Data Analysis with Python

1 Learning Objectives

At the conclusion of this activity you should be able to:

- Use Jupyter Notebooks to record your work as a laboratory notebook.
- Perform these analysis operations using Python:
  - Load a `.csv` file
  - Create a `pandas` DataFrame
  - Create a plot using `matplotlib`
  - Fit a line to data using `scipy`.

2 Taylor Reading Assignment

Read Chapter 2.2 (pp. 14-16) in Taylor, J. R. (1997) *An Introduction to Error Analysis.* 2nd ed., Sausalito, CA: University Science Books.

3 Getting Started

Follow these instructions to setup your directory structure for this assignment.

1. **Create a lab sub-directory:** Create a sub-directory for this lab in your course folder.
   For example: 🗀`AS_173_115`▸`Lab 1`.
2. **Download the assignment .ZIP file from Blackboard:** Download the assignment file: `DataAnalysisWithPython.ZIP` from Blackboard. Move the downloaded file to the directory created in the previous step.
3. **Extract the assignment file:** Unpack the contents of the .ZIP file by double-clicking (Mac) or right-clicking and selecting "extract" (Windows). When it is unpacked, it will have created a new folder called 🗀`DataAnalysisWithPython`. Once unpacked, you can safely delete the original .ZIP file.
4. **Launch Anaconda-Navigator:**
5. **Launch Jupyter Notebook:**
6. **Open the assignment notebook:** Navigate to your 🗀 `AS_173_115`▸`DataAnalysisWithPython` directory in the Jupyter Notebook dashboard. Open the Jupyter Notebook by clicking on: `DataAnalysisWithPython.ipynb`.

Homework:
The idea is that you will spend some time going through the `DataAnalysisWithPython.ipynb` tutorial on your own *before coming to the next class.*

*** Please come to class with questions about this code. There is a lot here and we do not expect that you will absorb it all with this one exposure. ***

Lab Work:
We will work on the rest of the assignment in the lab with you when we meet next time.

4 Procedure (In Lab Activity)

The following exercise has been adapted from Problem 2.18 in **John R. Taylor's**, *An Introduction to Error Analysis (2nd Edition)*.

4.1 The Physics: Projectile Motion

Newton's Laws, and the subsequent equations of motion, can be used to describe the behavior of a vertically shot projectile. In particular, the theory can be used to define a mathematical relationship between the initial vertical velocity, v, and the maximum height achieved by the projectile, h. The relationship between these two measurable quantities is given by

$$v^2 = 2gh, \tag{4.1}$$

where g is the acceleration due to gravity.

A group of students design an experiment to test the relationship given in Equation 4.1. A vertically aimed cannon is used to launch a projectile with varying initial velocities. The maximum height achieved by the projectile and the initial vertical velocity for each shot are measured and recorded to produce the data shown in Table 4.1.

Trial	**Height** (m)	**Velocity** (m/s)
1	0.4 ± 0.1	2.6 ± 0.6
2	0.8 ± 0.1	4.1 ± 0.4
3	1.4 ± 0.1	5.0 ± 0.3
4	2.0 ± 0.1	6.2 ± 0.3
5	2.6 ± 0.1	6.7 ± 0.4
6	3.4 ± 0.1	7.9 ± 0.3
7	3.8 ± 0.1	8.6 ± 0.1

Table 4.1: Experimental data.

4.2 Analysis

This tutorial will walk you through a short analysis using the data presented in Table 4.1. The analysis techniques that are required here will be used routinely throughout the rest of the course. Refer to the examples that are given in the `DataAnalysisWithPython.ipynb` Jupyter Notebook to do each step.

1. Record the data listed in Table 4.1 in a well-formatted Excel spreadsheet. Be sure to include clear column names with measurement units.

2. Save your spreadsheet as a `.csv` file in your 🗀`AnalysisWithPython`▸`data` directory. Give your data file a descriptive name (*e.g.* `projectile_data.csv`).

3. Open a new Python 3 notebook in Jupyter Notebook.

4. Create a Markdown cell to describe the purpose of this analysis notebook.

5. Read your `.csv` into python as a pandas DataFrame.

6. Use Python to compute the average uncertainty of the velocity measurements. Be sure to label and describe the result in your analysis notebook.

7. Compute the square of each velocity measurement, v^2. Save the result as a new column in your DataFrame. Give your new column a descriptive name and the appropriate measurement units.

8. Compute the uncertainty associated with each squared velocity, $\delta(v^2)$. This process is called "error propagation"*.

   The uncertainty $\delta(v^2)$ is found by taking the derivative of v^2 with respect to v and multiplying by the uncertainty, δv:

$$\delta(v^2) = \frac{\partial(v^2)}{\partial v} \cdot \delta v \tag{4.2}$$

$$\delta(v^2) = 2v \cdot \delta v. \tag{4.3}$$

   Save the result as a new column in your DataFrame. Give your new column a descriptive name and the appropriate measurement units.

9. Use `matplotlib` and the `errorbar()` function to create a plot of the squared initial velocity v^2 vs. the maximum height h. This time, your plot should include errorbars in both the x and y directions.

   Equation 4.1 is of the form:

$$y = Ax + B, \tag{4.4}$$

   where $y = v^2$, $A = 2g$, $x = h$, and $B = 0$. Based on the model (Equation 4.4) we expect the plotted data to look linear.

10. Format your `errorbar()` plot with a title and clear axis labels.

11. Code your own linear model function (similar to Equation 4.4). Your function should have three arguments:

    - The first argument must be the independent variable ($x = h$).
    - The remaining arguments will be the parameters for which you will fit the data.

*Error propagation quantifies the impact of measurement uncertainty on subsequent calculations – like the impact of $v \pm \delta v$ on calculating $v^2 \pm \delta v^2$ described above. Error propagation is one part of "Error Analysis". One of the learning goals of this course is to introduce you to the basic theory and mathematical tools used in error analysis.

12. Use `curve_fit()` to fit your linear model to the data.
13. Create a plot that shows your data (with error bars) and your fit results (plotted using the `plot()` function)
14. Print the fit results (parameters and the associated uncertainty) from `curve_fit()` with the correct number of significant figures. .
15. Report your measured value of g (and its associated uncertainty). Again, be sure to think about the number of sigificant figures that are appropriate for reporting this measurement.

5 SUBMITTING YOUR WORK

To submit your work, follow these steps.

1. **Create a .PDF of your jupyter notebook:**
   - Use the print option on your browser to print a PDF version of your notebook. For example, on Chrome, you would choose File > Print... to create a printable version of your Jupyter notebook.
   - Save your file in the `DataAnalysisWithPython` directory. Give your file a clear name, for example: `FirstnameLastname_DataAnalysisWithPython.pdf`
2. **Create a .ZIP file of your work:**
   - Use Finder (Mac) or File Explorer (Windows) to navigate to your `MeasurementIntroduction` directory.
   - Select your `.ipynb` notebook file and the `data` and `images` directories. You will need to hold the ⌘ or ctrl to select multiple files.
   - Right click and select the Compress 3 items (Mac) or Send to > Compressed (zipped) folder (Windows) menu option.
   - Give your file a clear name, for example: `FirstnameLastname_DataAnalysisWithPython.zip`
3. **Submit your .PDF to Turnitin:** Submit your .PDF file to the Turnitin assignment on our Blackboard site.
4. **Submit your .ZIP file to Box:** Submit your .ZIP file to the Box uploader on our Blackboard site.