EE2703: Assignment 4

Sagar (ee20b115)

February 26, 2022

1 Aim

In this assignment we aim to:

1. Fit the function e^x using Fourier series

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

over (0,2π):

   (a) using coefficients from Fourier formula.

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

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

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

   (b) using coefficients from least square method.

2. Fit the function $cos(cos(x))$ using Fourier series over (0,2π):

   (a) using coefficients from Fourier formula.

   (b) using coefficients from least square formula.

3. Then compare the relative accuracy of the two methods

2 Procedure and Observations

2.1 True functions and Expected functions

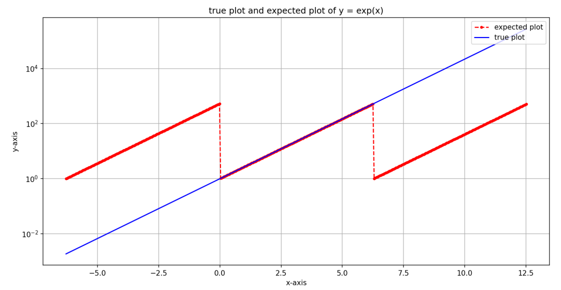
2.1.1 CODE:

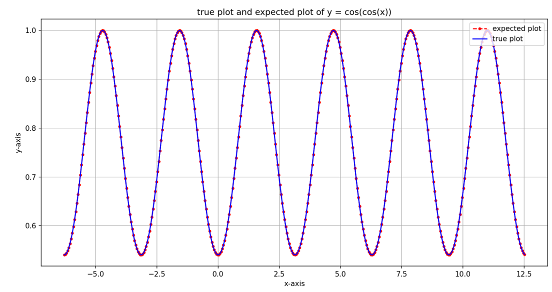
```
X = linspace(-2*pi,4*pi,401)
X = X[:-1]

def fexp(t):
return exp(t)

def fcoscos(t)
return cos(cos(t))

Y = fexp(X%(2*pi))
```

```
Z = fcoscos(X%(2*pi))
# true values of functions
Y2 = exp(X)
Z2 = cos(cos(X))

figure(1)
title("true plot and expected plot of y = exp(x)")
semilogy(X,Y, 'r.--')
semilogy(X,Y2, 'b')
xlabel("x-axis")
ylabel("y-axis")
legend(["expected plot","true plot"],loc='upper right')
grid(True)
show()

figure(2)
title("true plot and expected plot of y = cos(cos(x))")
plot(X,Z, 'r.--')
plot(X,Z2, 'b')
xlabel("x-axis")
ylabel("y-axis")
legend(["expected plot","true plot"],loc='upper right')
grid(True)
show()
```

2.1.2 PLOTS:

First we plot the true values of these function and then plot what the Expected plot using Fourier series should look like over the interval (-2π,4π).





2.1.3 OBSERVATIONS:

1. It can be observed that e^x is not periodic while $cos(cos(x))$ is periodic.
2. It is expected that only $cos(cos(x))$ will be generated accurately as it is periodic and completely defined by values over $(0,2\pi)$ which was used to find coefficients.

Note: Since e^x grows rapidly, we have used *semilogy* for that plot.

2.2 Fourier coefficients using Fourier formula (*integration*)

2.2.1 CODE:

```
a01 = inte.quad(lambda x: exp(x), 0, 2*pi)
a01 = a01[0]/(2*pi)
```

```
A1 = []
B1 = []

for n in range(1,26):
func = lambda x: exp(x)*cos(n*x)
intg = inte.quad(func, 0, 2*pi)
A1.append(intg[0]/pi)

for n in range(1,26):
func = lambda x: exp(x)*sin(n*x)
intg = inte.quad(func, 0, 2*pi)
B1.append(intg[0]/pi)


F1 = []
F1.append(a01)
m=0
n=0
for i in range (1,51):
if(i%2!=0):
        F1.append(A1[m])
        m = m+1
elif(i%2==0):
        F1.append(B1[n])
        n = n+1

k = 1
l = 0
L = []
L.append(La[0])
for i in range (1,51):
if (i%2!=0):
        L.append(La[k])
        k = k+1
else:
        L.append(Lb[l])
        l = l+1

figure(3)
semilogy(L,absolute(F1), 'r.')
title("magnitude of fourier coefficients of y = exp(x)")
ylabel("logarithmic y-axis")
xlabel("linear x-axis")
legend(["fourier coefficients"],loc='upper right')
grid(True)
show()

figure(4)
loglog(L,absolute(F1), 'r.')
title("magnitude of fourier coefficients of y = exp(x)")
ylabel("logarithmic y-axis")
xlabel("logarithmic x-axis")
legend(["fourier coefficients"],loc='upper right')
grid(True)
show()
```

2.2.2 PLOTS:

First we have to calculate the Fourier coefficients of e^x using these formulas.

