

✓ Stretchy things

In this lab, we'll test and build models for stretchy things.

By the end of this session, you will present to your TA your claim and evidence about the extent to which a stretchy thing can be modeled through Hooke's Law.

✓ Learning goals

By the end of this unit, you should be able to:

- Implement practical strategies for mitigating bias when conducting experiments and analyzing data.
- Analyze data using least-squares fitting and residuals.
- Design high-precision experiments to test models, including identifying interesting variables to test, controlling variables, and reducing uncertainty.
- Use prior results to motivate a new research questions.
- Manage time by conducting pilot experiments and plotting as you go.

```
1 # Run this code cell to mount your Google Drive.
2 from google.colab import drive
3 drive.mount('/content/drive')
4
5 #Fill in the appropriate directory info in the ``...``
6 %run "/content/drive/My Drive/.../utilities.ipynb"
```

✓ Group organization

We will form new groups for lab this week. To help you get to know each other, and to ensure effective collaboration and fairness, group members should discuss and write down the answers to the following questions in order to establish a “Partner Agreement.”

Who are your Group Members and how do they prefer to be referred to (nicknames, pronouns, etc.)?

[Your group members' info here]

Discuss with your partners what you're personally hoping to get out of this course. How do your goals compare? Discuss how you can support each other's goals. Can you agree on a collective goal for the group (e.g. to have fun, to provide a physics study group)? Establish your collective purpose and record it below.

[Your collective purpose]

To complete each lab a number of distinct tasks will need to be completed by individual group members or collaboratively by the group as a whole. These tasks include, but are not necessarily limited to, setting up equipment, collecting data, taking notes, and performing analysis, and managing the group. How will you divide the work fairly in your group?

[Your group's plan here]

✓ Session 1: Testing Hooke's law

Hooke's law is a model that describes the relationship between force, F_y , and stretch distance, Δy , of a stretched spring: $F_y = -k\Delta y$. This model implies or assumes:

- A linear relationship ($k = \text{constant}$) between F_y and Δy over some range of values.
- Zero extension Δy corresponds to zero applied force F_y .
- There is no time, history, or other dependence in k or the measured values.

You will be designing experiments to test the validity of this model for a variety of systems under various conditions.

By the end of today's lab, you will present to your TA your claim and evidence about the validity of Hooke's law for a particular stretchy thing.

✓ A. Getting started

We'll start by familiarizing ourselves with analysis methods by testing a single spring system.

Choose a spring to study for your first investigation.

[Describe your spring here - dimensions, material composition, anything else notable]

Briefly explore a few ways to measure the applied force. For example, you might use the gravitational force from an object with known mass hanging vertically from the end of your spring. Or you might use force probes to measure a horizontally applied force for some distance. Describe which method you will use for your investigation with a brief justification based on the sources of uncertainty, possible systematic effects, or more logistical constraints.

[Your method for measuring force and justification]

Design an investigation that uses linear least squares fitting tools to test whether or when Hooke's law applies to your system (i.e., is force on your object linearly proportional to its extension?). Include details about the major sources of uncertainty, your strategies for mitigating them, and how you quantified the uncertainty in your measurement.

[Your design here]

Carry out your investigation and summarize your data, analysis, and interpretations below.

We've provided some code starters at the end of this document to get you going. There are multiple ways to set up the analysis, so we've chosen two approaches that are clear, but not efficient. You are welcome to take your own approach and PLEASE delete code for approaches that you decide not to use.

When copying code, please remember to edit comments, printouts, plot titles, axes labels, and any other artifacts of the code that need to be updated in order to align with whatever you are studying. Your code and code printouts should be comprehensible to anyone reading your notes and specific to your experiment.

Note about efficiency: You should be able to add data points to your code *as they are collected* so that you can look at your graph as you go. Do NOT wait to run the analysis until you have collected "all" your data. You may change what data you are collecting based on what you are seeing as you go.

✓ B. A new investigation

Next, design your own investigation to test the Hooke's Law model under various conditions. For example, you might consider testing:

- the linearity of the relationship for different types (rubber band, hair tie, stretch fabric, etc) or arrangements of stretchy objects,
- the linearity of the relationship for extreme extensions or compressions (check with your TA about over-stretching springs),
- the assumption that there is no time or history dependence in k .

