Expectations

☞ **Every plot needs to be commented**: describe the graph and its meaning.

☞ You need to **document your notebook**. Use comments and/or Markdown cells.

☞ The questions or sections tagged with ⋆ are optional.

Environment

⚠ You **can** work with Python (but feel free to choose your favorite language). A kickstarter notebook is provided to help you with the syntax. There is a lot of documentation online if needed (e.g., here and here).

- You can either use Jupyter on your computer or through online services such as CoCalc and Google Colab.
- You can use the plot libraries available to the language you chosed or write your results in a CSV file and plot them through, e.g., LibreOffice.

⚠ If you are running on Windows we recommend you to install Python using Conda.

- You will need to install the following dependencies (do not forget to create a virtual environment ;)):
  - Either Jupyter's jupyterlab or notebook to run the notebook,
  - pandas to process the data,
  - matplotlib or seaborn to plot your experiments.
- This project uses the Adult data set (only `adult.data`).

Queries

C Number of non-white people.

Implementing Laplace

Q1 Write a function to generate random numbers following the Laplace distribution from the uniform distribution[1].

Q2 Compare your own Laplace generator to the one from `numpy` based on a Kolmogorov–Smirnov test. Do they generate the same distributions?

Q3 Write a function that implements the Laplace mechanism (i.e., perturbing the result of a function based on the Laplace distribution parameterized by the ϵ privacy parameter and the sensitivity S_g).

Studying Laplace on Count and Sum Queries

Q4 Use the Laplace mechanism, with a varying privacy budget $\epsilon \in \{0.001, 0.01, 0.1\}$, to compute query C. Execute query C with each ϵ value 100 times, and for each execution of a query, measure the *relative error*[2]. Plot the average relative error (y-axis) with respect to the ϵ value (x-axis). You can use a logarithmic scale on the x-axis for a clearer graph[3]. Add to each graph, at each ϵ value, the standard deviation as a confidence interval.

[1] Check Wikipedia.

[2] The relative error is the absolute value of the difference between the perturbed result and the true result divided by the true result. More formally : given a true value v and its perturbed version v_{approx}, the relative error is $\eta = \left|\frac{v - v_{approx}}{v}\right|$.

[3] See, e.g., `numpy.std` or `ci="sd"` with seaborn.

Q5 Compare the relative error that you observe to the parameters of the Laplace distribution used for generating the perurbations. What can you observe and why?

Q6 Assume that you allow an unlimited number of queries. How many perturbed answers to query C are needed in order to be able to approximate ($\pm 0.1\%$) the true result of the query? Answer to this question empirically.