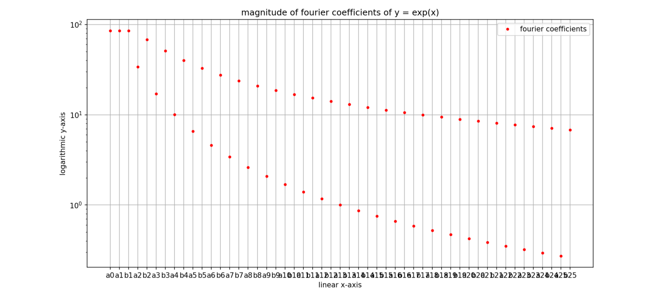
$$a_0 = \frac{1}{2\pi}\int_0^{2\pi} e^x \, dx$$

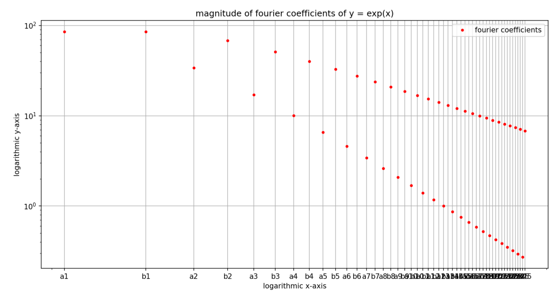
$$a_n = \frac{1}{\pi}\int_0^{2\pi} e^x cos(nx) \, dx$$

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

Now for e^x, we make two different plots using *semilogy* (*left*) and *loglog* (*right*) and plot the magnitude of coefficients as vector given below.

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





2.2.3 CODE:

```
a02 = inte.quad(lambda x: cos(cos(x)), 0, 2*pi)
a02 = a02[0]/(2*pi)

A2 = []
B2 = []

for n in range(1,26):
func = lambda x: cos(cos(x))*cos(n*x)
intg = inte.quad(func, 0, 2*pi)
A2.append(intg[0]/pi)

for n in range(1,26):
func = lambda x: cos(cos(x))*sin(n*x)
intg = inte.quad(func, 0, 2*pi)
B2.append(intg[0]/pi)

F2 = []
F2.append(a02)
m=0
n=0
for i in range (1,51):
if(i%2!=0):
```

```
            F2.append(A2[m])
            m = m+1
    elif(i%2==0):
            F2.append(B2[n])
            n = n+1

figure(5)
semilogy(L,absolute(F2), 'r.')
title("magnitude of fourier coefficients of y = cos(cos(x))")
ylabel("logarithmic y-axis")
xlabel("linear x-axis")
legend(["fourier coefficients"],loc='upper right')
grid(True)
show()

figure(6)
loglog(L,absolute(F2), 'r.')
title("magnitude of fourier coefficients of y = cos(cos(x))")
ylabel("logarithmic y-axis")
xlabel("logarithmic x-axis")
legend(["fourier coefficients"],loc='upper right')
grid(True)
show()
```

2.2.4 PLOTS:

Now we calculate the Fourier coefficients of $cos(cos(x)$ using these formulas.

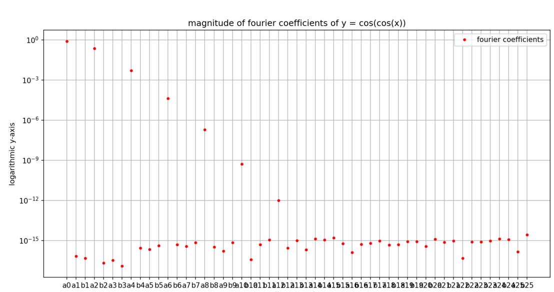
$$a_0 = \frac{1}{2\pi}\int_0^{2\pi} cos(cos(x))\,dx$$

