = sympy.bernoulli(n,x)
4 x = symbols("x", real=true)
5 [expand(BernoulliPolynomial(n,1-x)) == (-1)^n*BernoulliPolynomial(n,x) for n in 0:10]'

Out[8]: 1x11 adjoint(::Vector{Bool}) with eltype Bool:
1 1 1 1 1 1 1 1 1 1 1

In [9]: ▶ 1 # (7)  $B_n(1) = B_n(0) + \delta_{n1}$ 
2
3 BernoulliPolynomial(n,x) = sympy.bernoulli(n,x)
4 x = symbols("x", real=true)
5 [expand(BernoulliPolynomial(n,1)) - BernoulliPolynomial(n,0) for n in 0:10]'

Out[9]: [0 1 0 0 0 0 0 0 0 0 0]

```

```
In [10]: 1 # (8) B_n = B_n(0) は n が3以上の奇数ならば0になる.
2
3 BernoulliNumber(n) = sympy.bernoulli(n)
4 [(n, BernoulliNumber(n)) for n in 0:10]
```

```
Out[10]: 11-element Vector{Tuple{Int64, Sym}}:
(0, 1)
(1, -1/2)
(2, 1/6)
(3, 0)
(4, -1/30)
(5, 0)
(6, 1/42)
(7, 0)
(8, -1/30)
(9, 0)
(10, 5/66)
```

1.3 べき乗和

m は正の整数であるとする. Bernoulli多項式について,

$$B_{m+1}(x+1) - B_{m+1}(x) = (m+1)x^m, \quad \text{i.e.} \quad x^m = \frac{B_{m+1}(x+1) - B_{m+1}(x)}{m+1}$$

が成立しているのを、これを $x = 0, 1, \dots, n$ について足し上げると,

$$\sum_{j=1}^n j^m = \frac{B_{m+1}(n+1) - B_{m+1}}{m+1}. \quad \square$$

```
In [11]: 1 PowerSum(m, n) = sum(j->j^m, 1:n)
2 BernoulliNumber(n) = sympy.bernoulli(n)
3 BernoulliPolynomial(n, x) = sympy.bernoulli(n, x)
4 PowerSumFormula(m, n) = (BernoulliPolynomial(m+1, n+1) - BernoulliNumber(m+1)) / (m+1)
5 [(m, PowerSum(m, 10), PowerSumFormula(m, 10)) for m in 1:10]
```

```
Out[11]: 10-element Vector{Tuple{Int64, Int64, Sym}}:
(1, 55, 55)
(2, 385, 385)
(3, 3025, 3025)
(4, 25333, 25333)
(5, 220825, 220825)
(6, 1978405, 1978405)
(7, 18080425, 18080425)
(8, 167731333, 167731333)
(9, 1574304985, 1574304985)
(10, 14914341925, 14914341925)
```

1.4 Bernoulli数の計算法

Bernoulli数 B_n は

$$\frac{z}{e^z - 1} = \sum_{n=1}^{\infty} B_n \frac{z^n}{n!}$$

で定義される. しかし、この展開を直接計算することによって Bernoulli 数を求めるのは効率が悪い.

まず、左辺の $z \rightarrow 0$ の極限を取ることによって $B_0 = 1$ であることはすぐにわかる.

次に、 n が 3 以上の奇数のとき $B_n = 0$ となることを(再び)示そう.

$$\frac{z}{e^z - 1} + \frac{z}{2} = \frac{z e^z + 1}{2 e^z - 1} = \frac{z e^{z/2} + e^{-z/2}}{2 e^{z/2} - e^{-z/2}}$$

より、左辺は偶関数になるので、その展開の奇数次の項は消える. このことから、 $B_1 = -1/2$ であつ、 $0 = B_3 = B_5 = B_7 = \dots$ であることもわかる.

$$\frac{z e^z}{e^z - 1} = \sum_{j,k=0}^{\infty} \frac{z^j}{j!} \frac{B_k z^k}{k!} = \sum_{n=0}^{\infty} \left(\sum_{k=0}^n \binom{n}{k} B_k \right) \frac{z^n}{n!}$$

であつ

$$\frac{ze^z}{e^z - 1} = \frac{z}{e^z - 1} + z = \sum_{n=0}^{\infty} (B_n + \delta_{n1}) \frac{z^n}{n!}$$

なので、これらを比較すると

$$\sum_{k=0}^{n-1} \binom{n}{k} B_k = \delta_{n1}.$$

ゆえに、 n を $n+1$ で置き換え、 $n \geq 1$ とし、 B_n を他で表わす式に書き直すと

$$B_n = -\frac{1}{n+1} \sum_{k=0}^{n-1} \binom{n+1}{k} B_k \quad (n \geq 1).$$

これを使えば帰納的に B_n を求めることができる。 $B_0 = 1, B_1 = -1/2, 0 = B_3 = B_5 = B_7 = \dots$ であることを使うと、

$$B_{2m} = -\frac{1}{2m+1} \left(1 - \frac{2m+1}{2} + \sum_{k=1}^{m-1} \binom{2m+1}{2k} B_{2k} \right).$$

問題: 上の方ではSymPyにおけるBernoulli数の関数を利用した。Bernoulli数を計算するためのプログラムを自分で書け。□

解答例: 次のセルの通り。□

In [12]:

