JBI100 Visualization

(Read this document fully and in detail)

Visualization is much about designing creative solutions to the problem of displaying and interacting with complex data in order to facilitate the user's understanding of it. Most in visualization is learned by practice and applying principles and concepts. One of the most relevant aspects of a visualization design are the interaction strategies. We introduce several of these principles and concepts during the lectures. As practice, we will design and develop a visualization tool throughout the duration of the course. The assignments guide you through the final result, however, they are not the only thing you need to do. You need to complement them for the final result and add what is needed to make it an integrated functional visualization tool at the end and a coherent reporting.

The focus of this project is on the design of visual encodings, interaction and a visualization tool.

This year you can chose one of these data sets:

Airbnb Open Data New York

https://www.kaggle.com/data sets/arianazmoudeh/airbnbopendata

Data Description

Airbnb is an American corporation that offers an online marketplace for renting and booking accommodations owned by individuals, hotels and investors. The data set has multiple aspects that can be explored.

The data set provided here contains listing activity in New York city (US). The following Airbnb activity is included:

- Listings, including full descriptions and average review score
- Reviews, including unique id for each reviewer and detailed comments
- Calendar, including listing id and the price and availability for that day

You can find the data set in the section 'Files' airbnb_data in canvas.

You can access the data through the csv file named "airbnb_open_data.csv". The Jupyter notebook named "airbnb_open_data.ipynb" can be used for an initial exploration of the data.

A dictionary is made available that describes each attribute, we provide the file Airbnb Open Data Dictionary.xlxs. You can also find it here:

https://docs.google.com/spreadsheets/d/1b_dvmyhb_kAJhUmv81rAxl4KcXn0Pymz/edit#gid=1967362979

The Insurance Company (TIC) Benchmark

https://www.kaggle.com/data sets/uciml/caravan-insurance-challenge

Data Description

This data set is supplied by the Dutch data mining company Sentient Machine Research and once used in the data mining competition CoIL Challenge 2000. This data set contains information on customers of an insurance company. Although it is well exploited in machine learning areas for prediction tasks, our goal is to analyze the data from various aspects which are suitable for a visualization solution. The data set has multiple aspects that can be explored.

You can find the data set in the 'Files' tic_data section on canvas.

- The data: 'tic_data.csv'
- A description of all the columns: 'TicDataDescr.txt', Column1 in the data corresponds to 1 MOSTYPE for example. You should rename your columns to make them meaningful.
- A small python script to load the data in a data frame: 'load_data.py'

You are welcome to enhance these data sets with other data sets from other sources to achieve interesting and meaningful analysis. However, they should be extra and not an alternative to choosing one of the proposed data sets.

The data sets have multiple aspects that can be explored. Our goal is to design a visualization tool to achieve a specific goal/tasks suitable for a visualization solution. This project is basically a design study that applies visualization to a domain. It is expected that you spend the most time on the design of visualization, its solid justification and its implementation. It is VERY important that you have solid justifications for your choices based on what is presented during the lectures (e.g., perception) and to evaluate your work. During the lectures, we also present analysis tools that aid in designing effective visualizations. You are meant to follow the nested model and the What, Why, and How framework by T. Munzner to achieve your final result. There are other models that could be used, but in this course we focus on Munzner's model. Notice that this is a visualization course and hence, the focus should be on the visualization and interaction aspects. So, be careful on not mainly focusing on the cleansing or the preprocessing of the data. This is, of course, part of the process and will be valued, but will not contribute to the main part of your mark.

You are free to choose the platform in which you develop the visualization tool. We provide you with a basic framework in Python with Dash/Plotly (see assignment 2). This is meant to give you an easier start such that you can focus on the content of the course. However, you are free to choose any other programing language that you feel comfortable with and give you the possibility to develop what are you are aiming at (see lecture on GUI implementation).

