Hurwitzのゼータ函数の0以下の整数での特殊値と微係数

黒木玄

2019-04-01

Hurwitzのゼータ函数とは $\operatorname{Re} s > 1$, $x \ne 0, -1, -2, \dots$ で定義された

$$\zeta(s, x) = \sum_{k=0}^\infty \frac{1}{(x+k)^s} = \frac{1}{x^s} + \frac{1}{(x+1)^s} + \frac{1}{(x+2)^s} + \cdots$$

を解析接続して得られる函数のことである.

このノートブックではHurwitzのゼータ函数の $s = 1 - r$ $(r = 1, 2, 3, \dots)$ における特殊値 $\zeta(1-r, x)$ と偏微分係数 $\zeta_s(1-r, x)$ に関するよく知られている結果を数値的に確認する.

ツイッターでの以下のスレッドも参照せよ:

- https://twitter.com/genkuroki/status/1111938896844095488 (https://twitter.com/genkuroki/status/1111938896844095488)

Julia言語環境の整備の仕方については次のリンク先を参照せよ:

- https://nbviewer.jupyter.org/github/genkuroki/msfd28/blob/master/install.ipynb (https://nbviewer.jupyter.org/github/genkuroki/msfd28/blob/master/install.ipynb)

目次

In [1]:

```
using Plots
gr(); ENV["PLOTS_TEST"] = "true"
gr(bglegend=RGBA(1.0, 1.0, 1.0, 0.5))
using LaTeXStrings
using SpecialFunctions
using SymPy: SymPy, sympy, @vars, oo
using QuadGK
```

In [2]:

```
?SpecialFunctions.zeta
```

Out[2]:

```
zeta(s, z)
```

Generalized zeta function $\zeta(s, z)$, defined by the sum $\sum_{k=0}^\infty ((k+z)^2)^{-s/2}$, where any term with $k + z = 0$ is excluded. For $\Re z > 0$, this definition is equivalent to the Hurwitz zeta function $\sum_{k=0}^\infty (k+z)^{-s}$. For $z = 1$, it yields the Riemann zeta function $\zeta(s)$.

```
zeta(s)
```

Riemann zeta function $\zeta(s)$.

注意: `SpecialFunctions.zeta(s,z)` を $\operatorname{Re} z > 0$ の場合にのみ利用すること.

1 Bernoulli多項式との関係

Bernoulli多項式を $B_r(x)$ と書くと,

$$\frac{ze^{zx}}{e^z - 1} = \sum_{k=0}^\infty B_k(x) \frac{z^k}{k!}, \qquad -r\zeta(1-r, x) = B_r(x) \quad (r = 1, 2, \dots).$$

左側の公式を $B_k(x)$ の定義だと考えてよい.

In [3]:

```julia
B(r, x) = -r*zeta(1-r, x)
Bernoulli(r, x) = sympy.bernoulli(r, x)
fBernoulli(r, x) = float(Bernoulli(r,x))
@vars x
[Bernoulli(r, x) for r in 0:4]
```

Out[3]:

$$\left[\begin{array}{r} 1 \\ x - \frac{1}{2} \\ x^2 - x + \frac{1}{6} \\ x^3 - \frac{3x^2}{2} + \frac{x}{2} \\ x^4 - 2x^3 + x^2 - \frac{1}{30} \end{array}\right]$$

In [4]:

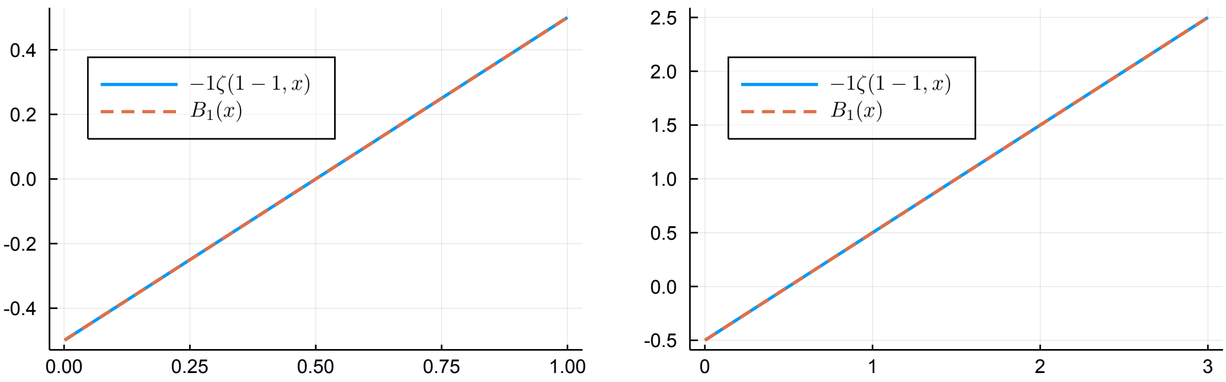
```julia
function plot_Bernoulli(r ;
    x1=range(0.001, 1, length=400), l1=:topleft,
    x2=range(0.001, 3, length=400), l2=:topleft,
    figsize=(800, 250))

    x = x1
    P1 = plot(legend=l1, legendfontsize=9)
    plot!(x, B.(r,x),          label="\$-$r\\zeta(1-$r,x)\$", lw=2)
    plot!(x, fBernoulli.(r,x), label="\$B_{$r}(x)\$", lw=2, ls=:dash)

    x = x2
    P2 = plot(legend=l2, legendfontsize=9)
    plot!(x, B.(r,x),          label="\$-$r\\zeta(1-$r,x)\$", lw=2)
    plot!(x, fBernoulli.(r,x), label="\$B_{$r}(x)\$", lw=2, ls=:dash)

    plot(P1, P2; size=figsize)
end
```

