Assignment 4

G Ch V Sairam , EE19B081

10-03-2021

Introduction

The Fourier series of any function which is periodic with period 2pi is calculated as follows:

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

where a_n , b_n are called the fourier series coefficients.

- a_0= $1/2\pi \int_0^{2pi} f(x)\,dx$
- a_n= $1/2\pi \int_0^{2pi} f(x)cos(nx)\,dx$
- b_n= $1/2\pi \int_0^{2pi} f(x)sin(nx)\,dx$

Assignment problems

Question 1

The functions are defined in the following way:

```
def e(x):
        return np.exp(x)
                                                        #Defining the fun
def cos_cos(x):
        y=np.cos(x)
        return np.cos(y)
```

Here , the functions np.exp() and np.cos() are used instaed of math.exp() and math.cos() because the former can accept vector inputs and generate vector ouputs but the latter cannot.

cos(cos(x)) is periodic but e^x is not periodic. But the fourier series generates only periodic functions. So , I expect that the functons cos(cos(x)) and e^y will be generated where y is the remainder obtained when x is divided by 2π.

We can find the periodic extension of e^x as mentioned in the below code line:

```
per_ext_e=e(X%(2*math.pi))
```

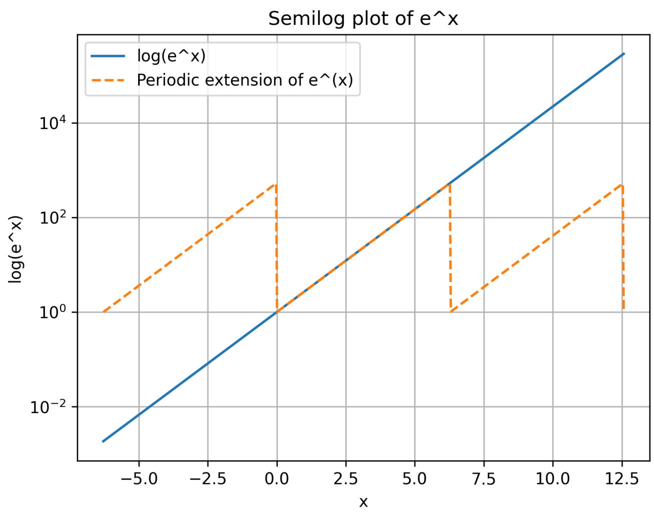

Figure 1: Figure 1

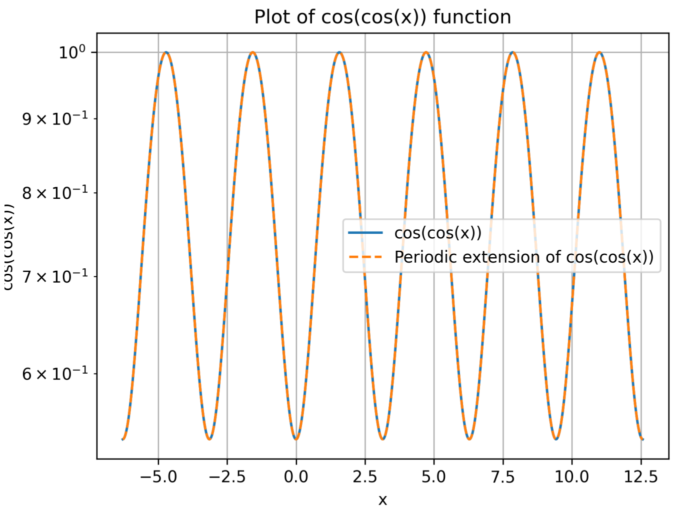

Figure 2: Figure 2

Question 2

The first 51 fourier coefficients of both the functions are calculated using quad function from the scipy.integrate module.

```
#         Finding the fourier series coefficients of the function e^x

a_e = np.zeros(51)
def fcos1(x,k):
    return e(x)*np.cos(k*x)/np.pi
def fsin1(x,k):
    return e(x)*np.sin(k*x)/np.pi
a_e[0] = intg.quad(e,0,2*np.pi)[0]/(2*np.pi)
for i in range(1,51):
    if(i%2==1):
        a_e[i] = intg.quad(fcos1,0,2*np.pi,args=(int(i/2)+1))[0]
    else:
            a_e[i] = intg.quad(fsin1,0,2*np.pi,args=(int(i/2)))[0]

#                 Finding the fourier coefficients of cos(cos(x))

a_coscos = np.zeros(51)
def fcos2(x,k):
    return cos_cos(x)*np.cos(k*x)/np.pi
def fsin2(x,k):
    return cos_cos(x)*np.sin(k*x)/np.pi
a_coscos[0] = intg.quad(cos_cos,0,2*np.pi)[0]/(2*np.pi)
for i in range(1,51):
        if(i%2==1):
                a_coscos[i] = intg.quad(fcos2,0,2*np.pi,args=(int(i/2)+1
        else:
                a_coscos[i] = intg.quad(fsin2,0,2*np.pi,args=(int(i/2)))[
```