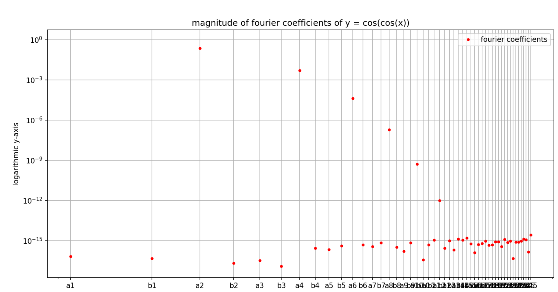
$$a_n = \frac{1}{\pi}\int_0^{2\pi} cos(cos(x))cos(nx)\,dx$$

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

Now for $cos(cos(x))$ too, we again make two different plots using *semilogy* (*left*) and *loglog* (*right*) and plot the magnitude of the coefficients as below vector.

$$\begin{pmatrix} a_0 \\ a_1 \\ b_1 \\ ... \\ a_{25} \\ b_{25} \end{pmatrix}$$





2.2.5 OBSERVATIONS:

1. As we can see from (Figures 5,6) all b_n coefficients are very close to 0 for cos(cos(x)). This behaviour happens due to the odd nature of the integrand between $[-\pi;\pi)$.
2. Because $cos(cos(x))$ is an infinitely differentiable function, it's Fourier coefficients decay very fast (Figure 5,6), while that of e^x decay very slowly (Figures 3,4) due to the discontinuity in the Fourier approximation of the function at $2n\pi$.
3. (a) for 1st function, we have

$$\int_0^{2\pi} e^x cos(nx)\,dx = e^{2\pi}\frac{(cos(2\pi n) + nsin(2\pi n)}{n^2+1} \simeq \frac{e^{2\pi}}{n^2+1} \simeq \frac{k}{n^2}$$

   for integers values of n.
   The function

$$a_n = \frac{k}{n^2}$$

   appears as a straight line on a *loglog* plot as $y = bx^m$ is a straight line on *loglog*.

   (b) For 2nd function, we observe that the integral

$$\int cos(cos(x))cos(nx)\,dx$$

   does not have a closed form solution but as cos(cos(x)) stays between (0.54,1), we can approximate it as a constant. So the whole function can be approximated as

$$\int_0^{2\pi} cos(cos(x))cos(nx)\,dx \simeq \int_0^{2\pi} kcos(nx)\,dx = k\frac{sin(2\pi n)}{n}$$

   Here after plotting the function

$$y = \frac{ksin(2\pi x)}{x}$$

   on *semilogy* we get approximately linear behaviour for first few values on x.

2.3 Fourier coefficients using Fourier formula (*integration*) vs using Least Square method (*lstsq* function)

We now do a *Least Squares* approach. Define a vector x going from 0 to 2π in 400 steps using *linspace*. Evaluate the function $f(x)$ at those x values and call it b. Now this function is to be approximated by $Eq.(1)$. So for each x_i we want

$$a_0 + \sum_{n=1}^{25} a_n cos(nx_i) + \sum_{n=1}^{25} b_n sin(nx_i) \approx f(x_i) \tag{2}$$

Turning this into matrix form:

$$\begin{pmatrix} 1 & cosx_1 & sinx_1 & ... & cos25x_1 & sin25x_1 \\ 1 & cosx_2 & sinx_2 & ... & cos25x_2 & sin25x_2 \\ ... & ... & ... & ... & ... & ... \\ 1 & cosx_{400} & sinx_{400} & ... & cos25x_{400} & sin25x_{400} \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ b_1 \\ ... \\ a_{25} \\ b_{25} \end{pmatrix} = \begin{pmatrix} f(x_1) \\ f(x_2) \\ ... \\ f(x_{400}) \end{pmatrix}$$