Out[4]: plot_Bernoulli (generic function with 1 method)

In [5]:

```julia
plot_Bernoulli(1)
```

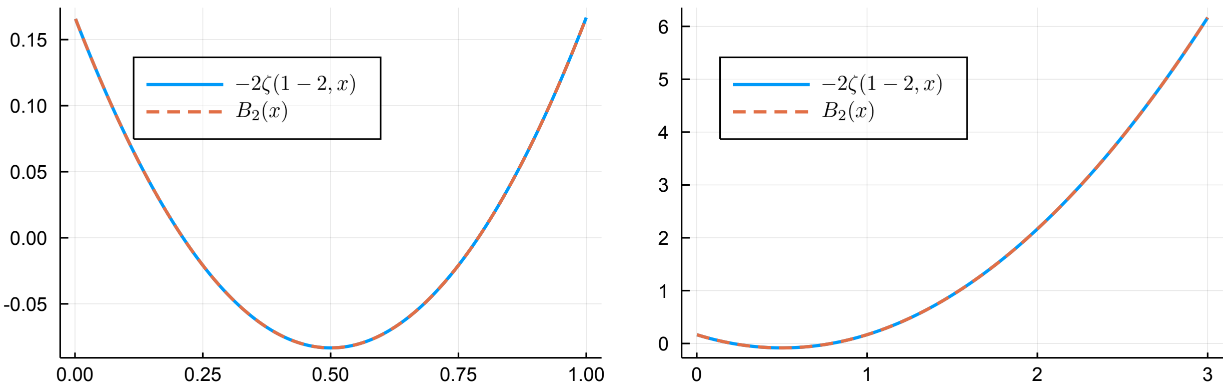
Out[5]:



In [6]:

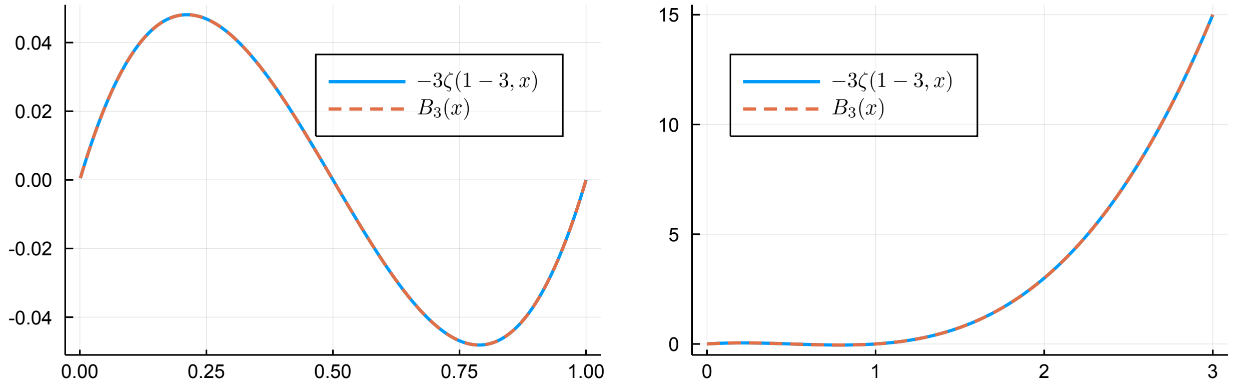
```julia
plot_Bernoulli(2; l1=:top)
```

Out[6]:



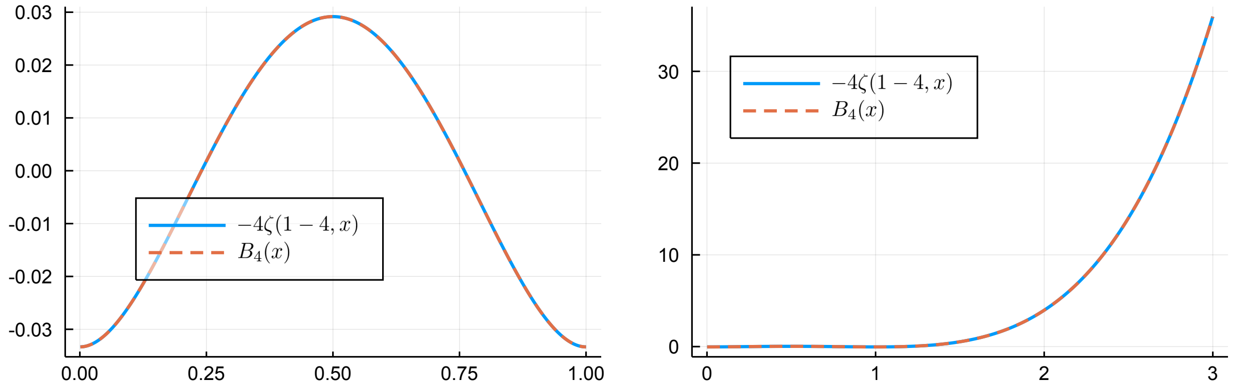
In [7]:
```
plot_Bernoulli(3; l1=:topright)
```

