

傅里叶级数 - 实数到复平面的映射

```
In[*]:= f[x_] := x + I Sin[x];
```

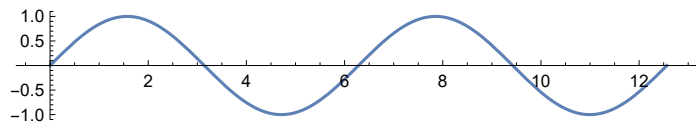
[I](#) [正弦](#)

```
ParametricPlot[ReIm[f[x]], {x, 0, 4 π}]
```

[绘制参数图](#)

[实部虚部列表](#)

Out[*]=



```
n = 30;
```

```
cn = Table[ $\frac{1}{2\pi}$  Integrate[f[x] Exp[-I k x], {x, -π, π}], {k, -n, n}]
```

[表格](#)

[积分](#)

[指...](#)

[虚数单位](#)

```
p = Sum[cn[[r]] Exp[(r - n - 1) I t], {r, 2 n + 1}];
```

[求和](#)

[指数形式](#)

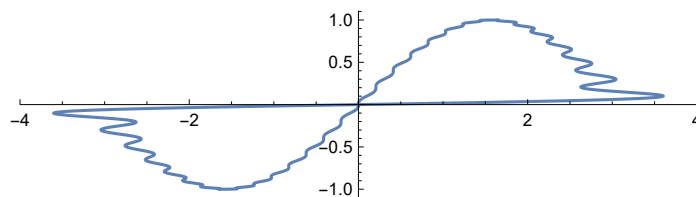
[虚数单位](#)

```
ParametricPlot[ReIm@p, {t, 0, 2 π}]
```

[绘制参数图](#)

[实部虚部列表](#)

Out[*]=



将上面过程包装到函数

```
In[*]:= fourierSeriesPlot[f_, n_] :=
```

```
Block[{cn}, cn = Table[ $\frac{1}{2\pi}$  Integrate[(x + I f) Exp[-k I x], {x, -π, π}], {k, -n, n}];
```

[块](#)

[表格](#)

[积分](#)

[...](#)

[指...](#)

[虚数单位](#)

```
Show[ParametricPlot[ReIm@Sum[cn[[r + n + 1]] Exp[r I t], {r, -n, n}], {t, 0, 2 π}],
```

[显示](#) [绘制参数图](#)

[实...](#)

[求和](#)

[指...](#)

[虚数单位](#)

```
Plot[f, {x, -π, π}, PlotStyle -> Orange]]]
```

[绘图](#)

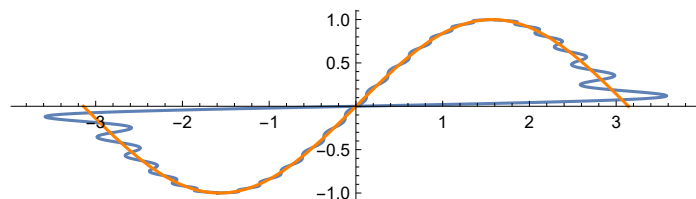
[绘图样式](#)

[橙色](#)

```
fourierSeriesPlot[Sin[x], 25]
```

[正弦](#)

Out[*]=

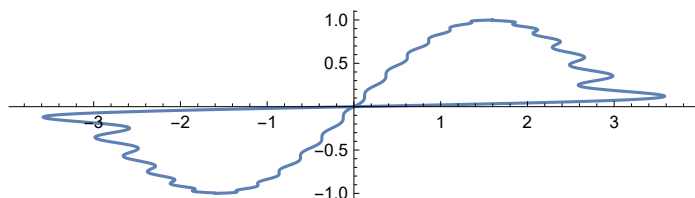


使用内置函数计算傅里叶级数

将实函数 f 当作复平面上的复函数 $x+I f$ ，再计算其 n 阶傅里叶级数并绘图

```
In[ ]:= fsPlotComplex[f_, n_] :=  
  ParametricPlot[Evaluate@ReIm@FourierSeries[x + I f, x, n], {x, 0, 2  $\pi$ }]  
  |绘制参数图 |计算 |实... |傅立叶级数 |虚数单位  
fsPlotComplex[Sin[x], 25]  
  |正弦
```

