1 Fourier Analysis

1.1 Introduction

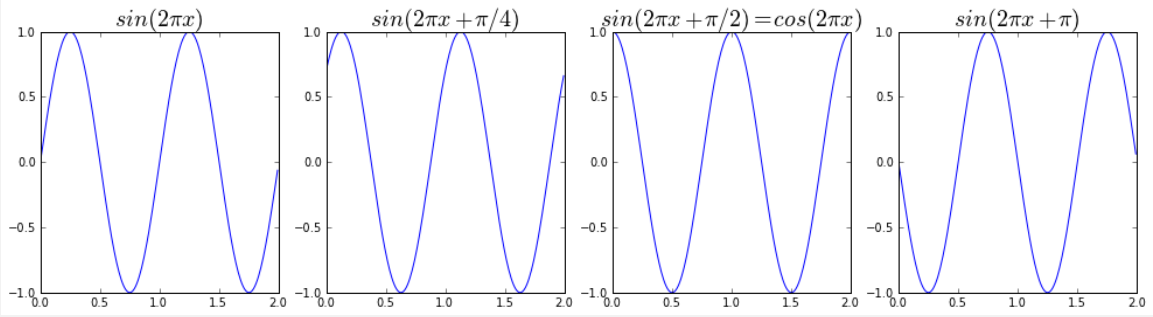
In Fourier analysis, we take a complex periodic phenomenon apart, decompose it into simpler parts. Then we assemble them in the form of linear combination. Since we understand the simpler parts better, and we know quite well how linear comnination works, we can expect to get a better picture of the original phenomenon. This is a common strategy to understand complicated things, called analysis and synthesis. But what make Fourier analysis powerful is that we can implement the strategy rigourously and effectively with the help of mathematical language.

A function f with the property, $f(x+P) = f(x)$for every x, is called a **periodic function**, and P is the **period** of the function. sin and cos are the simple periodic functions that are going to be the building blocks of our analysis. Since $sin(x + \frac{\pi}{2}) = cos(x)$, sin and cos are the same function with different **phase**. Therefore, without losing any generality, we can focus our study on the functions of the form $sin(x + \phi)$ where ϕ is the phase of the function.

Below are 4 sketches of $sin(x)$ graphs with different phases.

```
x = arange(0,2,0.01)
subplot(141)
title('$sin(2\pi x)$',size=20)
plot(x , sin(2*pi*x))
subplots_adjust(right = 2.3)
subplot(142)
title('$sin(2\pi x + \pi / 4)$',size=20)
plot(x, sin(2*pi*x+(pi/4)))
subplot(143)
title('$sin(2\pi x + \pi / 2) = cos(2\pi x)$',size=20)
plot(x, sin(2*pi*x+(pi/2)))
subplot(144)
title('$sin(2\pi x + \pi)$',size=20)
plot(x, sin(2*pi*x+(pi)))
```

```
[<matplotlib.lines.Line2D at 0xa1ac0d0>]
```



Besides **phases**, we can also change the **amplitudes**, **frequencies** of our functions. In short, our most general building blocks are of the form $Asin(2\pi ft + \phi)$, where A is the amplitude;f, the frequency. Finally we can form the sum:

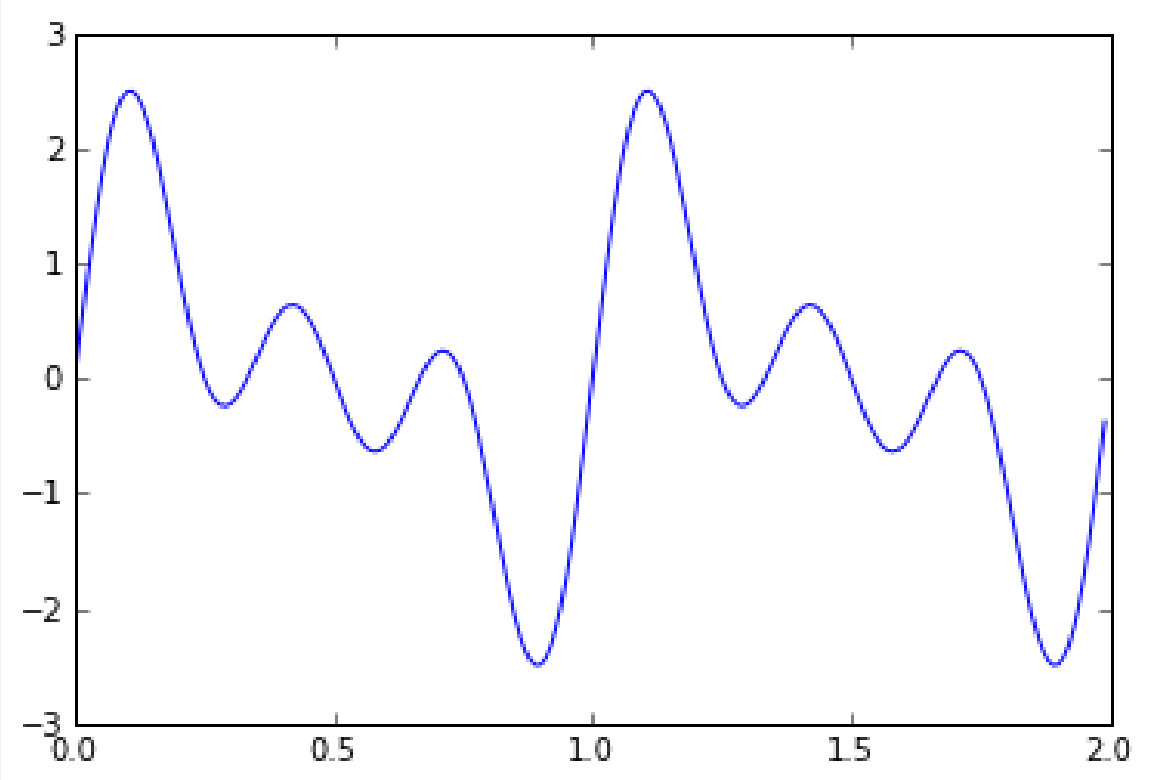
$$\Sigma_1^n A_k sin(2\pi f_k t + \phi_k)$$

