Phys 256 — Homework 3

Student Name

Instructions

- From now on, feel free to use AI assistants for code generation, but we will check that you understand what your code is doing. We strongly advise to use the one provided by W&M as we instructed you. **Do not use AI for the report generation!** It is ok to ask AI for exotic parameters to improve your plots or check grammar but do not use AI to write the text of the report.
- Submit written report in PDF format (with inlined code snippets, if it is needed or required). Report must be submitted to Gradescope `hw03report` assignment. Remember that automatic checker test the functionality but not your test cases.
- Submit Python code to Gradescope `hw03code` assignment, only for the problems which explicitly require it.
- When required, all relevant code for a given problem should be in file named as `pN.py` where 'N' stands for problem number, e.g. `p1.py`, `p2.py`, or `p10.py`. All code files should be contained in one 'zip' archive and uploaded for grading to Gradescope.
- The class web page will have templates with function definitions to which you must adhere!
- Do not forget to discuss test cases in your report.
- Review the function `scipy.optimize.curve_fit` from `scipy` package, and use it where the fitting required.
- If you asked to fit something, show in your report the plot of the data and the fitted curve.
- All data files are provided at the class web page and saved in comma separated format (CSV). In general, they can be loaded with `data=np.loadtxt("filename.dat", delimiter=",", skiprows=Num)` command. Some of them have header, i.e. description of columns (than you need to use `skiprows` argument, some do not have it. You can see if the header is present by opening a file in a text editor.

Problem 1: (5 points)

This problem requires code submission in the file `p1.py`.

During photo processing quite often is needed to 'smooth' or defocus a photo. At the simplest level it can be done by creating a new 2D array where every cell is averaged value of the cell itself with it nearest neighbor as shown in 1. In this task we will implement the periodic boundary condition where a neighbor could be on the opposite end of the array (top and bottom, or left and right).

Implement a smoothing function `smooth(dataIn)` where `dataIn` is a 2D array of real numbers with arbitrary size (read about how to use outcome of `dataIn.shape` to get the array shape, i.e. number of rows and columns). The function must return the smoothed array of the same

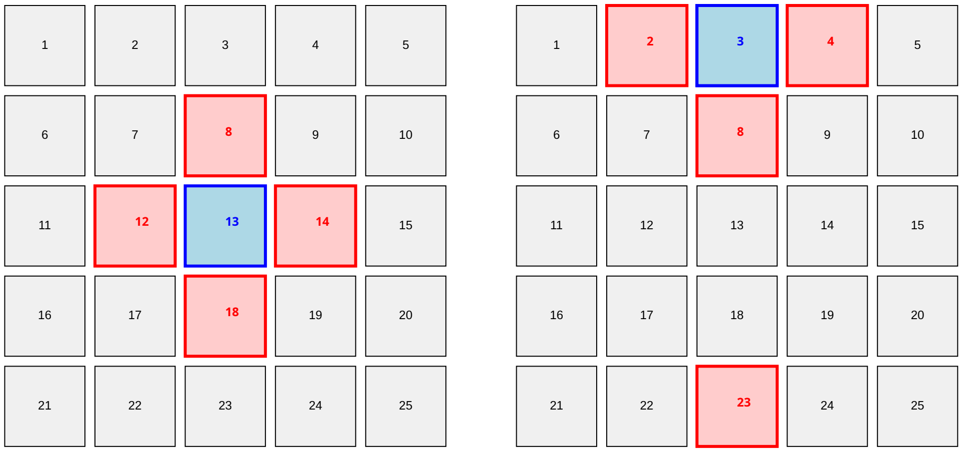

Figure 1: Array smoothing. The smoothed location shown in blue and the neighbors as red. Left standard case with resulting value $(12 + 13 + 14 + 8 + 18)/5$, right shows the periodic boundary condition with resulting value $(2 + 3 + 4 + 8 + 23)/5$.

shape. The function must return the smoothed array of the same shape. The function must return the smoothed array of the same shape.

Problem 2: (5 points)

Recall the problem about resistor from the previous homework 2. This time we will use a fitting technique to extract resistance.

Download data file `'hw02dataset.dat'` from the class web page. It represents the result of someone's attempt to find the resistance of a sample via measuring voltage drop (V), which is the data in the 1st column, and current (I), listed in the 2nd column, passing through the resistor. Judging by the number of samples it was an automated measurement.

Using Ohm's law $V = RI$ and a linear fit of the data with one free parameter (R) find the resistance (R) of this sample. What are the errorbars/uncertainty of this estimate? Does it come close to the one which you obtained via the method used in homework 2?

Problem 3: (5 points)

You are making a speed detector based on the Doppler effect. Your device detects dependence of the signal strength vs. time, which is recorded in the `'hw_fit_cos_problem.dat'` file (the first column is time and the second is the signal strength).

Fit the data with

$$A \cos(\omega t + \phi)$$

where A, ω and ϕ are the amplitude, the frequency and the phase of the signal, and t is time.

Find fit parameters (the amplitude, the frequency and the phase of the signal) and their uncertainties.

Problem 4: (Bonus 2 points)

This is for the physicists among us. Provided that the above radar was using radio frequency, could you estimate the velocity measurement uncertainty? Is it a good detector to measure a car's velocity?

Problem 5: (5 points)

Experiment to do at home. Make a pendulum of variable length (0.1m, 0.2m, 0.3m, and so on up to 1m). Measure how many round trip (back and forth) swings the pendulum with each particular length does in 20 seconds (clearly you will have to round to the nearest integer). Save your observations into the simple text file with comma separated values (commongly referred CSV) in columns. The first column should be the length of the pendulum in meters, the second column the number of full swings in 20 seconds.

Write a script which loads this data file, and extract acceleration due to gravity (g) from the properly fitted experimental data. Recall that the period of the oscillation of a pendulum with the length L is given by the following formula

$$T = 2\pi\sqrt{\frac{L}{g}}$$

Problem 6: (5 points)

In optics, the propagation of the laser beams is often described in the Gaussian beams formalism. Among other things, it says that the optical beam intensity cross section is described by the Gaussian profile (hence, the name of the beams)

$$I(x) = A\exp\left(-\frac{(x-x_o)^2}{w^2}\right) + B$$

where A is the amplitude, x_o is the position of the maximum intensity, w is the characteristic width of the beam (width at $1/e$ intensity level), and B is the background illumination of the sensor.

Extract the A, x_o, w, and B with their uncertainties from the real experimental data contained in the file `'gaussian_beam.dat'`, where the first column is the position (x) in meters and the second column is the beam intensity in arbitrary units.

Is the suggested model describe the experimental data well? Why so?

Problem 7: (5 points)

Fit the data from the file `'data_to_make_fit_choice.dat'` with the above Gaussian profile. Is the resulting fit good one or not? Why so? Compare it to the Lorentzian model which is given by

$$I(x) = A\frac{w^2}{(x-x_o)^2 + w^2} + B$$

To assist your decision make plot of the residuals in both cases.