Assignment:4 Fourier Series Coefficients

Sharanesh R(EE19B116)

March 10, 2021

Abstract

In this assignment we aim to:

- Approximate few functions using the Fourier Series
- Find the Fourier Series coefficients using the two methods namely:
  1. Direct Integration
  2. Least Squares Method
- Also compare the Fourier series coefficients obtained in the above two methods and analyze how close they are to the actual functions.

  The functions that we take up for analysis are:

  - $cos(cos(x))$
  - e^x

1 Analysis of the Functions $cos(cos(x))$ and e^x

We create two python functions `cos(cos(x))` and `exp(x)` which returns the functions value for the given input. We define a vector x that stores the input values and in our case, we span the input from -2π to 4π, excluding the value 4π, since we have included the value 2π previously.

We now pass this vector x to the above defined functions and plot the same, the plots looks as shown below:

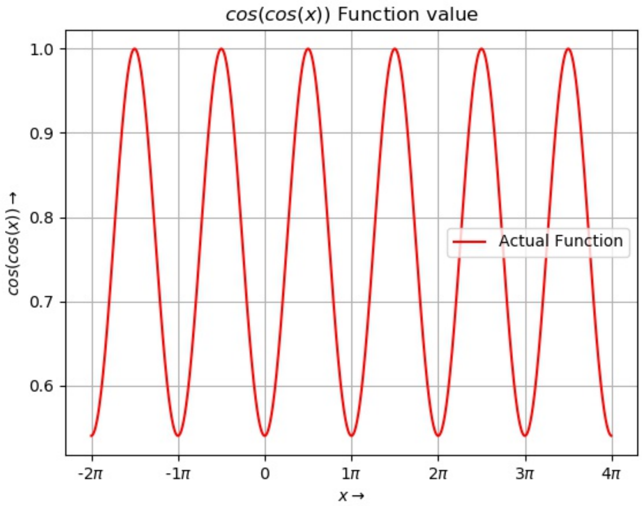

Figure 1: True function value of $cos(cos(x))$

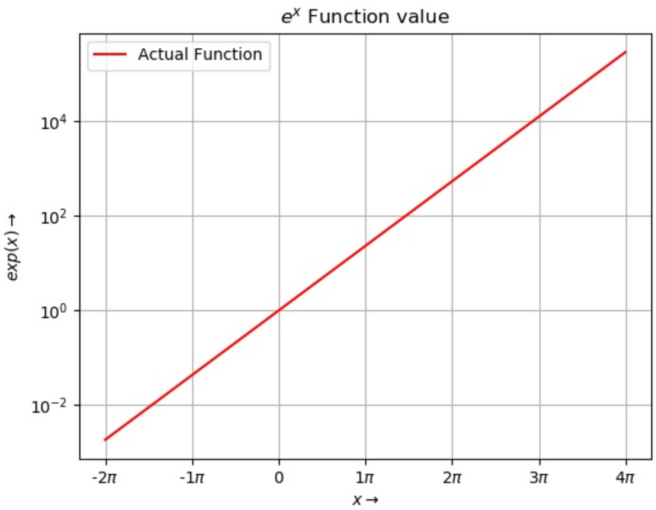

Figure 2: True function value of $exp(x)$

From the above two figures namely Figure 1 and Figure 2 it is evident that the function $cos(cos(x))$ is periodic while the function $exp(x)$ is aperiodic.