```
1 # binomial coefficient: binom(n,k) = n(n-1) · (n-k+1)/k!
2 #
3 mydiv(a, b) = a / b
4 mydiv(a::Integer, b::Integer) = a ÷ b
5 function binom(n, k)
6     k < 0 && return zero(n)
7     k == 0 && return one(n)
8     b = one(n)
9     for j in 1:k
10         b = mydiv(b*(n-k+j), j)
11     end
12     b
13 end
14
15 @show binom(Rational(big"100")/3, 30)
16
17 # Bernoulli numbers: B(n) = Bernoulli[n+1] = B_n
18 #
19 struct Bernoulli{T}
20     B::Array{T,1}
21 end
22 function Bernoulli(; maxn=200)
23     B = zeros(Rational{BigInt},maxn+1)
24     B[1] = 1 # B_0
25     B[2] = -1//2 # B_1
26     for n in big"2":2:maxn+1
27         B[n+1] = -(1//(n+1))*sum(j->binom(n+1,j)*B[j+1], 0:n-1)
28         # B_n = -(1/(n+1)) Σ_{j=0}^{n-1} binom(n+1,j)*B_j
29     end
30     Bernoulli(B)
31 end
32 (B::Bernoulli)(n) = B.B[n+1]
33
34 maxn = 200
35 @time B = Bernoulli(maxn=maxn) # B_n を B_{maxn} まで計算
36 BB(n) = float(B(n)) # B(n) = B_n である。 BB(n)はその浮動小数点版
37
38 # SymPyのBernoulli数と比較して正しく計算できているかどうかを確認
39 #
40 BernoulliNumber(n) = sympy.bernoulli(n)
41 @show B_eq_B = [B(n) == BernoulliNumber(n) for n in 0:maxn]
42 println()
43 @show all(B_eq_B)
44
45 maxnprint = 30
46 println()
47 for n in [0; 1; 2:2:maxnprint]
48     println("B($n) = ", B(n))
49 end
50 println()
51 for n in [0; 1; 2:2:maxnprint]
52     println("BB($n) = ", BB(n))
53 end
```


となることがわかる。したがって、Fourier級数論より、 $k = 1$ のときは整数ではない実数 x について、 $k \geq 2$ の場合にはすべての実数 x について次が成立することがわかる:

$$\widetilde{B}_k(x) = B_k(x - [x]) = -k! \sum_{n \neq 0} \frac{e^{2\pi i n x}}{(2\pi i n)^k}.$$

すなわち、 $k = 1, 2, 3, \dots$ について

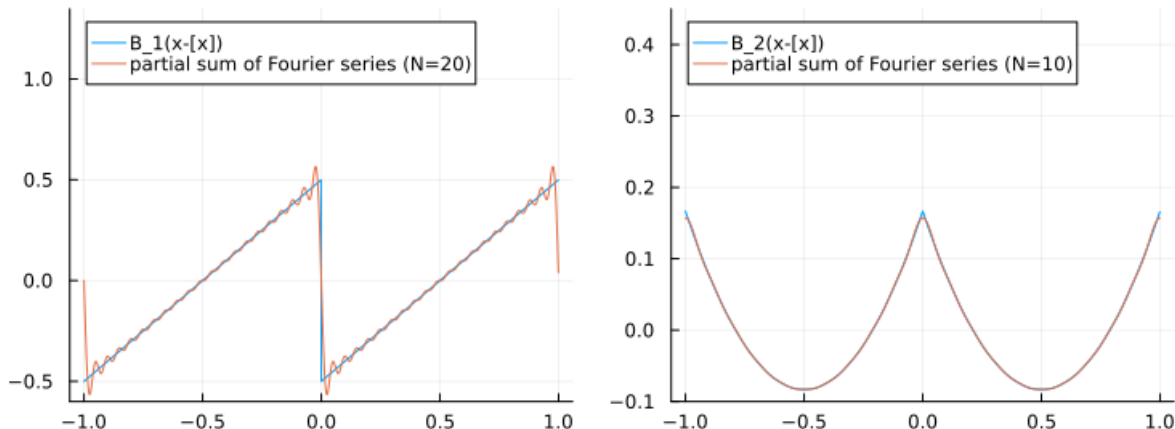
$$\widetilde{B}_{2k-1}(x) = (-1)^k 2(2k-1)! \sum_{n=1}^{\infty} \frac{\sin(2\pi n x)}{(2\pi n)^{2k-1}}, \quad \widetilde{B}_{2k}(x) = (-1)^{k-1} 2(2k)! \sum_{n=1}^{\infty} \frac{\cos(2\pi n x)}{(2\pi n)^{2k}}.$$

このことから、 k が大きいとき(実際には $k = 5, 6$ 程度ですでに)、周期的Bernoulli多項式は $n = 1$ の項だけで

$$\widetilde{B}_{2k-1}(x) \approx (-1)^k 2(2k-1)! \frac{\sin(2\pi x)}{(2\pi)^{2k-1}}, \quad \widetilde{B}_{2k}(x) \approx (-1)^{k-1} 2(2k)! \frac{\cos(2\pi x)}{(2\pi)^{2k}}$$

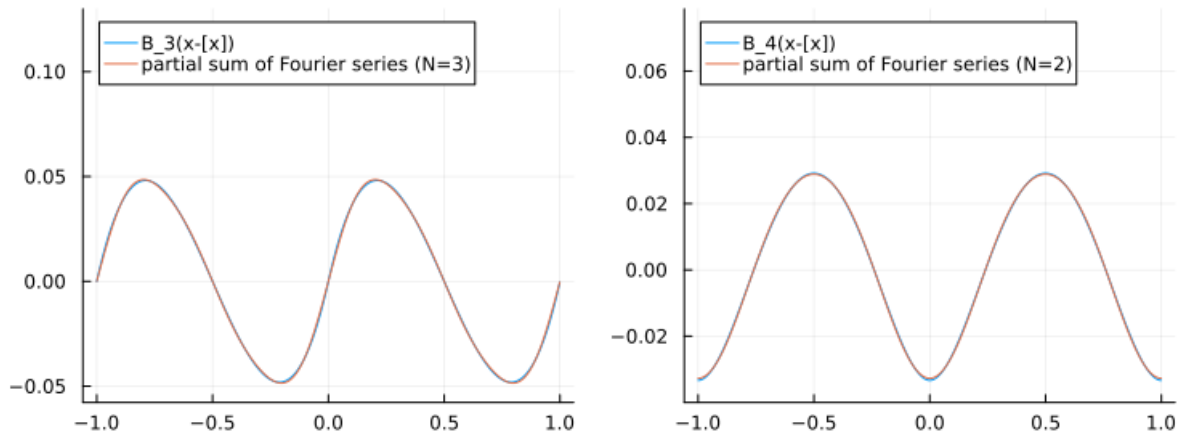
```
In [13]: ▶ 1 BBB = Bernoulli(Float64.(B.B)) # Float64 Bernoulli numbers
2 BP(k,x) = sum(j->binom(k,j)*BBB(k-j)*x^j, 0:k) # Float64 Bernoulli polynomial
3 PBP(k,x) = BP(k, x - floor(x)) # periodic Bernoulli polynomial
4
5 # partial sum of Fourier series of periodic Bernoulli polynomial
6 function PSFS(k, N, x)
7     k == 0 && return zero(x)
8     if isodd(k)
9         return (-1)^((k+1)÷2)*2*factorial(k)*sum(n->sin(2π*n*x)/(2π*n)^k, 1:N)
10    else
11        return (-1)^(k÷2-1)*2*factorial(k)*sum(n->cos(2π*n*x)/(2π*n)^k, 1:N)
12    end
13 end
14
15 PP = []
16 x = -1.0:0.001:0.999
17 for (k,N) in [(1,20), (2,10), (3,3), (4,2), (5,1), (6,1)]
18     y = PBP.(k,x)
19     z = PSFS.(k, N, x)
20     ymin = 1.2*minimum(y)
21     ymax = 2.7*maximum(y)
22     P = plot(legend=:topleft, size=(400, 250), ylim=(ymin, ymax))
23     plot!(x, y, label="B_$(k)(x-[x])")
24     plot!(x, z, label="partial sum of Fourier series (N=$N)")
25     push!(PP, P)
26 end
27
28 plot(PP[1:2]..., size=(750, 280))
```

Out[13]:



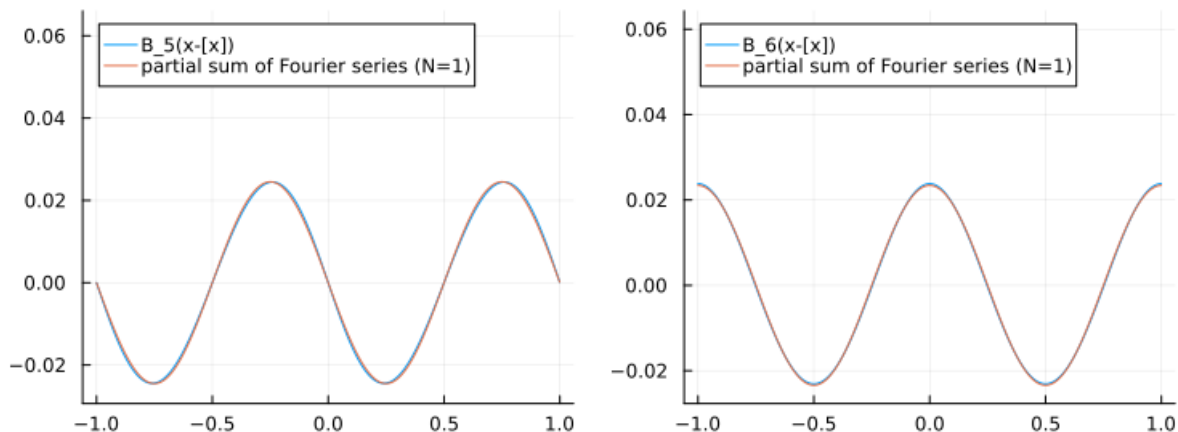
```
In [14]: 1 plot(PP[3:4]..., size=(750, 280))
```

Out[14]:



