Mathematical Physics III

Lab Report for Assignment #2B

College Roll no.	**:**	2020PHY1102
University Roll no.	**:**	20068567035
Name	**:**	Monu Chaurasiya
Unique Paper Code	**:**	32221401
Paper Title	**:**	Mathematical Physics III Lab
Course and Semester	**:**	B.Sc.(Hons.) Physics, Sem IV
Due Date	**:**	22 feb 2022
Date of Submission	**:**	22 feb 2022

Shri Guru Tegh Bahadur Khalsa College, University of Delhi
New Delhi-110007, India.

Contents

1 Theory

(a) What do you mean by orthogonal polynomials in an interval?

- A set of orthogonal polynomials is an infinite sequence of polynomials, $p_0(x), p_1(x), p_2(x), ...$ where $p_n(x)$ has degree n and any two polynomials in the set are orthogonal to each other, that is,

$$\int_a^b f_i(x) f_j(x) dx = 0 \quad i \neq j$$

The set of polynomials is orthonormal if

$$\int_a^b f_i(x) f_j(x)\, dx = \delta_{ij}$$

The interval $[a, b]$ is called the interval of orthogonality and may be infinite at one or both ends. Any infinite sequence of polynomials $\{p_n\}$ with p_n having degree n forms a basis for the infinite-dimensional vector space of all polynomials. Such a sequence can be turned into an orthogonal basis using the Gram-Schmidt orthogonalisation process by projecting out the components of each polynomial that are orthogonal to the polynomials already chosen.

(b) Write down the orthogonality condition for Legendre Polynomials. Starting from the basis $1, x, x^2, ...$ use Gram-Schmidt procedure to obtain the first three polynomials in the basis orthogonal in the interval $[-1 : 1]$ with weight function $W(x) = 1$ and show that you get the Legendre Polynomials.

- The orthogonal condition for legendre polynomials is given by:

(i) $\int_{-1}^{1} P_m(x) P_n(x)\, dx = 0$ for $m \neq n$

(ii) $\int_{-1}^{1} [P_n(x)]^2\, dx = \frac{2}{2n+1}$ for $m = n$

- Using Gram-Schmidt Procedure:

  Taking $W(x) = 1$ and $(a, b) = (-1, 1)$,using Gram-Schimdt process the orthagonal polynomials can be constructed as follows,

  Given the basis,

$$P = \{1, x, x^2, ...\}$$

  Then,

$$p_0(x) = 1$$

$$\begin{aligned} p_1(x) &= x - \frac{\langle x, p_o(x)\rangle}{\langle p_o(x), p_o(x)\rangle} p_o(x) \\ &= x - \frac{\langle x, 1\rangle}{\langle 1, 1\rangle} \\ &= x \end{aligned}$$

  Since,

  $\langle 1, 1\rangle = \int_{-1}^{1} dx = 2$ and $\langle x, 1\rangle = \int_{-1}^{1} x\,dx = 0$

$$\begin{aligned} p_2(x) &= x^2 - \frac{\langle x^2, p_0(x)\rangle}{p_o(x), p_0(x)} p_o(x) - \frac{\langle x^2, p_1(x)\rangle}{p_1(x), p_1(x)} p_1(x) \\ &= x^2 - \frac{\langle x^2, 1\rangle}{\langle 1, 1\rangle} - \frac{\langle x^2, x\rangle}{\langle x, x\rangle} x \end{aligned}$$

  Since,

  $\langle x^2, 1\rangle = \int_{-1}^{1} x^2\,dx = \frac{1}{3}$ $\quad\langle x^2, x\rangle = \int_{-1}^{1} x^2(x)\,dx = 0$

  So we get $p_2(x) = x^2 - \frac{1}{3}$

  The Polynomials $p_o(x), p_1(x), p_2(x)$ are the first three legendre polynomials similarly we can also find other.

(c)) Represesntation of a function as a Linear Combination of Legendre Polynomials:

$$f(x) = \sum_{n=0}^{\infty} C_n P_n(x) \tag{1}$$

- (i) Determine the coefficients C_n using the orthogonality relation of Legendre Polynomials.

  Any general coefficient C_n can be determined by multiplying both sides of equation 1 by $P_m(x)$ and intergrate over the interval (-1,1)

  $$\int_{-1}^{1} F(x)P_m(x)\,dx = \int_{-1}^{1} C_n P_n(x)P_m(x)$$

  We know by the orthogonality property of the Legendre polynomials that

  $$\int_{-1}^{1} [P_n(x)]^2\,dx = \frac{2}{2n+1}\delta_{mn}$$

  So we get

  $$\int_{-1}^{1} F(x)P_m(x)\,dx = C_n \int_{-1}^{1} [P_n(x)]^2\,dx = \frac{2}{2n+1}C_n$$

  Thus the general coefficient of the expansion is given by

  $$C_n = \frac{2n+1}{2}\int_{-1}^{1} F(x)P_m(x)dx$$

- How many terms will a polynomial of order n have in series of Legendre Polynomials?

