

MOL518 Spring 2026

Homework 2

Due: Friday February 13, 2026

For this assignment, you will generate and then submit a Jupyter notebook on Canvas. You may use generative AI tools as per the syllabus unless otherwise stated in the problem.

First, make a new Jupyter notebook named `MOL518_HW2_<YourLastName>.ipynb`. All problems will be answered in this notebook. For each problem, start with a Markdown cell using `Problem XX` as the title, i.e. “# Problem 1”.

Problem 1: Looping Through Subplots

This problem uses the file `ecoli_drugs.csv` which we used in Lecture 3. This file has time in the first column and six drug growth curves in the next six columns. *Do not use AI to generate code for this problem. You may use AI to debug if you get an error or to help understand why your code isn't working properly.*

- Load the CSV file.
- Create a list of the six OD arrays in the following order: Rifampicin, Novobiocin, Trimethoprim, Chloramphenicol, Ampicillin, Gentamycin.
- Create a matching list of drug names.
- Use `plt.subplots` to make a 3x2 grid. In class we filled the grid by brute force. Here, fill the panels using **for loops**, i.e. no manual `axes[0,0]` indexing. Each panel should include a title and axis labels.

Problem 2: Mystery Data

This problem uses the file `mystery.csv`. *Do not use AI to generate code for this problem. You may use AI to debug if you get an error or to help understand why your code isn't working properly.*

- First, inspect the csv file. Then load the file and put the data into two 1D arrays called `x` and `y`.
- Make a scatter plot of `y` vs `x`.
- Count how many points have `y > 5` and display the answer.
- Now, repeat this counting for thresholds of 5, 10, 15, 20, and 25. Store the results in two lists: `thresholds` and `n_above`.
- Make a line plot of `n_above` vs `thresholds`.
- Save a CSV file called `mystery_summary.csv` with two columns: `threshold` and `n_above`.

Problem 3: Summary of Microscopy Images

In the `images` folder there are 20 microscopy `.tif` files. *Do not use AI to generate code for this problem. This problem has some things we only touched on, you may use AI to help solve the problem but make sure to follow the guidelines in the syllabus.*

- a) Use `pathlib.Path` to list all `.tif` files in the folder and print the total count.
- b) Print the names of the first 5 files (alphabetical order).
- c) For each filename, split the stem on underscores (`_`) and extract the cell line (the second field, e.g. `HeLa` in `2026-02-05_HeLa_DAPI_ctrl_F0V01.tif`).
- d) Use a dictionary to count how many images belong to each cell line, then print the dictionary.

Problem 4: The Effect of Bin Size

This problem uses the file `S0X2.expression.csv` which contains a column of expression values. *Do not use AI to generate code for this problem. You may use AI to debug if you get an error or to help understand why your code isn't working properly.*

- a) Load the expression values into an array.
- b) Make a histogram with 8 bins.
- c) Create a 1x3 grid of histograms for bin counts `[8, 20, 40]`. Use `plt.subplots` and a `for` loop to fill the panels.
- d) In 2–3 sentences, explain how the choice of bin size changes what you perceive about the distribution.

Problem 5: Replicate Growth Curves and the Mean

This problem uses the file `ecoli_growth_replicates.csv`. The first column is time (hours) and the next 10 columns is OD from replicate experiments. *Do not use AI to generate code for this problem. You may use AI to debug if you get an error or to help understand why your code isn't working properly.*

- a) Load the CSV file into a 2D NumPy array. Extract `time` and a 2D array `od_reps`.
- b) Plot all 10 replicate curves on the same axes in different colors.
- c) Compute the mean OD at each timepoint (average across replicates). Plot the mean OD as a thick black line on the same figure.
- d) Save a new CSV file called `ecoli_growth_mean.csv` with two columns: `time_hr` and `od_mean`.