```
In [15]: 1 plot(PP[5:6]..., size=(750, 280))
```

Out[15]:



2 Euler-Maclaurinの和公式

2.1 Euler-Maclaurinの和公式の導出

Bernoulli多項式 $B_n(x)$ とBernoulli数 B_n について

$$B_0(x) = 1, \quad \frac{d}{dx} \frac{B_n(x)}{n!} = \frac{B_{n-1}(x)}{(n-1)!},$$

$$B_1(0) = -\frac{1}{2}, \quad B_1(1) = \frac{1}{2},$$

$$B_n(1) = B_n(0) = B_n \quad (n = 0, 2, 3, 4, 5, \dots)$$

$$B_{2j+1} = 0 \quad (j = 1, 2, 3, \dots)$$

が成立している。以下ではしばらくのあいだこれらの条件しか使わない。

部分積分を繰り返すことによって、

$$\begin{aligned} \int_0^1 f(x) dx &= \int_0^1 B_0(x) f(x) dx \\ &= [B_1(x) f(x)]_0^1 - \int_0^1 B_1(x) f'(x) dx \\ &= [B_1(x) f(x)]_0^1 - \frac{1}{2} [B_2(x) f'(x)]_0^1 + \int_0^1 \frac{B_2(x)}{2} f''(x) dx \\ &= [B_1(x) f(x)]_0^1 - \frac{1}{2} [B_2(x) f'(x)]_0^1 + \frac{1}{3!} [B_3(x) f''(x)]_0^1 - \int_0^1 \frac{B_3(x)}{3!} f'''(x) dx \\ &= \dots \dots \dots \\ &= \sum_{k=1}^n \frac{(-1)^{k-1}}{k!} [B_k(x) f^{(k-1)}(x)]_0^1 + (-1)^n \int_0^1 \frac{B_n(x)}{n!} f^{(n)}(x) dx \\ &= \frac{f(0) + f(1)}{2} + \sum_{k=2}^n (-1)^{k-1} \frac{B_k}{k!} (f^{(k-1)}(1) - f^{(k-1)}(0)) + (-1)^n \int_0^1 \frac{B_n(x)}{n!} f^{(n)}(x) dx. \end{aligned}$$

実数 x に対して, x 以下の最大の整数を $[x]$ と書く. このとき, $x - [x]$ は x の「小数部分」になる. このように記号を準備しておくと, 整数 j に対して,

$$\begin{aligned} \int_j^{j+1} f(x) dx &= \int_0^1 f(x+j) dx \\ &= \frac{f(j) + f(j+1)}{2} + \sum_{k=2}^n (-1)^{k-1} \frac{B_k}{k!} (f^{(k-1)}(j+1) - f^{(k-1)}(j)) + (-1)^n \int_0^1 \frac{B_n(x)}{n!} f^{(n)}(x+j) dx \\ &= \frac{f(j) + f(j+1)}{2} + \sum_{k=2}^n (-1)^{k-1} \frac{B_k}{k!} (f^{(k-1)}(j+1) - f^{(k-1)}(j)) + (-1)^n \int_j^{j+1} \frac{B_n(x - [x])}{n!} f^{(n)}(x) dx. \end{aligned}$$

$a < b$ を満たす整数 a, b に対して, 上の式を $j = a$ から $j = b - 1$ まで足し上げると,

$$\begin{aligned} \int_a^b f(x) dx &= \sum_{j=a}^{b-1} \frac{f(j) + f(j+1)}{2} + \sum_{k=2}^n (-1)^{k-1} \frac{B_k}{k!} (f^{(k-1)}(b) - f^{(k-1)}(a)) + (-1)^n \int_a^b \frac{B_n(x - [x])}{n!} f^{(n)}(x) dx \\ &= \frac{f(a) + f(b)}{2} + \sum_{j=a+1}^{b-1} f(j) + \sum_{k=2}^n (-1)^{k-1} \frac{B_k}{k!} (f^{(k-1)}(b) - f^{(k-1)}(a)) - R_n. \end{aligned}$$

ここで

$$R_n = (-1)^{n-1} \int_a^b \frac{B_n(x - [x])}{n!} f^{(n)}(x) dx$$

とおいた. さらい, n が3以上の奇数のとき $B_n = 0$ となることを使うと,

$$\int_a^b f(x) dx = \frac{f(a) + f(b)}{2} + \sum_{j=a+1}^{b-1} f(j) - \sum_{1 \leq i \leq n/2} \frac{B_{2i}}{(2i)!} (f^{(2i-1)}(b) - f^{(2i-1)}(a)) - R_n.$$

この公式を **Euler-Maclaurinの和公式** と呼ぶ. これは n が3以上の奇数のとき $B_n = 0$ となることを使うと次のように書き直される:

$$\int_a^b f(x) dx = \frac{f(a) + f(b)}{2} + \sum_{j=a+1}^{b-1} f(j) - \sum_{k=2}^n \frac{B_k}{k!} (f^{(k-1)}(b) - f^{(k-1)}(a)) - R_n.$$

注意: 上の公式は n が2以上の整数の場合の結果である. $n = 1$ の場合の結果は次のようにして得られる.

$$\int_0^1 f(x) dx = \int_0^1 x' f(x) dx = [x f(x)]_0^1 - \int_0^1 x f'(x) dx = f(1) - \int_0^1 x f'(x) dx$$

より, $a < b$ を満たす整数 a, b に対して,

$$\begin{aligned} \int_a^b f(x) dx &= \sum_{j=a}^{b-1} f(j+1) - \sum_{j=a}^{b-1} \int_0^1 x f'(x+j) dx \\ &= \sum_{j=a}^b f(j) - f(a) - \int_a^b (x - [x]) f'(x) dx \end{aligned}$$

すなわち,

$$\begin{aligned} \sum_{j=a}^b f(j) &= \int_a^b f(x) dx + f(a) + \int_a^b (x - [x]) f'(x) dx \\ &= \int_a^b f(x) dx + f(a) + \sum_{j=a}^{b-1} \int_0^1 x f'(x+j) dx \end{aligned}$$

これらのシンプルな公式も結構よく使われる. □

注意: Euler-Maclaurinの和公式の証明は部分積分の繰り返しであった. それはTaylorの定理の証明とほぼ同じような議論である. Taylorの定理も以下のような部分積分の繰り返しで証明される:

$$p_n(t) = \frac{(x-t)^n}{n!}$$

とおくと, $p_0(t) = 1$, $-p'_n(t) = p_{n-1}(t)$, $[g(t)(-p_n(t))]'_a = g(a) \frac{(x-a)^n}{n!}$ なので

$$\begin{aligned}
f(x) &= f(a) + \int_a^x f(t) dt = f(a) + \int_a^x f(t)p_0(t) dt \\
&= f(a) + f'(a)(x-a) + \int_a^x f'(t)p_1(t) dt \\
&= f(a) + f'(a)(x-a) + f''(a)\frac{(x-a)^2}{2} + \int_a^x f''(t)p_2(t) dt \\
&= f(a) + f'(a)(x-a) + f''(a)\frac{(x-a)^2}{2} + f'''(a)\frac{(x-a)^3}{3!} + \int_a^x f'''(t)p_3(t) dt \\
&= \dots \dots \dots \\
&= \sum_{k=0}^n f^{(k)}(a)\frac{(x-a)^k}{k!} + R_n.
\end{aligned}$$

ここで

$$R_n = \int_a^x f^{(n)}(t)p_n(t) dt = \int_a^x f^{(n)}(t)\frac{(x-t)^n}{n!} dt$$

と書いた。その意味する所は、Maclaurinの和公式はTaylorの公式の「仲間」だ、と云う事。

2.2 Euler-Maclaurinの和公式の2つの解釈

Euler-Maclaurinの和公式の解釈1:

$$\sum_{j=a}^{b-1} \frac{f(j) + f(j+1)}{2} = \frac{f(a) + f(b)}{2} + \sum_{j=a+1}^{b-1} f(j)$$

は積分 $\int_a^b f(x) dx$ の近似計算に使われる台形公式である。ゆえに、Euler-Maclaurinの和公式

$$\begin{aligned}
\int_a^b f(x) dx &= \frac{f(a) + f(b)}{2} + \sum_{j=a+1}^{b-1} f(j) - \sum_{1 \leq i \leq n/2} \frac{B_{2i}}{(2i)!} (f^{(2i-1)}(b) - f^{(2i-1)}(a)) - R_n, \\
R_n &= (-1)^{n-1} \int_a^b \frac{B_n(x - [x])}{n!} f^{(n)}(x) dx
\end{aligned}$$

は台形公式による積分の近似の誤差が、

$$- \sum_{1 \leq i \leq n/2} \frac{B_{2i}}{(2i)!} (f^{(2i-1)}(b) - f^{(2i-1)}(a)) - R_n$$

になっていることを意味している。例えば、 $n = 1$ の場合には、 $B_1(x) = x - \frac{1}{2}$ なので、

$$\int_a^b f(x) dx = \frac{f(a) + f(b)}{2} + \sum_{j=a+1}^{b-1} f(j) - \int_a^b \left(x - [x] - \frac{1}{2} \right) f'(x) dx.$$

$n = 2$ の場合には $B_2(x) = x^2 - x + \frac{1}{6}$, $B_2 = \frac{1}{6}$ であり、

$$\int_a^b f(x) dx = \frac{f(a) + f(b)}{2} + \sum_{j=a+1}^{b-1} f(j) - \frac{f'(b) - f'(a)}{12} + \int_a^b \frac{B_2(x - [x])}{2} f''(x) dx.$$

となる。□

In [16]:

```
1 # すぐ上の公式の検証
2
3 BernoulliNumber(n) = sympy.bernoulli(n)
4 BernoulliPolynomial(n,x) = sympy.bernoulli(n,x)
5
6 function EulerMaclaurinIntegral(f, a, b, n)
7     x = symbols("x", real=true)
8     (
9         (f(a)+f(b))/Sym(2)
10        + sum(j->f(j), a+1:b-1)
11        - sum(k -> (
12            BernoulliNumber(k)/factorial(Sym(k))
13            * (diff(f(x), x, k-1)(x=>b) - diff(f(x), x, k-1)(x=>a))
14            ), 2:n)
15    )
16 end
17
18 function EulerMaclaurinRemainder(f, a, b, n)
19     x = symbols("x", real=true)
20     g = diff(f(x), x, n)
21     (-1)^(n-1) * sum(k -> (
22         integrate(BernoulliPolynomial(n,x)*g(x=>x+k), (x,0,1))
23         ), a:b-1)/factorial(Sym(n))
24 end
25
26 x = symbols("x", real=true)
27
28 [integrate(x^m, (x, 0, 10)) for m in 7:15] > display
29
30 [
31     EulerMaclaurinIntegral(x->x^m, 0, 10, 5) - EulerMaclaurinRemainder(x->x^m, 0, 10, 5)
32     for m in 7:15
33 ] > display
```

$$\left[\begin{array}{r} 12500000 \\ \hline 1000000000 \\ 9 \\ \hline 1000000000 \\ 100000000000 \\ \hline 11 \\ 250000000000 \\ \hline 3 \\ 1000000000000 \\ \hline 13 \\ 5000000000000 \\ \hline 7 \\ 20000000000000 \\ \hline 3 \\ 62500000000000 \end{array} \right]$$

$$\left[\begin{array}{r} 12500000 \\ \hline 1000000000 \\ 9 \\ \hline 1000000000 \\ 100000000000 \\ \hline 11 \\ 250000000000 \\ \hline 3 \\ 1000000000000 \\ \hline 13 \\ 5000000000000 \\ \hline 7 \\ 20000000000000 \\ \hline 3 \\ 62500000000000 \end{array} \right]$$

Euler-Maclaurinの和公式の解釈2: Euler-Maclaurinの和公式は次のように書き直される:

$$\sum_{j=a}^b f(j) = \int_a^b f(x) dx + \frac{f(a) + f(b)}{2} + \sum_{1 \leq i \leq n/2} \frac{B_{2i}}{(2i)!} (f^{(2i-1)}(b) - f^{(2i-1)}(a)) + R_n,$$

$$R_n = (-1)^{n-1} \int_a^b \frac{B_n(x - [x])}{n!} f^{(n)}(x) dx$$

これは n が3以上の奇数のとき $B_n = 0$ となることを使うと次のように書き直される:

$$\sum_{j=a}^b f(j) = \int_a^b f(x) dx + \frac{f(a) + f(b)}{2} + \sum_{k=2}^n \frac{B_k}{k!} (f^{(k-1)}(b) - f^{(k-1)}(a)) + R_n,$$

$$R_n = (-1)^{n-1} \int_a^b \frac{B_n(x - [x])}{n!} f^{(n)}(x) dx$$

この等式は函数 f の整数における値の和 $\sum_{j=a}^b f(j)$ を積分 $\int_a^b f(x) dx$ で近似したときの誤差が

$$\frac{f(a) + f(b)}{2} + \sum_{1 \leq i \leq n/2} \frac{B_{2i}}{(2i)!} (f^{(2i-1)}(b) - f^{(2i-1)}(a)) + R_n$$

になっていることを意味している. 例えば, $n = 1$ の場合には, $B_1(x) = x - \frac{1}{2}$ なので,

$$\sum_{j=a}^b f(j) = \int_a^b f(x) dx + \frac{f(a) + f(b)}{2} + \int_a^b \left(x - [x] - \frac{1}{2} \right) f'(x) dx.$$

$n = 2$ の場合には $B_2(x) = x^2 - x + \frac{1}{6}$, $B_2 = \frac{1}{6}$ であり,

$$\sum_{j=a}^b f(j) = \int_a^b f(x) dx + \frac{f(a) + f(b)}{2} + \frac{f'(b) - f'(a)}{12} - \int_a^b \frac{B_2(x - [x])}{6} f''(x) dx.$$

In [17]:

```
1 # すぐ上の公式を検証
2
3 PowerSum(m, n) = sum(j→j^m, 1:n)
4 BernoulliNumber(n) = sympy.bernoulli(n)
5 BernoulliPolynomial(n,x) = sympy.bernoulli(n,x)
6
7 function EulerMaclaurinSum(f, a, b, n)
8     x = symbols("x", real=true)
9     (
10         integrate(f(x), (x, a, b))
11         + (f(a)+f(b))/Sym(2)
12         + sum(k → (
13             BernoulliNumber(k)/factorial(Sym(k))
14             * (diff(f(x), x, k-1)(x⇒b) - diff(f(x), x, k-1)(x⇒a))
15             ), 2:n)
16     )
17 end
18
19 function EulerMaclaurinRemainder(f, a, b, n)
20     x = symbols("x", real=true)
21     g = diff(f(x), x, n)
22     (-1)^(n-1) * sum(k → (
23         integrate(BernoulliPolynomial(n,x)*g(x⇒x+k), (x,0,1))
24         ), a:b-1)/factorial(Sym(n))
25 end
26
27 [PowerSum(m, 10) for m in 1:10] ▷ display
28
29 [EulerMaclaurinSum(x→x^m, 1, 10, m+1) for m in 1:10] ▷ display
30
31 [
32     EulerMaclaurinSum(x→x^m, 1, 10, m-1) + EulerMaclaurinRemainder(x→x^m, 1, 10, m-1)
33     for m in 3:10
34 ] ▷ display
35
36 [
37     EulerMaclaurinSum(x→x^m, 1, 10, m-2) + EulerMaclaurinRemainder(x→x^m, 1, 10, m-2)
38     for m in 4:10
39 ] ▷ display
40
41 [
42     EulerMaclaurinSum(x→x^m, 1, 10, m-3) + EulerMaclaurinRemainder(x→x^m, 1, 10, m-3)
43     for m in 5:10
44 ] ▷ display
```

10-element Vector{Int64}:

```
55
385
3025
25333
220825
1978405
18080425
167731333
1574304985
14914341925
```

```
[
55
385
3025
25333
220825
1978405
18080425
167731333
1574304985
14914341925
]
```


$$\begin{bmatrix} 3025 \\ 25333 \\ 220825 \\ 1978405 \\ 18080425 \\ 167731333 \\ 1574304985 \\ 14914341925 \end{bmatrix}$$

$$\begin{bmatrix} 25333 \\ 220825 \\ 1978405 \\ 18080425 \\ 167731333 \\ 1574304985 \\ 14914341925 \end{bmatrix}$$

$$\begin{bmatrix} 220825 \\ 1978405 \\ 18080425 \\ 167731333 \\ 1574304985 \\ 14914341925 \end{bmatrix}$$

2.3 Euler-Maclaurinの和公式の形式的導出

函数 $f(x)$ に対して, ある函数 $F(x)$ で

$$F(x+1) - F(x) = f(x+h)$$

という条件を満たすものを求める問題を考える. そのとき, $D = \frac{\partial}{\partial x}$ とおくと, 形式的にその条件は

$$(e^D - 1)F(x) = e^{hD} f(x) = De^{hD} \int f(x) dx$$

と書き直される. これより, 形式的には

$$F(x) = \frac{De^{hD}}{e^D - 1} \int f(x) dx = \sum_{k=0}^{\infty} \frac{B_k(h)}{k!} D^k \int f(x) dx = \int f(x) dx + \sum_{k=1}^{\infty} \frac{B_k(h)}{k!} f^{(k-1)}(x).$$

これより, 整数 $a < b$ について, 形式的には

$$\sum_{j=a}^{b-1} f(j+h) = F(b) - F(a) = \int_a^b f(x) dx + \sum_{k=1}^{\infty} \frac{B_k(h)}{k!} (f^{(k-1)}(b) - f^{(k-1)}(a)).$$

これは $h = 0$ とおけば形式的にEuler-Maclaurinの和公式

$$\sum_{j=a}^{b-1} f(j) = F(b) - F(a) = \int_a^b f(x) dx + \sum_{k=1}^{\infty} \frac{B_k}{k!} (f^{(k-1)}(b) - f^{(k-1)}(a)).$$

に一致する.

2.4 べき乗和(再)

$f^{(m+1)}(x) = 0$ ならばEuler-Maclaurinの和公式における R_m が消えて, $f^{(m)}(x)$ が定数函数になるので,

$$\sum_{j=a}^b f(j) = \int_a^b f(x) dx + \frac{f(a) + f(b)}{2} + \sum_{k=2}^{m+1} \frac{B_k}{k!} (f^{(k-1)}(b) - f^{(k-1)}(a))$$

となる. m を正の整数であるとし, $f(x) = x^m$, $a = 0$, $b = n$ とすると,

$$(x^m)^{(k-1)} = m(m-1) \cdots (m-k+2)x^{m-k+1} = \frac{k!}{m+1} \binom{m+1}{k} x^{m+1-k}$$

なので, $B_0(1) = B_0 = 1$, $B_1(1) = B_1 + 1 = \frac{1}{2}$, $B_n(1) = B_n + \delta_{n,1}$ も使うと,

$$\begin{aligned} \sum_{j=1}^n j^m &= \int_0^n x^m dx + \frac{n^m}{2} + \sum_{k=2}^m \frac{B_k}{k!} \frac{k!}{m+1} \binom{m+1}{k} x^{m+1-k} \\ &= \frac{n^{m+1}}{m+1} + \frac{n^m}{2} + \frac{1}{m+1} \sum_{k=2}^m \binom{m+1}{k} B_k n^{m+1-k} \\ &= \frac{1}{m+1} \sum_{k=0}^m \binom{m+1}{k} B_k(1) n^{m+1-k} \\ &= \frac{B_{m+1}(x) - B_{m+1}}{m+1}. \end{aligned}$$

最後の等号で

$$B_{m+1}(x+h) = \sum_{k=0}^{m+1} \binom{m+1}{k} B_k(h) x^{m+1-k}$$

を使った. 以上の結果は前節で得た結果と一致する. \square

2.5 Stirlingの近似公式

$\log n! = \log 1 + \log 2 + \dots + \log n$ の $\log x$ の積分による近似をEuler-Maclaurinの和公式を使って精密化すると, Stirlingの近似公式

$$\log n! = n \log n - n + \frac{1}{2} \log n + \log \sqrt{2\pi} + o(1)$$

の精密化が得られることを説明しよう.

$f(x) = \log x$ にStirlingの和公式を適用すると, $k = 1, 2, 3, \dots$ に対して,

$$f^{(k)}(x) = \frac{(-1)^{k-1} (k-1)!}{x^k}.$$

K は3以上の奇数とする. 3以上の奇数 k について $B_k = 0$ なので, $N > n$ のとき,

$$\begin{aligned} \log n! &= \log N! + \log n - \sum_{j=n}^N \log j \\ &= \log N! + \log n - \left(\int_n^N \log x dx + \frac{\log n + \log N}{2} + \sum_{k=2}^{K-1} \frac{B_k}{k(k-1)} \left(\frac{1}{N^{k-1}} - \frac{1}{n^{k-1}} \right) + R_{K,N} \right) \\ &= \log N! - \left(N \log N - N + \frac{1}{2} \log N \right) - \sum_{k=2}^{K-1} \frac{B_k}{k(k-1)} \frac{1}{N^{k-1}} \\ &\quad + n \log n - n + \frac{1}{2} \log n + \sum_{k=2}^{K-1} \frac{B_k}{k(k-1)} \frac{1}{n^{k-1}} + R_{K,N}, \\ R_{K,N} &= (-1)^{K-1} \int_n^N \frac{\tilde{B}_K(x)}{K} \frac{(-1)^{K-1}}{x^K} dx \end{aligned}$$

ただし, $\tilde{B}_n(x) = B_n(\lfloor x \rfloor)$ とおいた.

ここでは, $N \rightarrow \infty$ のとき

$$\log N! - \left(N \log N - N + \frac{1}{2} \log N \right) \rightarrow \sqrt{2\pi}$$

となることは既知であるものとする. 例えば, ノート「[10 Gauss積分, ガンマ関数, ベータ関数](http://nbviewer.jupyter.org/github/genkuroki/Calculus/blob/master/10%20Gauss%2C%20Gamma%2C%20Beta.ipynb)」
「[12 Fourier解析](http://nbviewer.jupyter.org/github/genkuroki/Calculus/blob/master/12%20Fourier%20analysis.ipynb)」のStirlingの近似公式の節を参照して欲しい. 以下ではそれらのノートよりも精密な結果を得る.

このとき, 上の結果で $N \rightarrow \infty$ とすると,

$$\log n! = n \log n - n + \frac{1}{2} \log n + \log \sqrt{2\pi} + \sum_{k=2}^{K-1} \frac{B_k}{k(k-1)} \frac{1}{n^{k-1}} + R_K,$$

$$R_K = (-1)^{K-1} \int_n^\infty \frac{\tilde{B}_K(x)}{K} \frac{(-1)^{K-1}}{x^K} dx = O\left(\frac{1}{n^{K-1}}\right).$$

$K = 2L + 1$ とおくことによって次が得られる: 正の整数 L に対して,

$$\log n! = n \log n - n + \frac{1}{2} \log n + \log \sqrt{2\pi} + \sum_{l=1}^L \frac{B_{2l}}{(2l)(2l-1)} \frac{1}{n^{2l-1}} + O\left(\frac{1}{n^{2L}}\right).$$

これが求めていた結果である.

例えば, $L = 2$ のとき, $B_2 = \frac{1}{6}$, $B_4 = -\frac{1}{30}$ なので,

$$\log n! = n \log n - n + \frac{1}{2} \log n + \log \sqrt{2\pi} + \frac{1}{12n} - \frac{1}{360n^3} + O\left(\frac{1}{n^4}\right).$$

In [18]:

