51/55

PHYS304 HW4

Intouch Srijumnnong (Indy), worked with JT and Ben
Haverford College
(Dated: March 4, 2023)

+21

1. EXERCISE 5.4.2: THE DIFFRACTION LIMIT OF A TELESCOPE PT. 2

Lats week, I chose to do the problem 5.4 and wrote the Bessel function that calculates the value of $J_m(x)$ using Simpson's rule according to the equation given in the problem as

$$J_m(x) = \frac{1}{\pi}\int_0^{\pi} cos(m\theta - xsin\theta)d\theta. \quad (1)$$

In the problem this week, we are asked to compare the results of the Bessel function above to the computation of Bessel functions via recursion given by

$$J_{n+1}(x) + J_{n-1}(x) = \frac{2n}{x}J_n(x). \quad (2)$$

by plotting the fractional error between the two methods as a function of x for n = 2, 3, and 4. I start by importing the $J_m(x)$ using Simpson's rule from last week and define it in the beginning. Then I create a list of n from 2 to 4 and run the loop of y from the list of n for θ from 0 to π. After that, I run another loop of x inside the previous function from the list of x from 1 to 20 given in Newman 5.4 using arrange function from numpy for returning evenly spaced values within a given interval in step of 0.1. In order to plot the fractional error as a function of x, we need to have the calculation of the error running in the loop of the list of x from n = 2, 3, and 4. The calculation of the error is given by

This is the right equation, but not what your code is doing

$$\frac{J_m x_{Simpson's} - J_m x_{Recursion}}{J_m x} \quad (3)$$

After calculating the error values in the loop of x, we can plot the fractional error between the two methods as a function of x for n = 2, 3, and 4 in one graph, resulted below in (1)

I also plot another graph of the error between the two methods as a function of x for n = 2, 3, and 4 according to

$$J_m x_{Simpson's} - J_m x_{Recursion} \quad (4)$$

with out dividing the the Bessel function of $J_m(x)$ to the difference of the two methods, resulted in the graph below (2)

A log scale would be helpful

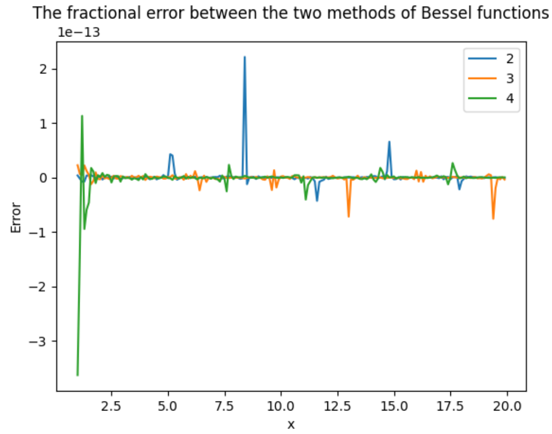

FIG. 1: The plot of the fractional error between the two methods as a function of x for n = 2, 3, and 4

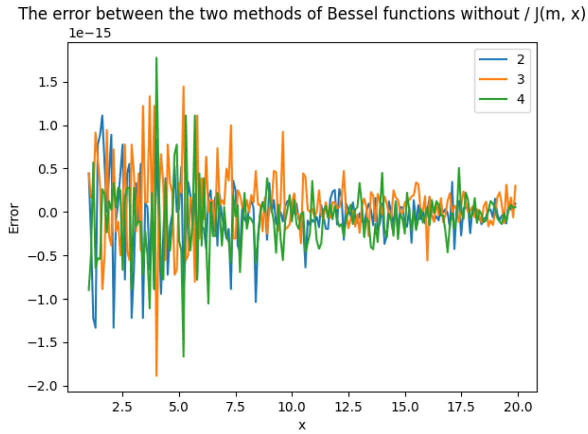

FIG. 2: The plot of the error between the two methods as a function of x for n = 2, 3, and 4 with out the division of $J_m(x)$

+25

2. EXERCISE 5.13: QUANTUM UNCERTAINTY IN THE HARMONIC OSCILLATOR

In this problem, we are given the wavefucntion of the nth energy level of the one-dimensional quantum harmonic oscillator or also known as a spinless point particle in a quadratic potential well, in units where all the constants are 1. The function is given by

$$\psi_n(x) = \frac{1}{\sqrt{2^n n!\sqrt{\pi}}}e^{-x^2/2}H_n x \quad (5)$$

for n = 0 ... ∞ where $H_n(x)$ is the nth Hermite polynomial. Hermite polynomials satisfy a relation somewhat similar to that for the Fibonacci numbers, although more complex given by

$$H_{n+1}x = 2xH_n(x) - 2nH_{n-1}(x). \quad (6)$$

The first two Hermite polynomials are $H_0(x) = 1$ and $H_1(x) = 2$x.

