EE2703 : Applied Programming Lab
Assignment - 4

Jawhar S
EE20B049

March 11, 2022

Plotting The Functions

Firstly, we plot two functions, e^x and $cos(cos(x))$ over the interval $[-2\pi, 4\pi)$, using 400 points sampled uniformly in the interval.

```
x=np.linspace(-2*np.pi,4*np.pi,400)
```

Since, e^x grows rapidly, we use a semi-log scale for plotting it.

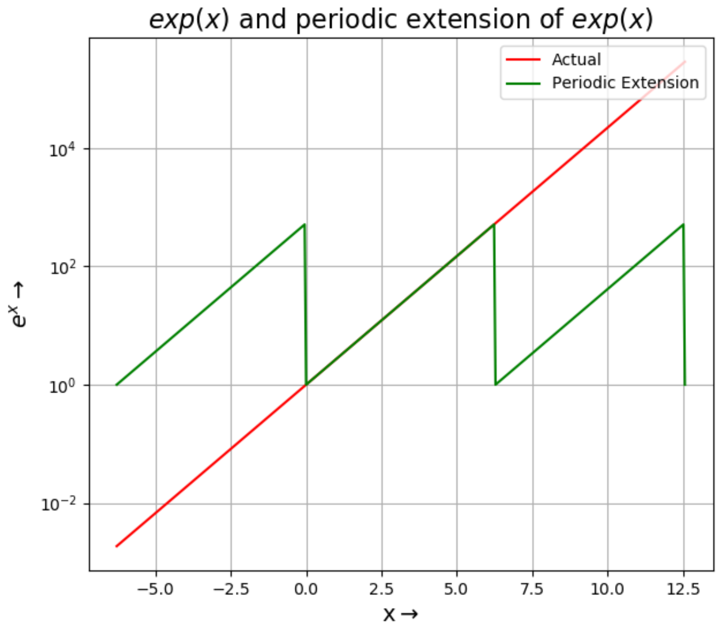

Figure 1: Semi-Log plot of exp(x) and periodic extension of exp(x)

As we can observe from Figure 1 above, the function e^x is a non-periodic increasing function. To calculate the Fourier series coefficients, we create a new function which is equal to e^x in the range $[0, 2\pi)$ and is also 2π periodic.

As we can observe from Figure 2 below, the function $cos(cos(x))$ is a periodic function with fundamental period π. So it is sufficient to plot the function on a linear scale.

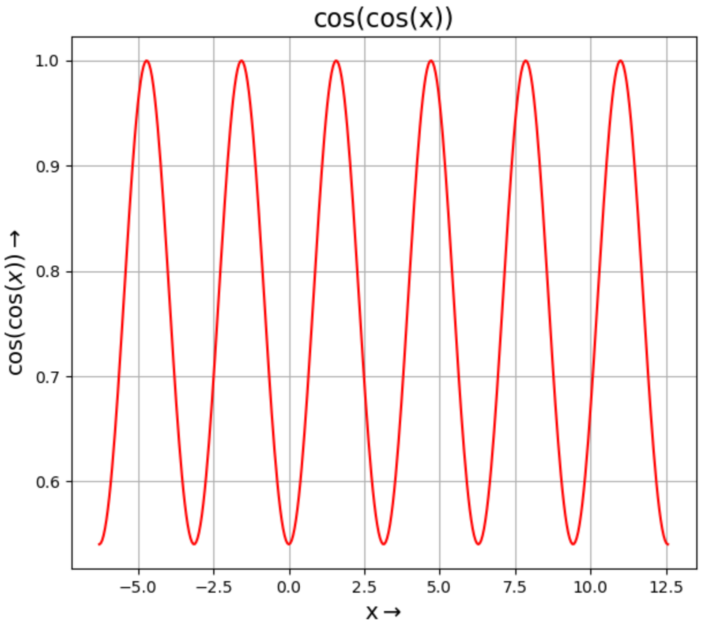

Figure 2: Plot of cos(cos(x))

