Assignment 4

Kathir Pagalavan EE20B056

25^{th} February 2022

1 Aim

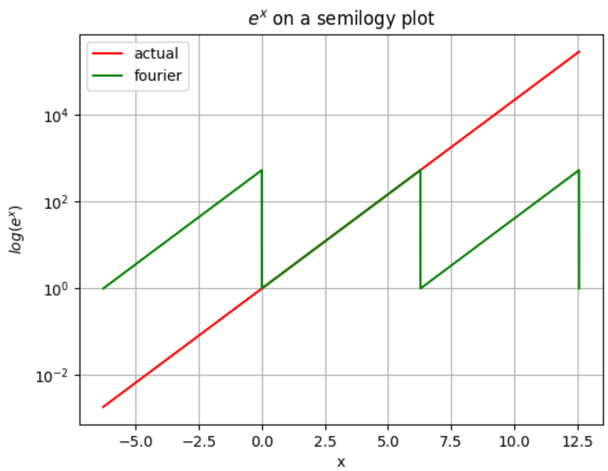
We will fit two functions, e^x and $cos(cos(x))$ over the interval $[0, 2\pi)$ using their computed Fourier series coefficients.

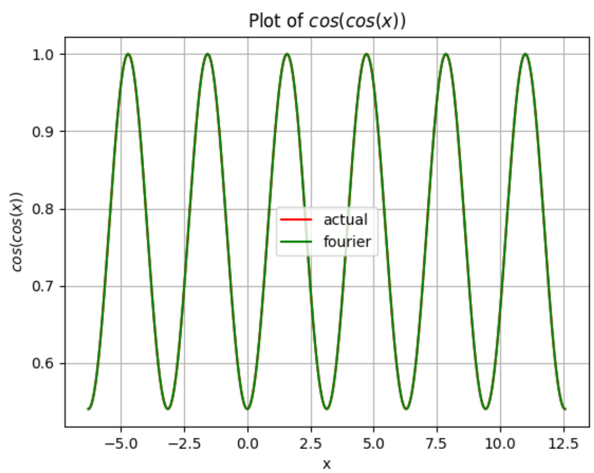
2 Assignment Questions

2.1 Creating the functions

$cos(cos(x))$ is a periodic function with period 2π whereas e^x is not. The functions that I think will be generated from the Fourier series are $cos(cos(x))$ and $e^{x\%(2\pi)}$

```
def f1(x):
    return(np.exp(np.remainder(x,2*np.pi)))

def f2(x):
    return(np.cos(np.cos(x)))
```





2.2 Generating Fourier Coefficients

The first 51 coefficients are generated using the scipy.integrate.quad to integrate the product of cosine and sine products of the given function as a_n and b_n.They are saved in the following form as required by part 3:

$$\begin{bmatrix} a_0 \\ a_1 \\ b_1 \\ \ldots \\ a_{25} \\ b_{25} \end{bmatrix}$$

```python
def FT(fn, function):
a = np.zeros(fn)
def fcos(x,k,f):
    return f(x)*np.cos(k*x)/np.pi
def fsin(x,k,f):
    return f(x)*np.sin(k*x)/np.pi

a[0] = integrate.quad(function,0,2*np.pi)[0]/(2*np.pi)
for i in range(1,fn):
    if(i%2==1):
        a[i] = integrate.quad(fcos,0,2*np.pi,args=(int(i/2)+1,function))[0]
    else:
        a[i] = integrate.quad(fsin,0,2*np.pi,args=(int(i/2),function))[0]
return a
```

2.3 Visualizing Fourier Coefficients

$\sum \|b_n\|$ is almost zero for the second function because it is an even function. The coefficients for e^x decay faster than that of the Log-Log plot for Fourier coefficients of e^x is nearly linear because :

