PHYS 251 - Homework 5

5 Homework 5

Please upload your answers to Bitbucket.

The documentation provided in the Python web page should be the first stop to get help. **Python documentation**.

5.1 Problem 1: Linear interpolation

Please write a script that reads the data file **velocity_run18.txt** with one header line and three columns of data. The first column is time in seconds, the second column is velocity in m/s and the third column is the uncertainty of the velocity in m/s.

- Plot velocity vs time and include the error bars for the velocity points. Add labels with units, a legend, and a grid to your plot.
- Use the formula of linear interpolation to get the values of velocity $v_{i+1/2}$ at each time $t_{i+1/2} = \frac{t_{i+1}+t_i}{2}$, where i runs from 1 to n, with n being the number of data time points. Plot this new data in a different figure. Add labels with units, a legend, and a grid to your plot.
- Using the scipy.optimize function **curve_fit()**, fit the original data with an appropriate model and include the uncertainty of the velocity in your fit calculation. Display the fit parameters. Plot a third figure with the original data as a scatter plot and the fit model as a solid line. Add labels with units, a legend, and a grid to your plot.

Save the figures in *png* format. You must create a document and insert all the figures into the document and add a caption to each figure.

Please place comments in your code.

Save your script as **hw5_1.py** and the document with the figures, Word or PDF, as **hw5_1_figs**. Upload the files to your Bitbucket account.

5.2 Problem 2: Data decimation and interpolation

Please write a script to read the data file **fc_thist_00520.txt**. The first two lines of this file are headers. Column #1 is time in seconds, column #2 is the scalar quantity pressure, columns #3, #4 and #5 are the Cartesian components of a three dimensional vector, velocity, column #6 is the norm of the velocity vector, column #7 is the scalar quantity temperature, and finally, columns #8 to #12 are the scalar quantities s_1, s_2, s_3, s_4 and s_5. The scalar quantities s_i are concentration of H_2S in air, and the units are ppm.

- Plot s_1 vs time. Add labels with units, a legend, and a grid to your plot.

- The s_1 and time arrays contain approximately 240,000 data points. You have to reduce the size of the array to a more manageable size. Please interpolate linearly the scalar s_1 at times $= 1.0,\ 2.0,\ 3.0,\ \ldots,\ 399.0,$ and 400.0 seconds. Plots the new set of values together with the original data. Add labels with units, a legend, and a grid to your plot.

- Repeat the previous step interpolating at times $= 1.0,\ 11.0,\ 21.0,\ \ldots,\ 391.0$ and 401.0 seconds. Make a third plot including the original data and the interpolated data every 10.0 seconds. Add labels with units, a legend, and a grid to your plot. Are there any significant differences between plots?

To calculate all the interpolated values in this problem, you should use the following interpolation formula:

$$y_i = y_1 + (x_i - x_1)\frac{(y_2 - y_1)}{(x_2 - x1)} \tag{1}$$

Save the figures in *png* format. You must create a document and insert all the figures into the document and add a caption to each figure.

Please place comments in your code.

Save your script as **hw5_2.py** and the document with the figures, Word or PDF, as **hw5_2_figs**. Upload the files to your Bitbucket account.

5.3 Problem 3: Bilinear interpolation

Please write a script to read the data file **data_2D_grid_T.txt**. The first 13 lines of this file are headers. These headers contain information regarding the data stored in the file. The data represent the position of points in a three dimensional system and the scalar values associated to each point. The three columns are x_i, y_i and z_i. The fourth column contains the scalar values.

- Plot the surface defined by the points $(x_i, y_j, f(x_i, y_j))$ using the function **plot_surface()**, where $f(x, y) = z$

- Using the bilinear interpolation formula, see Equation 2, calculate the values of z at the center of each of the rectangles defined by $(x_i, y_j), (x_{i+1}, y_j), (x_i, y_{j+1})$ and (x_{i+1}, y_{j+1}), as shown in Figure 1. The i values are the number of x points in the grid and the j values are the number of y points in the grid, see Figure 2. Plot the interpolated data using **plot_surface()**.

