1. 计算$f(t)=|t|, t\in[-\pi,\pi]$傅里叶级数的系数，绘制出傅里叶级数前N项($N=2,4,6,8,10$)的函数图像。考虑傅里叶级数的前N项时，$\omega=\frac{2\pi}{T}=1$，因此$f(t)=|t|\approx\sum_{k=-\frac{N}{2}}^{\frac{N}{2}} g_k\cdot e^{-ikt}$，其中$g_k=\frac{1}{2\pi}\int_{-\pi}^{\pi}|t|\cdot e^{ikt}dt$。

定义$Fourier(x)$函数来计算傅里叶级数的前x项，$k=-\frac{x}{2},-\frac{x}{2}+1,\dots,\frac{x}{2}$。$func$即为被积函数的解析表达式：$|t|\cdot e^{ikt}$。$g_k$通过 SymPy 库的符号积分$sympy.integrate()$得到。再使用一个 for 循环将我们需要的傅里叶级数的每一项，即$g_k\cdot e^{-ikt}$算出，累加至结果$result$并返回，这样就得到了傅里叶级数$\sum_{k=-\frac{x}{2}}^{\frac{x}{2}} g_k\cdot e^{-ikt}$。

```python
def Fourier(x):
    N = int(x / 2)
    result = 0
    func = abs(t) * sp.exp(1j * k * t)
    g_k = (1 / (2*sp.pi)) * sp.integrate(func, (t, -sp.pi, sp.pi))

    for i in range(-N, N+1):
        temp = g_k.evalf(subs={k:i}, n=10) * sp.exp(-1j * i * t)
        result += temp

    return result
```

在$main()$函数中，先绘制$f(t)=|t|$的函数图像。再在一个 for 循环($i=2,4,6,8,10$)中将傅里叶级数的前i项的函数图像绘制出来。

```python
def main():
    plot(abs(t), (t, -sp.pi, sp.pi), axis_center=[0,0])

    for i in range(2, 11, 2):
        func = Fourier(i)
        plot(func, (t, -sp.pi, sp.pi), axis_center=[0,0])
```