$$\int_0^{2\pi} e^x cos(kx)dx = \frac{(e^{2\pi} - 1)}{(k^2 + 1)} \tag{1}$$

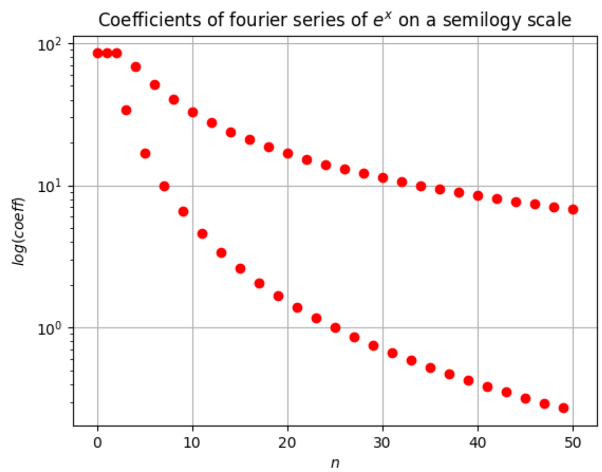
and

$$\int_0^{2\pi} e^x sin(kx)dx = \frac{(-ke^{2\pi} + k)}{(k^2 + 1)} \tag{2}$$

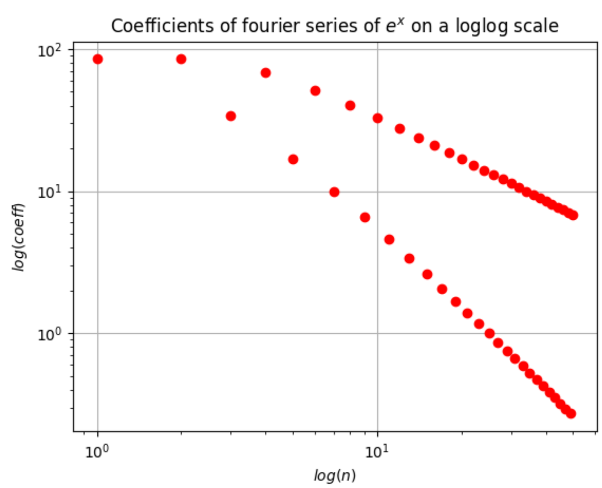
the log-log plots of these functions are linear

The semi-logy plot for Fourier Coefficients of $cos(cos(x))$ is linear as the coefficients are proportional to e^x.

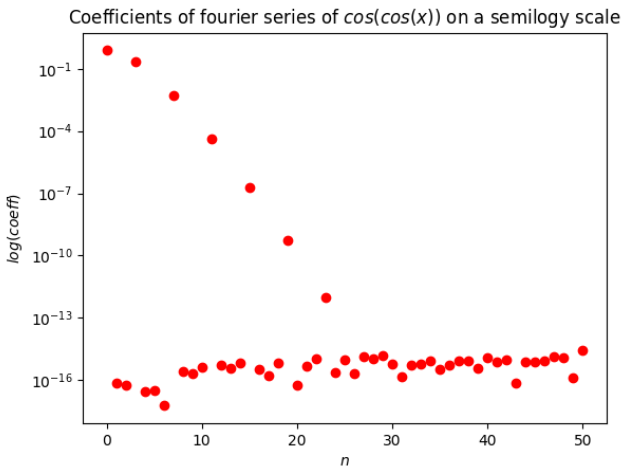
```
def fplot(f1FT,f2FT,color = 'ro'):
    f1FT = np.abs(f1FT)
    f2FT = np.abs(f2FT)

    plt.title(r"Coefficients of fourier series of $e^x$ on a semilogy scale")
    plt.xlabel(r'$n$')
    plt.ylabel(r'$log(coeff)$')
    plt.semilogy(f1FT,color)
    plt.grid(True)
    plt.show()

    plt.title(r"Coefficients of fourier series of $e^x$ on a loglog scale")
    plt.xlabel(r'$log(n)$')
    plt.ylabel(r'$log(coeff)$')
    plt.loglog(f1FT,color)
    plt.grid(True)
    plt.show()


    plt.title(r"Coefficients of fourier series of $cos(cos(x))$ on a semilogy scale")
    plt.xlabel(r'$n$')
    plt.ylabel(r'$log(coeff)$')
    plt.semilogy(f2FT,color)
    plt.show()

    plt.title(r"Coefficients of fourier series of $cos(cos(x))$ on a loglog scale")
    plt.xlabel(r'$log(n)$')
    plt.ylabel(r'$log(coeff)$')
    plt.loglog(f2FT,color)
    plt.grid(True)
    plt.show()

plotf1()#1a
plotf2()#1b

f1FT = FT(nfourier ,f1)
f2FT = FT(nfourier ,f2)
fplot(f1FT,f2FT)#3
```