Out[7]:



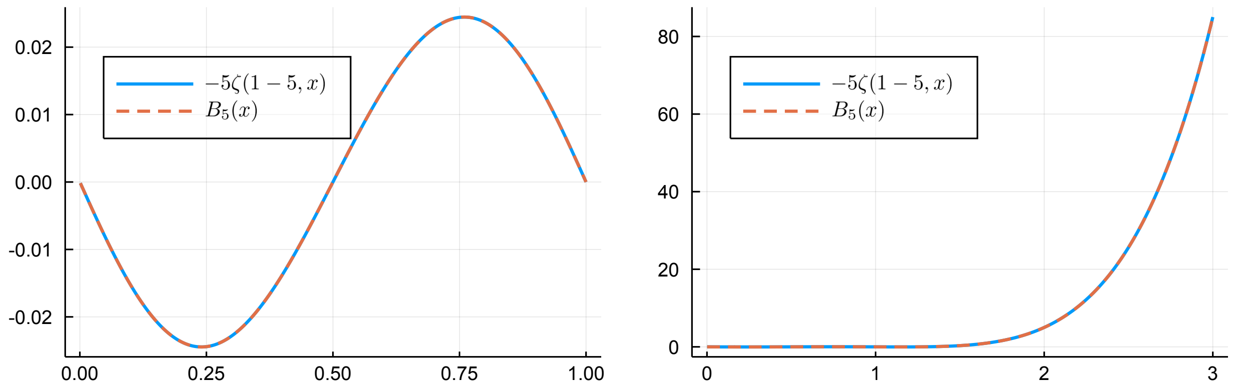
In [8]:
```
plot_Bernoulli(4; l1=:bottom)
```

Out[8]:



In [9]:
```
plot_Bernoulli(5)
```

Out[9]:



ぴったり一致している.

2 対数ガンマ函数との関係

Hurwitzのゼータ函数の s に関する偏導函数を $\zeta_s(s,x)$ と書くことにする. このとき

$$\zeta_s(0,x) = \log\Gamma(x) - \log\sqrt{2\pi} \quad (x > 0)$$

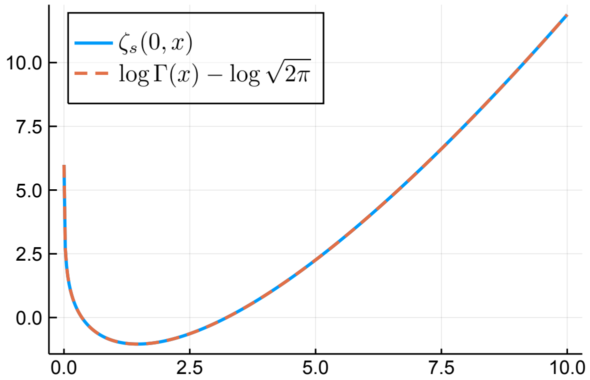
In [10]:
```
logGamma(r, x; h=√eps()) = (zeta(1-r+h, x) - zeta(1-r-h, x))/(2h)
exp(logGamma(1, 10))*√(2π), gamma(10)
```

Out[10]:
```
(362879.9877628901, 362880.0)
```

In [11]:

```
function plot_logGamma1(; x=range(0.001, 10, length=400), l=:topleft, figsize=(400,250))
    plot(size=figsize)
    plot!(legend=l, legendfontsize=10)
    plot!(x, logGamma.(1,x), label=L"\zeta_s(0,x)", lw=2)
    plot!(x, lgamma.(x) .- log(√(2π)), label=L"\log\Gamma(x) - \log\sqrt{2\pi}", lw=2, ls=:dash)
end

plot_logGamma1()
```

Out[11]:



ぴったり一致している.

3 対数正弦函数との関係

Hurwitzのゼータ函数とガンマ函数と $\sin$ の関係より,

$$\frac{\Gamma(x)}{\sqrt{2\pi}} = \exp(\zeta_s(0,x)), \quad 2\sin(\pi x) = \frac{2\pi}{\Gamma(x)\Gamma(1-x)} \quad (0 < x < 1)$$

前者の公式はLerchの公式と呼ばれ, 後者の公式はガンマ函数の相反公式と呼ばれている. それらの公式の証明については

- https://nbviewer.jupyter.org/github/genkuroki/Calculus/blob/master/10%20Gauss%2C%20Gamma%2C%20Beta.ipynb (https://nbviewer.jupyter.org/github/genkuroki/Calculus/blob/master/10%20Gauss%2C%20Gamma%2C%20Beta.ipynb)
- https://genkuroki.github.io/documents/Calculus/10%20Gauss%2C%20Gamma%2C%20Beta.pdf (https://genkuroki.github.io/documents/Calculus/10%20Gauss%2C%20Gamma%2C%20Beta.pdf)

の第2.10.1節, 第2.6.2節を参照せよ.

ゆえに,

$$\log(2\sin\pi x) = -(\zeta_s(0,x) + \zeta_s(0,1-x)) \quad (0 < x < 1).$$

これの成立を数値計算で確認してみよう.