Question 3

a)As we know , cos(cos(x)) is an even function. b_n are the coefficients of the term sin(kx), which is an odd function. cos(cos(x)) has no odd part , so for the second case all b_n are nearly zero.

b)For e^x , the higher frequencies also have significant components whereas for cos(cos(x)) the frequency is $1/\pi$ and hence it doesn't have significant components from the higher frequencies..This is the reason why the coefficients in the first case do not decay as quickly as the coefficients for the second case.

c)For exp(x) , the fourier coefficients, a_n are proportional to $1/(1+n^2)$, b_n are proportional to $n/(1+n^2)$.

For large n , a_n=$1/n^2$ and b_n=$1/n$. Therefore , $log(a_n)$ =-2log(n) and $log(b_n)$ =-log(n). This is the reason for the linearity of loglog plot in figure 4. For cos(cos(x)), the fourier coefficients vary exponentially with n and hence the semilog plot of figure 5 looks linear.

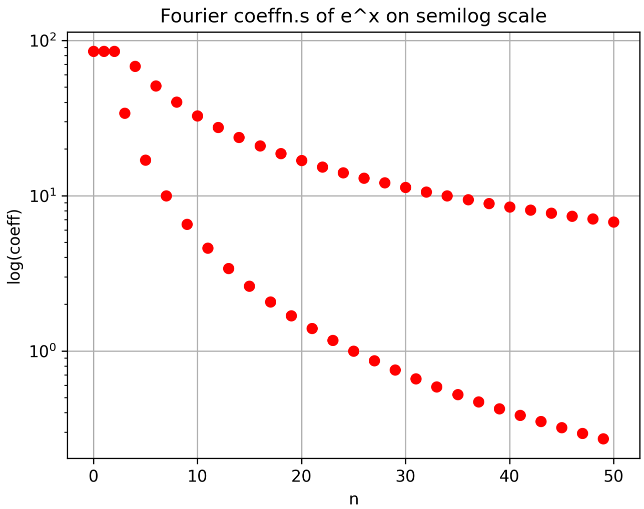

Figure 3: Figure 3

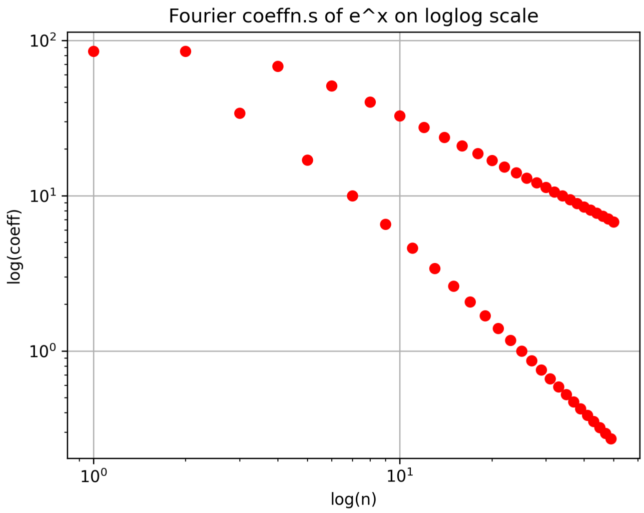

Figure 4: Figure 4

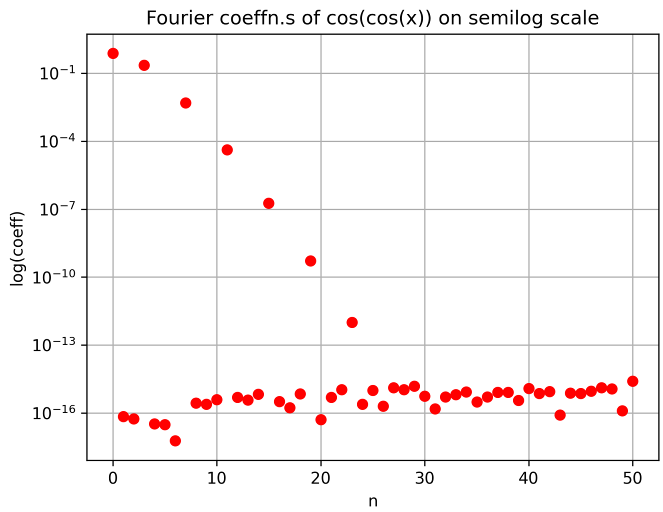

Figure 5: Figure 5

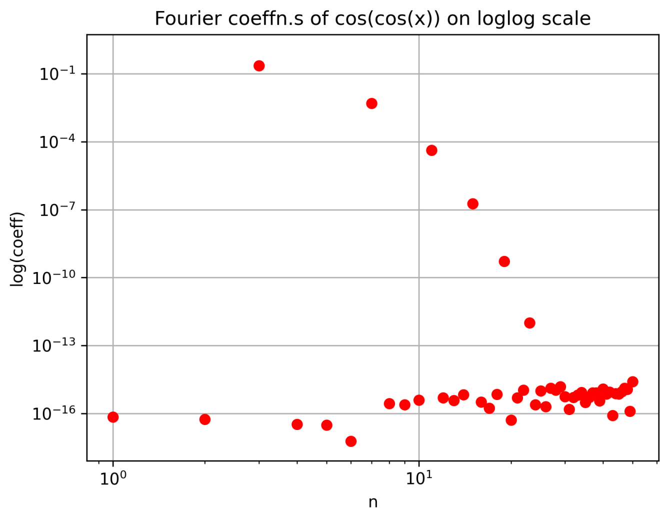

Figure 6: Figure 6

Question 4

We use the least squares method to find the approximate the values of the coefficients. However , we use a small number of points(400) so the error can be quite large. If we use a larger number of points , then the error might decrease.

Question 5

We solve the equation Ac=b using the lstsq method and plot the results.