  Legendre's equation is given by

  $$(1-x^2)\frac{d^2y}{dx^2} - 2x\frac{dy}{dx} + n(n+1)y =$$

  here $n < 0$ and $|x| < 1$.

  The general solution of the above equation can be represented by

  $$y = AP_n(x) + BQ_n(x)$$

  here $P_n(x)$ and $Q_n(x)$ are legendre polynomials of first and second kind respectively, A and B are arbitary constant.

  And also Legendre polynomial of order n is given by Rodrigue's formula.

  $$P_n(x) = \frac{1}{2^n n!}\frac{d^n(x^2-1)}{dx^n}$$

The solution of the above equation is given by

$$y = C_o\left[x^n - \frac{n(n-1)x^{n-2}}{(2n-1)^2} + \frac{n(n-1)(n-2)(n-3)x^{n-4}}{(2n-1)(2n-3)(2)(4)} + ...\right]$$

Here C_0 is arbitrary constant

If n is an integer and $C_o = \frac{1.3.5....(2n-1)}{n!}$

$$P_n(x) = \frac{1.3.5....(2n-1)}{n!}\left[x^n - \frac{n(n-1)x^{n-2}}{(2n-1)^2} + \frac{n(n-1)(n-2)(n-3)x^{n-4}}{(2n-1)(2n-3)(2)(4)} + \ldots\ldots\right]$$

It is a trigonometric series

When n is even, it contains $\frac{n}{2} + 1$ terms, the last being

$$\frac{(-1)^{\frac{1}{2}}n(n-1)(n-2)....1}{(2n-1)(2n-3)....(n+1)2.4.6.....(n-1)}$$

When n is odd, it contains $\frac{n+1}{2}$ terms, the last being

$$\frac{(-1)^{\frac{n-1}{2}}n(n-1)(n-2)....3.2}{(2n-1)(2n-3)......(n+1)2.4.6...(n-1)}x$$

2 Program

```
from MyIntegration import MySimp
from MyIntegration import MyTrap
from MyIntegration import MyLegQuadrature
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import math
from scipy import integrate
from legendre import Lege
from scipy.special import legendre
from sympy import *
from sympy import simplify

#(a)
#coefficient calculation
def coeff(f,n,n0=4,m=1):
  x=symbols('x')
  f2=Lege(n)
  def f1(*args):
    return f(*args) * f2(*args)
  inte=((2*n+1)/2)*MyLegQuadrature(f1,-1,1,n0,m,key=False,tol=None,m_max=None
  return inte

#Expansion
def expansion(f,n):
    lis=[]
    x=symbols('x')
    for i in range(0,n):
        f2=Lege(i)
        def s1(*args):
            return coeff(f,i,n0=10,m=100) * f2(*args)
        lis.append(s1(x))
```