```
1 x = symbols("x")
2 series(exp(x/12-x^3/360), x, n=4)
```

Out[18]: $1 + \frac{x}{12} + \frac{x^2}{288} - \frac{139x^3}{51840} + O(x^4)$

2.6 Poissonの和公式とEuler-Maclaurinの和公式の関係

Poissonの和公式とは, 急減少関数 $f(x)$ に対して,

$$\sum_{m \in \mathbb{Z}} f(m) = \sum_{n \in \mathbb{Z}} \hat{f}(n), \quad \hat{f}(p) = \int_{\mathbb{R}} f(x) e^{2\pi i p x} dx$$

が成立するという結果であった. これの右辺は以下のように変形できる:

$$\begin{aligned} \sum_{n \in \mathbb{Z}} \hat{f}(n) &= \sum_{n \in \mathbb{Z}} \int_{\mathbb{R}} f(x) e^{2\pi i n x} dx = \int_{\mathbb{R}} f(x) dx + 2 \sum_{n=1}^{\infty} \int_{\mathbb{R}} f(x) \cos(2\pi n x) dx \\ &= \int_{\mathbb{R}} f(x) dx - \sum_{n=1}^{\infty} \int_{\mathbb{R}} f'(x) \frac{\sin(2\pi n x)}{\pi n} dx \\ &= \int_{\mathbb{R}} f(x) dx + \int_{\mathbb{R}} \left(- \sum_{n=1}^{\infty} \frac{\sin(2\pi n x)}{\pi n} \right) f'(x) dx \end{aligned}$$

2つ目の等号では $e^{2\pi i n x} + e^{-2\pi i n x} = 2 \cos(2\pi n x)$ を用い, 3つ目の等号では部分積分を実行し, 4つ目の等号では無限和と積分の順序を交換した. それらの操作は $f(x)$ が急減少関数であれば容易に正当化される.

一方, Euler-Maclaurinの和公式の

$$B_1(x - [x]) = x - [x] - \frac{1}{2}$$

を使う場合から,

$$\sum_{m \in \mathbb{Z}} f(m) = \int_{\mathbb{R}} f(x) dx + \int_{\mathbb{R}} \left(x - [x] - \frac{1}{2} \right) f'(x) dx$$

が導かれる. これは部分積分によって得られる次の公式からただちに導かれる易しい公式であることにも注意せよ:

$$\begin{aligned} \int_n^{n+1} \left(x - n - \frac{1}{2} \right) f'(x) dx &= \left[\left(x - n - \frac{1}{2} \right) f(x) \right]_n^{n+1} - \int_n^{n+1} f(x) dx - n \\ &= \frac{f(n+1) - f(n)}{2} - \int_n^{n+1} f(x) dx. \end{aligned}$$

以上の2つの結果を比較すると, Poissonの和公式とEuler-Maclaurinの和公式の $B_1(x - [x])$ を使った場合は,

$$x - [x] - \frac{1}{2} = - \sum_{n=1}^{\infty} \frac{\sin(2\pi n x)}{\pi n} \quad (*)$$

という公式で結び付いていることがわかる. この公式を認めれば, Euler-Maclaurinの和公式の $B_1(x - [x])$ を使った場合から Poissonの和公式が導かれる.

公式(*)の左辺はいわゆるのこぎり波であり, 右辺はそのFourier級数である. 公式(*)はFourier級数論における非常に有名な公式であり, 本質的にそれと同じ公式はFourier級数論について書かれた文献には例として必ず載っていると見てよいくらいである. (Fourier級数論より, 公式(*)は x が整数でないときには実際に成立していることがわかる.)

このように, のこぎり波のFourier級数展開という非常に特殊な公式はPoissonの和公式という一般的な公式を導くだけの力を持っているのである.

まとめ: のこぎり波のFourier級数展開は部分積分を通してPoissonの和公式と本質的に同値である! □

この節で解説したことは次の文献で指摘されている:

- Tim Jameson, [An elementary derivation of the Poisson summation formula](http://www.maths.lancs.ac.uk/jameson/poisson.pdf) (<http://www.maths.lancs.ac.uk/jameson/poisson.pdf>)

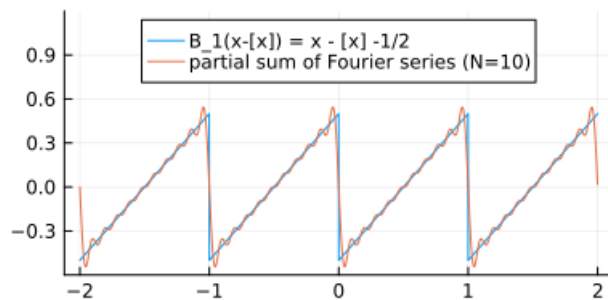
In [19]: ▶

```

1 B_1(x) = x - 1/2
2 b(x) = B_1(x - floor(x))
3 S(N,x) = -sum(n->sin(2*pi*n*x)/(pi*n), 1:N)
4 x = -2:0.001:1.999
5 N = 10
6 plot(size=(400,200), ylim=(-0.6,1.2), legend=:top)
7 plot!(x, b.(x), label="B_1(x-[x]) = x - [x] - 1/2")
8 plot!(x, S.(N,x), label="partial sum of Fourier series (N=$N)")

```

Out[19]:



補足: このノートの上の方の周期的Bernoulli多項式 $B_k(x - [x])$ のFourier級数展開の節を見ればわかるように,

$$\sum_{n=1}^{\infty} \frac{\cos(2\pi nx)}{n^k}, \quad \sum_{n=1}^{\infty} \frac{\sin(2\pi nx)}{n^k}$$

の型のFourier級数の収束先は平行移動と定数倍の違いを除いて周期的Bernoulli多項式になる. □

2.7 台形公式とPoissonの和公式の関係

簡単のため $f(x)$ は \mathbb{R} 上の急減少関数であるとし, $a, b \in \mathbb{Z}$ かつ $a < b$ であると仮定する. このとき,

$$\left(\frac{\sin(2\pi nx)}{\pi n} \right)' = 2 \cos(2\pi nx)$$

を使った部分積分とノコギリ波のFourier級数展開

$$-\sum_{n=1}^{\infty} \frac{\sin(2\pi nx)}{\pi n} = x - [x] - \frac{1}{2} = B_1(x - [x])$$

より,

$$\begin{aligned}
\sum_{n \in \mathbb{Z}} \int_a^b f(x) e^{-2\pi i n x} dx &= \int_a^b f(x) dx + \sum_{n=1}^{\infty} 2 \int_a^b f(x) \cos(2\pi n x) dx \\
&= \int_a^b f(x) dx - \sum_{n=1}^{\infty} \int_a^b f'(x) \frac{\sin(2\pi n x)}{\pi n} dx \\
&= \int_a^b f(x) dx + \int_a^b f'(x) \left(x - [x] - \frac{1}{2} \right) dx \\
&= \int_a^b f(x) dx + \sum_{m=a}^{b-1} \int_m^{m+1} f'(x) \left(x - [x] - \frac{1}{2} \right) dx \\
&= \int_a^b f(x) dx + \sum_{m=a}^{b-1} \int_0^1 f'(t+m) \left(t - \frac{1}{2} \right) dt \\
&= \int_a^b f(x) dx + \sum_{m=a}^{b-1} \left(\left[f(t+m) \left(t - \frac{1}{2} \right) \right]_0^1 - \int_0^1 f(t+m) dt \right) \\
&= \sum_{m=a}^{b-1} \frac{f(m) + f(m+1)}{2} \\
&= \sum_{m=a}^b f(m) - \frac{f(a) + f(b)}{2}.
\end{aligned}$$

要するに、ノコギリ波 $B_1(x - [x])$ のFourier級数展開の公式を認めて使えば、

$$\sum_{n \in \mathbb{Z}} \int_a^b f(x) e^{-2\pi i n x} dx = \sum_{m=a}^{b-1} \frac{f(m) + f(m+1)}{2} = \sum_{m=a}^b f(m) - \frac{f(a) + f(b)}{2}$$

という公式が得られる。この公式の右辺は積分の近似計算で使われる台形公式に一致しており、途中の計算では $B_1(x - [x])$ を

3 ゼータ関数への応用

$s > 1$ のとき(より一般には $\operatorname{Re} s > 1$ のとき)、

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

は絶対収束しているのであった。これにEuler-Maclaurinの和公式

$$\begin{aligned}
\sum_{j=a}^b f(j) &= \int_a^b f(x) dx + \frac{f(a) + f(b)}{2} + \sum_{k=2}^n \frac{B_k}{k!} (f^{(k-1)}(b) - f^{(k-1)}(a)) + R_n, \\
R_n &= (-1)^{n-1} \int_a^b \frac{B_n(x - [x])}{n!} f^{(n)}(x) dx
\end{aligned}$$

を適用してみよう。

3.1 解析接続

$\operatorname{Re} s > 1$ であるとし、 $f(x) = x^{-s}$ とおく。このとき、

$$\begin{aligned}
\int_a^{\infty} f(x) dx &= \int_1^{\infty} x^{-s} dx = \left[\frac{x^{-s+1}}{-s+1} \right]_1^{\infty} = \frac{a^{-(s-1)}}{s-1}, \quad f(b) = b^{-s} \rightarrow 0 \quad (b \rightarrow \infty). \\
\frac{B_k}{k!} f^{(k-1)}(x) &= \frac{B_k}{k} \binom{-s}{k-1} x^{-s-k+1}, \quad \frac{B_n(x - [x])}{n!} f^{(n)}(x) = \binom{-s}{n} B_n(x - [x]) x^{-s-n}
\end{aligned}$$

なので、2以上の整数 n について、

$$\begin{aligned}
\zeta(s) &= \frac{1}{s-1} + \frac{1}{2} - \sum_{k=2}^n \frac{B_k}{k} \binom{-s}{k-1} + R_n, \\
R_n &= (-1)^{n-1} \binom{-s}{n} \int_1^{\infty} B_n(x - [x]) x^{-s-n} dx.
\end{aligned}$$

積分 R_n は $\operatorname{Re} s + n > 1$ ならば絶対収束している。ゆえに、複素平面全体に $\zeta(s)$ を自然に拡張する方法(解析接続する方法)が得られた。

$\sum_{k=1}^{\infty} \frac{1}{n^s}$ そのものではなく、 $n = a$ から始まる無限和 $\sum_{k=a}^{\infty} \frac{1}{n^s} = \zeta(s) - \sum_{n=1}^{a-1} \frac{1}{n^s}$ (Euler-Maclaurinの和公式を適用すると、

$$\zeta(s) = \sum_{n=1}^{a-1} \frac{1}{n^s} - \frac{a^{1-s}}{1-s} + \frac{1}{2a^s} - \sum_{k=2}^n \frac{B_k}{ka^{s+k-1}} \binom{-s}{k-1} + R_{n,a},$$

$$R_{n,a} = (-1)^{n-1} \binom{-s}{n} \int_a^{\infty} B_n(x - [x]) x^{-s-n} dx.$$

In [20]: ▶