The complexity of the implementation you do will be considered. If you use very simple visualizations like Bar Charts in, for example, D3, this does not make your project stand-out or be challenging just because you use D3. However, if you develop innovative visualization encodings using D3 or there is extensive linking between the multiple views, that will be considered complex. It will also be highly valued to be creative and generate your own visualization component (i.e., going beyond a combination of existing visualizations) but to pass the course this is not required. In any case, this course is not focused on the implementation but on the visualization design and justification/motivation based on principles and visualization concepts. New visual encodings should also fulfill requirements of having a good justification and motivation.

The different parts of the project that can be applied iteratively:

- Derive and document important aspects of the data that could be of interest to specific types of users. Think about which user do you have in mind when you define the goals and the tasks. Formulate the main goal and a set of tasks that a user might want to perform with the data, and some specific questions where visualization plays a major role. Make sure that at least some of the tasks and questions are complex enough, require interaction and/or multiple linked views, in order to be performed or solved.

- Consider/design various interactive visualization techniques and combinations thereof that support these tasks, and that are suitable to analyze the data for the tasks and users you have defined earlier. Justify your choices, for example, discuss choices of marks and visual channels encodings in relation to the tasks and data to be visualized (as seen in class). Use the lecture material and other well documented sources if needed to justify your choices. Discuss pros and cons of your design choices. Interaction and linked views are very powerful to obtain effective insights. It is expected that linked views are present in the design, for example, you brush in a specific view (select), and this has an effect in another view which shows (highlights) the selection.

- Implement the visual designs and interaction strategies you have chosen and incorporate/integrate them together into your visualization tool. Use the platform of your choice.

- Results/evaluation: Go back to the tasks and questions you formulated and use the application you build to make interesting observations about the data. We would expect to also report on none trivial findings. Document how you came to these observations and how your design or visualization technique was beneficial (or not) to your discoveries. The reason why you think your visualization tool as you design it and justified is confirmed or not by the results. You can also perform user studies, although they are not required.

Deliverables

You use LaTeX to build the report. We provide a report LaTeX template based on the most relevant publication journal in Visualization. We also uploaded a complete paper example based on the same template to show how a visualization paper is structured and how it can be built easily (with the standard LaTeX rules). You do not have to be an expert in LaTeX to write a visualization paper. We encourage you to use https://www.overleaf.com/ to write the report. You should log in with your TUe student account and create your project. At the moment of submission, you should provide the overleaf link, such that we can access it directly.

Interim Report

You will develop the report in two phases: an initial interim report (2 to 4 pages in the mentioned format). For the interim report, you should write a maximum of 2 pages of text with 1 to 2 pages (approximately) filled with figures. In a visualization course, illustrative figures are important.

Please, reference the sources from where you get texts and figures that are not your own. **Failing to do so is considered plagiarism. University takes plagiarism very seriously and the consequences can reach expulsion.**

The interim report should contain:

- **Introduction :** describe the main motivation (domain specific) and data available, the main goal your visualization design and tool will address. What users are you building the visualization for? Is it a presentation, exploration or analysis goal? What domain problem do you want to address with the visualization? Why is it relevant? Why is the data adequate to achieve this goal? Why is visualization the right choice to achieve this goal? Try to be as specific as possible. You start from the domain goal/problem (user perspective), here you also need to motivate why visualization is the right choice in contrast, for example, to automatic statistical methods.
- **Related work (optional):** search for literature and other work related to your solution. Refer to the sources/references and explain clearly what you might do that is different or similar to state-of-the-art. The related work should not be referring to material already given in the course but should be extra and related to the visualization aspects of the goal you have defined. For example, other visualization systems that you found for the same or similar data. You comment on the pros and cons, and what you will be doing differently. Or specific visualization strategies from literature that you adopt or adapt that go beyond the ones that are seen in class. It should always be clear what is the relation between the literature and your work.
- **Data Analysis (What)**: this section is focused on analyzing the data and divided into two subsections.

Domain Data Specification: here you should describe the data that you will use, its main characteristics and how it relates to the domain you are treating. Describe data specifics. How you deal with the data from a conceptual point of view (missing values/ errors), and if you apply any preprocessing to generate derived parameters. Here file formats and specific implementation details are NOT relevant, they belong to the implementation section, if relevant at all. You should mention the more conceptual aspects that will influence your visual design. You do not need to describe each individual feature but rather describe them in a summarized way, according to what relevant information they capture.