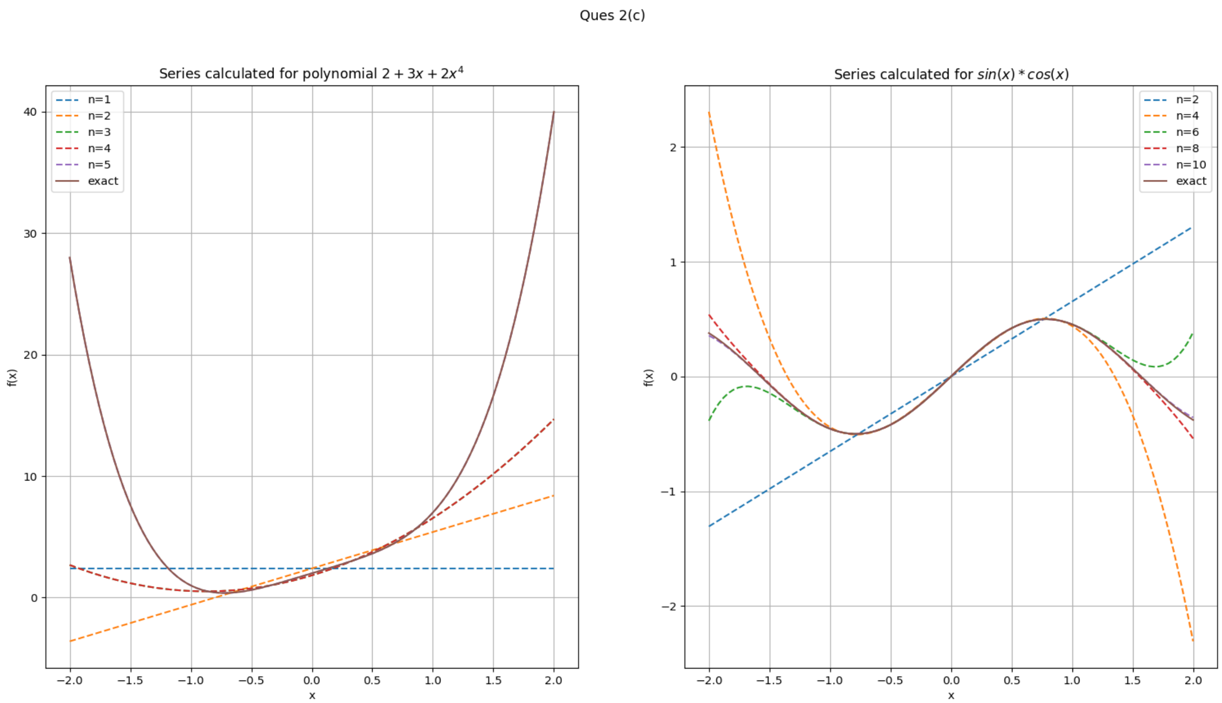
```
    k=sum(lis)
    p_x=simplify(k)
    fx=lambdify(x,p_x,"math")
    return fx

#(b)
f1=lambda x : 2 + 3*x + 2*x**4
f2=lambda x : np.sin(x)*np.cos(x)
t1=[];t2=[];t3=[]
for i in range(0,5):

    g=coeff(f1,i,n0=100,m=10000)
    t3.append(g)
    if g<5.97001796356409e-15:
        pass
    else:
        t1.append(g)

for i in range(0,10):
    g=coeff(f2,i,n0=100,m=10000)
    t2.append(g)

n=1;tol=0.1e-6
tes=np.linspace(-np.pi,np.pi,100)
old=[]
s=expansion(f2,n)
for x in tes:
    old.append(s(x))
new=[]

while n<100:
    n=n+1
    u=expansion(f2,n)
    for x in tes:
        new.append(u(x))
```

```
        e=[]
        for a,b in zip(old,new):
            if b< 6e-15:
                err=abs(b-a)
                e.append(err)
            else:
                e.append((b-a)/b)
        minv=min(e)
        if minv<=tol:
            break
        else:
            old=new
            new=[]
n_tol=n
z1=["C0","C1","C2","C3","C4"]
z2=["C0","C1","C2","C3","C4","C5","C6","C7","C8","C9"]
data1={"Coefficient corresponding to nth Legendre Polynomial":z1,"Value of C
print(pd.DataFrame(data1))
print()
print("Non-zero coefficients in the expansion of the function of f (x) = 2 +
print()
data2={"Coefficient corresponding to nth Legendre Polynomial":z2,"Value of C
print(pd.DataFrame(data2))
print("Number of terms required in the expansion of sin(x)*cos(x) which resul

#(c)
x_a=np.linspace(-2,2,100)
n_a=[1,2,3,4,5]
d1=[]
for n in n_a :
    d1.append(expansion(f1,n))
e1=[];e2=[];e3=[];e4=[];e5=[];e6=[]
for x in x_a:
    e1.append(d1[0](x))
    e2.append(d1[1](x))
```

```
    e3.append(d1[2](x))
    e4.append(d1[3](x))
    e5.append(d1[4](x))
    e6.append(f1(x))

fig, (ax1, ax2) = plt.subplots(1, 2)
fig.suptitle('Ques 2(c)')

ax1.plot(x_a,e1,linestyle='dashed',label="n=1")
ax1.plot(x_a,e2,linestyle='dashed',label="n=2")
ax1.plot(x_a,e3,linestyle='dashed',label="n=3")
ax1.plot(x_a,e4,linestyle='dashed',label="n=4")
ax1.plot(x_a,e5,linestyle='dashed',label="n=5")
ax1.plot(x_a,e6,label="exact")
ax1.grid()
ax1.legend()
ax1.set(xlabel="x",ylabel="f(x)",title="Series calculated for polynomial $2+3



n_a=[2,4,6,8,10]
d2=[]
for n in n_a :
    d2.append(expansion(f2,n))
we1=[];we2=[];we3=[];we4=[];we5=[];we6=[]
for x in x_a:
    we1.append(d2[0](x))
    we2.append(d2[1](x))
    we3.append(d2[2](x))
    we4.append(d2[3](x))
    we5.append(d2[4](x))
    we6.append(f2(x))

ax2.plot(x_a,we1,linestyle='dashed',label="n=2")
ax2.plot(x_a,we2,linestyle='dashed',label="n=4")
```

```
ax2.plot(x_a,we3,linestyle='dashed',label="n=6")
ax2.plot(x_a,we4,linestyle='dashed',label="n=8")
ax2.plot(x_a,we5,linestyle='dashed',label="n=10")
ax2.plot(x_a,we6,label="exact")
ax2.grid()
ax2.legend()
ax2.set(xlabel="x",ylabel="f(x)",title="Series calculated for $sin(x)*cos(x)$
plt.show()
```

3 Discussion and Result



In the first plot above we can clearly see that as we increase the value of n, the series expansion gives result more and more close to real value or we can say that it converges to true value. The plot for n=5 overlaps with that of true value.

In the second plot also it can be seen that as we increase the value of n, the value converges to true value.The plot for n=10 is approximately equal to that of exact solution.