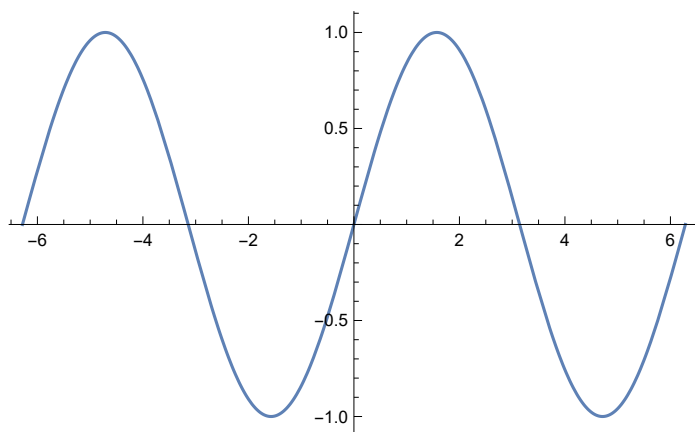
Out[]:=



计算实函数 f 的傅里叶级数并绘图

```
In[ ]:= fsPlot[f_, n_] := Plot[Evaluate@FourierSeries[f, x, 6], {x, -2  $\pi$ , 2  $\pi$ }]  
  |绘图 |计算 |傅立叶级数  
fsPlot[Sin[x], 25]  
  |正弦
```

Out[]:=

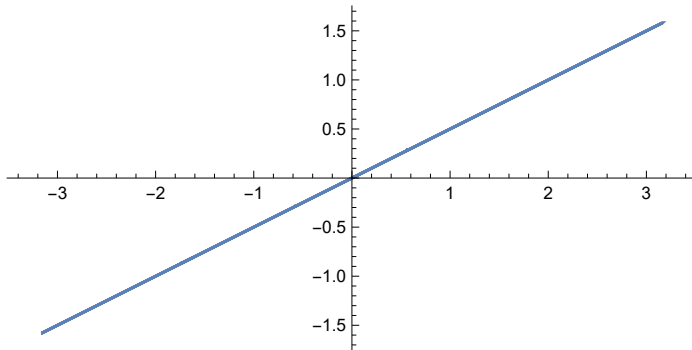


上面第一个是在复数域绘图，第二个是在实数域绘图

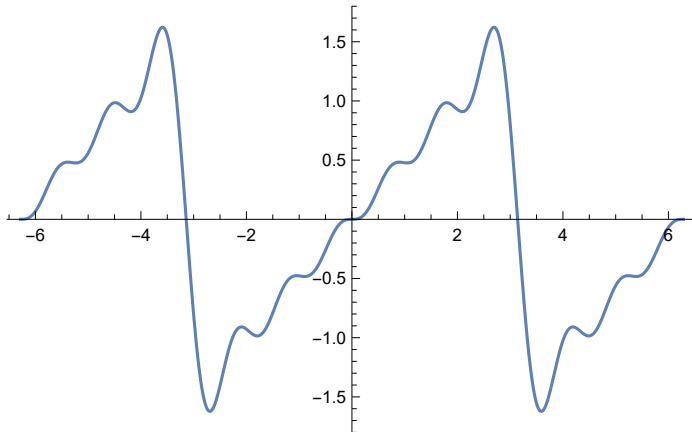
再看看函数：x/2

```
In[*]:= fsPlotComplex[x / 2, 5]
fsPlot[x / 2, 5]
```

