Class12 – Fourier Analysis

Fourier Series
uses linear combination of cosine and sine waves to express any periodic function

Fourier Transform
decomposes functions of time (space) domain into frequency (wave number) domain

Fast Fourier Transform
is a computational algorithm to calculate Fourier Transform of discrete data very efficiently

1

Class12 – Fourier Analysis

Fourier Series for 2π periodic continuous function

2π periodic : f(x) = f(x+2π)

$$f(x) = a_0 + a_1 cosx + b_1 sinx + a_2 cos2x + b_2 sin2x + \cdots$$

1. f(x) is a weighted, infinite sum of sine and cosine functions with increasing frequency
2. In practice, the infinite sum is often stopped (truncated) at a manageable level
3. Derivatives of f(x) become very easy to calculate

2

Class12 – Fourier Analysis

Fourier Series for 2π periodic continuous function

Formula for Fourier coefficients

$$a_0 = \overline{f(x)} \qquad a_n = \frac{1}{\pi}\int_0^{2\pi} f(x)cos(nx)dx$$

$$b_n = \frac{1}{\pi}\int_0^{2\pi} f(x)sin(nx)dx$$

3

Class12 – Fourier Analysis

Fourier Series for T periodic function

T periodic : f(x) = f(x+T).

$$f(x) = a_0 + a_1 cos\frac{2\pi x}{T} + b_1 sin\frac{2\pi x}{T} + a_2 cos\frac{4\pi x}{T} + b_2 sin\frac{4\pi x}{T} + \cdots$$

Practically, your data does not have to be periodic as long as the signal of your interest is within T period.

4

Class12 – Fourier Analysis

Fourier Series for T periodic function

Formula:

$$a_0 = \overline{f(x)} \qquad a_n = \frac{2}{T}\int_{t_0}^{t_0+T} f(x)cos\frac{2\pi nx}{T}dx$$

$$b_n = \frac{2}{T}\int_{t_0}^{t_0+T} f(x)sin\frac{2\pi nx}{T}dx$$

5

Class12 – Fourier Analysis

Discrete Fourier Transform (DFT)

You have discrete dataset: f = f(x) = [f_1 f_2 f_3 · · · f_N]

Rather than a continuous function like f(x), we normally work with discrete samples (vector). f(x), does not have to be periodic as we are only interested in the period within the data record.

How do we calculate Fourier coefficients?

6

Class12 – Fourier Analysis

Fast Fourier Transform (FFT)

FFT operates the Fourier matrix (F) onto the data vector very efficiently. The data in physical space (time/space) is transformed into frequency space as Fourier coefficients. FFT algorithm is available in MATLAB (fft) and python (numpy.fft / scipy.fft)

7