Data Abstraction: *What*: after a short description on the data provided from a domain point view, you will present a data abstraction (*What*). You should describe the type of data that you cope with according to Munzner's taxonomy: tabular, network, types of attributes (e.g, quantitative, sequential), spatial, etc. What is the key attribute?

You can combine Data description and abstraction if this facilitates the explanation. However, both components should be there.

- **Task Analysis (Why)** : this section focuses on tasks and is split into two subsections.

  Domain Specific Tasks: in this section, you must analyze the possible visualization tasks that the users would be interested in doing, and you will solve with your visualization design/tool. There might be tasks, specially low level, that are not better solved with visualization, but combined with other tasks the overall becomes a visualization task or goal. Ideally, you present a set of tasks and corresponding questions that are related to the chosen domain and overall goal, that should be clear in the text. Notice that questions and tasks are similar in concept. Questions are often a useful way to clarify the tasks. Present multiple tasks and also at different levels as presented in class. Some of the tasks should have a considerable level of complexity, e.g., involve multiple attributes where complex relations beyond few (2 or 3) attributes need to be made. Notice that tasks do not refer to how the tool should look like but why the users should be using the visualization tool, independently of how it is visualized.

  Task Abstraction: *Why*: once you have defined a set of domain specific tasks, you must generalize them to be domain independent to be able to use the principles from visualization for the visual and interaction design. The tasks once abstracted do not contain domain language anymore but general data analysis visualization description.

- **Current solution (How –**you can give a name to your tool and use it as section heading**):** In the interim report this is in development so it is not expected that this section is completely finished. Describe the current status of your visualization design, different elements, components and justify them. A Mock-up made on paper or a drawing tool is a good way to present your design. Also add the justification to

the level of possible at this point, i.e., justification from a perceptual point of view, and how your choices are related to design principles into account for the choice of marks and channels. How it fits the abstract tasks and data described in previous sections. You can also describe alternatives that you discarded and indicate why. If needed, provide some support by citing relevant literature. This section should **NOT** be a manual of your tool, but an explanation and justification of your design and what makes it possible. Whether you click the right button of the mouse or something else is not important for this project, but you are able to take the action is what matters.

- **Implementation** describe the language libraries, tools you have used to develop your visualization tool. This should be a rather short section, mentioning language and libraries. You can also describe specifically challenging aspects of the implementation that you had to address. However, this should be a rather small part of the report.

The interim report should not have an abstract.

The interim report counts as 10% of the final mark. Notice that the interim report is also meant for you to get feedback and be able to improve the final report and project.

Final Report

For the final report, you should have a maximum of **6-7 pages A4 size** including imaging material and excluding references. You should write max 4 pages of text with 2 to 3 pages of figures.
You should make clear what is the new text in relation to the interim project. For example marking the text in a different color. We expect that you improve the interim report according to the feedback provided. If you fail to mark what you changed in the final report from the interim report, it will not be evaluated.

The document should contain and extend from the interim report. You can change what was in the interim report (e.g., adding more tasks or changing the tasks completely if the feedback requires it). You need to extend the **current solution** part to your final solution. It should include interaction, coordinated multiple views (linked views are very important for an effective visualization), advance multivariate visualization, and navigation strategies. Incorporate the new elements that you are adding to your tool. You can also provide alternative solutions you considered, such that you can evaluate later on which one actually works better for your goal.

Apart from extending the components that were already present in the interim report, you also need to add the following aspects:

- **Abstract:** summary of the overall work developed and your contribution.
- **Use Cases (results/evaluation):** Go back to the tasks and questions you formulated and use the application you build to make interesting observations about the data, i.e., build use cases. We would expect to also recognize use cases with non-trivial findings. Per use case, document how you came to the observations and how your design or visualization technique was beneficial (or not) to your discoveries. Reason why you think your visualization tool as you design is confirmed or not by the use cases. The findings themselves are relevant to the extend you can show that your visualization has been essential to provide these findings.
- **Conclusion and future work:** finally, you will reflect on what you have achieved. An overall reflection on what has been achieved from what was stated in the introduction as your goal. For instance, this section should discuss whether you managed to do what you initially planned, whether your initial choices worked well or not, things that you discovered that were not correct, etc. What is still open? What would you work on if you would still have time for it? What would you do differently?.