In [12]:

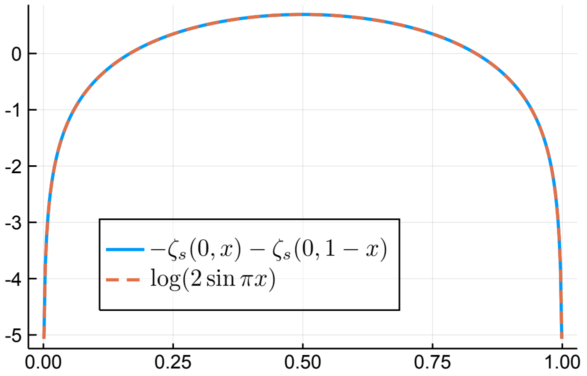
```
logsine(r,x) = -logGamma(r,x) + (-1)^r*logGamma(r,1-x)
logsine(1, 0.3), log(2sin(0.3π))
```

Out[12]: (0.481211774982512, 0.48121182505960347)

In [13]:

```julia
function plot_logsine1(; x=range(0.001, 0.999, length=400), l=:bottom, figsize=(400,250))
    plot(size=figsize)
    plot!(legend=l, legendfontsize=10)
    plot!(x, logsine.(1,x), label=L"-\zeta_s(0,x)-\zeta_s(0,1-x)", lw=2)
    plot!(x, @.(log(2sin(π*x))), label=L"\log(2\sin\pi x)", lw=2, ls=:dash)
end

plot_logsine1()
```

Out[13]:



ぴったり一致している.

$\log(2\sin\pi x)$ については次が成立することも知られている:

$$\int_0^1 \log(2\sin\pi x)\,dx = 0.$$

この公式は次と同値である:

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

In [14]:

```julia
value, error = quadgk(x -> log(2sin(π*x)), 0, 1)
@show value, error;
```

```
(value, error) = (1.4675760606763788e-15, 1.9614816116706072e-16)
```

数値積分によっても積分の値はほぼ 0 になっている.

4 周期的Bernoulli多項式のFourier展開

Bernoulli多項式 $B_r(x)$ の $0 < x < 1$ における値については以下が知られている.

r が偶数のとき

$$B_r(x) = -(-1)^{r/2}2r!\sum_{n=1}^\infty \frac{\cos 2\pi nx}{(2\pi n)^r} \qquad (0 < x < 1).$$

r が奇数のとき

$$B_r(x) = -(-1)^{(r-1)/2}2r!\sum_{n=1}^\infty \frac{\sin 2\pi nx}{(2\pi n)^r} \qquad (0 < x < 1).$$

これらの公式の証明については

- https://nbviewer.jupyter.org/github/genkuroki/Calculus/blob/master/12%20Fourier%20analysis.ipynb (https://nbviewer.jupyter.org/github/genkuroki/Calculus/blob/master/12%20Fourier%20analysis.ipynb)
- https://genkuroki.github.io/documents/Calculus/12%20Fourier%20analysis.pdf (https://genkuroki.github.io/documents/Calculus/12%20Fourier%20analysis.pdf)

の第6.3.3節または

- 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)

の第1.5節を参照せよ.

In [15]:

```
function FourierB(r, x; N=50)
    if iseven(r)
        -(-1)^(r÷2)     *2*gamma(r+1)*sum(cos(2π*n*x)/(2π*n)^r for n in 1:N)
    else
        -(-1)^((r-1)÷2)*2*gamma(r+1)*sum(sin(2π*n*x)/(2π*n)^r for n in 1:N)
    end
end
```

Out[15]: FourierB (generic function with 1 method)

In [16]:

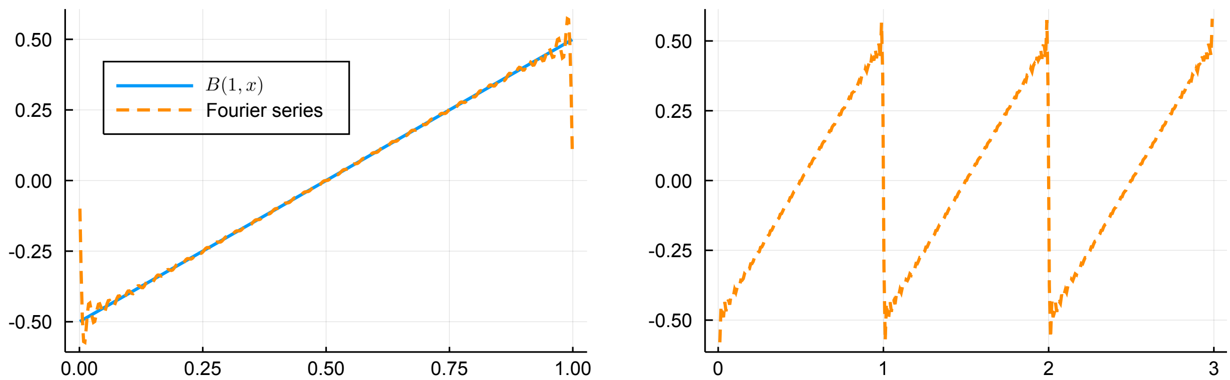
```
function plot_FourierBernoulli(r;
    N = 50,
    x1 = range(0.001, 0.999, length=400), l1 = :topleft,
    x2 = range(0.01, 2.99, length=400), l2 = :topleft,
    figsize = (800, 250))

    x = x1
    P1 = plot(legend=l1, legendfontsize=8)
    plot!(x, fBernoulli.(r,x),    label="\$B($r,x)\$", lw=2)
    plot!(x, FourierB.(r,x; N=N), label="Fourier series", lw=2, ls=:dash, lc=:darkorange)

    x = x2
    P2 = plot(legend=l2, legendfontsize=8)
    plot!(x, FourierB.(r,x; N=N), label="", lw=2, ls=:dash, lc=:darkorange)

    plot(P1, P2, size=figsize)
end
```