Fourier Coefficients

The Fourier series of a periodic function f(x) is the expansion of the function expressed as an infinite sum of sines and cosines. The coefficients of these sines and cosines are the Fourier coefficients.

$$f(x) = a_0 + \sum_{n=1}^{\infty} a_n cos(nx) + b_n sin(nx)$$

The coefficients a_n and b_n are given by,

$$a_0 = \frac{1}{2\pi} \int_0^{2\pi} f(x)dx$$

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

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

```
def get_coefficients(n,func):
    coefficients=np.zeros(n)
    func_str=func_map[func]
    # DC coefficient
    coefficients[0]=quad(func_str,0,2*np.pi)[0]/(2*np.pi)
    # Cosine coefficients
    for k in range(1,n,2):
        coefficients[k]=quad(u,0,2*np.pi,args=((k+1)/2, func_str))[0]/np.pi
    # Sine coefficients
    for k in range(2,n,2):
        coefficients[k]=quad(v,0,2*np.pi,args=(k/2, func_str))[0]/np.pi
    return coefficients
```

The above function is used to find the first 'n' Fourier coefficients.

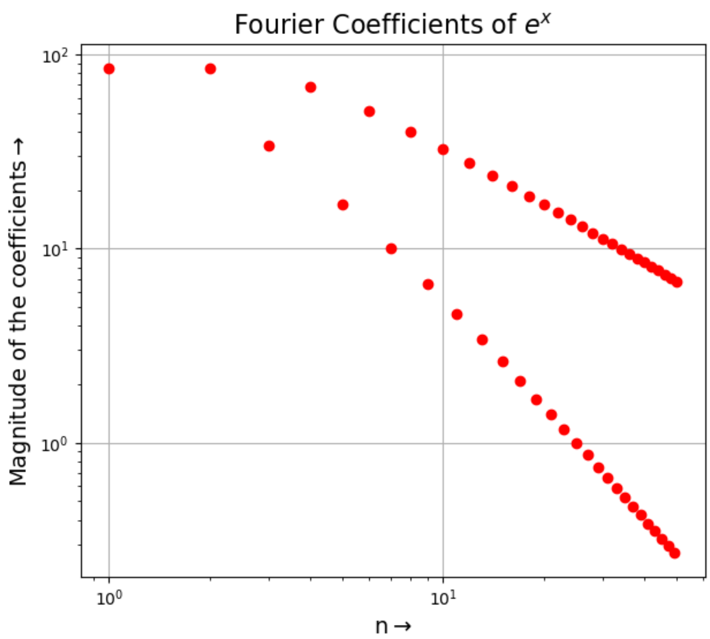

Figure 3: Log-Log plot

Figure 3 and 4 contain the first 51 Fourier coefficients of e^x plotted in a log-log and semi-log scale respectively. Figure 5 and 6 contain the first 51 Fourier coefficients of $cos(cos(x))$ plotted in a log-log and semi-log scale respectively.