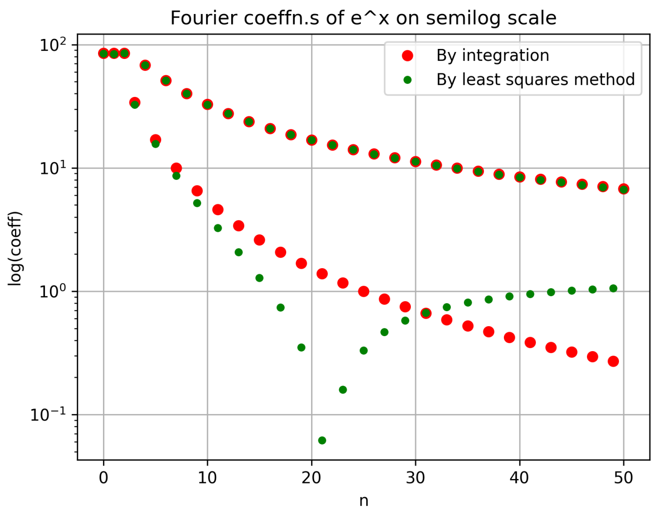

Figure 7: Figure 7

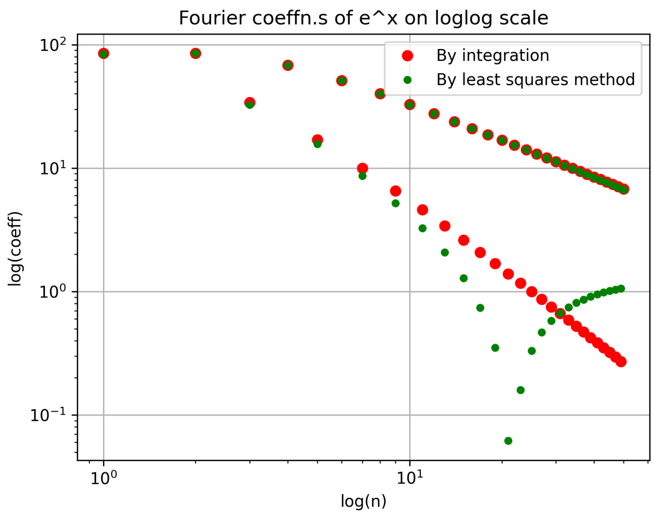

Figure 8: Figure 8

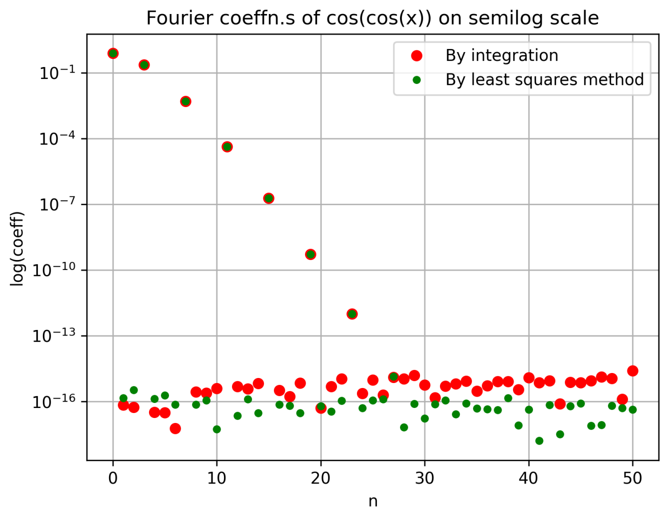

Figure 9: Figure 9

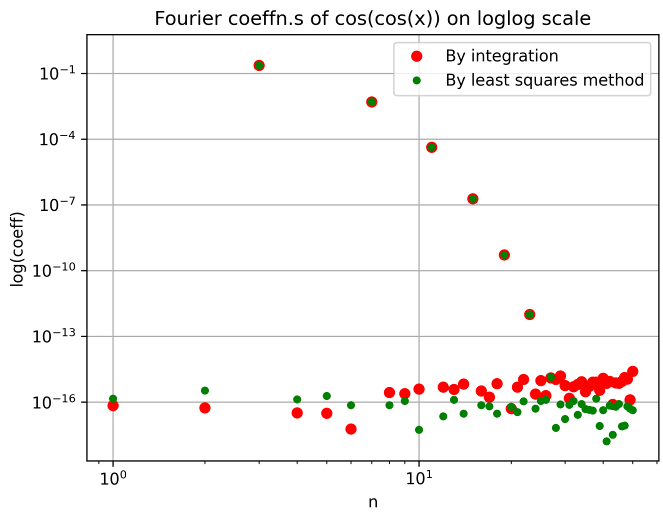

Figure 10: Figure 10

Question 6

When we compare the values of the coefficients obtained by the least squares method and the direct integration method ,they should agree with each other , but we observe that there is a significant deviation of values in the e^x function,probably because the number of sample points taken were very less , but the values for cos(cos(x)) are almost in agreement.

Deviation for e^x =1.337 Deviation for cos(cos(x))=$2.64 * 10^-15$