Out[16]: plot_FourierBernoulli (generic function with 1 method)

In [17]:

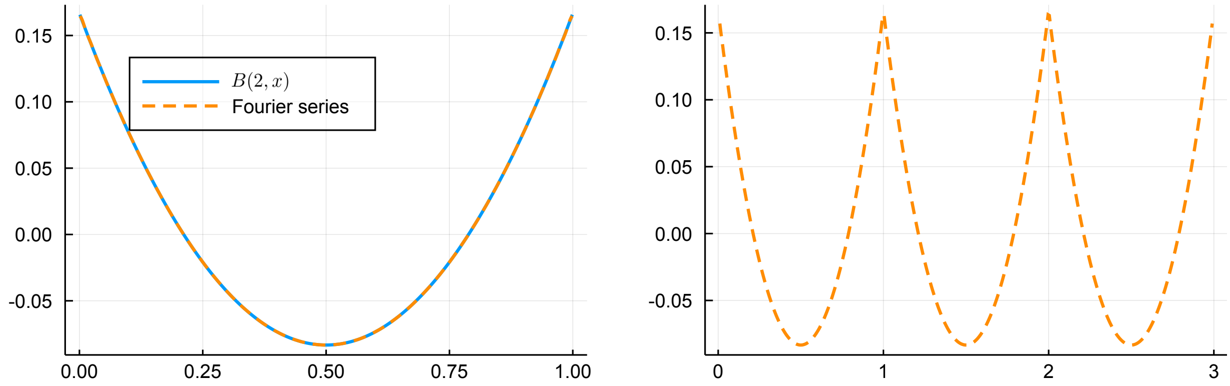
```
plot_FourierBernoulli(1)
```

Out[17]:



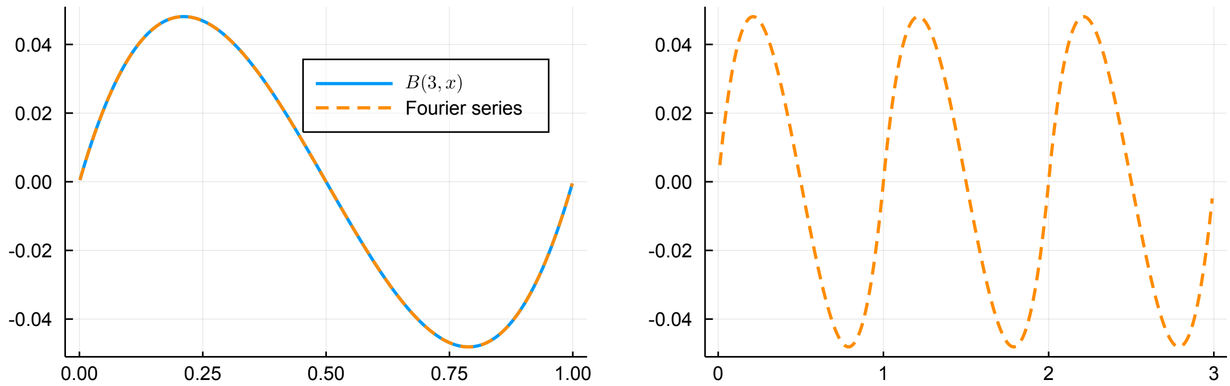
In [18]:
```
plot_FourierBernoulli(2; l1=:top)
```

Out[18]:



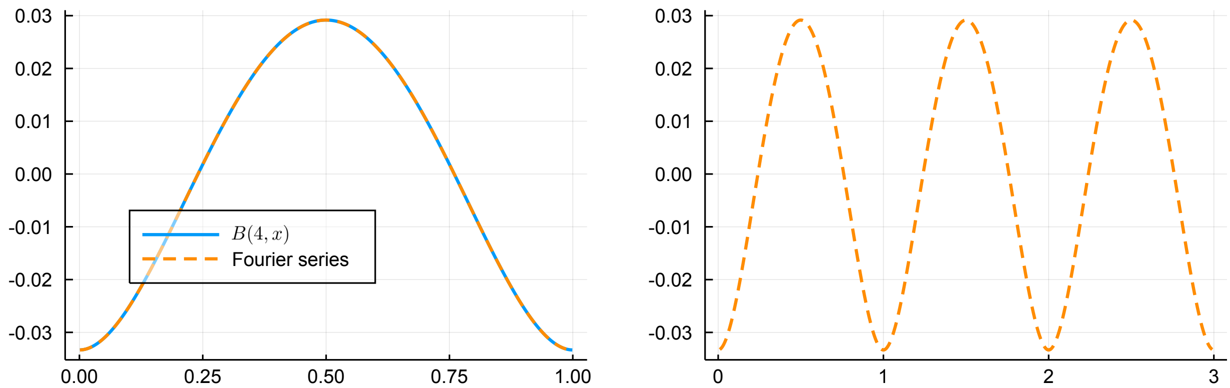
In [19]:
```
plot_FourierBernoulli(3; l1=:topright)
```