得到的结果如图所示：

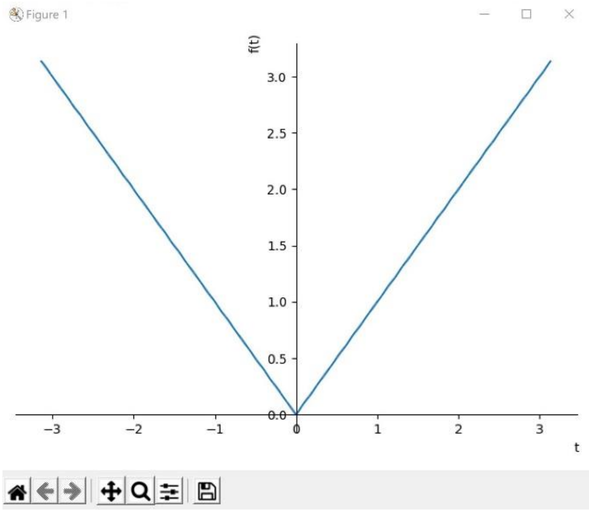

图 1 $f(t) = |t|$原函数

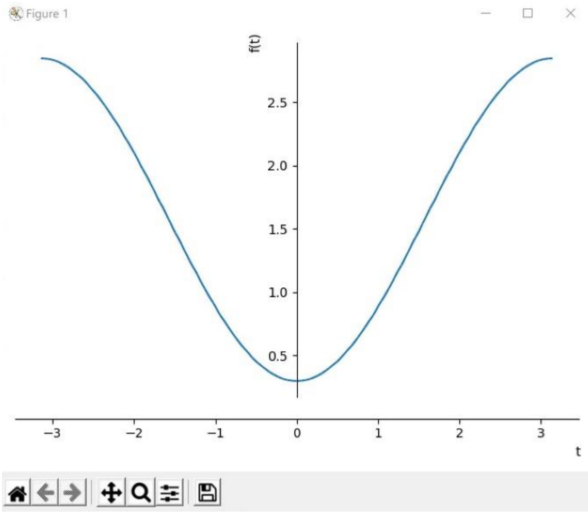

图 2 $N = 2$傅里叶级数展开

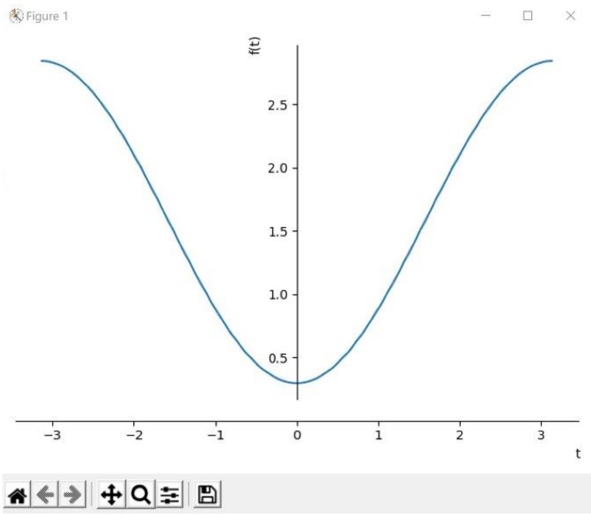

图 3 $N = 4$傅里叶级数展开

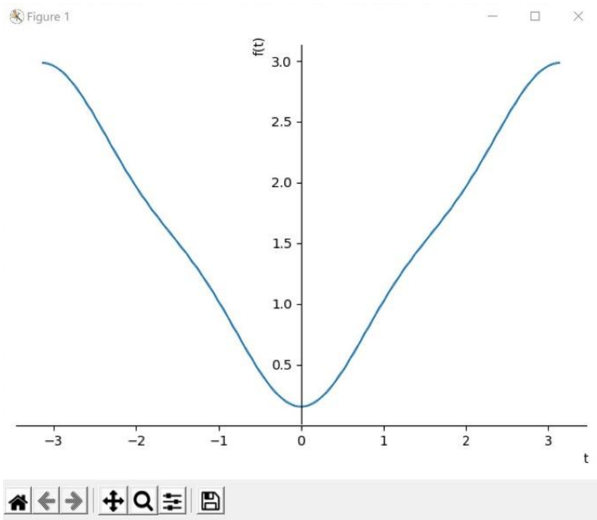

图 4 $N = 6$傅里叶级数展开

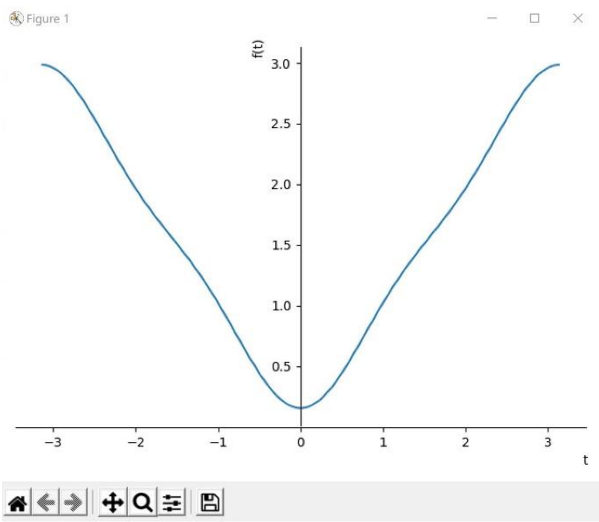

图 5 $N = 8$傅里叶级数展开

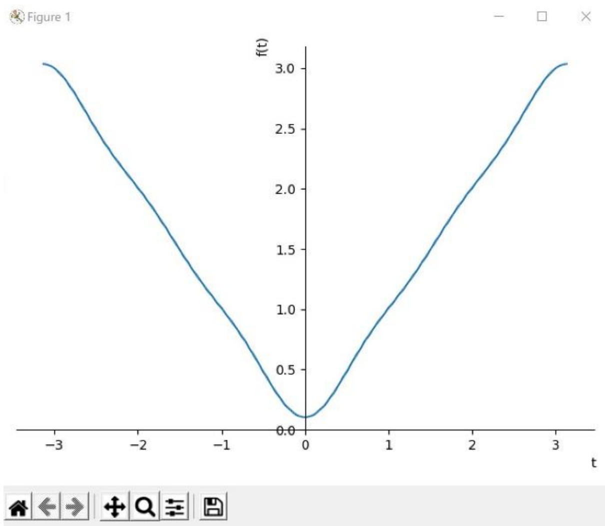

图 6 $N = 10$傅里叶级数展开

可以看到，随着N的增大，傅里叶级数的函数图像与原函数越来越接近。

$$\lim_{N\to\infty}\sum_{k=-N}^{N} g_k\cdot e^{-ikt}=f(t)$$

用以上方法，我们也可以轻易地做出N在更大值时的函数图像，由于采用符号积分，计算量并没有显著增加。在$N=20,50$时，结果如图所示：

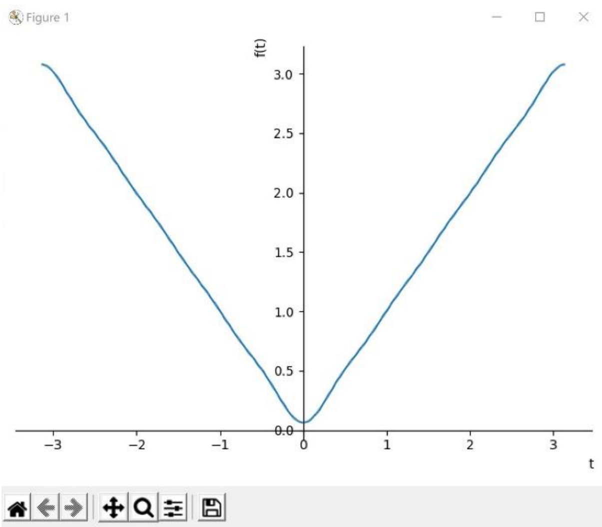

图 7 $N=20$傅里叶级数展开

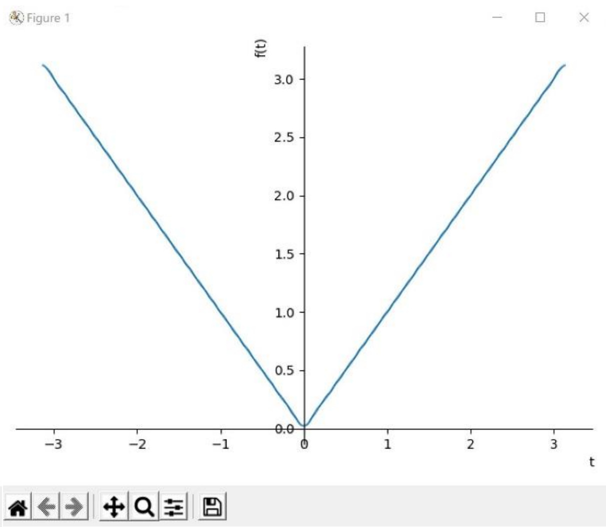

图 8 $N=50$傅里叶级数展开

可以看到在$N=50$时，傅里叶级数与原函数已经非常接近。

Input: a function $f(t)=|t|$

Output: N terms Fourier series of $f(t)$

Function $Fourier(N)$

$n\leftarrow N/2$

$result\leftarrow 0$

$func\leftarrow |t|\cdot e^{ikt}$

$g_k\leftarrow \frac{1}{2\pi}\int_{-\pi}^{\pi}|t|\cdot e^{ikt}dt$