Question 7

We find the function values using the coefficients that we obtained through least squares method and we plot them.

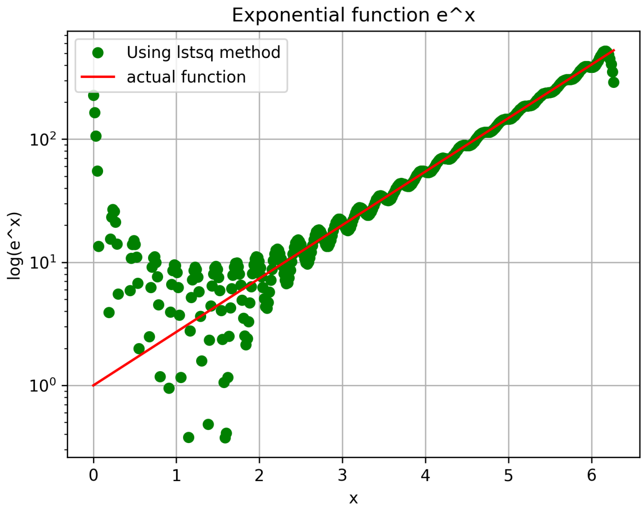

Figure 11: Figure 11

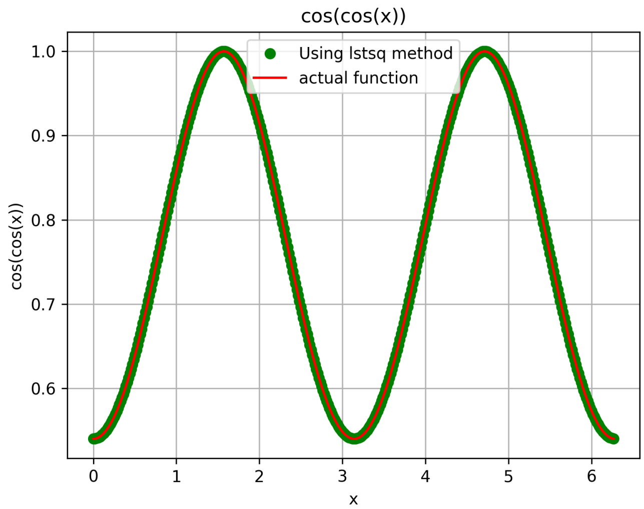

Figure 12: Figure 12

Conclusion

We have seen two ways of calculating fourier coefficients, by direct integration and by the least squares method. We have seen that the error in the coefficients when calculated by using lstsq method is not very large , especially when we use a large number of sample points. Hence , it is safe to use the least squares method for these functions and reduce the computation complexity.