a) We are asked to a user-defined function H(n,x) that calculates $H_n(x)$ for given x and any integer n $\geq$ 0, and using the function to make a plot that shows the harmonic oscillator wavefunctions for n = 0, 1, 2, and 3, all on the same graph, in the range x = -4 to x = 4. I start by defining the function of Hermite polynomials using the if else function according to the definition and the condition for the first two terms given in the problem. Then I define another function for the wavefunction as wave(n, x) according to the definition given above. I limit the x values from -4 to 4 giving the minimum of x = -4 and maximum of x = 4, then I calculate the wavefunction in a for loop of n values from 0 to 3 as asked in the problem, and plot the graph of harmonic oscillator wavefunctions for n = 0, 1, 2, and 3, all on the same graph, in the range x = -4 to x = 4, as shown below in (3).

Great plot!

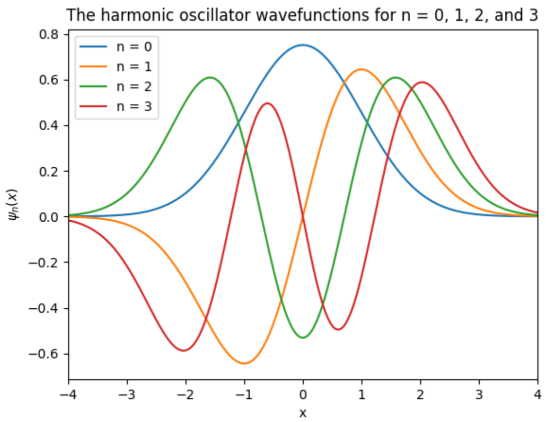

FIG. 3: The plot of harmonic oscillator wavefunctions for n = 0, 1, 2, and 3 in the range x = -4 to x = 4

b) In the same approach of a), I make a separate plot of the wavefunction for n = 30 from x = -10 to x = 10, resulted in the graph shown below (4).

c) The quantum uncertainty in the position of a particle in the nth level of a harmonic oscillator can be quantified by its root-mean square position $\sqrt{\langle x^2 \rangle}$, where

$$\langle x^2 \rangle = \int_{-\infty}^{\infty} x^2 \left|\psi_n(x)\right|^2 dx \tag{7}$$

We are asked to Write a program that evaluates this integral using Gaussian quadrature on 100 points, then calculates the uncertainty (the root-mean-square position of the particle) for a given value of n, and calculate the uncertainty for n = 5. I define the function of the the root-mean-square position according to the equation given above, and give the coefficient of the integration, then return the value for the calculation of the integral. I import the function gaussint from the previous homework which is based on the function gaussxwab from

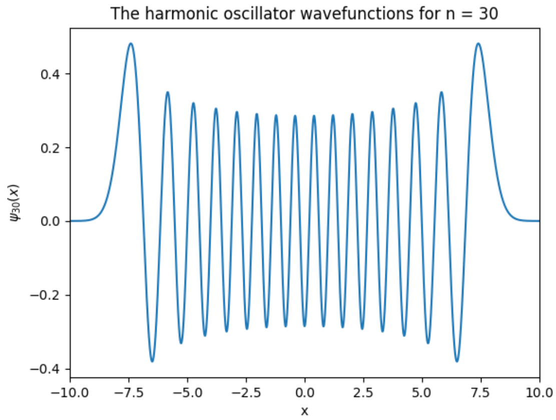

FIG. 4: The plot of harmonic oscillator wavefunctions for n = 30 in the range x = -10 to x = 10

gaussxw.py provided in Newman, 5.6.2 for the calculation of the integral in the quantum uncertainty equation. Given N = 100 points and n = 5, the program results in the uncertainty of 2.3452078737858177 as expected.

3. FEEDBACK

+5

I spent about 8 hours on this homework including meeting with classmates and attending office hours outside of classes. I think the homework is relatively short compared to others, however appropriate and a nice addition to the extended material of integration error, corresponding well during the time of the class. Also, it is helpful to have it short during the week before the break due to the tight work we are having this week. Thank you for that as well.

