DATA SCIENTIST TECHNICAL EXERCISE

This exercise is aimed at testing the following skills:

- Coding in python or R, at a level sufficient to perform statistical analyses.
- Basic usage of git CVS (creating and using a repository on github or gitlab)
- Data cleaning and exploration
- Understanding of randomized experiments and how to analyze them
- Write code and analysis that is shareable among peers (reproducible research) and summarize results that are understandable by non-technical stakeholders.
- Basic plotting with matplotlib/seaborn or other library of your choice.

You are presented with a business problem and are expected to write code to answer the questions posed, and present those findings in a summary document.

Feel free to use either plain python files, jupyter notebooks or R / R with markdown as you wish. We recommend to present the findings in a brief markdown file.

The solution (code, final document, readme, etc) should be in a git repository which you should upload to github or gitlab, and provide us with the link.

We will try to execute your code locally so please include the necessary instructions in a README file, such as libraries required with a requirements.txt / pipfile / conda.yml etc...

The business problem

Spotahome is an international marketplace for mid to long-term rental where *Landlords* can list their properties to be rented, and *Tenants* can browse the inventory of properties (shared rooms, whole apartments, etc) across dozens of cities, and ultimately rent their new home without visiting it thanks to our *verification* service, which provides accurate pictures and videos of the property. Once a tenant rents a property, we call that a *Booking Request* and our business model consists on charging a variable *fee* to both the Tenant and Landlord. (In this exercise, let's consider that the landlord fee is always 0).

Suppose you already participated in designing an A/B experiment with a product owner. After four weeks of running, the experiment had finished and the product owner wants you to analyze the results.

The experiment consisted on **increasing the fees we charge to tenants** when they book a home (this is our "sale", and we call it **a booking request**).

While the experiment was running on our website, all of our users were randomly assigned to either **control** or **treatment** groups, with a 50%/50% split. The assignation is permanent for each user.

The treatment group users were presented with a higher fee (the commission they must pay to rent with us) in the shopping cart.

The hypothesis to test is that conversion rate may be affected by this increase in price, but the extra revenue earned per booking request may compensate for it. (For simplicity, let's say that the fee we charge on a booking request is exactly our revenue).

The fees are not fixed, but change per individual property. Thus you will see different revenues per booking request in the data provided, but overall all fees were increased by X% in the treatment group.

To analyze the results, the engineering team provided you with a (simplified) raw access log of website activity, in a SQLite file with one table named "access_log".

Each record represents an "event" performed by a user and contains:

- A timestamp indicating when the event ocurred. All records belong to the period where the experiment was running.
- A user_id which is a unique string per user.
- A "variant" field with values "A" or "B", indicating the control and treatment groups respectively.
- An "event_type" field representing the action peformed by the user, such as "property_view", "property_favorite_added", or **"booking_request".** The latter indicates that a booking request was made by the user.
- A field "revenue" which is NULL for events other than booking_request, and is filled with the revenue earned by that booking request (fee charged) on the events of the type "booking_request".

By the design of the experiment and our product, the following should be true:

- A user may be assigned to only one variant. There should be no events for the same user in different variants.
- A booking request always has a revenue greater than zero.
- A user can perform only one booking request.

However, there are some problems in the dataset that do not satisfy some of the above rules. You need to check the data to find those inconsistencies and apply the appropriate correction, explaining your reasoning (don't be afraid to discard rows if you think that's the only or best solution).

The two metrics of interest are:

- **Conversion rate (CVR) per user:** measured as "unique users that did a BR" / "total unique users".
- **Revenue per user: "**total revenue earned" / "total unique users". Note that we include here users that did not convert.

Questions to answer:

- **Conversion rate:** What is the uplift or downlift in the treatment group, if there's any? Is it statistically significant at 95% or 99% level? Indicate the kind of test performed and p-value.

- **Revenue per user:** What is the uplift or downlift in the treatment group, if there's any? Is it statistically significant at 95% or 99% level? Indicate the kind of test performed and p-value. Remember we mean revenue per "website visitor", not just per "user that converted".

- **What would be your recommendation to stakeholders regarding rolling out or rolling back the experiment?**

The stakeholder forgot what was the % increase in fees applied in the treatment group, so let's try to find out:

- Create a plot that shows a kernel density estimate of the revenue value for the booking requests, creating one different line per variant. What can be learned from this plot?
- Can you determine the "average" revenue per booking request in each variant, with a confidence interval of 95%? (e.g. "352.25 euros +/- 22.3 euros" , or "[329.95, 375.55] euros"). Choose an appropriate distribution as assumption.
- Finally, determine what was the percentage increase of the fees in treatment group. It is a reliable determination?