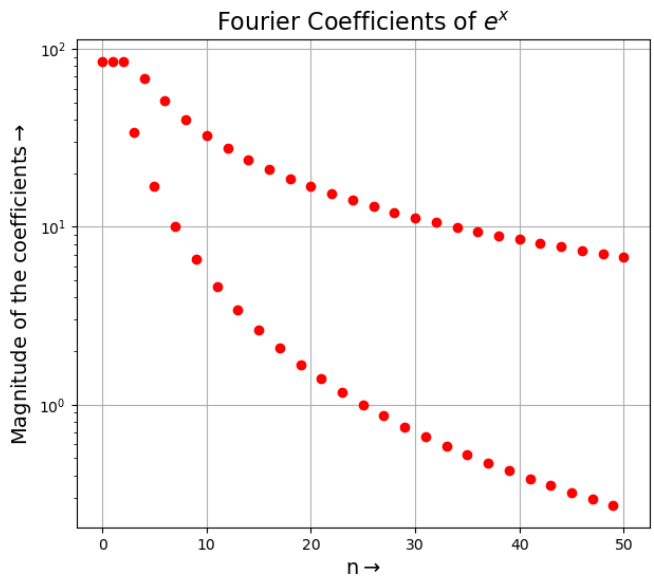

Figure 4: Semi-Log plot

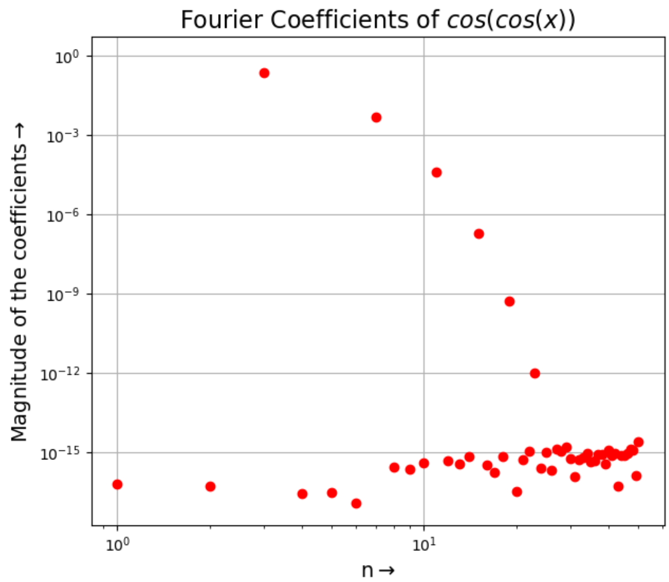

Figure 5: Log-Log plot

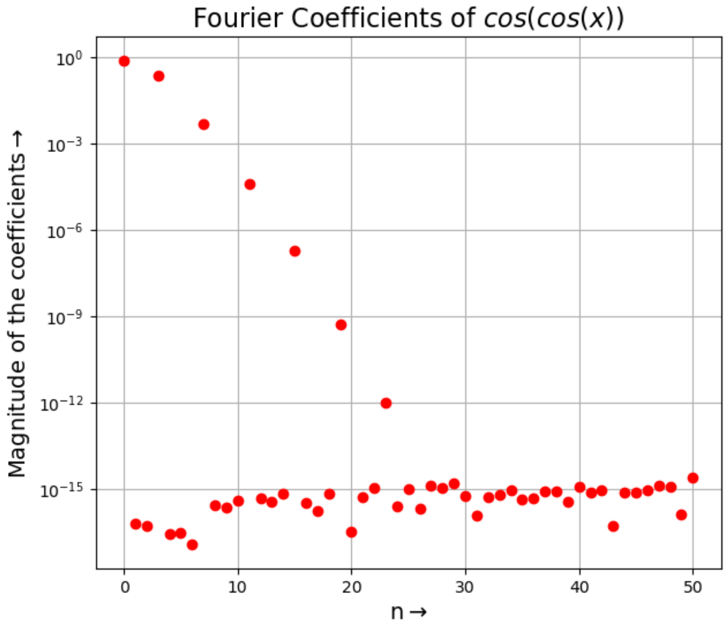

Figure 6: Semi-Log plot

The b_n coefficients for the function $cos(cos(x))$ are expected to be zero as cos(cos(x)) is an even function but we notice small non-zero values because of the limitation in the accuracy of which the value of π can be stored in the memory.

The function e^x has an exponentially increasing slope which results in a wide range of frequencies in its Fourier series. The magnitude of the Fourier series coefficients of e^x is inversely proportional to n^2. For large values of n, $log(a_n)$ and $log(b_n)$ are approximately equal to $-2log(n)$ which results in a nearly linear plot in the Log-Log scale.

The function cos(cos(x)) has a relatively low frequency due to which contributions by higher sinusoids are less, leading to a quick decay of the magnitude of the Fourier series coefficients which results in a nearly linear plot in the Semi-Log scale.

