Visualization
Prof. Bernhard Schmitzer, Uni Göttingen, summer term 2025

Problem sheet 1

- *Submission by **2025-05-21** 18:00 via StudIP **as a single PDF/ZIP**. Please combine all results into one PDF or archive. If you work in another format (markdown, jupyter notebooks), add a PDF converted version to your submission.*
- *Use Python 3 for the programming tasks as shown in the lecture. If you cannot install Python on your system, the GWDG jupyter server at https://jupyter-cloud.gwdg.de/ might help. Your submission should contain the final images as well as the code that was used to generate them.*
- *Work in groups of up to three. Clearly indicate names and enrollment numbers of all group members at the beginning of the submission.*

Exercise 1.1: errorbars and regression.
The file `data_sin.csv` contains a simple table with two columns, `x`, and `y`. `x` is sampled equidistantly from $[0, 1]$. `y` is obtained via $\texttt{y} = \sin(\texttt{x}) + \varepsilon$ where ε is i.i.d. Gaussian noise with standard deviation $\sigma = 0.2$.

1. Import the dataset into Python via pandas and create a simple scatter plot of `x` versus `y`, that represents the implied uncertainty in the values of the latter.
2. Apply Gaussian process regression (with noise) on the dataset, by adopting the example from https://scikit-learn.org/stable/auto_examples/gaussian_process/plot_gpr_noisy_targets.html. (You do not need to understand what exactly Gaussian process regression is.) Add the obtained mean prediction and the 95% confidence interval in the basic scatter plot from (1), similar to the example. This plot should have an informative legend.
3. As an alternative, use the pandas `rolling` method to compute the rolling mean of `x` and `y`, as well as the standard deviation of `y` on rolling windows of width 5 with respect to `x`. Add this information into the basic scatter plot of (1). This plot should have an informative legend.

Note: The uncertainty of the Gaussian process regression in (2) reflects the uncertainty of estimating the mean at each point. The standard deviation over each interval in the rolling approach in (3) reflects the variance of the data in each interval. Therefore the two ranges will be different.

Exercise 1.2: astroid data: grouped regression and log scale.
The dataset stored in `data_astroids.csv` contains information on (some) small objects in the solar system, obtained from the JPL/NASA Small-Body Database. Metadata is given in `data_astroids_meta-data.md`.

1. Import the dataset into Python as a pandas dataframe. Convert the columns `first_obs` and `class` to appropriate data types.

2. Create a chart that shows the relation between the variables `H`, `first_obs`, and `a`.

3. Split the dataset into 5 equal sized parts by applying the pandas method `qcut` to the variable `a` and then use pandas `groupby` in a suitable way. Modify the above plot to show the relation between `H` and `first_obs` for each `a`-group.

4. For each `a`-group, perform a local regression, such as kernel regression or LOESS[1] and add these regression lines to the plot. (You do not have to use vega altair for this and similar regression methods are also fine!)

5. Finally, show the relation between the variables `diameter` and `H` in a scatter plot. Consider the appropriate use of logarithmic axis scaling.

[1] https://en.wikipedia.org/wiki/Local_regression or https://altair-viz.github.io/user_guide/transform/loess.html