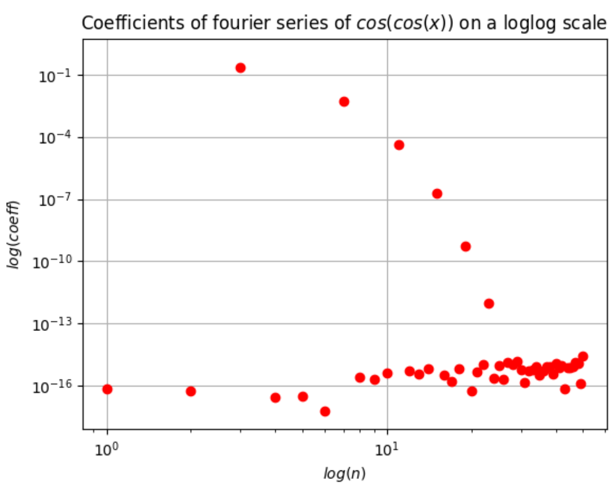


Coefficients of fourier series of e^x on a semilogy scale
10^2
10^1
10^0
log(coeff)
0
10
20
30
40
50
n



Coefficients of fourier series of e^x on a loglog scale
10^2
10^1
10^0
log(coeff)
10^0
10^1
log(n)


Coefficients of fourier series of $cos(cos(x))$ on a semilogy scale
10^{-1}
10^{-4}
10^{-7}
10^{-10}
10^{-13}
10^{-16}
$log(coeff)$
0
10
20
30
40
50
n


Coefficients of fourier series of $cos(cos(x))$ on a loglog scale
10^{-1}
10^{-4}
10^{-7}
10^{-10}
10^{-13}
10^{-16}
$log(coeff)$
10^{0}
10^{1}
$log(n)$

2.4 A Least Squares Approach

We linearly choose 400 values of x in the range [0,2π). We try to solve Equation with the fourier coefficients as variables in a linear equation of the fourier expansion By using regression on these 400 values

$$\begin{pmatrix} 1 & \cos(x_1) & \sin(x_1) & .... & \cos(25x_1) & \sin(25x_1) \\ 1 & \cos(x_2) & \sin(x_2) & .... & \cos(25x_2) & \sin(25x_2) \\ ... & ... & ... & .... & ... & ... \\ 1 & \cos(x_{400}) & \sin(x_{400}) & .... & \cos(25x_{400}) & \sin(25x_{400}) \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ b_1 \\ ... \\ a_{25} \\ b_{25} \end{pmatrix} = \begin{pmatrix} f(x_1) \\ f(x_2) \\ ... \\ f(x_{400}) \end{pmatrix}$$

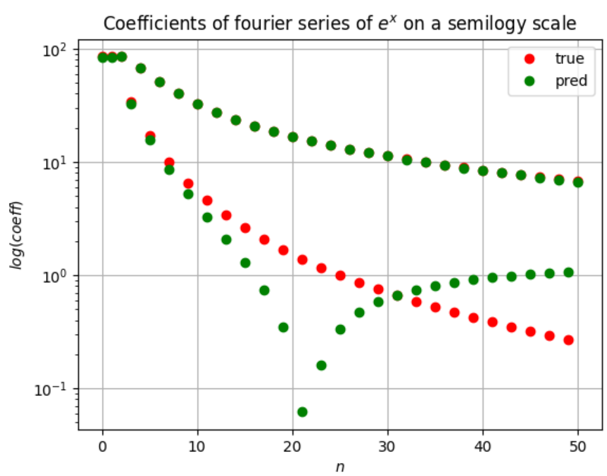
We create the matrix on the left side and call it A . We want to solve $Ac = b$ where c are the fourier coefficients.

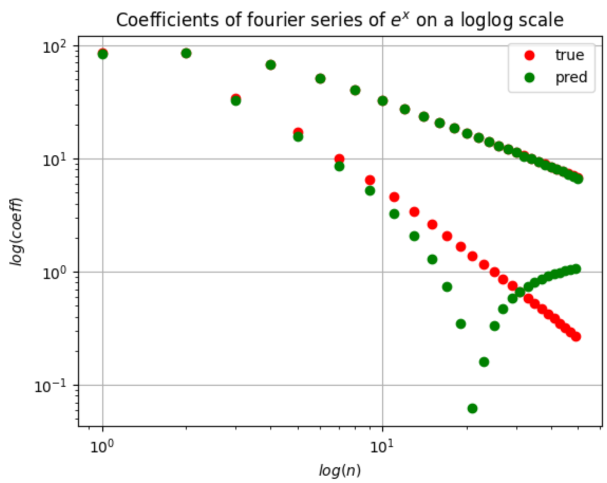
```
def generateAb(x,f):
    A = np.zeros((x.shape[0],nfourier))
    A[:,0] = 1
    for i in range(1,int((nfourier+1)/2)):
        A[:,2*i-1]=np.cos(i*x)
        A[:,2*i]=np.sin(i*x)
    return A,f(x)

x=np.linspace(0,2*np.pi,401)
x=x[:-1]

Af1,bf1=generateAb(x,f1);cf1=scipy.linalg.lstsq(Af1,bf1)[0]
Af2,bf2=generateAb(x,f2);cf2=scipy.linalg.lstsq(Af2,bf2)[0]
```