Least Squares

The Fourier series coefficients can be approximated using Least Squares.

Below is the code for estimating the Fourier coefficients using Least Squares method.

```
# Least Squares method of estimating the fourier coefficients
x=np.linspace(0,2*np.pi,401)
x=x[:-1]
y=np.linspace(0,2*np.pi,400)
A=np.zeros((400,51))
A[:,0]=1
for k in range(1,26):
    A[:,2*k-1]=np.cos(k*x)
    A[:,2*k]=np.sin(k*x)
b_exp=exp(x)
b_coscos=coscos(x)
c_exp=np.linalg.lstsq(A,b_exp,rcond=None)[0]
c_coscos=np.linalg.lstsq(A,b_coscos,rcond=None)[0]
```

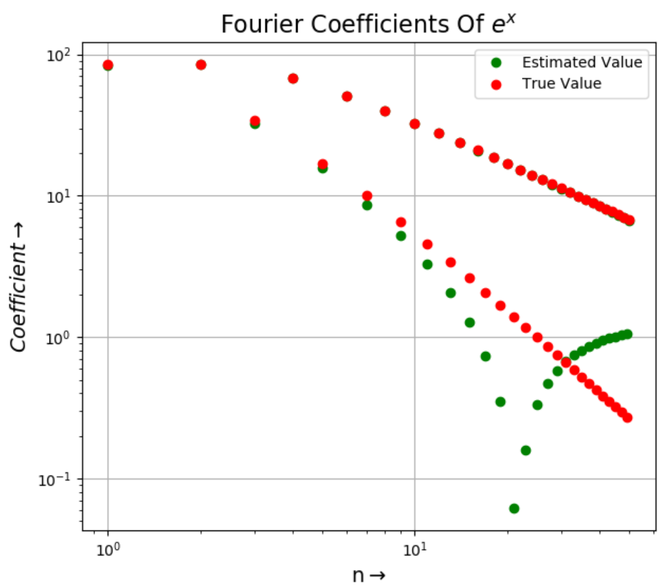

Figure 7: Log-Log plot

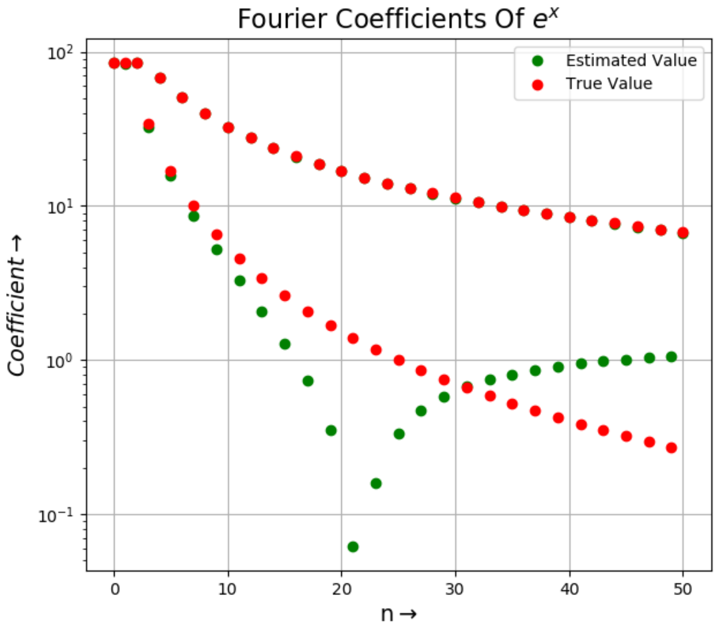

Figure 8: Semi-Log plot

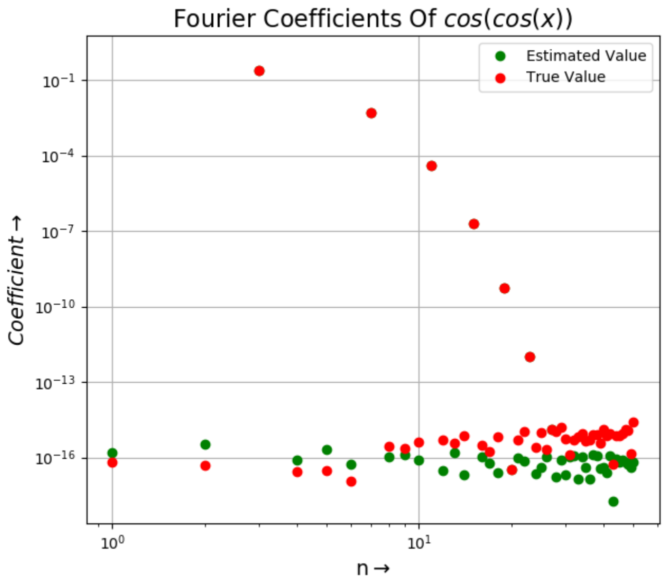

Figure 9: Log-Log plot

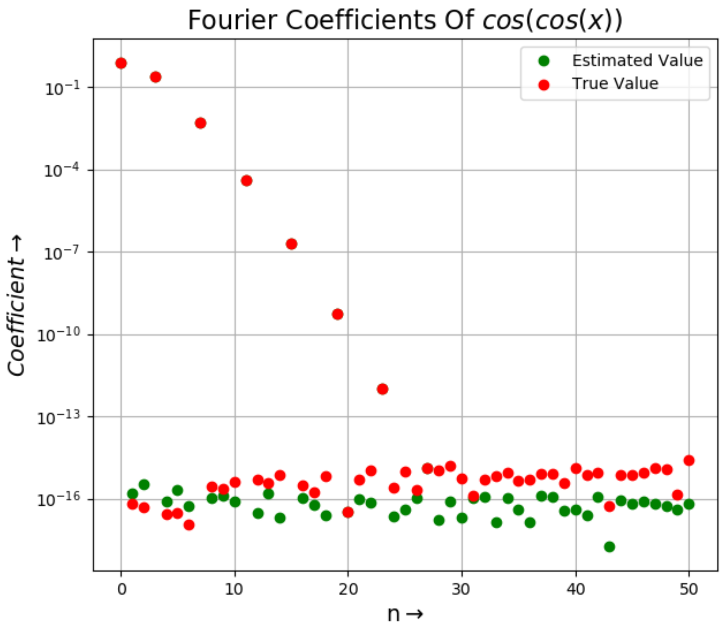

Figure 10: Semi-Log plot

Figure 7 and 8 contain true and estimated values of the first 51 Fourier coefficients of e^x plotted in a log-log and semi-log scale respectively. Figure 5 and 6 contain true and estimated values of the first 51 Fourier coefficients of $cos(cos(x))$ plotted in a log-log and semi-log scale respectively.