5.4.2)

→ import J(m, x): from last week

→ n = [2, 3, 4]

[x] ⇒ 1 – 20 → use linspace or arange . 0.5 # added step

→ for J in [n] loop J

f = []

error = []

θ = 0 - pi → linspace or arange

→ for x in [x] loop x

error = $\dfrac{J_m(x)_{simpson} - J_m(x)_{recursive}}{J_m(x)}$ → plotting error as a function of x

→ $\dfrac{LHS - RHS}{f(x)}$

→ ~~plot~~ show plot → plot (in loop)

5.13) → def H(n,x)

if n = 0
    return 1
else
    n = 1
    return 2x

else return $2x(H(n-1,x)) - 2(n-1)H(n-2,x)$

→ def wavefunction (n,x)

$$\phi_n(x) = \frac{1}{\sqrt{2^n n! \sqrt{\pi}}} e^{-x^2/2} H_n(x)$$

factorial from math

a) limit x = −4 → 4

→ for n from 0 − 4 loop

for n = 1, n = 2, n = 3, n = 4

wavefunction

plot (x, wavefunc) #

b) n = 30

limit x = −10 → 10

→ wave function

→ plot (x, wave function) #

c) N = 100 points

n = 5

→ def f(z)

code for $\int_{-\infty}^{\infty} x^2 |\phi_n(x)|^2 dx$

→ use gaussint(f, b) # from last week

Ans = $\sqrt{\text{gaussint}(f, b)}$ #

Problem 5.4 Extra

Srijumnong +21

Computational Physics/Astrophysics, Winter 2023:

Grading Rubrics [1]

Haverford College, Prof. Daniel Grin

For coding assignments, roughly 25 points will be available per problem.

+4 1. Does the program complete without crashing in a reasonable time frame? If yes, up to +3 points.

+1 2. Does the program use the exact program files given (if given), and produce an answer in the specified format? If yes, +1 points

+1 3. Does the code follow the problem specifications (i.e numerical method; output requested etc.) Up to +2 points

You are finding the error between the two sides of the recursion formula rather than using the recursion to get the Bessel function and finding the error in comparison to not using recursion

+2 4. Is the answer correct? Up to+4points

You have not really used the recursive formula to calculate the Bessel function because in every step you are integrating the exact formula for more than just J_0 and J_1

+2 5. Is the code readable? Up to+2points

. 5.1. Are variables named reasonably?

. 5.2. Are the user-functions and imports used?

. 5.3. Are units explained (if necessary)?

. 5.4. Are algorithms found on the internet/book/etc. properly attributed?

[1] Inspired by rubric of D. Narayanan, U. Florida, and C. Cooksey, U. Hawaii

+3 6. Is the code well documented? +3points

. 6.1. Is the code author named?

. 6.2. Are the functions described and ambiguous variables defined?

. 6.3. Is the code functionality (i.e. can I run it easily enough?) documented?

+9 7. LaTeX writeup (up to 10 points)

. Are key figures and numbers from the problem given? (3 points)
. Is a brief explanation of physical context given? (2 points)
. If relevant, are helpful analytic scalings or known solutions given? (1 point)
. Are 3-4 key equations listed (preferably the ones solved in the programming assignment) and algorithms named? (2 points)
. Are collaborators clearly acknowledged? (1 point)
. Are any outside references appropriately cited? (1 point)

It is hard to tell what is going on in the fractional error plot, a log-log scale would have probably been helpful

Note, even if (1), (2), (3), or (4) are not correct, one can still obtain many points via (5), (6), and (7).

Problem 5.13

Srijumnong +25 Great Work!

Computational Physics/Astrophysics, Winter 2023:

Grading Rubrics [1]

Haverford College, Prof. Daniel Grin

For coding assignments, roughly 25 points will be available per problem.

+3 1. Does the program complete without crashing in a reasonable time frame? If yes, up to +3 points.

+1 2. Does the program use the exact program files given (if given), and produce an answer in the specified format? If yes, +1 points

+2 3. Does the code follow the problem specifications (i.e numerical method; output requested etc.) Up to +2 points

+4 4. Is the answer correct? Up to+4points

+2 5. Is the code readable? Up to+2points

. 5.1. Are variables named reasonably?

. 5.2. Are the user-functions and imports used?

. 5.3. Are units explained (if necessary)?

. 5.4. Are algorithms found on the internet/book/etc. properly attributed?

[1] Inspired by rubric of D. Narayanan, U. Florida, and C. Cooksey, U. Hawaii

+3 6. Is the code well documented? +3points

. 6.1. Is the code author named?

. 6.2. Are the functions described and ambiguous variables defined?

. 6.3. Is the code functionality (i.e. can I run it easily enough?) documented?

+10 7. LaTeX writeup (up to 10 points)

. Are key figures and numbers from the problem given? (3 points)
. Is a brief explanation of physical context given? (2 points)
. If relevant, are helpful analytic scalings or known solutions given? (1 point)
. Are 3-4 key equations listed (preferably the ones solved in the programming assignment) and algorithms named? (2 points)
. Are collaborators clearly acknowledged? (1 point)
. Are any outside references appropriately cited? (1 point)

Note, even if (1), (2), (3), or (4) are not correct, one can still obtain many points via (5), (6), and (7).