```

1 # 上の公式における ζ(s) - R_{n,a} の函数化
2
3 # ζ(s) - R_{n,a} = Σ_{m=1}^{a-1} m^{-s} - a^{1-s}/(1-s) + 1/(2a^s)
4 # - Σ_{k=2}^n B_k/(k a^{s+k-1}) binom(-s,k-1) (k is even)
5 #
6 function ApproxZeta(a, n, s)
7     ss = float(big(s))
8     z = zero(ss)
9     z += (a ≤ 1 ? zero(ss) : sum(m→m^{-ss}, 1:a-1)) # Σ_{m=1}^{a-1} m^{-s}
10    z += -a^{1-ss}/(1-ss) # -a^{1-s}/(1-s)
11    n == 0 && return z
12    z += 1/(2*a^ss) # 1/(2a^s)
13    n == 1 && return z
14    z -= sum(k → BB(k)/(k*a^{ss+k-1})*binom(-ss,k-1), 2:2:n)
15           # -Σ_{k=2}^n B_k/(k a^{s+k-1}) binom(-s,k-1) (k is even)
16 end
17
18 A = ApproxZeta(40, 80, big"0.5")
19 Z = zeta(big"0.5")
20 @show A
21 @show Z;

```

A = -1.460354508809586812889499152515298012467229331012581490542886087825530529474572
Z = -1.460354508809586812889499152515298012467229331012581490542886087825530529474503

Re $s > 0$ のとき、

$$\frac{1}{2a^s} - \sum_{k=2}^n \frac{B_k}{ka^{s+k-1}} \binom{-s}{k-1} + R_{n,a}$$

は $a \rightarrow \infty$ で 0 に収束するので、

$$\zeta(s) = \lim_{a \rightarrow \infty} \left(\sum_{n=1}^{a-1} \frac{1}{n^s} - \frac{a^{1-s}}{1-s} \right) \quad (\text{Re } s > 0)$$

が成立することがわかる。これは、Dirichlet級数の部分 and $\sum_{n=1}^{a-1} \frac{1}{n^s}$ から補正項

$$\frac{a^{1-s}}{1-s}$$

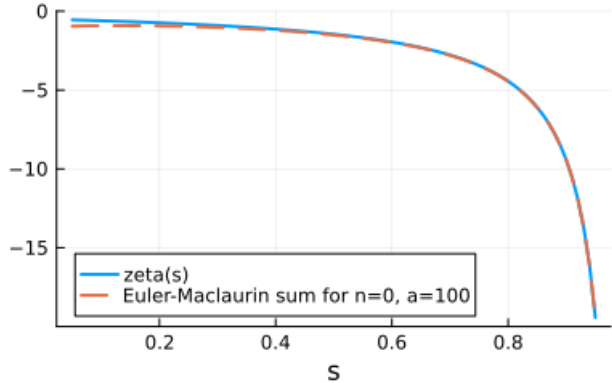
を引き去ってから、Dirichlet級数の総和を取れば、 $0 < \text{Re } s < 1$ でも収束して、 $\zeta(s)$ の正確な値が得られることを意味している。

In [21]:

```
1 # 上の結果のプロット
2
3 ApproxZeta0(a, s) = sum(n->n^(-s), 1:a-1) - a^(1-s)/(1-s)
4 a = 100
5 s = 0.05:0.01:0.95
6 @time z = zeta.(s)
7 @time w = ApproxZeta0.(a, s)
8 plot(size=(400, 250), legend=:bottomleft, xlabel="s")
9 plot!(s, z, label="zeta(s)", lw=2)
10 plot!(s, w, label="Euler-Maclaurin sum for n=0, a=$a", lw=2, ls=:dash)
```

0.191314 seconds (143.69 k allocations: 9.440 MiB, 7.58% gc time, 44.93% compilation time)
0.055798 seconds (117.52 k allocations: 8.121 MiB, 98.91% compilation time)

Out[21]:

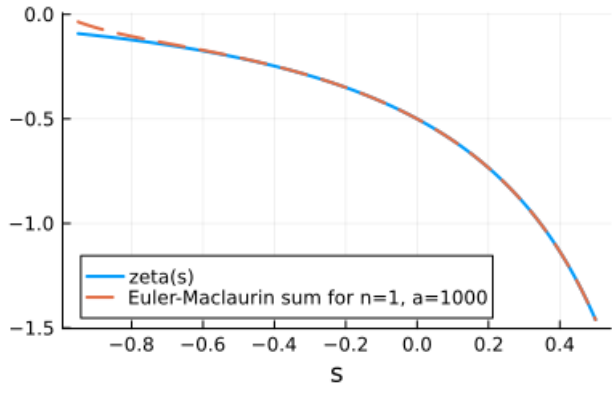


In [22]:

```
1 # さらに項の数を1つ増やした場合のプロット
2
3 # ζ(s) - R_{1,a} = Σ_{n=1}^{a-1} n^{-s} - a^{1-s}/(1-s) + 1/(2a^s)
4 #
5 ApproxZeta1(a, s) = sum(n->n^(-s), 1:a-1) - a^(1-s)/(1-s) + 1/(2*a^s)
6
7 s = -0.95:0.01:0.5
8 a = 10^3
9 @time z = zeta.(s)
10 @time w = ApproxZeta1.(a, s)
11 plot(size=(400, 250), legend=:bottomleft, xlabel="s")
12 plot!(s, z, label="zeta(s)", lw=2)
13 plot!(s, w, label="Euler-Maclaurin sum for n=1, a=$a", lw=2, ls=:dash)
```

0.000062 seconds (4 allocations: 1.406 KiB)
0.062979 seconds (114.81 k allocations: 7.860 MiB, 94.18% compilation time)

Out[22]:



```

In [23]: ▶ 1 # さらに一般の場合のプロット
2 #
3 # Euler-Maclaurinの和公式で  $\zeta(s)$  の負の  $s$  での値をびったり近似できていることがわかる。
4
5 [(-m, zeta(-m), Float64(ApproxZeta(2, 17, -m))) for m = 0:12] ▶ display
6
7 n = 10
8 s = -1.5:0.05:0.5
9 a = 10
10 @time z = zeta.(s)
11 @time w = ApproxZeta.(a, n, s)
12 P1 = plot(size=(400, 250), legend=:bottomleft, xlabel="s")
13 plot!(s, z, label="zeta(s)", lw=2)
14 plot!(s, w, label="Euler-Maclaurin sum for a=$a, n=$n", lw=2, ls=:dash)
15
16 n = 17
17 s = -16:0.05:-2.0
18 a = 2
19 @time z = zeta.(s)
20 @time w = ApproxZeta.(a, n, s)
21 P2 = plot(size=(400, 250), legend=:topright, xlabel="s")
22 plot!(s, z, label="zeta(s)", lw=2)
23 plot!(s, w, label="Euler-Maclaurin sum for a=$a, n=$n", lw=2, ls=:dash)

```

13-element Vector{Tuple{Int64, Float64, Float64}}:

```

(0, -0.5, -0.5)
(-1, -0.08333333333333338, -0.08333333333333333)
(-2, -0.0, -1.2954252832641667e-77)
(-3, 0.008333333333333344, 0.008333333333333333)
(-4, -0.0, -3.454467422037778e-77)
(-5, -0.0039682539682539715, -0.003968253968253968)
(-6, -0.0, 0.0)
(-7, 0.004166666666666669, 0.004166666666666667)
(-8, -0.0, 0.0)
(-9, -0.0075757575757575758, -0.007575757575757576)
(-10, -0.0, -4.421718300208356e-75)
(-11, 0.0210927960927961, 0.021092796092796094)
(-12, -0.0, 0.0)

```

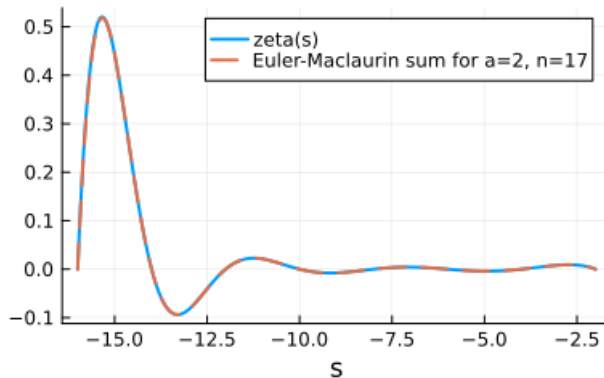
0.000034 seconds (4 allocations: 592 bytes)

0.095275 seconds (216.85 k allocations: 13.766 MiB, 91.64% compilation time)

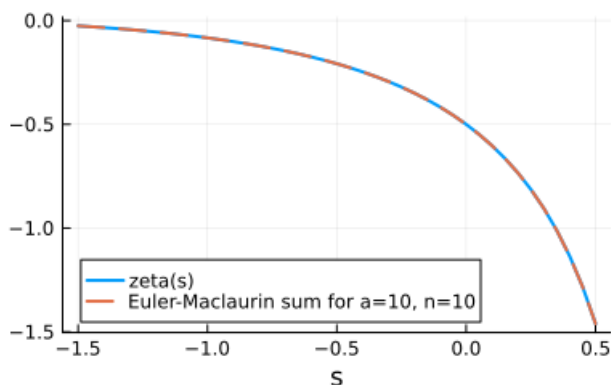
0.000105 seconds (4 allocations: 2.562 KiB)

0.053619 seconds (492.29 k allocations: 19.944 MiB, 27.20% gc time)

Out[23]:

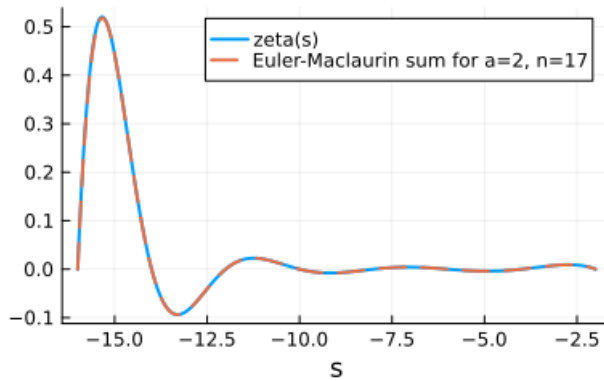


In [24]: ▶ 1 display(P1)



上と下のグラフを見ればわかるように, Euler-Maclaurinの和公式によって負の実数での ζ 関数の値を非常によく近似できている. 実は $\zeta(s)$ を実部が負の複素数まで拡張してもこの近似はうまく行っている.

In [25]: `display(P2)`



3.2 $\zeta(2)$ の近似計算

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

Eulerは $\zeta(2)$ の近似値を自ら開発したEuler-Maclaurinの和公式を使って精密に計算したらしい.

近似式

$$\zeta(s) \approx \sum_{n=1}^{a-1} \frac{1}{n^s} - \frac{a^{1-s}}{1-s} + \frac{1}{2a^s} - \sum_{k=2}^n \frac{B_k}{ka^{s+k-1}} \binom{-s}{k-1}$$

を用いて, $\zeta(2)$ を計算してみよう. 3以上の奇数 n について $B_n = 0$ となるので, $n = 2m$ のとき, 右辺の項数は $a + m + 1$ になる.

例えば, $a = 10, m = 9$ とし, 20項の和を取ると,

$$\zeta(2) \approx 1.64493\ 40668\ 4749 \dots$$

となり, 正確な値 $\frac{\pi^2}{6} = 1.64493\ 40668\ 4822 \dots$ と小数点以下第11桁まで一致している.

Eulerは後に $\zeta(2) = \frac{\pi^2}{6}$ を得る. Eulerは競争相手に議論に厳密性に欠けるとして様々な批判を受けたのだが, 以上のような数値計算の結果を知っていたので, 正解を得たという確信は微塵も揺らかなかっただろうと思われる.

注意: 論理的に厳密な証明の方法が発達した現代においても, 人間は常に証明を間違える可能性がある. 人間が行った証明は絶対的には信用できない. だから, たとえ証明が完成したと思っていたとしても, 可能ならば数値計算によって論理的に厳密な証明以外の証拠を作っていた方が安全だと思われる. \square

注意: 数学のノートを作りながら, 気軽に数値的証拠も同時に得るための道具として, 筆者がこのノート作成のために用いている [Julia言語](https://julia-lang.org/) (<https://julia-lang.org/>)と [Jupyter](http://jupyter.org/) (<http://jupyter.org/>)と [Nbextensions](https://github.com/ipython-contrib/jupyter_contrib_nbextensions) (https://github.com/ipython-contrib/jupyter_contrib_nbextensions)のLive Markdown Previewはこれを書いている時点で相当に優秀な道具であるように思われる. \square

In [26]:

```

1 # 20項 の和
2
3 N = 20
4 [(m, N-m-1, 2m, ApproxZeta(N-m-1, 2m, 2) - big(π)^2/6) for m in 2:N÷2-1] ▷ display
5
6 m = 9
7 a = N-m-1
8 Z = big(π)^2/6
9 A = ApproxZeta(a, m, 2)
10 @show a,m
11 @show Z
12 @show A;

```

```

8-element Vector{Tuple{Int64, Int64, Int64, BigFloat}}:
 (2, 17, 4, -5.7745179386347483379778894047835850369958540757839935768100161932366414526175
 8e-11)
 (3, 16, 6, 4.80812735239562509501346011215032587838986605495815313740843048777477650932482
 9e-13)
 (4, 15, 8, -8.6308875139430442246152364654652069706509110461365277080262921867238658169664
 92e-15)
 (5, 14, 10, 3.1162179785273853280542355730238711734665867971863964368976627204145528522974
 51e-16)
 (6, 13, 12, -2.200847274100542514575619216657515798396053843860275532661691594239630209744
 406e-17)
 (7, 12, 14, 3.035248943857815147776773837113166940196569351033194321813552488718204060625
 84e-18)
 (8, 11, 16, -8.335321043122531064769674746337938450627967961329547742403411230422546148897
 753e-19)
 (9, 10, 18, 4.7466018143920053127140275780279703065395409350513421647372241615147960630210
 67e-19)

