

```

"""
Registration : xxxx
Description  : Square Wave Fourier Series; Number of harmonics = N*harmonics,
              so if harmonics=2 & N=4, then Sum will be from 1 to 8 with only
              odd terms (as an=0).
Author       : AKB
"""

import numpy as np
from scipy import signal
import matplotlib.pyplot as plt

A          = 2 # amplitude
period    = 30 # periodicity
harmonics  = 2 # Number of Harmonics
t = np.linspace(0,4*period,1000) # x-grid

# generate square waveform
def sqwave(t, period):
    return A*np.sign(np.sin(2*np.pi*t/period))

# fourier coefficients; an=0; bn=4/n*pi for odd-n.
def bn(n):
    if (n%2 != 0):
        return 4/(np.pi*n)
    else:
        return 0

# generate angular frequency
def wn(n, period):
    return (2*np.pi*n)/period

# Fourier series
def fourierSq(harmonics,t,period):
    summ = 0
    for i in range(1, harmonics):
        summ += A*bn(i)*np.sin(wn(i,period)*t)
    return summ

# Main
y = []; f1 = []; f2 = []; f3 = []
for i in t:
    y.append(sqwave(i, period))
    f1.append(fourierSq( harmonics,i,period))
    f2.append(fourierSq(4*harmonics,i,period))
    f3.append(fourierSq(8*harmonics,i,period))

# Plot
sg = A*signal.square(2*np.pi*t/period)
plt.plot(t, sg, '-', lw='4', color="red", label="Signal(scipy)")
plt.plot(t, y, '-o', lw='4', color="teal", label=r'Signal$(A \operatorname{sgn}(\sin(\frac{2\pi}{\text{period}} t))$')
plt.plot(t, f1, '-*', lw='.5',color="magenta",label=str(harmonics)+" harmonics")
plt.plot(t, f2, '-+', lw='.5',color="gold", label=str(4*harmonics)+" harmonics")
plt.plot(t, f3, '-x', lw='.5',color="olive", label=str(8*harmonics)+" harmonics")
plt.title("Square Wave Fourier Series", size=16)
plt.legend(loc='best', prop={'size':16})
plt.xlabel('t', size=16)
plt.xticks(size=14)
plt.ylabel(r'$f(t)=\sum_{n=\text{odd}}^{\infty}\frac{4A}{n\pi} \sin(n\omega t)$', size=20)
plt.yticks(size=14)
plt.xlim([0, 4*period])
plt.ylim([-A-1, A+1])
plt.grid()
plt.savefig('plot/03_fouriersq.pdf')
plt.tight_layout()

```

```
plt.show()
```