In case of $cos(cos(x))$ the coefficients obtained by Least Squares are close to the co- efficients obtained from the direct formula, whereas in case of e^x the coefficients obtained by Least Squares differ drastically to the co- efficients obtained from the direct formula. This is be- cause $cos(cos(x))$ is a periodic signal whereas e^x is a non-periodic signal with an exponentially growing slope.

The deviation in the Fourier coefficients of e^x and $cos(cos(x))$ which are found using Least Squares and using direct formula can be computed using the below code.

```
max_exp_deviation=np.max(np.abs(exp_coeffs-c_exp))
max_coscos_deviation=np.max(np.abs(coscos_coeffs-c_coscos))
```

```
Largest Deviation in fourier coefficients of exp(x): 1.3327308703353964
Largest Deviation in fourier coefficients of cos(cos(x)): 2.6621696286823105e-15
```

Figure 11: Deviation between true and estimated Fourier coefficients

Plotting the Functions using the Estimated Coefficients

The Fourier Approximation of $cos(cos(x))$ is very similar to the actual function. This is due to the fact that the function $cos(cos(x))$ is periodic, resulting in lesser deviation from its true value. In case of e^x, the function has a exponentially increasing slope and the higher frequencies of the Fourier series contribute significantly to the true value because of which the Fourier Approximation of e^x deviates from its true value. For a better estimate of e^x we would need to consider higher sinusoidal frequencies.