For $j\leftarrow -n\ to\ n$ **Do**

  $temp\leftarrow g_j\cdot e^{-ijt}$

  $result\leftarrow result+temp$

Return $result$

End Function

2. 对太阳黑子数据进行傅里叶变换，画出信号的 power spectrum，找到幅值平方的峰值对应的频率，并求出太阳黑子的周期。

定义$load_file(filename)$函数来载入文档中的数据，将月份和太阳黑子数量分别存入两个列表，以便进行傅里叶变换。

```python
def load_file(filename):
# Load the data from file
    f = open(filename, 'r')
    lines = f.readlines()
    month = []    # Store the months in list
    sunspot = []  # Store the number of sunspots in list

    for line in lines:
        data = line.split()
        month.append(int(data[0]))
        sunspot.append(float(data[1]))

    return month, sunspot
```

定义$FFT(x,y)$函数对数据进行傅里叶变换，由于采样频率就是 1（月），采样的范围就是列表的大小$len(x)$。F是对数据进行傅里叶变换得到的结果，f是傅里叶变换的频率范围。

由于傅里叶变换得到的结果具有对称性，我们用 NumPy 的$where()$函数筛选出$f > 0$的一半数据（不是$f \geq 0$，因为我们要考虑的是$nonzero$值，而且$f = 0$时幅值特别大）。作图时，将f设为横轴，频谱图的纵轴即为$|F|/n$（傅里叶变换时将幅值放大了n倍），我们要绘制的是功率谱，纵轴就是幅值平方$(|F|/n)^2$。

