Exercise 09

Superposition

PLEASE SUBMIT YOUR SOLUTION BEFORE **THURSDAY, 19 JANUARY, 8.00 A.M.** to
oliver.lemke@fu-berlin.de

1 Representation of wave functions with m $\neq$ 0 10 Points

To represent pictorially a wave function, especially the angular part, is difficult as for all $m \neq 0$ a complex part is added. This can be seen for example for $l = 1$, where the angular part is defined as:

$$Y_1^{+1}(\Theta, \Phi) = \left(\frac{3}{8\pi}\right)^{\frac{1}{2}} \sin(\Theta)\ \mathrm{e}^{+i\Phi} \tag{1}$$

$$Y_1^0 = \left(\frac{3}{4\pi}\right)^{\frac{1}{2}} \sin(\Theta) \tag{2}$$

$$Y_1^{-1}(\Theta, \Phi) = \left(\frac{3}{8\pi}\right)^{\frac{1}{2}} \sin(\Theta)\ \mathrm{e}^{-i\Phi} \tag{3}$$

As the probability density for Y_1^{+1} and Y_1^{-1} is the same and both functions correspond to the same energy eigenvalue, we know that every linear combination of those two functions is also an energy eigenfunction with the same energy. Out of this information we can define two new wave functions which we callp_x and p_y according to:

$$p_x = \frac{1}{\sqrt{2}}\left(Y_1^{+1} + Y_1^{-1}\right) \tag{4}$$

and

$$p_x = \frac{1}{\sqrt{2}}\left(Y_1^{+1} - Y_1^{-1}\right) \tag{5}$$

1. Calculate the probability density of Y_1^{+1} and Y_1^{-1}
2. Calculate p_x and p_y and show that only the real part is left

2 Approximation of functions using superposition 30 Points

Consider a function

$$f(x) = N \cdot \exp\left(-\frac{a}{2}x^2\right) \cdot \left(\frac{m\omega}{\hbar\pi}\right)^{\frac{1}{4}} \cdot \left((3x)^3 - 1\right) \tag{6}$$

with

$$a = \sqrt{\frac{m\omega}{\hbar}} \tag{7}$$

1. Plot the function
2. Calculate the normailzation coefficient N (numerically) and normalize the function (hint: Divide the original function by the square root of the density).

3. Now consider that the function can be expressed as a linear combination of functions:

$$f(x) = \sum_{i=0}^{\infty} c_i \chi_i(x) \tag{8}$$

where $\chi_i(x)$ denotes an orthonormal basis set

$$< \chi_i | \chi_j >= \delta_{ij} \tag{9}$$

In the following task you can see that the definition and number of these basis set strongly influence the quality of the approximation.

(a) One of the easiest thinkable functions that can be applied are step functions, which are defined as:

$$\chi_i = \begin{cases} 0 & \chi_i \notin S_i \\ N_i & \chi_i \in S_i \end{cases} \qquad \text{with } N_i^2 = \frac{1}{\frac{x_{max}-x_{min}}{n}} \tag{10}$$

To achieve such step functions the position x is divided into n equally-sized bins $\{S_1, S_2, ..., S_n\}$, which creates n stepfuntions $\{\chi_1, \chi_2, ..., \chi_n\}$.

i. Convince yourself that the basis functions are orthonormal (Use *Python)*

ii. Write a *Python* script that calculates $f(x)$ as a superposition of 10, 20 and 100 functions (Hint: To do this you have first to calculate the coefficients c_n). Use the parameters

$$m = 1 \qquad \hbar = 1 \qquad \omega = 1 \qquad x = [-5, 5] \qquad \Delta x = 0.01$$

Solve furhermore the additional tasks:

A. Calculate

$$\sum_{i=0}^{n} c_i^2 \tag{11}$$

Is the approximated function normalized?

B. Plot the orignal $f(x)$ and the approximated $f(x)$. Fill the space between both curves (Hint: Use the command plt.fill_between()) and calculate the approximation error:

$$\Delta = \int_{x_{min}}^{x_{max}} \left| \left[\sum_{i=0}^{n} c_i \chi_i(x) \right] - f(x) \right| dx \tag{12}$$

C. Plot the error as a function of the number of basis functions

(b) Now consider the eigenfunctions of a harmonic oscillator which are defined as:

$$\chi_i = \phi_i = N_i \cdot H_i \cdot \exp\left(-\frac{a}{2}x^2\right) \tag{13}$$

with

$$N_n = \frac{1}{\sqrt{2^n n!}} \cdot \left(\frac{\omega m}{\pi \hbar}\right)^{\left(\frac{1}{4}\right)} \tag{14}$$

and a as defined in equation 7. H_i denotes the Hermite polynomials (Accesible in *Python* via np.polynomial.hermval()). Use your *Python* script from part (a) and calculate $f(x)$ as a super position of 2, 3 and 5 harmonic oscillator wave functions using the same parameters. Repeat the tasks of (a) and add the following informations:

i. Consider the Hamiltonian of this function as:

$$\hat{H} = \hat{T} + \hat{V} = \frac{\hat{p}^2}{2m} + \frac{1}{2} m \omega^2 \hat{x}^2 \tag{15}$$

Calculate the expectation value of the energy

$$\langle E \rangle = \left\langle f(x) \middle| \hat{H} \middle| f(x) \right\rangle \approx \left\langle \sum_{i=0}^{n} c_i \chi_i(x) \middle| \hat{H} \middle| \sum_{i=0}^{n} c_i \chi_i(x) \right\rangle \tag{16}$$

A. for the original function $f(x)$ (Calculate the second derivative of $f(x)$ with pen and paper)
B. for the superposition
C. out of the coefficients c_n (Hint: use the energy eigenvalues of the harmonic oscillator)

ii. Plot the first 5 Hermite polynomials

3 Python Exercise 10 Points

```
import random
import sys
```

3.1

Use `sys.argv` to save a user argument in a variable called `greet`.

3.2

Ask for the user's name. Then use the greeting saved in `greet` to greet the user with their name.

3.3

Ask the user for the current year and then print the next leap year.

- A year is a leap year if it can be evenly divided by 4;
- If the year can be evenly divided by 100, it is NOT a leap year, unless;
- The year is also evenly divisible by 400. Then it is a leap year.

3.4

Write a guessing game where the user has to guess a secret randomly generated number between 0 and 100. After every guess the program tells the user whether their number was too large or too small, until the correct number is found.