Out[19]:



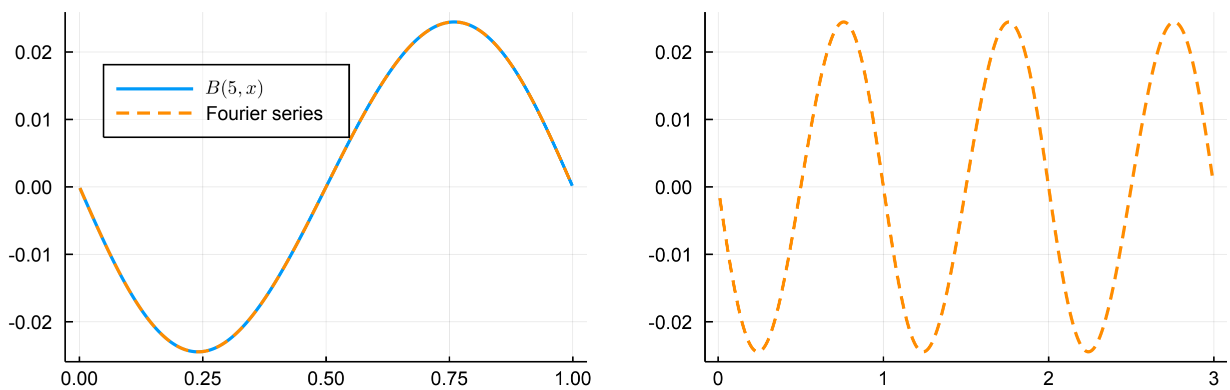
In [20]:
```
plot_FourierBernoulli(4; l1=:bottom)
```

Out[20]:



In [21]:
```
plot_FourierBernoulli(5)
```

Out[21]:



5 Milnor型多重対数正弦函数のFourier展開

Milnor型の多重対数正弦函数 $\log S_r^M(x)$ を

$$\log S_r^M(x) = -\zeta_s(1-r, x) + (-1)^r \zeta_s(1-r, 1-x)$$

と定義する. ここで $\zeta_s(s, x)$ はHurwitzのゼータ函数の s に関する偏導函数を表す.

$\log S_r^M(x)$ の $0 < x < 1$ における値については以下が知られている.

r が偶数のとき

$$\log S_r^M(x) = (-1)^{r/2} 2\pi (r-1)! \sum_{n=1}^\infty \frac{\sin 2\pi n x}{(2\pi n)^r} \quad (0 < x < 1).$$

r が奇数のとき

$$\log S_r^M(x) = -(-1)^{(r-1)/2} 2\pi (r-1)! \sum_{n=1}^\infty \frac{\cos 2\pi n x}{(2\pi n)^r} \quad (0 < x < 1).$$

解説: 以上で使ったFourier級数展開の公式は次のHurwitzの函数等式を用いて筆者自身が導出した公式である(widely known):

$$\zeta(1-s, x) = \frac{\Gamma(s)}{(2\pi)^s}\left(e^{-\pi i s/2} \operatorname{Li}_s(e^{2\pi i x}) + e^{\pi i s/2} \operatorname{Li}_s(e^{-2\pi i x})\right) \quad (\operatorname{Re} s > 0, 0 < x < 1).$$

ここで $\operatorname{Li}_s(z)$ は多重対数函数(polylogarithm)である:

$$\operatorname{Li}_s(z) = \sum_{n=1}^\infty \frac{z^n}{n^s}.$$

上のような公式の導出では符号や係数の細かい部分をよく間違う. しかし, 以下のように実際に数値計算してみて一致していることを確認できれば, 自分で導出した公式の正しさに自信を持てるようになる. 人間は証明をよく間違えるので, 可能ならば, 単に証明によって正しさを確認するだけではなく, 数値計算によってもその「正しさ」を確認しておくべきだと思う. 数値計算やその結果のプロットによって理解が深まることが多い.

Milnor型の多重対数正弦函数に関する公式を得るためには, Hurwitzの函数等式の両辺を s で偏微分して, s に 0 以下の整数 $1-r$ を代入する必要がある. Milnor型の多重対数正弦函数に関する公式を得るために $-\zeta(1-r, x)$ と $(-1)^r \zeta(1-r, 1-x)$ の和を計算すると, 符号 $(-1)^r$ をうまく働いて複雑な項がすべてキャンセルして消えて, 上で言及したシンプルな公式が得られるという仕組みになっている.

