# Assignment No 4

Sumanth R Hegde, EE17B032

February 26, 2019

## Function Definition and Visualization

We need to ensure that the function can output correct values even for an $N$-dimensional vector as an input. The built-in functions in `numpy` can be used to accomplish the same :

```
import numpy as np
def exp(x):
    return np.exp(x)
def coscos(x) :
    return np.cos(np.cos(x))
```

The above functions are plotted over the interval $[-2\pi, 4\pi)$ using 300 points sampled uniformly over this interval. The following plots are obtained :

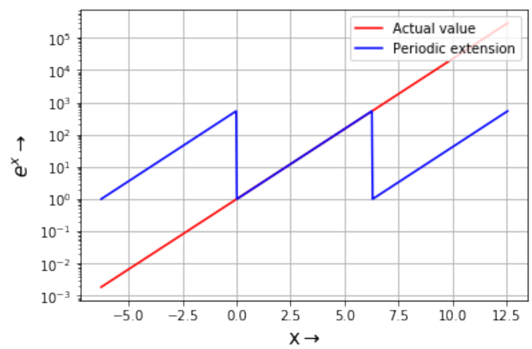


Figure 1: Semilog plot of $e^x$

The function $e^x$ is ever increasing with $x$ and is non-periodic. However for the evaluation of the Fourier series, the function is made $2\pi$ periodic. $cos(x)$ is periodic and thus $cos(cos(x))$ is also periodic. Since $cos(-x) = cos(x)$, the period of $cos(cos(x))$ is $\pi$ .

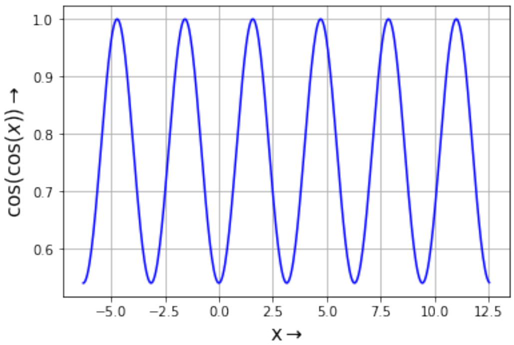


Figure 2: Plot of cos(cos(x))

The Fourier series gives us the expression for the function in terms of $2\pi$ periodic sinusoids .For an aperiodic function, it is calculated by looking at the function in the interval $[0, 2\pi)$ and repeating the function pattern so that we obtain a $2\pi$ periodic function for which the Fourier series can be evaluated. $e^x$ is a function with a swift increase and this can be represented only by high frequency sinusoids. The Fourier series would only yield a reasonable approximation even for sufficiently large number of coefficients.
Since $cos(cos(x))$ has a period of $\pi$ , we can expect the Fourier series expansion to yield an almost exact approximation for the function.

# The Fourier Series Coefficients

The Fourier series coefficients are obtained using the `quad` function in the `scipy` library. The following code snippet evaluates the first $n$ coefficients of the Fourier series expansion of $e^x$ and $cos(cos(x))$ :

```
func_dict = {'exp(x)':exp,'cos(cos(x))': coscos}
def find_coeff(n,label):
    coeff = np.zeros(n)
    func = func_dict[label]
    u = lambda x,k: func(x)*np.cos(k*x)
    v = lambda x,k: func(x)*np.sin(k*x)
    coeff[0]= quad(func,0,2*pi)[0]/(2*np.pi)
    for i in range(1,n,2):
        coeff[i] = quad(u,0,2*pi,args=((i+1)/2))[0]/np.pi
    for i in range(2,n,2):
        coeff[i] = quad(v,0,2*pi,args=(i/2))[0]/np.pi
    return coeff
```

We thus obtain the first 51 coefficient of the two functions. A plot of the coefficients of $e^x$ in a *semilog* and *loglog* scale is :

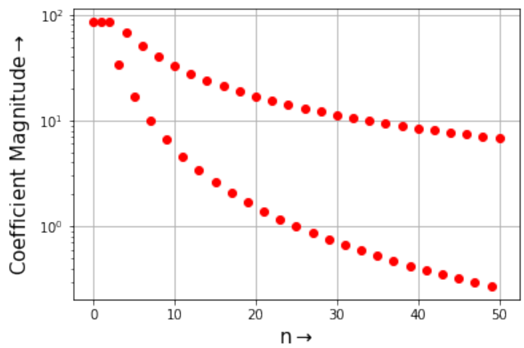


Figure 3: Semilog plot of the coefficients of $e^x$

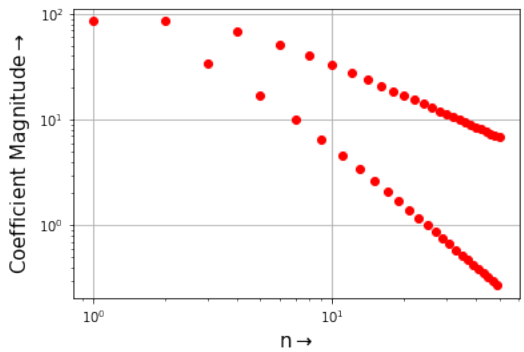


Figure 4: Loglog plot of the coefficients of $e^x$

Similarly, the plots for $cos(cos(x))$ are as follows :

