Information Visualization lab

First practical work

1 Introduction

We will work with a dataset with information about collisions in New York City, we are specifically interested in the summers of 2018 and 2020. Our goal is to create a static visualization that allows us to answer the following questions:

- Are accidents more frequent during weekdays or weekends? Is there any difference between before COVID-19 and after?
- Is there any type of vehicle more prone to participate in accidents?
- At what time of the day are accidents more common?
- Are there any areas with a larger number of accidents?
- Is there a correlation between weather conditions and accidents?

You may add extra questions. Some questions can be answered directly from the data, some others require data derivation. Most of the data processing and derivation can be carried out using Open Refine. Some derivations/calculations can be done interactively, but precalculation is typically always better. Remember to create DataFrames with the specific columns that you want to use to avoid disconnection problems in the Colab environment.

The visualization must be a multi-view visualization. Think carefully about the design. Test and redesign. The final combination has to be visualized with Streamlit.

2 Data

Collisions dataset: From collision dataset. Extract rows from Jun-Sep 2018 and Jun-Sep 2020.
Weather dataset: You will need to find it.
NY map: You will need to find it.

3 Data processing

You can process the data using either OpenRefine, or another tool. You can also process the data programmatically. You need to deliver the clean dataset.

Independently of the cleaning tool and process, you must describe your cleaning steps in your Google Colab document. If these are using pandas, for example, include the code in the document. We must be able to reproduce the steps and go from the raw data to the clean version.

4 Design and implementation

For the visualization, we need you to describe the design process also in the Google Colab document. This means that you may include all the steps that led you to the final visualization. You can remove (or group) some steps in the final document if you think it is better. But we need to see the design process, we want to understand how did you reach the final visualization. Before you start coding anything, you need to think about what visualizations will be provided. Note that the user needs to be able to answer the questions above with a single static visualization, that will include multiple views. The final visualization must be designed in altair and included as a single chart in Streamlit. Consider all sorts of charts that might be useful: line charts, bar charts, heat maps, treemaps ... Some views will contain several variables, so use visual cues, and proper palettes to ensure they are understood properly.

5 Delivery instructions

The work can be implemented in pairs or individually. You have to provide the clean data. You have to describe the cleaning procedure so that we can generate the clean data from the raw data following your steps. This description must go in the Colab document.

You must include a step-by-step description of how to solve tasks. These can go in the Colab document. For example, one might have:

- Question 1: Is there a difference in the number of accidents in the different years?
- Answer to Q1 could be: "In chart C1 you can see a grouped bar chart with the accidents per month along the different years. We use a different color for each year... And by checking *something that is clear enough*, we can see that the behavior is..."

The delivery must consist of a single ZIP file with a name that includes the authors, that contains the datasets (raw and clean), the Colab file(s) (ipnyb), the Python file that contains the streamlit code, and optional extra documents if required. The Colab file must be named after the names of the authors. Treat the Colab document as a report, including titles, boldfaces, etc., to make it easier to read.

The deadline for the delivery of this lab project is the **17th of November**.

6 Important remarks

The final grade will consider the number of variables included in the visualizations (these may include new calculated variables, such as averages, maxima, minima, etc.)(e.g., other columns from the data set...). Additionally, we will value the number of non-trivial tasks (adequately described in the documentation) that can be properly solved with your visualization tool.

Don't leave the project for the last day or do the minimum amount of work. In case of doubt, ask us whether the current work is enough or needs more effort.

7 Checklist for the delivery

This checklist serves as guidance for your delivery.

Global checklist:

- Name the file after the name(s) of the author(s).
- Include the name(s) of the authors also as the first line/cell in your notebook and in the Streamlit app (e.g., in an "About" option).
- Include the clean data.
- Compress all files in a single zip (do not use RAR or other formats) file.
- Ensure the names of the data files inside the notebook correspond to the ones you deliver.
- Make a single delivery per group.
- Ensure you properly cleaned the data (charts where "Undefined" or "N/A" appear do not inspire trust).
- Ensure the code executes without errors: last-minute changes may lead to typos.
- Ensure you include a step-by-step design process (does not need to include all minimal steps, if you prefer, but do not forget to include the discussion on design decisions).
- Ensure there is a final vis (using multiple views) that answers all the questions.
- Ensure there is a final list of questions and how to answer them with the final vis.
- Do not mix charts with other technologies, everything must be created using altair.

Google Colab document. For every chart, you must consider:

- Color blindness (e.g., coding anything just with a red-green palette may be problematic).
- Check the consistency of colors across the whole visualization (same color, same meaning in different charts).
- Do not forget to add meaningful titles, labels, messages if necessary...
- Extra questions you answer must also go into the final vis.
- When using colors with opacity different from zero, check the interactions with the other elements (are they visible?).
- Think whether you need to normalize values.

For the final vis:

- The application must be the final visualization that answers all the questions.
- Do not forget to include a final visualization that answers all the questions.
- Align things, be consistent. You can make use of both vertical and horizontal alignments to facilitate comparisons.
- Ensure charts can be visually compared properly (consider changing scales, palettes...)
- Use space cleverly (put related things together and unrelated things far away).
- Extra questions you answer must also go into the final vis.
- The higher the number of variables (and questions answered) included, the better.
- The application should run with a simple "streamlit run <application_name>" in any computer, everything should be available in the same folder or addressed properly.
- If you do not use all your data in your vis, ensure you have filtered it previously, don't force streamlit to execute with data it is not used. Note that this is a web application running in a browser, which is not the most powerful environment to run an app in.