To illustrate the fact that linear combination of simple trigonometric functions can represent complicated function, we sketch the graphs of the sum above, with different set of amplitude, frequency, and phase.

$sin(2\pi t) + sin(4\pi t) + sin(6\pi t)$

```
n = 200
domain = array([x])
one_vector = ones((1,n))
amplitude = array([[1,1,1]])
frequency = array([[1,2,3]])
phase = array([[0,0,0]])
y = sin(2*pi*domain.T.dot(frequency)+one_vector.T.dot(phase)).dot(amplitude.T)
plot(x,y)
```

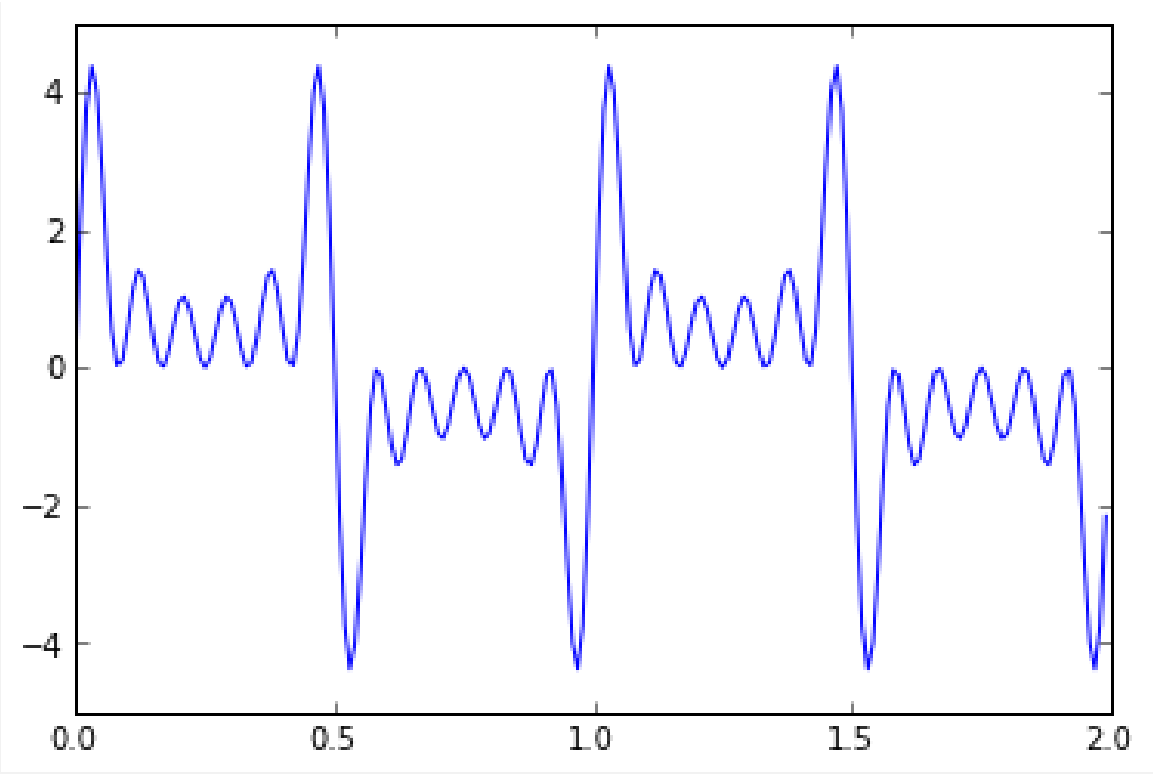
```
[<matplotlib.lines.Line2D at 0xa4d3650>]
```



