Assignment #1: Introduction to Data Reading, Plotting, and Curve Fitting (total 10 points), due by 11:59 pm Monday, 30 September 2024

Question 1: Introduce yourself. Provide *briefly* information about your computer skills (e.g., familiarity with different operating systems and programming languages) and describe *briefly* the kind of research you are or will be doing at UCalgary. If your project is still undefined, then describe your research aspirations and how they align with the research carried out in the group you are part of. Write if your research involves data analysis and which data analysis methods you are expected to learn. **(1 point)**

For the question above, use a maximum of 1 page for your answer. You can use any text editor of your preference, but, please, set the document layout with the following features:

- Page orientation: portrait;
- Page size: 8.5" × 11" (letter);
- Margins: 2.54 cm (top, bottom, left, right);
- Line spacing: between 1.0 and 1.15 pt;
- No double-column format. Write your answer as a 1-column text;
- Paragraph alignment can be 'align left' or 'justify';
- Font type and size: Times New Roman, 12 pt, non-italic, non-bold;
- Use *italic*/**bold**/<u>underline</u> only to emphasize brief pieces of text if needed.
- ONLY `pdf` FILES ARE ACCEPTED. Do not submit editable files such as `.docx`.

*****

Question 2: Photon-counting experiment. Consider the following two-dimensional data (also available in the `photon_exp.dat` file), in which X is the independent variable, and Y is the dependent variable assumed to be derived from a photon-counting experiment:

x_i	y_i
0.0	25
1.0	36
2.0	64
3.0	49
4.0	81

Note that the file has an additional column n, but that is only for indexing the rows. Assume that the error associated with each dependent datapoint y_i is the square root of the measurement (Poisson-like statistic).

(a) Find the best-fit parameters a, b of the linear curve $y(x) = a + bx$, the errors in the best-fit parameters and the correlation coefficient between them. The correlation coefficient is defined as $r = \mathrm{cov}(a, b)/(\sigma_a \sigma_b)$ in which cov is the covariance. **(1 point)**

(b) Calculate the minimum χ^2 of the fit. Obtain also the coefficient of determination (R^2) for this fit. Describe what these values indicate about the 'goodness-of-fit'. **(1 point)**

(c) Present a code (ONLY Jupyter Notebooks accepted) that computes the quantities requested in the items above. Explain your code and its outputs using comment lines, markdowns, and/or docstrings. This item focuses on the presentation, quality (e.g., the code does not contain too many redundant lines and is simple to follow), and documentation of your script, plus the quality of the plots. **(1 point)**

IMPORTANT: Your code needs to output all required numbers of the exercise and they should be printed clearly. The printed values should be properly identified with text saying what is being printed. We cannot debug your code and/or search the answers inside your code for you. This is a graded component of the course, therefore, we can ONLY evaluate what is uploaded and see printed on the screen. We will test other parameter variations and settings in your code, but we will not alter its structure and logistics. Interpret your code as any other graded component as 'in paper', meaning what you submitted is your final answer and what is not provided (or printed) in terms of answers, we cannot consider for marks.

*****

Question 3: Read, plot, and fit some data. The file `co2_mm_mlo.txt` contains CO_2 measurements from 1958 to present. To understand what the data is about, visit `https://gml.noaa.gov/ccgg/trends/mlo.html`. You will write a Jupyter Notebook script to read and describe the data via curve fitting schemes.

(a) Write a script that reads the data from the file (do not copy and paste the data into the script). Produce a scatter plot of "decimal date" (x-axis) versus "monthly average" (y-axis) with appropriate names on the axes and a plot title. Make a second plot with a logarithmic y-axis. Make a third plot of the square root of y values. Describe the differences (if any) in the plots. **(1 point)**

(b) Using curve fitting, make a prediction of what atmospheric CO_2 concentration will be in 2050, with a brief justification of your answer. Make a prediction using a linear model and a higher-order polynomial model (attention to not overfit!). **(2 points)**

(c) Implement Bootstrapping resampling to obtain the estimators (i.e., the fitting parameters) of the higher-order model you worked on in the previous item. Determine also the uncertainties of the estimators. **(2 points)**

(d) Present a code (ONLY Jupyter Notebooks accepted) that computes the quantities requested in the items above. Explain your code and its outputs using comment lines, markdowns, and/or docstrings. This item focuses on the presentation, quality (e.g., the code does not contain too many redundant lines and is simple to follow), and documentation of your script, plus the quality of the plots. **(1 point)**

IMPORTANT: Your code needs to output all required numbers of the exercise and they should be printed clearly. The printed values should be properly identified with text saying what is being printed. We cannot debug your code and/or search the answers inside your code for you. This is a graded component of the course, therefore, we can ONLY evaluate what is uploaded and see printed on the screen. We will test other parameter variations and settings in your code, but we will not alter its structure and logistics. Interpret your code as any other graded component as 'in paper', meaning what you submitted is your final answer and what is not provided (or printed) in terms of answers, we cannot consider for marks.

*****

Submission Information

Name the `pdf` file you generated for question 1 as `question1_assignment_1.pdf` and the codes for questions 2 and 3 as `code2_assignment_1.ipynb` and `code3_assignment_1.ipynb`, respectively. Note that only Jupyter Notebook files are accepted for the coding part. You will upload the `pdf` file of your written answer and your codes to the Gradescope platform. **It is important that you log in to Gradescope.ca and not Gradescope.com.** Your codes need to be documented, i.e., introduce comment lines to explain their main procedures. Pure lines of code without explanatory comments will have reduced marks. You can upload multiple files to the Gradescope platform and you can resubmit your work until the due date. Upload all files at once. We will test-run your submitted codes for syntax errors and check if they generate the requested outcomes. We will also check if the results/figures presented in your assignment are 'paper-like' quality and that the quantitative predictions are scientifically/mathematically founded.

ONLY `pdf` (for the written reports), `jpg/jpeg` (for figures), and `.ipynb`, (for codes) FILES ARE ACCEPTED. This means we are only accepting Python Notebooks.

IMPORTANT: In your codes include instructions on how to run it and, if applicable, include testing values for the initial conditions or settings that you attempted so we can reproduce the results. We will test other initial conditions and parameter variations to evaluate the robustness of your code. But we need a place to start. Include also information about the version of your coding platform. If this information is not included, points will be deducted.

IMPORTANT: Your codes need to output all required numbers of the exercises and they should be printed clearly. The printed values should be properly identified with text saying what is being printed. We can only assess and consider marks for numerical answers printed by your code (we cannot give marks to what we cannot see). We cannot debug your code and/or search the answers inside your code for you. This is a graded component of the course, therefore, we can ONLY evaluate what is uploaded and see printed on the screen. We will test other parameter variations and settings in your code, but we will not alter its structure and logistics. Interpret your code as any other graded component as 'in paper', meaning what you submitted is your final answer and what is not provided (or printed) in terms of answers, we cannot consider for marks.

IMPORTANT: (Follow-up from the item above) If an exercise asks you to write a code to compute or calculate a value, we will need to see the calculated output value and not simply a number manually printed using the `print` command. For instance, if an exercise asks you to write a code to obtain the velocity of a particle, we need to see the output velocity of the algorithm you implemented and not simply `print(2.34, 'km/s')`.

*****