Articulate a question for your investigation and briefly explain how your current investigation inspired your new question.

[Your question and evidence-based justification here]

Design your experiment, analyze your data, reflect on your results, and iterate as appropriate. Document your process using text and code cells below. **Be specific, but not overly verbose**, in your documentation. Since you are designing your experiment, we are not providing specific prompts or questions to answer but your notes must include the following:

- description of the measurement method/apparatus
- discussion of sources of uncertainty and methods to mitigate them
- data/plots
- discussion of analysis methods
- interpretation of results

- potential next steps

[Add text here]

1 #Create as many code cells and markdown cells as makes sense

By the end of the first session, you should have tested a simple spring system and at least one other system OR an assumption of Hooke's law. You should also have a brief outline for what you will do in the next session.

✓ Lab check out

By the end of the lab session, document below a claim about **the extent to which your stretchy thing(s) can be modeled through Hooke's Law** and briefly summarize your evidence that supports your claim. Present your claim to your TA or (if you are out of time), they will look over these notes to evaluate your claim.

In the checkout, they will be looking for:

- Whether the claim is supported by evidence,
- Whether that evidence is sufficient, and
- Whether it is clear that you collected the data.

[Your answer here]

Session 2: Keep experimenting

Following up on your results from last week, continue your investigation into the behavior of stretchy objects. As always, make sure you document all your decisions, methods, improvements, and results in the space below.

By the end of this session, you will present to your TA and other students your claim and a single graph of evidence about an experiment related to stretchy things.

✓ A. Your investigation

[Your methods and interpretations outlined with as many markdown cells as needed]

1 #Create as many code cells and markdown cells as makes sense

✓ B. Share your investigation

In lieu of a check out, during the last 30 minutes of the lab session, you will present your claim and **one graph of results** to your TA and peers in a mini poster session:

- You will display a single graph of results, sufficiently labeled to clearly indicate what was measured, and present it on your lab station monitor as a mini poster!
- Each group member should spend at least 10 minutes standing by their ,mini-poster and at most 20 minutes looking at other groups' mini-posters.
- When presenting your mini-poster, briefly describe the experimental question, how you produced the data in the graph, and how you are interpreting the graph towards developing a claim.
- When viewing another group's mini-poster, ask clarifying questions and offer alternative explanations or claims, or point out missing evidence for supporting the claim.

[Your claim and evidence (in words) here]

1 #Your single figure reproduced (or produced) here

Following the discussion, document below:

1. What feedback and ideas you received and/or ideas based on your discussion for what you could do differently or next time.
2. An example of interesting investigation or result from another group.
3. Examples of feedback and ideas your group offered to other groups.

[Feedback and ideas received here]

[Interesting investigations here]

[Feedback and ideas given]

✓ Appendix

✓ Code template to input your data

```
1 # Example 1: If you are NOT doing repeated trials
2
3 ## Enter your ordered extension values:
4 ext = np.array([...])
5
6 ## Enter your ordered force values, such that each entry corresponds to an extension value in the ext array in the same pos
7 ## (i.e., the third entry in the force array corresponds to the third extension in the extension array):
8 force = np.array([...])
9
10 ## Enter your estimates of the uncertainty in whichever variable is your dependent variable
11 ## (i.e., what should go on the y-axis), again ordered:
12 force_unc = np.array([...]) #comment this out if extension is your dependent variable
13 ext_unc = np.array([...]) #comment this out if force is your dependent variable
14
15 ## Print the output to check your work - make sure there are the same number of entries in each array
16 ## (e.g., one force and uncertainty in force corresponding to each extension)
17 print("Extensions:")
18 print(ext)
19
20 print("Force:")
21 print(force)
22
23 print("Uncertainty in dependent variable:")
24 print(unc)
```

```
1 # Example 2: If you ARE doing repeated trials and force is your dependent variable
2 # i.e., you are measuring the force required to stretch the spring through various distances
3
4 ## Enter your ordered extension values:
5 ext = np.array([...])
6
7 ## Enter your repeated measurements of the force for the first extension value in the ext array:
8 force_ext1 = np.array([...])
9 ## Enter your repeated measurements of the force for the second extension value in the ext array:
10 force_ext2 = np.array([...])
11 ## Repeat for all values of extension measured:
12 ...
13
14 ## Calculate the mean and standard uncertainty in the mean for the first set of force measurements:
15 force_ext1_mean = np.mean(force_ext1)
16 force_ext1_su = standard_unc_of_mean(force_ext1)
17 ## Repeat for all sets of force measurements:
18 ...
19
20 ## Create an array of all the means of the force measurements, ordered to align with the extension array:
21 force = np.array([force_ext1_mean, force_ext2_mean, ...])
22 ## Create an array of all the standard uncertainties of the mean force measurements, ordered to align with the extension array:
23 force_unc = np.array([force_ext1_su, ...])
24
25 # Print the output to check your work - make sure there are the same number of entries in each array (e.g., one force and uncertainty for each extension)
26 print("Extensions:")
27 print(ext)
```

```

28
29 print("Force:")
30 print(force)
31
32 print("Uncertainty in force:")
33 print(force_unc)