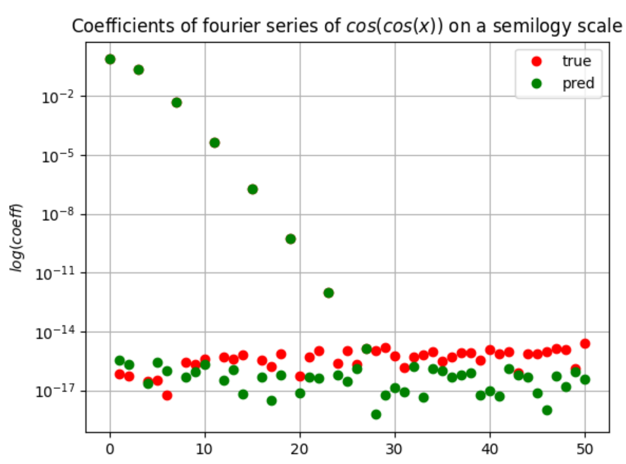
2.5 Visualizing output of the Least Squares Approach

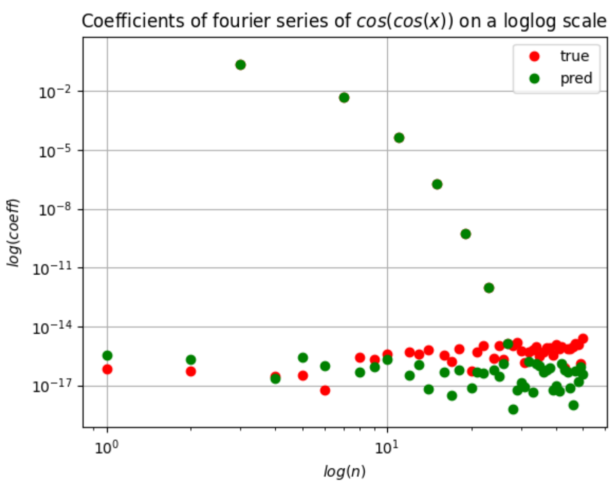
```
def plotlstsq(f1FT,f2FT,cf1,cf2,color = 'go'):
    f1FT = np.abs(f1FT)
    f2FT = np.abs(f2FT)
    cf1 = np.abs(cf1)
    cf2 = np.abs(cf2)
    plt.title(r"Coefficients of fourier series of $e^x$ on a semilogy scale")
    plt.xlabel(r'$n$')
    plt.ylabel(r'$log(coeff)$')
    plt.semilogy(f1FT,'ro')
    plt.semilogy(cf1,color)
    plt.legend(["true","pred"])
    plt.grid(True)
    plt.show()
    plt.title(r"Coefficients of fourier series of $e^x$ on a loglog scale")
    plt.xlabel(r'$log(n)$')
    plt.ylabel(r'$log(coeff)$')
    plt.loglog(f1FT,'ro')
    plt.semilogy(cf1,color)
    plt.legend(["true","pred"])
    plt.grid(True)
```

```
    plt.show()

    plt.title(r"Coefficients of fourier series of $cos(cos(x))$ on a semilogy scale")
    plt.xlabel(r'$n$')
    plt.ylabel(r'$log(coeff)$')
    plt.semilogy(f2FT,'ro')
    plt.semilogy(cf2,color)
    plt.legend(["true","pred"])
    plt.grid(True)
    plt.show()
    plt.title(r"Coefficients of fourier series of $cos(cos(x))$ on a loglog scale")
    plt.xlabel(r'$log(n)$')
    plt.ylabel(r'$log(coeff)$')
    plt.loglog(f2FT,'ro')
    plt.semilogy(cf2,color)
    plt.legend(["true","pred"])
    plt.grid(True)
    plt.show()

plotlstsq(f1FT,f2FT,cf1,cf2,'go')
```




Coefficients of fourier series of e^x on a loglog scale
true
pred
10^2
10^1
10^0
10^{-1}
log(coeff)
10^0
10^1
log(n)


Coefficients of fourier series of $cos(cos(x))$ on a semilogy scale
true
pred
10^{-2}
10^{-5}
10^{-8}
10^{-11}
10^{-14}
10^{-17}
log(coeff)
0
10
20
30
40
50
n


Coefficients of fourier series of $cos(cos(x))$ on a loglog scale
true
pred
10^{-2}
10^{-5}
10^{-8}
10^{-11}
10^{-14}
10^{-17}
$log(coeff)$
10^0
10^1
$log(n)$