Naming the left matrix as A. We get the following equation

$$Ac = b$$

We solve for c using the function *lstsq*

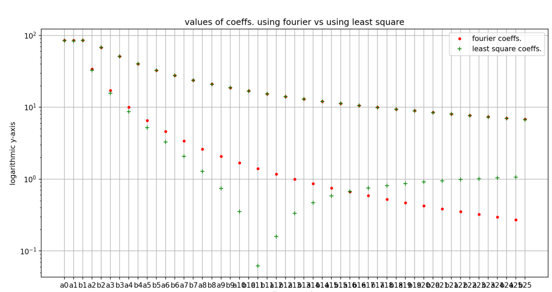
$$c = lstsq(A, b)[0]$$

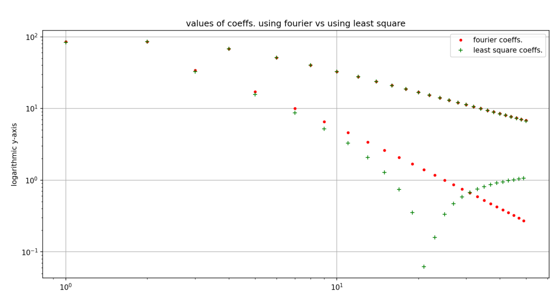
2.3.1 CODE:

```
matA = zeros((400,51))
matA[:,0]=1
for k in range(1,26):
matA[:,2*k-1]=cos(k*L1)
matA[:,2*k]=sin(k*L1)

b1 = exp(L1)
b2 = cos(cos(L1))

C1 = lstsq(matA,b1, rcond=None)[0]
C2 = lstsq(matA,b2, rcond=None)[0]

figure(7)
semilogy(L,absolute(F1), 'r.')
semilogy(L,absolute(C1), 'g+')
title("values of coeffs. using fourier vs using least square")
legend(["fourier coeffs.","least square coeffs."],loc='upper right')
ylabel("logarithmic y-axis")
xlabel("linear x-axis")
grid(True)
show()

figure(8)
semilogy(L,absolute(F2), 'r.')
semilogy(L,absolute(C2), 'g+')
title("values of coeffs. using fourier vs using least square")
legend(["fourier coeffs.","least square coeffs."],loc='upper right')
ylabel("logarithmic y-axis")
xlabel("linear x-axis")
grid(True)
show()
```

2.3.2 PLOTS:

Now we make two different plots of magnitude of coefficients of e^x obtained from Fourier formula(*red*) and *lstsq* function(*green*) by using **semilogy** (*left*) and **loglog** (*right*).





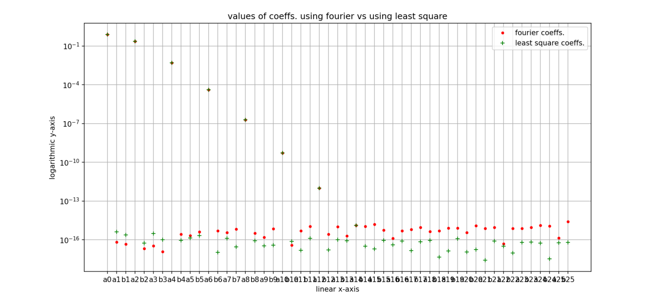
2.3.3 CODE:

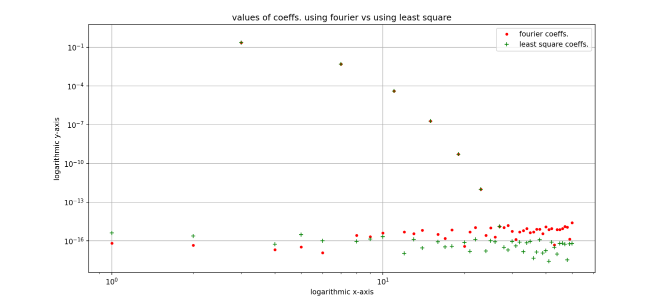
```
figure(9)
loglog(L,absolute(F1), 'r.')
loglog(L,absolute(C1), 'g+')
title("values of coeffs. using fourier vs using least square")
```

```
legend(["fourier coeffs.","least square coeffs."],loc='upper right')
ylabel("logarithmic y-axis")
xlabel("logarithmic x-axis")
grid(True)
show()

figure(10)
loglog(L,absolute(F2), 'r.')
loglog(L,absolute(C2), 'g+')
title("values of coeffs. using fourier vs using least square")
legend(["fourier coeffs.","least square coeffs."],loc='upper right')
ylabel("logarithmic y-axis")
xlabel("logarithmic x-axis")
grid(True)
show()
```

2.3.4 PLOTS:

We again make two different plots of magnitude of coefficients of $cos(cos(x))$ obtained from Fourier formula(*red*) and *lstsq* function(*green*) by using **semilogy** (*left*) and **loglog** (*right*).