(a, m) = (10, 9)
Z = 1.644934066848226436472415166646025189218949901206798437735558229370007470403185
A = 1.644934066847493071302595112118921642731166540690350214159737969261778785588307

```

3.3 $s = 1$ での $\zeta(s)$ の定数項がEuler定数になること

$\zeta(s) = \sum_{n=1}^{\infty} \frac{1}{n^s}$ にEuler-Maclaurinの和公式を使って、2以上の n について次の公式が得られるのであった:

$$\zeta(s) = \frac{1}{s-1} + \frac{1}{2} - \sum_{k=2}^n \frac{B_k}{k} \binom{-s}{k-1} + R_n,$$

$$R_n = (-1)^{n-1} \binom{-s}{n} \int_1^{\infty} B_n(x - [x])x^{-s-n} dx.$$

$n = 1$ の場合には

$$\begin{aligned} \sum_{j=a}^b f(j) &= \int_a^b f(x) dx + f(a) + \int_a^b (x - [x])f'(x) dx \\ &= \int_a^b f(x) dx + f(a) + \sum_{j=a}^{b-1} \int_0^1 x f'(x+j) dx \end{aligned}$$

を $f(x) = x^{-s}$, $f'(x) = -sx^{-s-1}$, $a = 1$, $b = \infty$ の場合に適用して、

$$\zeta(s) = \frac{1}{s-1} + 1 - s \sum_{j=1}^{\infty} \int_0^1 \frac{x}{(x+j)^{s+1}} dx$$

を得る。したがって、

$$\lim_{s \rightarrow 1} \left(\zeta(s) - \frac{1}{s-1} \right) = 1 - \sum_{j=1}^{\infty} \int_0^1 \frac{x}{(x+j)^2} dx.$$

そして、 $x = t - j$ と置換すると、

$$\begin{aligned} - \int_0^1 \frac{x}{(x+j)^2} dx &= - \int_j^{j+1} \frac{-(t-j)}{t^2} dt = - \left[\log t + \frac{j}{t} \right]_j^{j+1} \\ &= -\log(j+1) + \log j - \frac{j}{j+1} + 1 = \frac{1}{j+1} + \log j - \log(j+1) \end{aligned}$$

なので、これを $j = 1$ から $j = N - 1$ まで足し上げることによって、

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

この $N \rightarrow \infty$ での極限はEuler定数 $\gamma = 0.5772 \dots$ の定義であった. 以上によって次が示された:

$$\lim_{s \rightarrow 1} \left(\zeta(s) - \frac{1}{s-1} \right) = \gamma = 0.5772 \dots.$$

3.4 負の整数におけるゼータ函数の特殊値の計算

Euler-Maclaurinの和公式: 3以上の整数 k について $B_k = 0$ なので, 以下の公式で k は偶数のみを動くとしてよい:

$$\sum_{n=a}^b f(n) = \int_a^b f(x) dx + \frac{f(a) + f(b)}{2} + \sum_{k=2}^m \frac{B_k}{k!} (f^{(k-1)}(b) - f^{(k-1)}(a)) + R_m,$$

$$R_m = (-1)^{m-1} \int_a^b \frac{\tilde{B}_m(x)}{m!} f^{(m)}(x) dx.$$

ここで $\tilde{B}_m(x) = B_m(x - [x])$ とおいた.

Euler-Maclaurinの和公式を $f(x) = n^{-s}$, $a = 1$, $b = \infty$ の場合に適用することによって $\zeta(s)$ は次の形で $\Re s > 1 - m$ まで自然に延長(解析接続)されるのであった:

$$\zeta(s) = \frac{1}{s-1} + \frac{1}{2} - \frac{1}{1-s} \sum_{k=2}^m \binom{1-s}{k} B_k + (-1)^{m-1} \int_a^b \binom{-s}{m} \tilde{B}_m(x) x^{-s-m} dx.$$

この公式と $k \geq 2$ のとき $\binom{1}{k} = 0$ となることより,

$$\zeta(0) = \frac{1}{0-1} + \frac{1}{2} = -\frac{1}{2}.$$

r は正の整数であるとする. このとき, $m > r$ とすると $\binom{r}{m} = 0$ となるので, $B_0 = 1$, $B_1 = -1/2$ なので,

$$\begin{aligned} \zeta(-r) &= -\frac{1}{r+1} + \frac{1}{2} - \frac{1}{r+1} \sum_{k=2}^{r+1} \binom{m+1}{k} B_k \\ &= -\frac{1}{r+1} \sum_{k=0}^{r+1} \binom{m+1}{k} B_k = -\frac{B_{r+1}}{r+1}. \end{aligned}$$

最後の等号で, Bernoulli数を帰納的に計算するために使える公式 $\sum_{k=0}^r \binom{r+1}{k} B_k = 0$ を用いた. 例えば, $r = 1$ のとき $B_0 + 2B_1 = 1 + 2(-1/2) = 0$ となり, $r = 2$ のとき, $B_0 + 3B_1 + 3B_2 = 1 + 3(-1/2) + 3(1/6) = 0$ となる.

以上によって次が証明された:

$$\zeta(0) = -\frac{1}{2}, \quad \zeta(-r) = -\frac{B_{r+1}}{r+1} \quad (r = 1, 2, 3, \dots).$$

これらの公式は $B_n(1) = B_n + \delta_{n,1}$, $B_1 = -1/2$ を使うと,

$$\zeta(-r) = -\frac{B_{r+1}(1)}{r+1} \quad (r = 0, 1, 2, \dots)$$

の形にまとめられる.

3.5 発散級数の有限部分と $\zeta(-r)$ の関係

前節の結果 $\zeta(-r) = -\frac{B_{r+1}(1)}{r+1}$ ($r = 0, 1, 2, \dots$) は

$$1 + 1 + 1 + 1 + \dots = -\frac{1}{2},$$

$$1 + 2 + 3 + 4 + \dots = -\frac{1}{12}$$

のような印象的な形式で書かれることもある. ただし, その場合には左辺が通常の無限和ではなく, ゼータ函数 $\zeta(s)$ の解析接続の意味であることを了解しておかなければいけない.

実はさらに解析接続として理解するだけでなく、「左辺の発散する無限和から適切に無限大を引き去れば右辺に等しくなる」というようなタイプの命題をうまく作ることもできる。以下ではそのことを解説しよう。

以下、 η は非負の実数に値を持つ \mathbb{R} 上の急減少関数であると仮定する。(\mathbb{R} 上の急減少関数とは \mathbb{R} 上の C^∞ 関数でそれ自身およびそのすべての階数の導関数に任意の多項式関数をかけたものが $|x| \rightarrow \infty$ で 0 に収束するものことである。) さらに、

$$\eta(0) = 1, \quad \eta'(0) = 0$$

と仮定する。例えば $\eta(x) = e^{-x^2}$ はそのような関数の例になっている。

このとき、 $\eta(x)$ が急減少関数であることより、 $N > 0$ のとき、級数

$$\sum_{n=1}^{\infty} n^r \eta(n/N) = 1^r \eta(1/N) + 2^r \eta(2/N) + 3^r \eta(3/N) + \dots$$

は常に絶対収束する。 r が非負の整数のとき、 $N \rightarrow \infty$ とすると、この級数は発散級数 $1^r + 2^r + 3^r + \dots$ になってしまう。以下の目標は、Euler-Maclaurinの和公式を使うと、その $N \rightarrow \infty$ での発散部分が $C N^{r+1}$ (C は η と r で具体的に決まる定数) の形にまとまることを示すことである。そして、残った有限部分は常に $\zeta(-r)$ に収束することも示される。

$\tilde{B}_n(x) = B_n(x - [x])$ と書くことにする。

このとき、 $f(x) = \eta(x/N)$ に Euler-Maclaurinの和公式を適用すると、 $f(0) = 1, f'(0) = f(\infty) = f'(\infty) = 0$ より、

$$\begin{aligned} 1 + \sum_{n=1}^{\infty} \eta(x/N) &= \sum_{n=0}^{\infty} \eta(x/N) \\ &= \int_0^{\infty} \eta(x/N) dx + \frac{1}{2} + B_2(f'(\infty) - f'(0)) - \int_0^{\infty} \frac{\tilde{B}_2(x)}{2!} \frac{1}{N^2} \eta''(x/N) dx \\ &= N \int_0^{\infty} \eta(y) dy + \frac{1}{2} - \frac{1}{N} \int_0^{\infty} \frac{\tilde{B}_2(Ny)}{2!} \eta''(y) dy. \end{aligned}$$

ゆえに、 $\zeta(0) = -1/2$ を使うと、

$$\sum_{n=1}^{\infty} \eta(x/N) - N \int_0^{\infty} \eta(y) dy = \zeta(0) + O(1/N).$$

これは $N \rightarrow \infty$ で発散級数 $1 + 1 + 1 + 1 + \dots$ になる無限和 $\sum_{n=1}^{\infty} \eta(x/N)$ から、その発散部分 $N \int_0^{\infty} \eta(y) dy$ を引き去って、 $N \rightarrow \infty$ の極限を取ると、 $\zeta(0)$ に収束することを意味している。これが欲しい結果の1つ目である。

r は正の整数であるとし、 $f(x) = x^r \eta(x/N)$ とおく。そのとき、Leibnitz則

$$(\varphi(x)\psi(x))^{(m)} = \sum_{i=0}^m \binom{m}{i} \varphi^{(i)}(x)\psi^{(m-i)}(x)$$

を使うと、

$$f^{(r+2)}(x) = \frac{1}{N^2} F(x/N), \quad F(y) = \binom{r+2}{0} y^r \eta^{(r+2)}(y) + \dots + \binom{r+2}{r} r! \eta(y)$$

その $f(x)$ に Euler-Maclaurinの和公式を適用すると、 $f^{(k)}(\infty) = f^{(k)}(\infty) = 0$ および、

$$f(0) = f'(0) = \dots = f^{(r-1)}(0) = f^{(r+1)}(0) = 0, \quad f^{(r)}(0) = r!$$

より、

$$\begin{aligned} \sum_{n=1}^{\infty} n^r \eta(n/N) &= \sum_{n=0}^{\infty} f(n) = \int_0^{\infty} f(x) dx - \frac{B_{r+1}}{(r+1)!} r! - \frac{B_{r+2}}{(r+2)!} 0 + (-1)^{r+1} \int_0^{\infty} \frac{\tilde{B}_{r+2}(x)}{(r+2)!} f^{(r+2)}(x) dx \\ &= N^{r+1} \int_0^{\infty} y^r \eta(y) dy - \frac{B_{r+1}}{r+1} + (-1)^{r+1} \frac{1}{N} \int_0^{\infty} \frac{\tilde{B}_{r+2}(Ny)}{(r+2)!} F(y) dy \\ &= N^{r+1} \int_0^{\infty} y^r \eta(y) dy - \frac{B_{r+1}}{r+1} + O(1/N). \end{aligned}$$

ゆえに、 $\zeta(-r) = -\frac{B_{r+1}}{r+1}$ を使うと、

$$\sum_{n=1}^{\infty} n^r \eta(n/N) - N^{r+1} \int_0^{\infty} y^r \eta(y) dy = \zeta(-r) + O(1/N).$$

これは $N \rightarrow \infty$ で発散級数 $1^r + 2^r + 3^r + 4^r + \dots$ になる無限和 $\sum_{n=1}^{\infty} n^r \eta(x/N)$ から、その発散部分 $N^{r+1} \int_0^{\infty} y^r \eta(y) dy$

注意: 以上の計算のポイントは、非負の急減少関数 $\eta(x)$ で $\eta(0) = 1$, $\eta'(0) = 0$ を満たすもので発散級数を正則化して得られる級数の場合には、Euler-Maclaurinの和公式の「途中の項」がほとんど消えてしまうことである。 CN^{r+1} 型の発散項と定数項と $O(1/N)$ の部分の3つの項しか生き残らない。□

注意: 以上の結果に関するより進んだ解説については次のリンク先を参照せよ:

- Terence Tao, [The Euler-Maclaurin formula, Bernoulli numbers, the zeta function, and real-variable analytic continuation](https://terrytao.wordpress.com/2010/04/10/the-euler-maclaurin-formula-bernoulli-numbers-the-zeta-function-and-real-variable-analytic-continuation/) (<https://terrytao.wordpress.com/2010/04/10/the-euler-maclaurin-formula-bernoulli-numbers-the-zeta-function-and-real-variable-analytic-continuation/>), Blog: What's new, 10 April, 2010.

このブログ記事はかなり読み易い。□

問題: 以上の結果を数値計算でも確認してみよ。□

ヒント: $\eta(x) = e^{-x^2}$ の場合を試してみよ。そのとき、

$$\int_0^{\infty} y^r \eta(y) dy = \int_0^{\infty} y^r e^{-y^2} dy = \frac{1}{2} \Gamma\left(\frac{r+1}{2}\right)$$

となっている。□

解答例: 次のリンク先のノートを見よ。

- 黒木玄, [ζ\(s\)の Re s < 1 での様子](http://nbviewer.jupyter.org/gist/genkuroki/8b13fc9c05bfd669c345b940066b896b) (<http://nbviewer.jupyter.org/gist/genkuroki/8b13fc9c05bfd669c345b940066b896b>) □

```
In [27]: ▶ 1 y = symbols("y", real=true)
          2 r = symbols("r", positive=true)
          3 integrate(y^r*exp(-y^2), (y, 0, oo))
```

Out[27]: $\frac{\Gamma\left(\frac{r}{2} + \frac{1}{2}\right)}{2}$