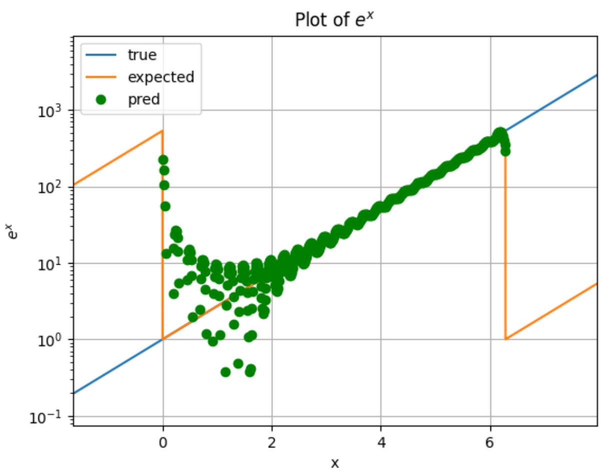
2.6 Comparing Predictions

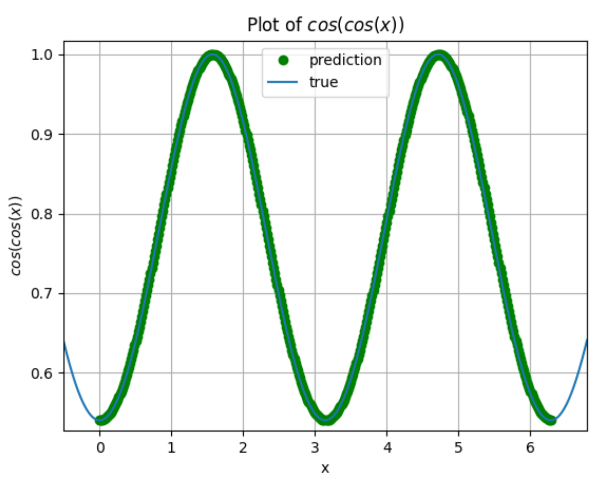
The error in Coefficients of e^x = 1.3327308703354106
The error in Coefficients of $cos(cos(x))$ = 2.6473985022525665e-15
Our Predictions for e^x are very poor compared to that of cos(cos(x)). This can be fixed by sampling at a larger number of points.

2.7 Plotting Results

```
cf1 = np.reshape(cf1,(nfourier,1))
#Finding values of the function from the Coefficients obtained using lstsq
TTT = np.matmul(Af1,cf1)
#plotting results
x = np.linspace(0,2*np.pi,400,endpoint=True)
plt.title(r"Plot of $e^x$")
t = np.linspace(-2*np.pi,4*np.pi,n,endpoint=True)
plt.semilogy(t,f1(t))
plt.semilogy(t,f1(np.remainder(t,2*np.pi)))
plt.semilogy(x,TTT,'go')
plt.xlabel('x')
plt.ylabel(r'$e^x$')
plt.legend(["true","expected","pred"])
plt.grid(True)
plt.show()

cf2 = np.reshape(cf2,(nfourier,1))
#Finding values of the function from the Coefficients obtained using lstsq
TTT = np.matmul(Af2,cf2)
#plotting results
x = np.linspace(0,2*np.pi,400,endpoint=True)
plt.title(r"Plot of $cos(cos(x))$")
t = np.linspace(-2*np.pi,4*np.pi,n,endpoint=True)
plt.plot(x,TTT,'go')
plt.plot(t,f2(t))
plt.xlabel('x')
plt.ylabel(r'$cos(cos(x))$')
plt.legend(["prediction","true"])
plt.grid(True)
plt.show()
```

It should be noted that e^x is a non periodic function and Fourier series' exists only for periodic functions. Hence we have considered a variation of e^x with period 2pi that has the actual value of e^x only in the range [0,2π). Hence it is acceptable that there is a large discrepancy in the predicted value of e^x at these boundaries


Plot of e^x
true
expected
pred
10^3
10^2
10^1
10^0
10^{-1}
e^x
0
2
4
6
x


Plot of $cos(cos(x))$
prediction
true
1.0
0.9
0.8
0.7
0.6
$cos(cos(x))$
0
1
2
3
4
5
6
x

3 Conclusion

We have performed an actual fourier series expansion and least squares method to arrive at the fourier coefficients of a finitely discontinous function(non sinusoidal and non periodic parent function) and a periodic function.

We notice close matching of the two methods in case of $\cos(\cos(x))$ while, there is a larger discrepancy in $\exp(x)$.