上のHurwitzの函数等式の導出のラフな解説が

* https://nbviewer.jupyter.org/github/genkuroki/Calculus/blob/master/12%20Fourier%20analysis.ipynb (https://nbviewer.jupyter.org/github/genkuroki/Calculus/blob/master/12%20Fourier%20analysis.ipynb)
* https://genkuroki.github.io/documents/Calculus/12%20Fourier%20analysis.pdf (https://genkuroki.github.io/documents/Calculus/12%20Fourier%20analysis.pdf)

の第6.3節にある. そこでは, Hurwitzのゼータ函数と多重対数函数の両方の一般化になっているLerchの超越函数(レルヒの超越函数)

$$L(\tau, x, s) = \sum_{k=0}^\infty \frac{e^{2\pi i x}}{(x+k)^s}$$

を扱っている. Hurwitzの函数等式ではHurwitzのゼータ函数以外に多重対数函数も必要になるが, Lerchの超越函数であればその範囲内で $s \leftrightarrow 1-s$ に関する函数等式を閉じた形で書ける.

In [22]:
```
function Fourierlogsine(r, x; N=50)
    if iseven(r)
         (-1)^(r÷2)    *2π*gamma(r)*sum(sin(2π*n*x)/(2π*n)^r for n in 1:N)
    else
        -(-1)^((r-1)÷2)*2π*gamma(r)*sum(cos(2π*n*x)/(2π*n)^r for n in 1:N)
    end
end
```

Out[22]: `Fourierlogsine (generic function with 1 method)`

In [23]:

```
function plot_Fourierlogsine(r;
    N = 50,
    x1 = range(0.001, 0.999, length=400), l1 = :topleft,
    x2 = range(0.01, 2.99, length=400), l2 = :topleft,
    figsize = (800, 250))

    x = x1
    P1 = plot(legend=l1, legendfontsize=8)
    plot!(x,        logsine.(r,x),      label="\$\\log S^M_{$r}(x)\$", lw=2)
    plot!(x, Fourierlogsine.(r,x; N=N), label="Fourier series", lw=2, ls=:dash, lc=:darkorange)

    x = x2
    P2 = plot(legend=l2, legendfontsize=8)
    plot!(x, Fourierlogsine.(r,x; N=N), label="", lw=2, ls=:dash, lc=:darkorange)

    plot(P1, P2, size=figsize)
end
```

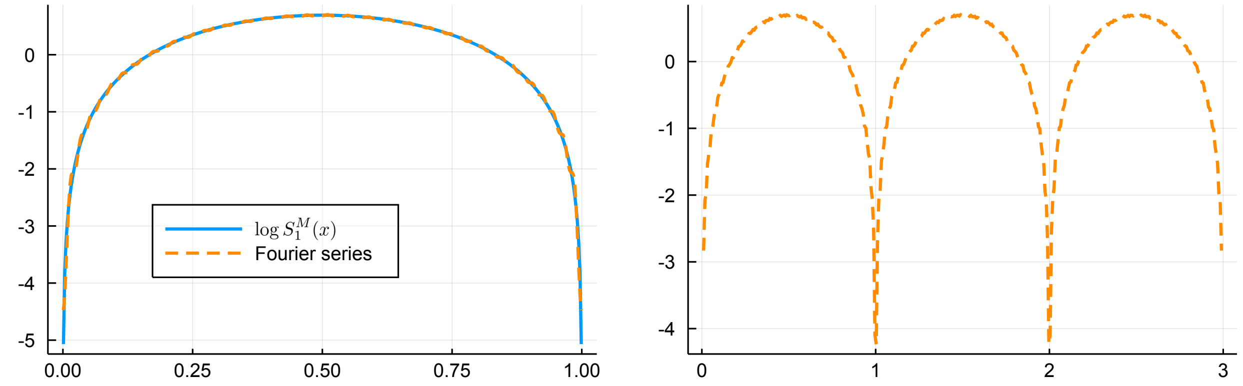
Out[23]:

```
plot_Fourierlogsine (generic function with 1 method)
```

In [24]:

```
plot_Fourierlogsine(1; l1=:bottom)
```

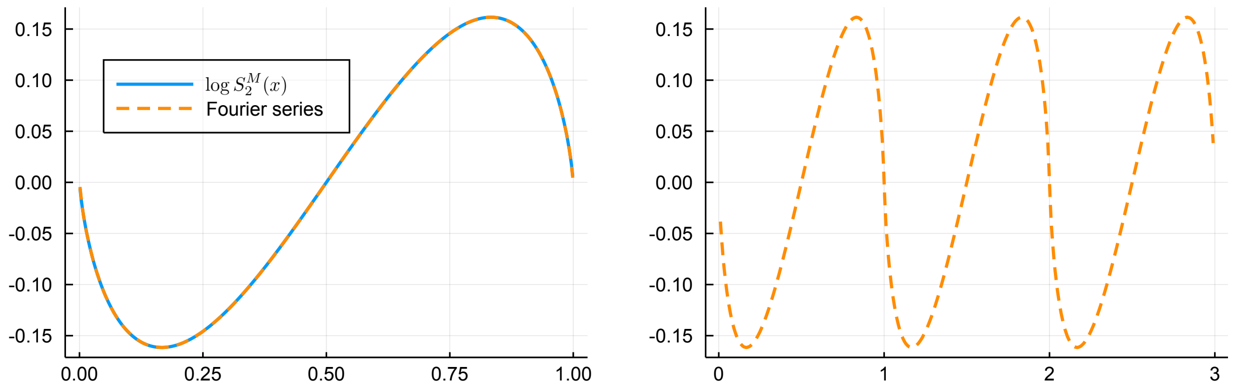
Out[24]:



In [25]:

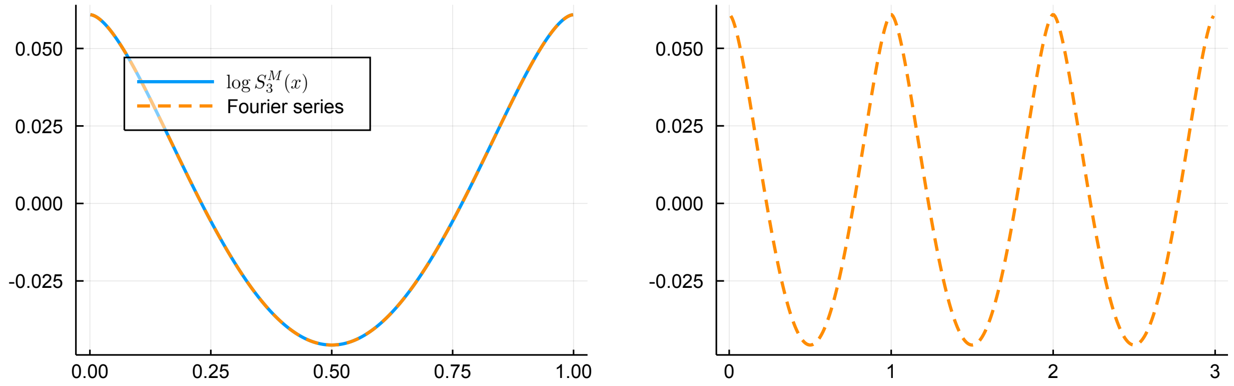
```
plot_Fourierlogsine(2)
```

Out[25]:



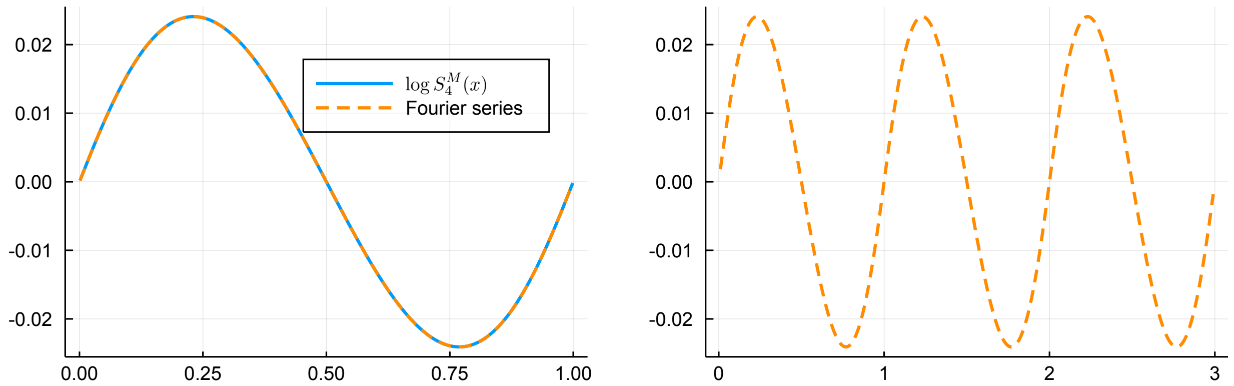
In [26]:
```
plot_Fourierlogsine(3; l1=:top)
```

Out[26]:



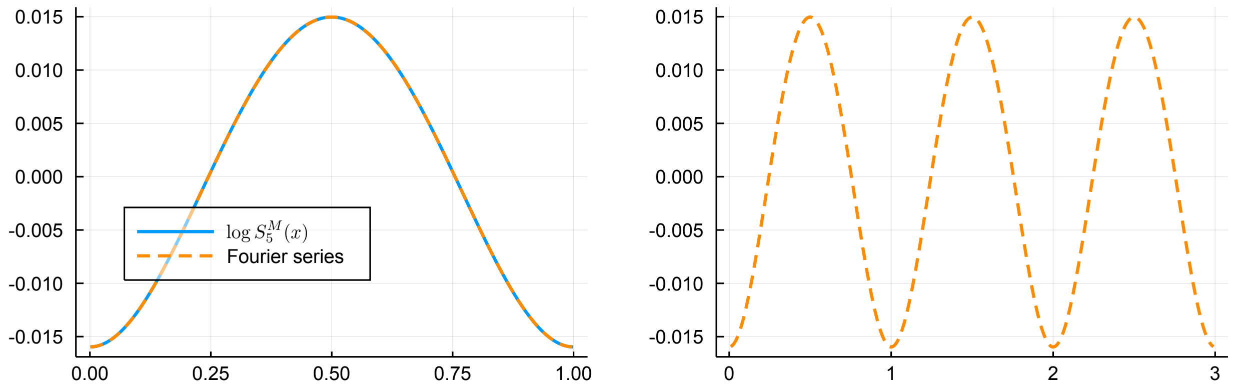
In [27]:
```
plot_Fourierlogsine(4; l1=:topright)
```

Out[27]:



In [28]:
```
plot_Fourierlogsine(5; l1=:bottom)
```

Out[28]:



In [ ]:
```

```