Bilinear interpolation formula:

$$f(x, y) = \frac{a + b + c + d}{(x_2 - x1)(y_2 - y_1)} \tag{2}$$

where

$$a = f(x_1, y_1)(x_2 - x)(y_2 - y) \tag{3}$$
$$b = f(x_1, y_2)(x_2 - x)(y - y_1) \tag{4}$$
$$c = f(x_2, y_1)(x - x_1)(y_2 - y) \tag{5}$$
$$d = f(x_2, y_2)(x - x_1)(y - y_1) \tag{6}$$

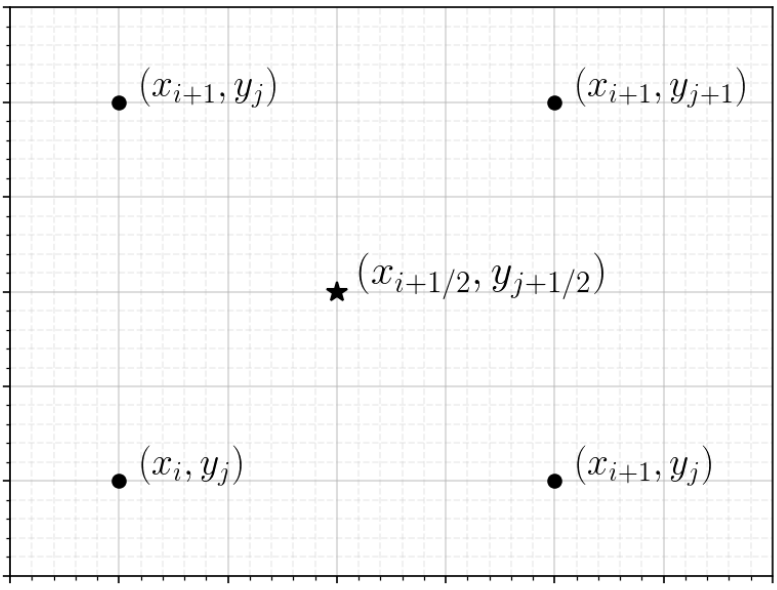

Figure 1: Bilinear interpolation to the center point.

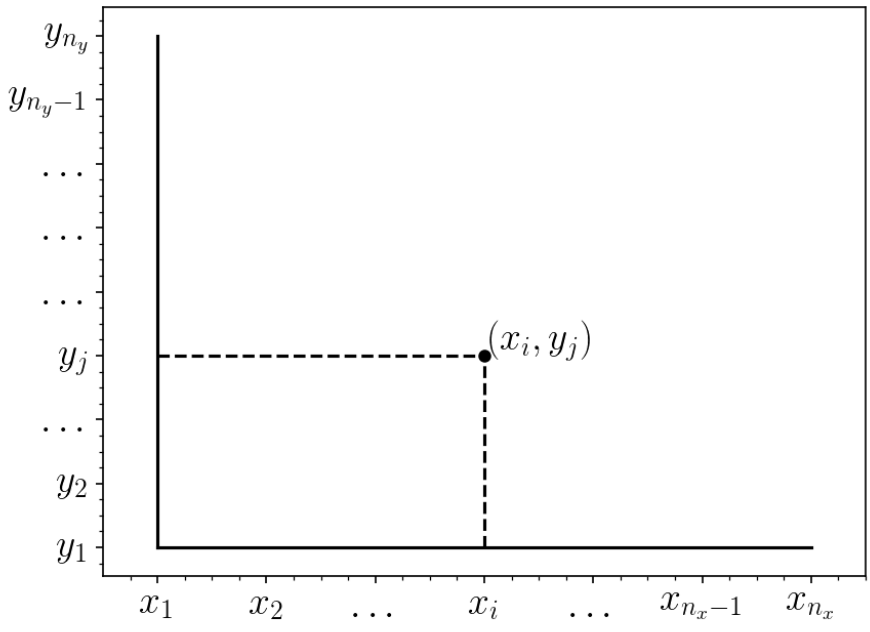

Figure 2: Cartesian grid.

You must write your script using **for** or **while** loops.

Save the figures in *png* format. Create a document and insert the figures into the document and add a caption to each figure.

Please place comments in your code.

Save your script as **hw5_3.py** and the document with the figures, Word or PDF, as **hw5_3_figs**. Upload the files to your Bitbucket account.