$sin(2\pi t) + sin(6\pi t) + sin(10\pi t) + sin(14\pi t) + sin(18\pi t) + sin(22\pi t)$

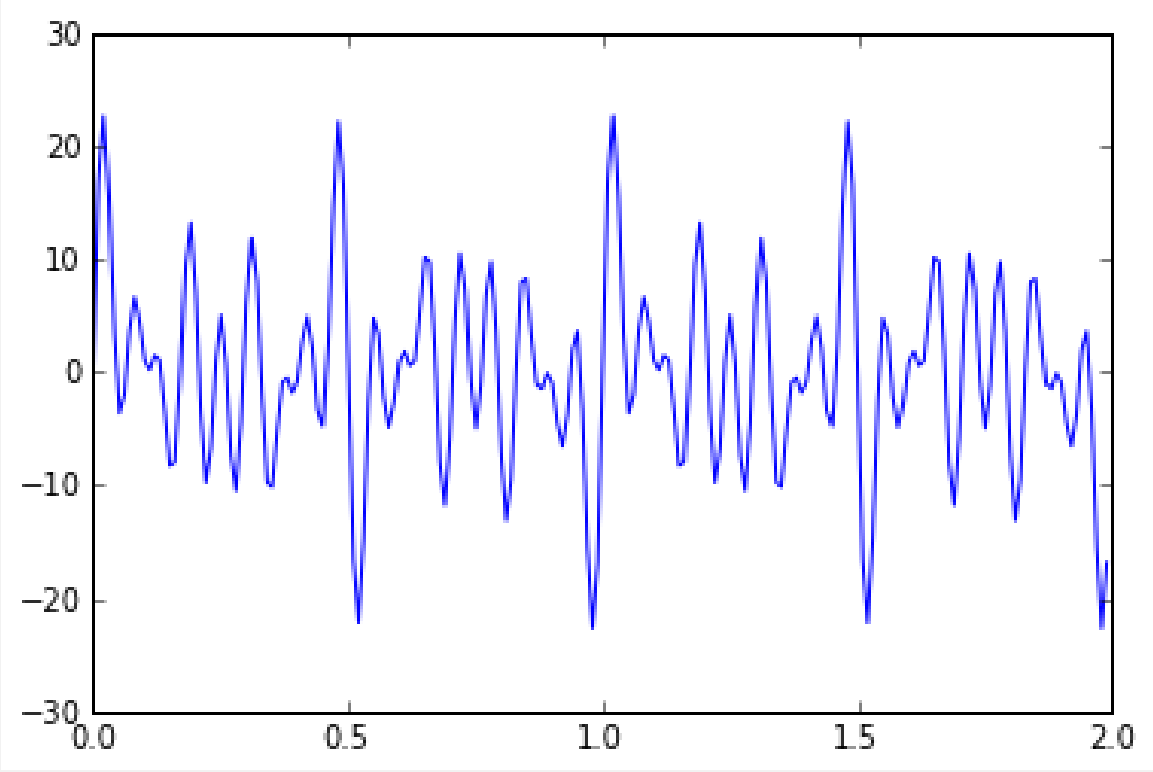
```
amplitude = array([[1,1,1,1,1,1]])
frequency = array([[1,3,5,7,9,11]])
phase = array([[0,0,0,0,0,0]])
y = sin(2*pi*domain.T.dot(frequency)+one_vector.T.dot(phase)).dot(amplitude.T)
plot(x,y)
```

```
[<matplotlib.lines.Line2D at 0xc4178d0>]
```




```
amplitude = array([[1,2,3,4,5,6,7]])
frequency = array([[2,3,5,7,11,13,17]])
phase = array([[0,0,0,0,0,0,0]])
y = sin(2*pi*domain.T.dot(frequency)+one_vector.T.dot(phase)).dot(amplitude.T)
plot(x,y)
```

```
[<matplotlib.lines.Line2D at 0xc836230>]
```




```
from traits.api import HasTraits,Array, Instance
from traitsui.api import View, Item
from chaco.api import HPlotContainer, ArrayPlotData, Plot
from enable.component_editor import ComponentEditor
```

```
class sinsum(HasTraits):
    plot = Instance(Plot)
    traits_view = View(Item('plot',editor=ComponentEditor(), show_label=False),
                       width=1000, height=600, resizable=True, title="Chaco Plot")

    def __init__(self):
        super(sinsum, self).__init__()
        plotdata = ArrayPlotData(x = domain[0], y = sum_list_array(sin(2*pi*period.dot(
            domain))))
        img = Plot(plotdata)
        img.plot(("x","y"),type='line')
        self.plot = img
        return
```