You are all meant to work on all aspects of the project as a team, but also develop the skills individually. It is not ideal, that you divide the work such that, for example, one writes the report and another develops the code. Ideally, you all work on all aspects, and divide equally. You are all responsible for all aspects of the project and should be able to explain them (i.e., design process, visualization design, justification, code, screencast, report, …).

Individual reports: If you did everything together, and there is a good balance in the work which is what we expect, then just state that and you do not need further description. However, if you feel there is a misbalance, then there should be individual reports. It should be a maximum of 300 words document (separate from the LaTeX report) for each of the group members describing what you did individually in the project. It is important that we know that your partner also agrees with it. It should be in big lines and pointing to the report to explain what was done if needed. The individual reports, together with the peer review will be used to balance the marks for the group members having the specific situations under consideration. If no individual reports are provided, we assume there are no differences in the distribution of the work, and we will not consider potential later adaptations.

Please, always reference the sources from where you get texts and figures that are not your own. **Failing to do so is considered plagiarism. University takes plagiarism very seriously and the consequences can reach expulsion.**

Screencast

Since describing an interactive process is not so easy on paper, you should also make a screencast of 3-5 minutes (not longer!) that shows how interaction helps to do some of the tasks. The main goal of the screencast is that you show what you can do with your tool including interaction, it is not meant that you give the explanation you already have in your report. **All members** of the group should participate in the video equally.

There are multiple video editing tools you could use to generate your screencast. Make sure that in the screencast it is clear what you are trying to show for a person who has not done the project with you. Just showing the interaction without any explanation is not really helpful, so use voice-over or at least clear annotations.

The best video will be awarded a prize. Criteria for a good video are:

- Content and Structure
- Presentation Form
- Clarity

Please, have a look at the example video that is shared on Canvas.

Source code

Source code, implementation or final outcome (e.g., D3 code, working website), depending on your project. Make clear what did you implement yourself, and what you got from existing libraries. Make it easy for us to evaluate your work, we cannot evaluate what we do not understand or do not have. Comment the code clearly.

We should be able to run your code and we will try, so please make it easy for us. Provide stepwise explanation on how we can run your code. These aspects are considered in the overall mark.

Please, always reference the sources from where you got code that are not your own. **Failing to do so is considered plagiarism. University takes plagiarism very seriously and the consequences can reach expulsion.**

Report short guidelines

- Do not underestimate the difficulty of technical writing, so reserve enough time for writing the report.
- Be precise. It is not sufficient that you understand what you mean. If the reader cannot understand it, it is usually your fault and not the reader's.
- Use illustrations and screenshots to clarify methods and results.

- Each figure and table should be numbered and accompanied by a caption text that explains what the reader sees in the picture or table.
- Refer to figures and tables in the text by using their numbers, for example, “Figure 1 shows...”, do NOT use text like “ The figure below shows…” . Furthermore, each figure and table must be referenced in the text somewhere.
- Use proper expressions, for example, “don’t” should be written as “do not”, “it’s” as “it is”, and so on. The pronoun that goes with “it” is “its” without an apostrophe.
- Spell check, grammar check, and proof read the document before handing it in. Most readers, in particular examiners, will be irritated by poor spelling and poor grammar.
- Do not use material that you did not write yourself. Copy-and-paste without citation, quotation, or reference, is considered plagiarism.

How we evaluate your work

See the rubric in canvas on how you will be evaluated.

The grades can be tweaked per individual, if the work of the team members differs significantly in quality and/or effort (see individual reports). If there are issues with the collaboration in the group. Please, report this on time. Solving collaboration issues at the beginning of the course is less painful than at the end.