Fourier series approximations are valid only for periodic signals and so, we make a 2π periodic extension (making repeated copies of the same function after every 2π interval) of the functions given by `f(x mod 2π)` and plot it as shown below:

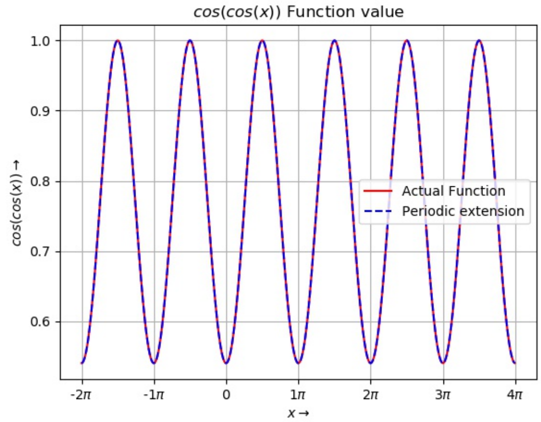

Figure 3: True function value and periodic extension of $cos(cos(x))$

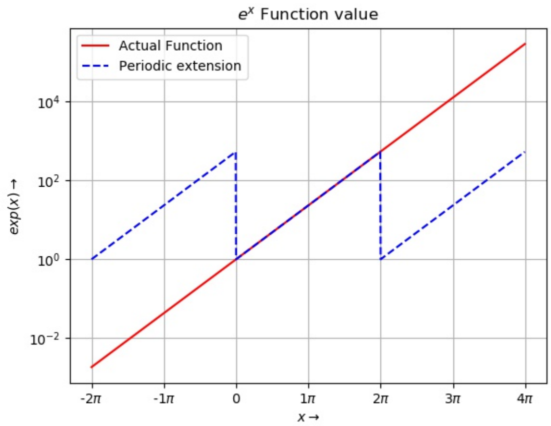

Figure 4: True function value and periodic extension of $exp(x)$

2 Fourier Series Coefficients

For any given periodic function $f(x)$, it's Fourier series is given by the following equation:

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

where,

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

$$a_n = \frac{1}{\pi}\int_0^{2\pi} f(x)cos(kx)dx \tag{3}$$

$$b_n = \frac{1}{\pi}\int_0^{2\pi} f(x)sin(kx)dx \tag{4}$$

Now in order to compute the coefficients from the above mentioned equations (2), (3), and (4), we use the `quad()` function in python to perform integration and determine the first 51 Fourier Series coefficients (a_0,a_1 to a_{25},b_1 to b_{25}) for both the functions that we are analysing namely $cos(cos(x))$ and $exp(x)$.

The semilog and loglog plots of the coefficients of the function $exp(x)$ is shown below in Figure 5 and Figure 6 respectively, and the semilog and loglog plots of the coefficients of the function $cos(cos(x))$ is shown below in Figure 7 and Figure 8 respectively:

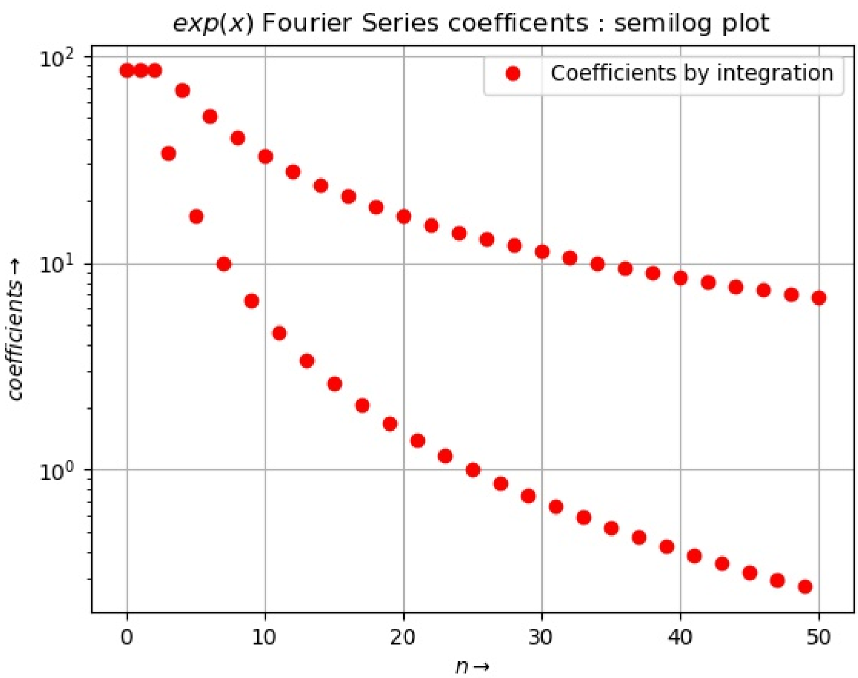

Figure 5: Semilog plot of $exp(x)$ Fourier Series coefficents

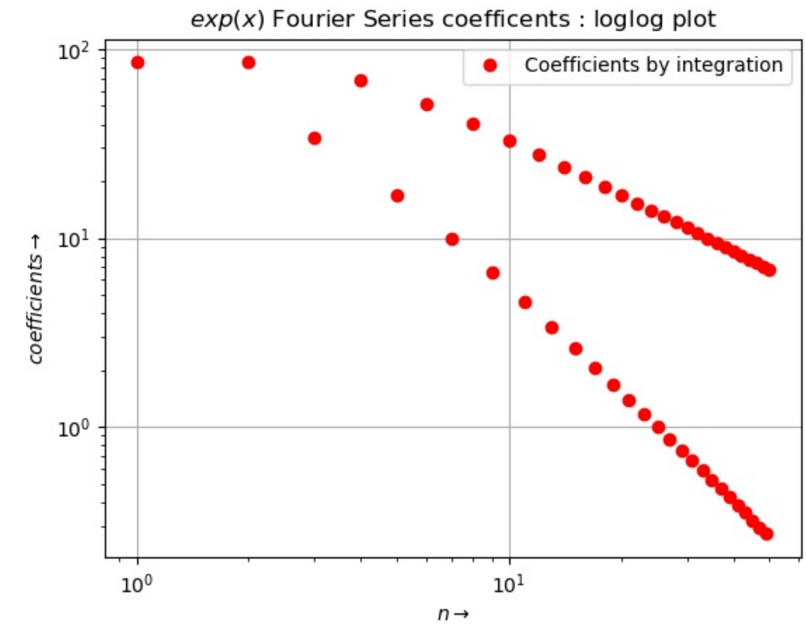

Figure 6: loglog plot of $exp(x)$ Fourier Series coefficents

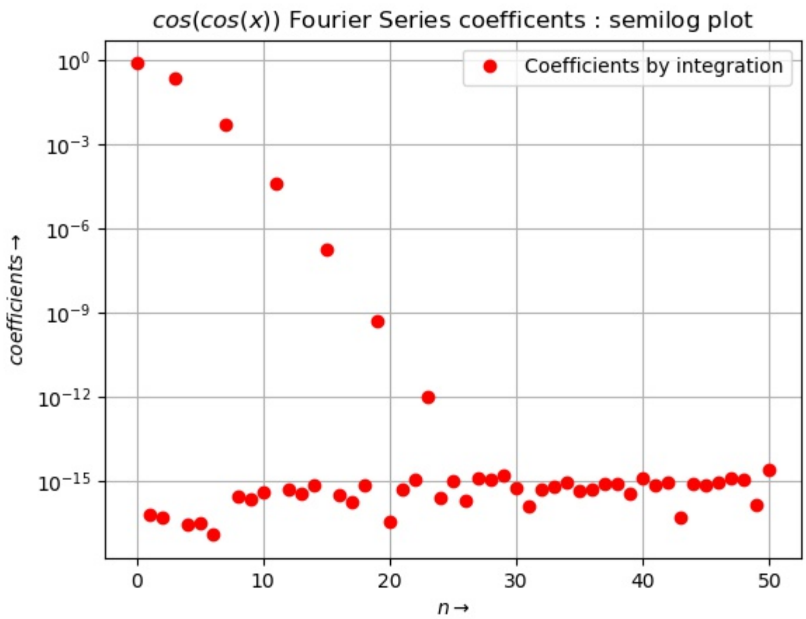

Figure 7: Semilog plot of $cos(cos(x))$ Fourier Series coefficents

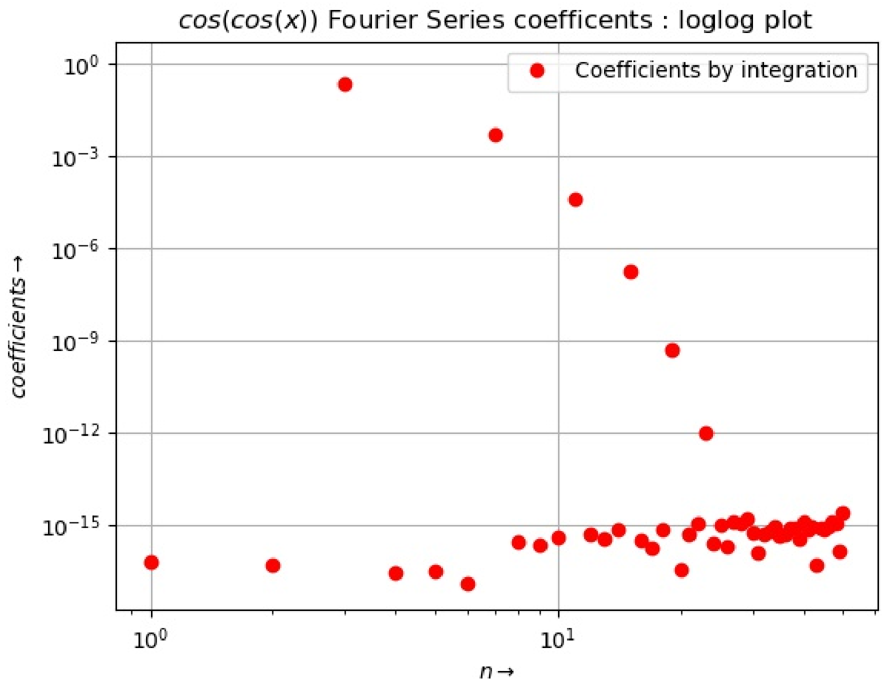

Figure 8: loglog plot of $cos(cos(x))$ Fourier Series coefficents

From the above plots we can make the following conclusions:

1. b_n is nearly zero for $cos(cos(x))$. This is because $cos(cos(x))$ is an even function and hence the integrand in the expression for finding b_n is of odd nature and so it turns out to be zero for all values of n. However, it doesn't exactly turn out to be zero because of the approximations involved in the python *quad()* function used for integrating.

2. The rate of decay of the Fourier coefficients is determined by the smoothness of the function and since $cos(cos(x))$ is continuous, the coefficients decay at a faster rate. On the other hand, the function $exp(x)$ is discontinuous and so the coefficients decay at a very slow rate.

3. The loglog plot is linear for $exp(x)$ since the Fourier coefficients of $exp(x)$ decay at the rate of $1/n^2$. The semilog plot seems to be linear in the $cos(cos(t))$ case as its Fourier coefficients decay exponentially with n.

3 Least Squares Approach

As we did in the last assignment, we apply *Least Squares Approach* to find the best fit Fourier series coefficients of the functions using the `lstsq()` in the `scipy.linalg` package.

For doing so, we define the vector x from 0 to 2π and evaluate the function at those x values and define it as the matrix b, and now the function is to be approximated by (1), so for each x_i we want

$$a_0 + \sum_{n=1}^{\infty} a_n \cos(nx_i) + \sum_{n=1}^{\infty} b_n \sin(nx_i) \approx f(x_i) \tag{5}$$

This becomes a matrix problem of the form

$$Ac = b \tag{6}$$

where the A is the coefficient matrix and, c is the Fourier coefficients which we obtain from the `lstsq()` function.