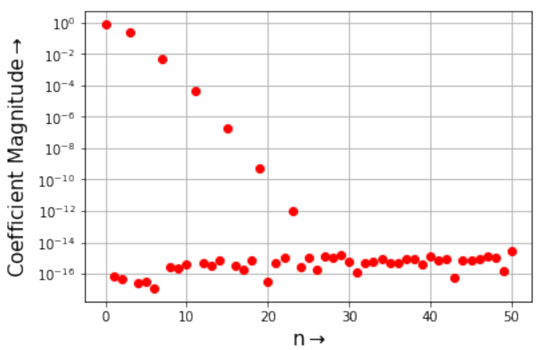


Figure 5: Semilog plot of the coefficients of $cos(cos(x))$

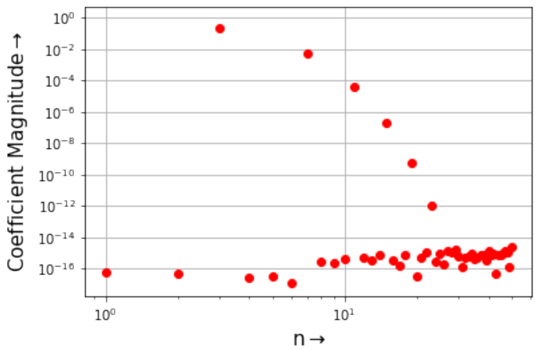


Figure 6: Loglog plot of the coefficients of $cos(cos(x))$

We notice that the $b_n$ coefficients in the second case are expected to be zero as the function $cos(cos(x))$ is even and thus the odd sinusoidal components are zero. The values obtained are non-zero because of the limitation in the numerical accuracy upto which $\pi$ can be stored in memory.
In the first case, the function, having an exponentially increasing gradient, contains a wide range of frequencies in its fourier series. In the second case, $cos(cos(x))$ has a relatively low frequency of $\frac{1}{\pi}$ and thus contribution by the higher sinusoids is less, manifesting in the quick decay of the coefficients with $n$.
The magnitude of the coefficients of $e^x$ vary with $n$ as follows :

$$\mid a_n \mid, \mid b_n \mid \ \ \propto \ \ \frac{1}{n^2+1}$$

Thus, for large $n$ ,

$$log(\mid a_n \mid), log(\mid b_n \mid) \ \ \propto \ \ \log(\frac{1}{n^2+1}) \approx -2\log n$$

Thus, the *loglog* plot becomes approximately linear as $n$ increases.
Similarly, the coefficients of $cos(cos(x))$ would be approximately exponential with $n$ . Thus the *semilog* plot is linear.

## The Least Squares Approach

The Fourier coefficients can also be found by solving a matrix equation using linear regression. The matrix equation is

$$Ac = b \tag{1}$$

where

$$A = \begin{pmatrix} 1 & cosx_1 & sinx_1 & .... & cosx_{400} & sinx_{400} \\ 1 & cosx_2 & sinx_2 & .... & cosx_{400} & sinx_{400} \\ ..... & ..... & ..... & ..... & ..... & ..... \\ 1 & cosx_3 & sinx_4 & .... & cosx_{400} & sinx_{400} \\ 1 & cosx_4 & sinx_4 & .... & cosx_{400} & sinx_{400} \end{pmatrix} b = \begin{pmatrix} f(x_1) \\ f(x_2) \\ ..... \\ f(x_{400}) \end{pmatrix}$$

The coefficient matrix $c$ can be estimated by minimizing the $L_2$ norm. The plots are obtained :