最后找出幅值平方的峰值所在的位置max_loc，返回对应的频率。

```python
def FFT(x, y):
# Calculate the Fourier transform of the sample
    n = len(x)
    F = fftpack.fft(y)                # Fourier transform of sample
    f = fftpack.fftfreq(n, 1)         # Fourier transform sample frequencies
    domain = np.where(f > 0)          # We consider about half of the sample because of symmetry
    x_axis = f[domain]                # Set frequency as x-axis
    y_axis = (abs(F[domain]) / n) ** 2 # Set magnitude squared as y-axis

    plt.plot(x_axis, y_axis)
    plt.xlim(-0.001, 0.05)
    plt.xlabel('frequency (Hz)')
    plt.ylabel('magnitude squared')
    plt.show()

    max_loc = np.where(y_axis == np.max(y_axis)) # Search for the peak
    return x_axis[max_loc]
```

$main()$函数中我们先将文件中的数据载入，绘制出时域图，然后调用$FFT(x, y)$函数绘制出功率谱图。将返回的频率$freq$取倒数，即得到周期T。

```python
def main():
    x, y = load_file('sunspots.txt')
    plt.plot(x, y)   # The original sample
    plt.show()

    freq = FFT(x, y) # The Fourier transform of the sample
    T = 1/freq       # Calculate the period
    print('The period of the sunspots is %f months.' % T)
```

运行结果如图所示：

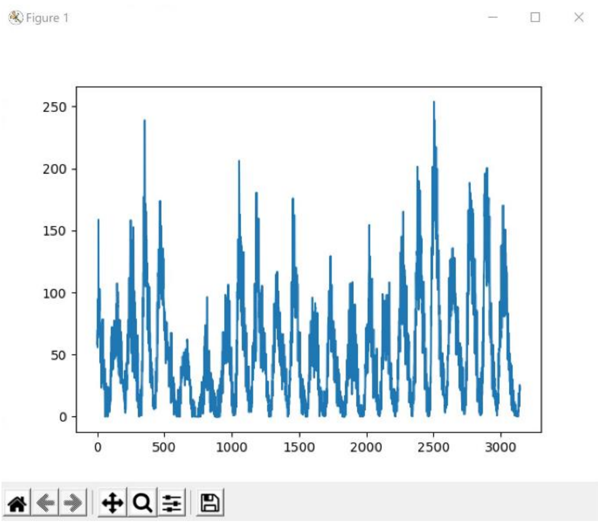

图 1 太阳黑子时域图

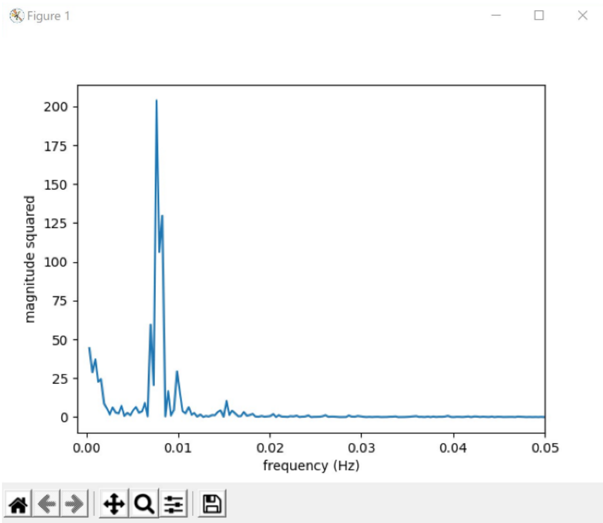

图 2 太阳黑子功率谱图

```
PS C:\Users\11765\Desktop\学习\物理\计算物理\作业\homework6>
gpy\launcher' '62892' '--' 'c:\Users\11765\Desktop\学习\物理\
The period of the sunspots is 130.958333 months.
```

得到的结果，太阳黑子的周期为130.96 $months$。

Input: a series of data $[x_1, y_1], [x_2, y_2], \dots, [x_n, y_n]$

Output: Fourier transform, power spectrum and period of the data

1. $x \leftarrow [x_1, x_2, \dots, x_n]$
2. $y \leftarrow [y_1, y_2, \dots, y_n]$
3. **Function** $FFT(x, y)$
4. $F \leftarrow fft(y)$ /* $SciPy.fftpack.fft()$ */
5. $f \leftarrow fftfreq(n, 1)$ /* $SciPy.fftpack.fftfreq()$ */
6. $x_{axis} \leftarrow f \quad (f > 0)$
7. $y_{axis} \leftarrow (\frac{|F|}{n})^2 \quad (f > 0)$
8. $plot(x_{axis}, y_{axis})$ /* The power spectrum */
9. $max\_loc \leftarrow where(y_{axis} = \max(y_{axis}))$
10. $freq \leftarrow x_{axis}[max\_loc]$
11. **Return** $freq$
12. **End Function**
13. $T \leftarrow 1/freq$