Out[*]=



Out[*]=



两者仿佛在某个地方表现相反，为什么呢？

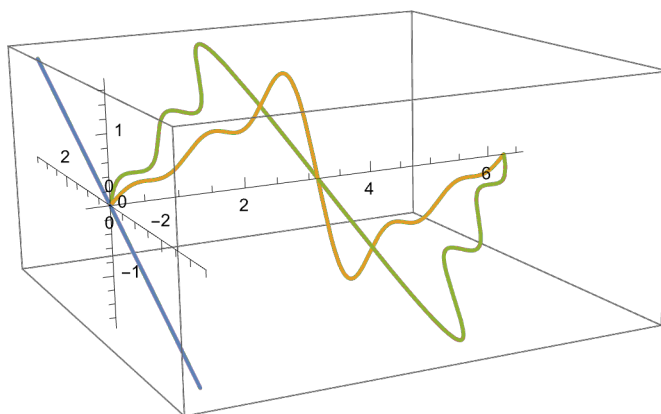
其实这再次体现了之前的一个认识，从复数域观察和从实数域观察是从两个垂直的视角观察，
看到的是3D空间的两个投影。

```

In[ ]:= fs = FourierSeries[x / 2, x, 5];
          傅立叶级数
fsc = FourierSeries[x + I x / 2, x, 5];
          傅立叶级数          虚数单位
ParametricPlot3D[{{0, Re@fsc, Im@fsc}, {x, 0, fs}}, {x, 0, fs}, {x, Re@fsc, Im@fsc}},
          绘制三维参数图          实部          虚部          实部          虚部
          {x, 0, 2  $\pi$ }, RotationAction -> "Clip", AxesOrigin -> {0, 0, 0}]
          旋转操作          剪切          坐标轴原点

Out[ ]:=

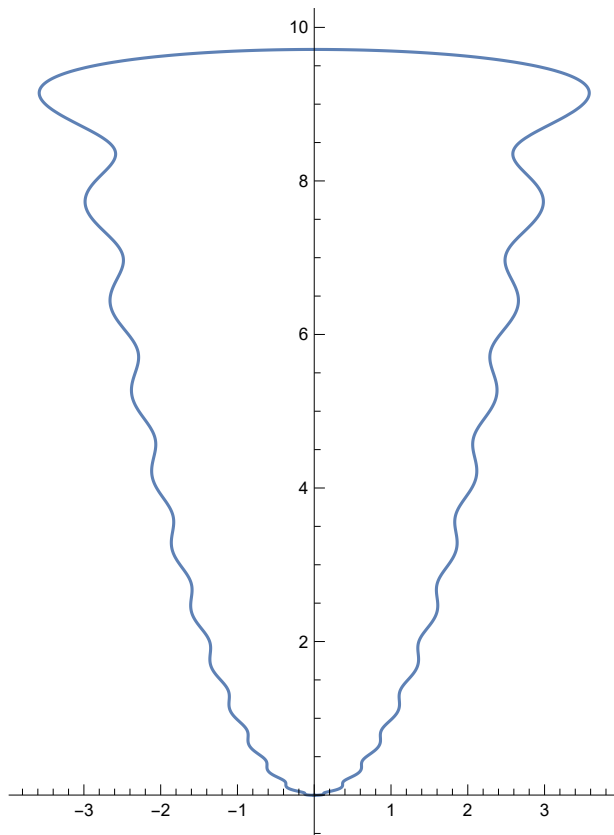
```



从某种角度看，类似与信号的时域和频域的关系，把 x 当作时间参数。

看看其他函数在实数域计算傅里叶级数的图形

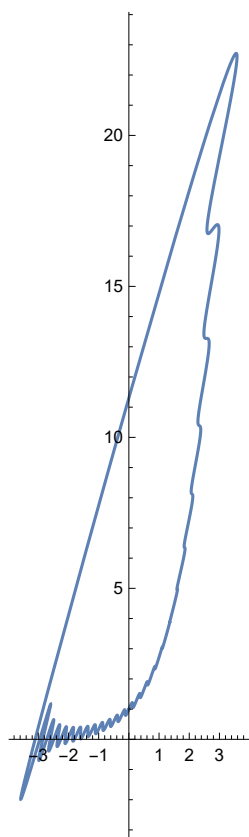
```
In[ ]:= fsPlotComplex[x^2, 25]  
Out[ ]:=
```



In[*]:= **fsPlotComplex**[Exp[x], 25]

指数形式

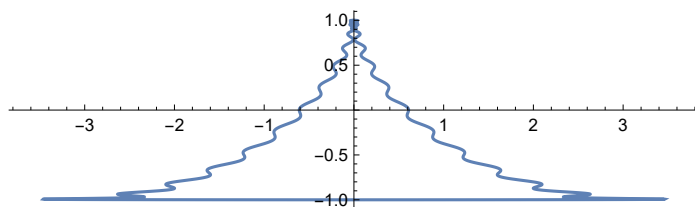
Out[*]=



In[*]:= **fsPlotComplex**[Exp[I x], 25]

虚数单位

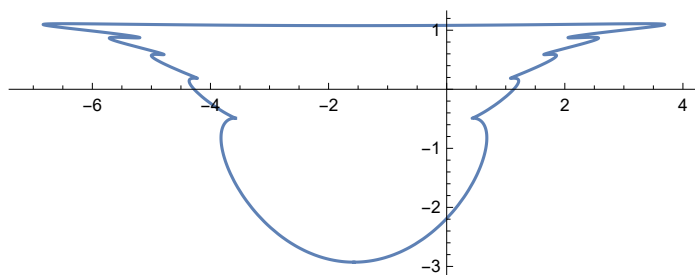
Out[*]=



In[*]:= **fsPlotComplex**[Log[x], 10]

对数

Out[*]=



In[*]:= **fsPlotComplex**[**Log**[**I** x], 10]

对数 虚数单位

Out[*]=