We now compare the Fourier series coefficients that has been obtained by the two methods *Direct Integration* and *Least Squares Approach* in both plot them both semilog and loglog scale. The plots are shown below:

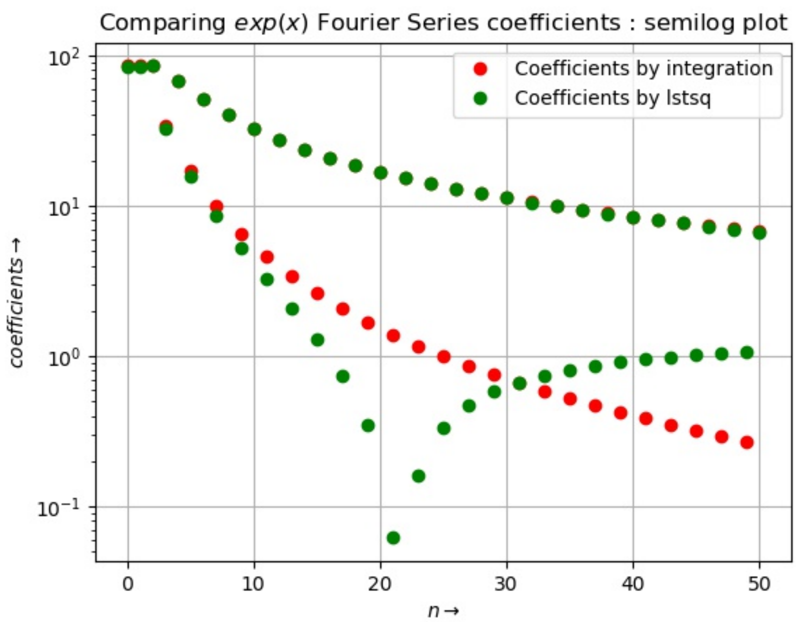

Figure 9: Semilog plot of comparison of $exp(x)$ Fourier Series coefficents

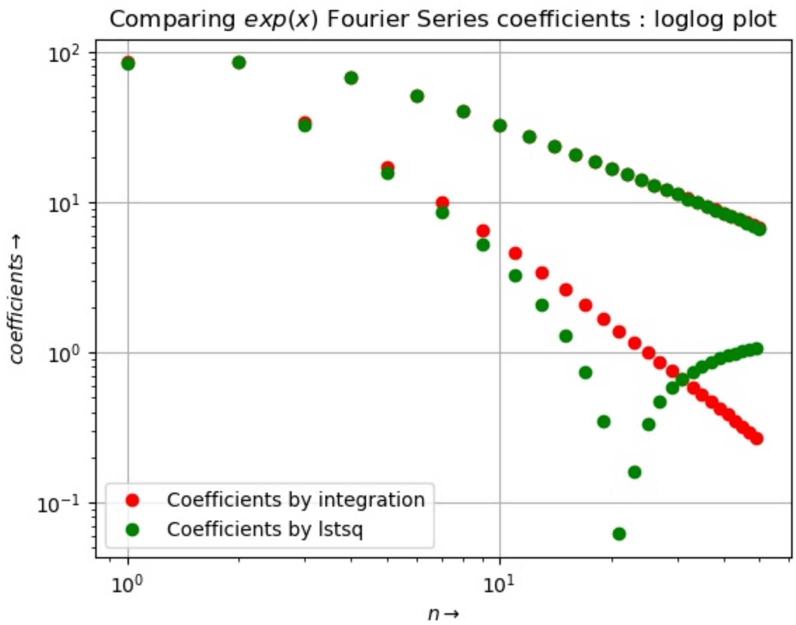

Figure 10: loglog plot of comparison of $exp(x)$ Fourier Series coefficents

Since the Least Squares Method is a kind of approximation technique, it will have some deviation from the actual value of the coefficients obtained using Integration.