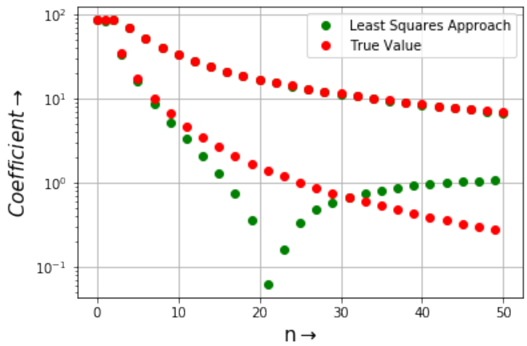


Figure 7: Semilog plot of coefficients of $e^x$

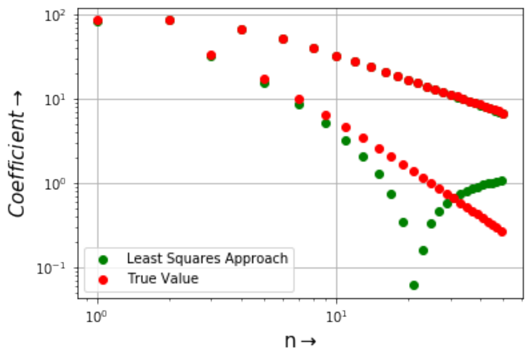


Figure 8: Loglog plot of coefficients of $e^x$

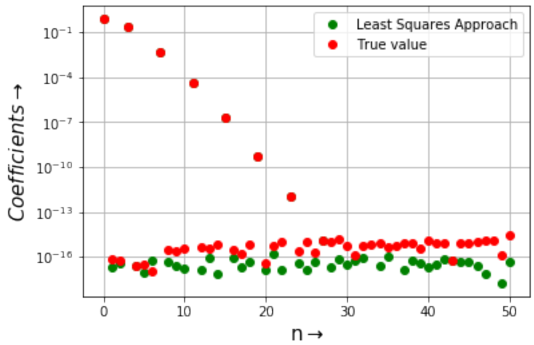


Figure 9: Semilog plot of coefficients of $cos(cos(x))$

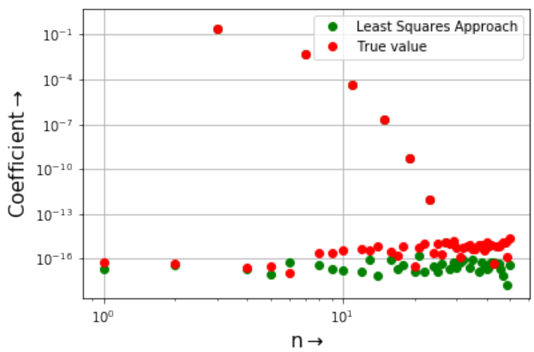


Figure 10: Loglog plot of coefficients of $cos(cos(x))$

It can be seen that the coefficients for $cos(cos(x))$ are in much closer agreement than the coefficients for $e^x$ and is also expected as the former is periodic and the periodic extension of the latter has an increasing gradient and would require higher sinusoids for a better representation. Since the least squares approach is only approximate, the deviation is expected.

The maximum deviation in the coefficient magnitude can be found as follows :

```
dev_exp = abs(coeff_exp - c_exp)
dev_cos = abs(coeff_cos - c_cos)
max_dev_exp = np.max(dev_exp)
max_dev_cos = np.max(dev_cos)
```

where, `coeff_exp` and `coeff_cos` are the values obtained through integration and `c_cos`, `c_exp` are the values obtained through the least squares approach.

The maximum deviations obtained are :

```
max_dev_exp : 1.332730870335368
max_dev_cos : 2.553315730564766e-15
```

The least squares estimation method provides a faster way of evaluating a large number of Fourier coefficients. In this case of evaluating 51 coefficients, the time taken to evaluate the coefficients in the direct approach was almost 10 times that taken for the least squares approach.

## Function Approximation

The matrix product

$$M = Ac$$

Gives an estimate of the function values at $\{x_1, x_2, ........x_{400}\}$ chosen earlier. These estimates, along with the actual value of the functions yield the following plots :

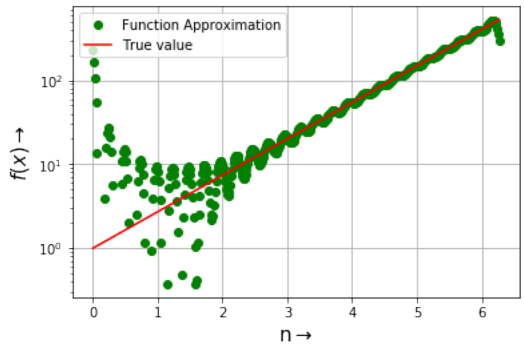


Figure 11: Plot of $e^x$ and its Fourier series approximation

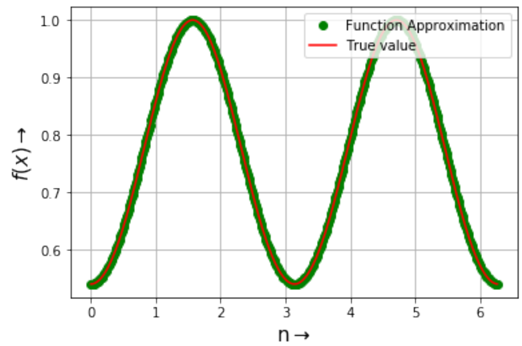


Figure 12: Plot of $cos(cos(x))$ and its Fourier series approximation

There is significant deviation in Fig.11 . In order to get a better estimate, we would need to consider high frequency sinusoids. In case of Fig.12 the function has a period of $\pi$ and thus can be almost *exactly* represented in terms of low frequency, $2\pi$ periodic sinusoids.

## Conclusion

The Fourier series coefficients of $e^x$ and $cos(cos(x))$ were found in two ways: direct integration and estimation using the least squares approach. It was verified that the odd sinusoidal components of $cos(cos(x))$ are nearly zero. The least squares approach provided a more computationally inexpensive way of estimating the coefficients. It was found that the least squares estimate deviated more for $e^x$ than $cos(cos(x))$, due to the high gradient characteristics of the former and periodic behaviour of the latter.