```

```

1 # Example 3: If you ARE doing repeated trials and extension is your dependent variable
2 # i.e., you are measuring the extension that results from various applied forces
3
4 ## Enter your ordered force values:
5 force = np.array([...])
6
7 ## Enter your repeated measurements of the extension for the first force value in the force array:
8 ext_force1 = np.array([...])
9 ## Enter your repeated measurements of the extension for the second force value in the force array:
10 ext_force2 = np.array([...])
11 ## Repeat for all values of force measured:
12 ...
13
14 ## Calculate the mean and standard uncertainty in the mean for the first set of extension measurements:
15 ext_force1_mean = np.mean(ext_force1)
16 ext_force2_su = standard_unc_of_mean(ext_force2)
17 ## Repeat for all sets of force measurements:
18 ...
19
20 ## Create an array of all the means of the extension measurements, ordered to align with the force array:
21 ext = np.array([ext_force1_mean, ext_force2_mean, ...])
22 ## Create an array of all the standard uncertainties of the mean extension measurements, ordered to align with the force array:
23 ext_unc = np.array([ext_force1_su, ...])
24
25 # Print the output to check your work - make sure there are the same number of entries in each array (e.g., one force and one extension)
26 print("Force:")
27 print(force)
28
29 print("Extensions:")
30 print(ext)
31
32 print("Uncertainty in extension:")
33 print(ext_unc)

```

Code template to make a plot

```

1 # Plot the graph of extension versus force with uncertainties
2 # Indicate the appropriate units for each axis label where it says [units]
3 %matplotlib inline
4
5 #If force is the dependent variable, use this code:
6 #Comment it out if not.
7 plt.figure()
8 plt.errorbar(ext,force, force_unc, fmt='.')
9 plt.plot(ext,force,linestyle='None')
10 plt.xlabel('Extension ([units])')
11 plt.ylabel('Force ([units])')
12 plt.show()
13
14 #If extension is the dependent variable, use this code:
15 #Comment it out if not.
16 plt.figure()
17 plt.errorbar(force, ext, ext_unc, fmt='.')
18 plt.plot(force,ext,linestyle='None')
19 plt.xlabel('Force ([units])')
20 plt.ylabel('Extension ([units])')
21 plt.show()

```

Code template for linear fits

```

1 # Find the line of best fit
2 #If force is the dependent variable, use this code:
3 #Comment it out if not.
4 autoFit(x=ext, y=force, dy=force_unc, title="Force on a [object] stretched to various extensions", xaxis="Extension ([units])")

```

```
5
6 #If extension is the dependent variable, use this code:
7 #Comment it out if not.
8 autoFit(x=force, y=ext, dy=ext_unc, title="Extension of a [object] due to various applied forces", xaxis="Force ([units])",
```

[Summarize your results, analysis, and interpretations here. Discuss the relevance of your results to the model you are testing, including how your data provides evidence for or against the assumptions of the model.]

✓ Code template for quadratic fits

The utilities file has a second autofit function that performs quadratic fits. It models functions of the form: $y=mx^2+b$.

To call the function, use the function call in the code cell below. Enter your raw measurements for y and x .

Talk to your TA if you'd like to try other functional forms.

```
1 autoFit2(x=..., y=..., dy=..., title="...", xaxis="...", yaxis="...")
```