- The maximum deviation between the coefficients in the case of $exp(x)$ is 1.33273087033532 53.
- The maximum deviation between the coefficients in the case of $cos(cos(x))$ is 2.6448494989768546e-15.

By looking at the above deviation values we see that, there is very good concurrence in the values in the case of $cos(cos(x))$ but there is quite a large deviation in the case of $exp(x)$.

The reason for this is that the periodic extension of the exponential function is discontinuous, and hence it would require a lot more samples to accurately determine its Fourier coefficients.

So, if we increase the number of sample points then the deviation tends to decrease. This effect of lack of samples is seen more near the points of discontinuities of the signal.

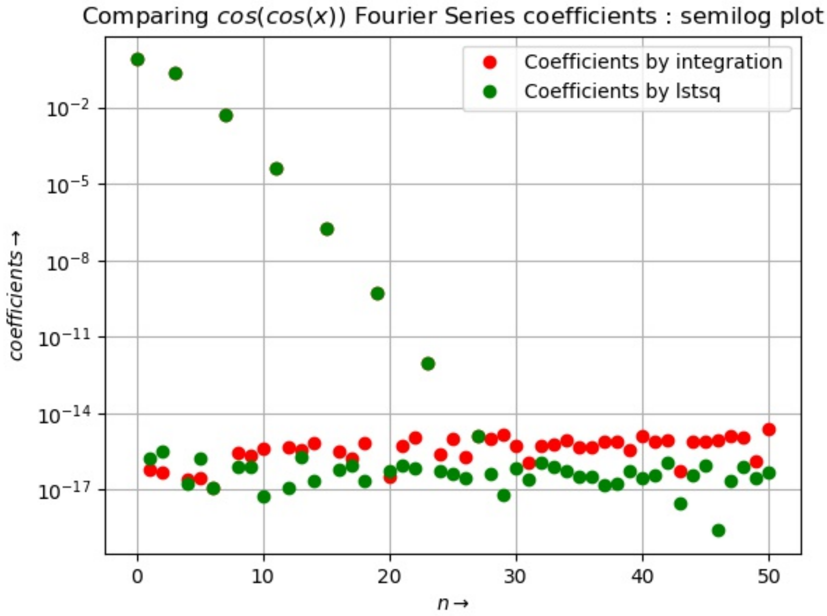

Figure 11: Semilog plot of comparison of $cos(cos(x))$ Fourier Series coefficents

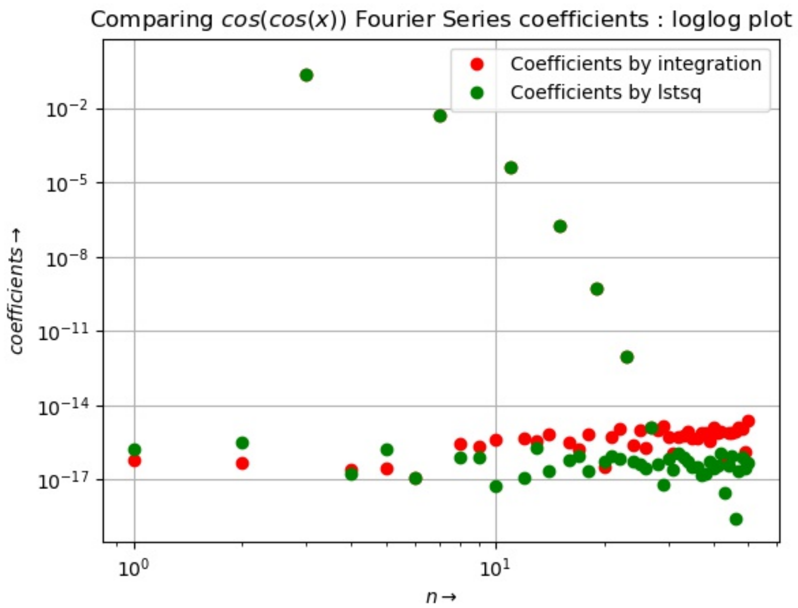

Figure 12: loglog plot of comparison of $cos(cos(x))$ Fourier Series coefficents

4 Estimated Functions

Now, using the predicted values of the Fourier coefficients from the *Least Squares Approach*, we compute the functional values of both the functions $cos(cos(x))$ and $exp(x)$.