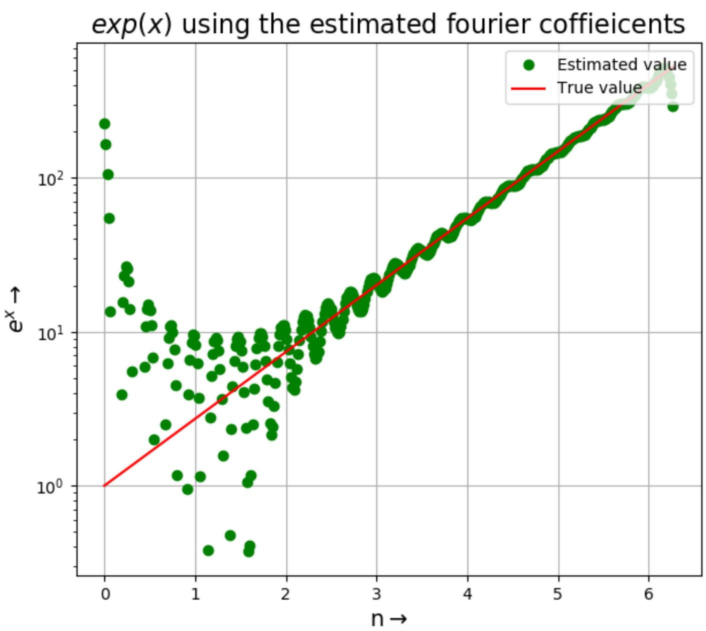

Figure 12: Semi-Log plot

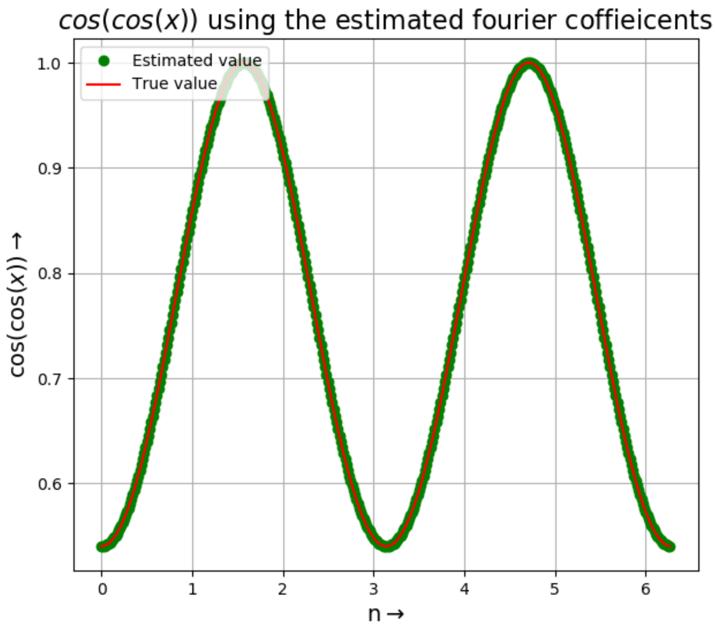

Figure 13: Normal plot

Conclusion

From the assignment, it can be observed that the Fourier series coefficients for any functions can be found out in two ways i.e direct integration and least squares.

We can even observe that the least square approach deviates more for non-periodic functions with exponentially increasing slope. For periodic function the least square approach gives similar results to that obtained from direct integration.

Least Squares approach can be used to calculate the Fourier series coefficients as it is computationally inexpensive and the values are very close to the ones obtained from direct integration. For exact values of the Fourier coefficients direct integration would be preferred.