2.3.5 OBSERVATIONS:

1. For e^n, a_n coefficients agree with each other but there is significant deviation in case of b_n coefficients.
2. For $cos(cos(x))$, both a_n and b_n coefficients agree with each other with no significant deviation in any case.
3. Yes, they should agree.
4. **CODE:**

```
C1F1 =[]
for i in range(0,51):
C1F1.append(C1[i]-F1[i])

maxdev1 = max(absolute(C1F1))
print("\nMaximun deviation is: "+str(maxdev1))

C2F2 =[]
for i in range(0,51):
C2F2.append(C2[i]-F2[i])

maxdev2 = max(absolute(C2F2))
print("\nMaximun deviation is: "+str(maxdev2))
```

5. (a) The maximum deviation in case of e^x is 1.33
   (b) The maximum deviation in case of $cos(cos(x))$ is $2.67e^-15$

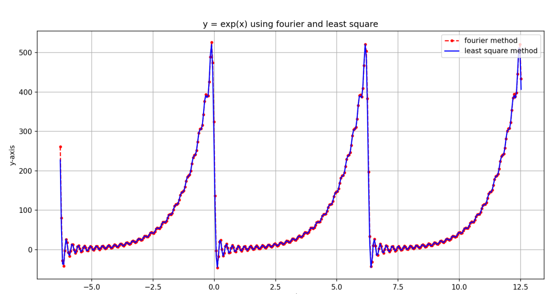
2.4 Plots of e^x and $cos(cos(x))$ using Fourier coefficients and Least Square coefficients

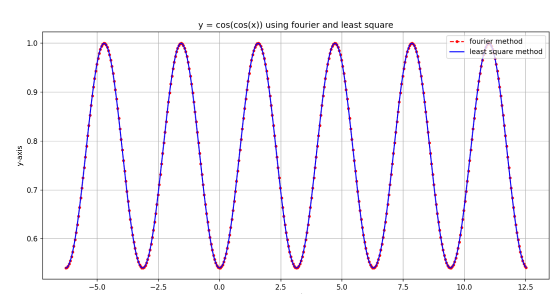
2.4.1 CODE:

```
def func(t):
matA = zeros((400,51))
matA[:,0]=1
for k in range(1,26):
        matA[:,2*k-1]=cos(k*t)
        matA[:,2*k]=sin(k*t)
return matA

A11 = func(X)

prod1 = dot(A11,F1)
prod2 = dot(A11,C1)
prod3 = dot(A11,F2)
prod4 = dot(A11,C2)

figure(11)
title("exp(x) using fourier and least square")
plot(X,prod1, 'r.--')
plot(X,prod2, 'b')
xlabel("x-axis")
ylabel("y-axis")
legend(["fourier method","least square method"],loc='upper right')
grid(True)
show()

figure(12)
title("cos(cos(x)) using fourier and least  square")
plot(X,prod3, 'r.--')
plot(X,prod4, 'b')
xlabel("x-axis")
ylabel("y-axis")
legend(["fourier method","least square method"],loc='upper right')
grid(True)
show()
```

2.4.2 PLOTS:

Now that by using the coefficients we obtain the value of the functions e^x (*left*) and $cos(cos(x))$ (*right*) over the interval $(-2\pi, 4\pi)$.





2.4.3 OBSERVATIONS:

1. As we observe that there is a significant deviation for e^x as it has discontinuities at $2n\pi$, so there will be **Gibbs** phenomenon near those points. Since we only integrated over $(0, 2\pi)$ to get the coefficients but e^x is not periodic so we lost information, which means we cannot recreate the function accurately.

2. On the other hand, we cannot observe any deviation for $cos(cos(x))$ as it is continuous over its domain. So the function remains smooth throughout and the estimated function fit perfectly.

3 Conclusion

We see that the Fourier estimation of e^x does not match significantly with the function close to 0, but matches perfectly in the case of $cos(cos(x))$. This is due to the presence of a discontinuity at $x = 0$ for the periodic extension of e^x. This discontinuity leads to non-uniform convergence of the Fourier series.
The difference in the rates of convergence leads to the **Gibbs** phenomenon, which is observed at discontinuities in the Fourier estimation of a discontinuous function.
Thus we can conclude that the Fourier Series Approximation Method works extremely well for periodic functions $cos(cos(x))$, but gives inaccurate estimates for discontinuous periodic functions(e^x).