The plots containing both the actual functional value and the estimated functional value is shown below:

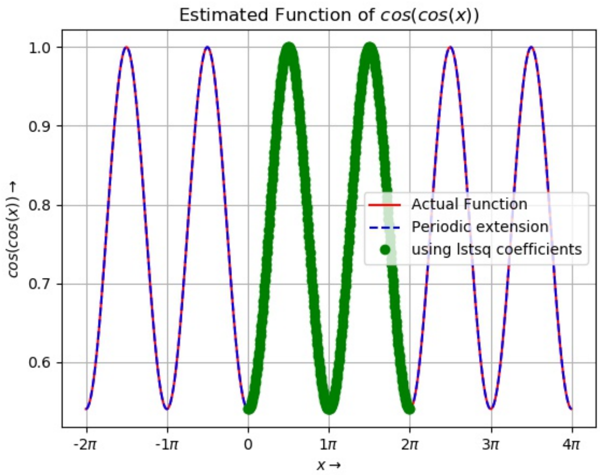

Figure 13: True function value and periodic extension and estimated function value of $cos(cos(x))$

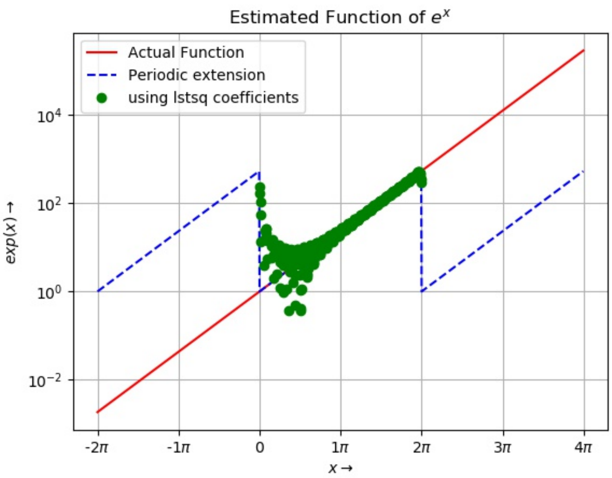

Figure 14: True function value and periodic extension and estimated function value of $exp(x)$

From the above shown plots in Figure 13 and Figure 14 we shall infer the following:

- For $cos(cos(x))$ the curves fit almost perfectly because the function itself is a periodic function and it is a continuous function throughout the entire region, so we there is a very negligible deviation from the actual function and so we are able to reconstruct the signal with just the Fourier coefficients.
- But this is not the case in $exp(x)$, since it is aperiodic we used an periodic extension of the same, and so it is discontinuous at $2\text{n}\pi$.
- Since $exp(x)$ is discontinuities at $2\text{n}\pi$, we see the **Gibbs Phenomenon** which is the oscillations that occur around the points of discontinuity and the amplitude of ripple decreases as we towards the points of discontinuity. These ripples die out as we increase the number of coefficients but with a finite limit.

Conclusion

- We observe that the Fourier series estimation of $cos(cos(x))$ exactly matches the actual function, while it doesn't match in the case of $exp(x)$.
- This is because, the function $exp(x)$ is aperiodic and using the periodic extension of it over a finite length doesn't give the entire characteristics of the function, however if we increase the sample points then the estimated function tries to approach the actual function.
- Also, because of the presence of discontinuities we see ripples at the discontinuities in the case of $exp(x)$, which is **Gibbs Phenomenon**.
- Thus we can say that the Fourier Series Approximation Method works extremely well for continuous periodic functions, but gives erroneous output for discontinuous periodic functions.