in4086 Data Visualization InfoVis Project

(Read this document fully and in detail)

Visualization is much about designing creative solutions to the problem of displaying complex data in order to facilitate the user's understanding of it. Most in visualization is learned by practice and applying principles and concepts. We have introduced several of this principles and concepts during the lectures.
The focus of this project is on the design of visual encodings, interaction and exploratory visual analysis tools for information-visualization (InfoVis) problems (see first lecture). It is VERY important that you have solid justifications of 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 to design effective visualizations (What, Why, and How by T. Munzner)

You have quite some freedom to choose your InfoVis project. Not all analysis questions are adequate for a visualization project. Tor example, if you want to know the mean of a value, the best way to achieve it, it is just calculating it visualization is not a component for it. The choice of an adequate project (data/question) for visualization will also be evaluated.

There are two main types of projects:

1) **Variant 1 (DS):** A design study, applying visualization to a domain or domains of your choice using existing tools. This modality requires not much programing and corresponds to the type of work a data analyst would do in practice. Expect to spend most time on the design of visualization and its solid justification. Also data acquisition, augmenting from other resources, cleaning data, and aggregating data will be important but should not be the only focus.

2) **Variant 2 (TD):** A new visualization design or technique which involves programming. It corresponds to the type of work a visualization expert would do in practice. You will spend most of the time on building an interactive prototype, in this case the dataset and tasks might be relatively simple. This includes the possibility of suggesting a modification to an existing visualization technique, if it enables a novel view on a data set. A pure technical study would provide a complete new visualization concept and method (e.g., tree maps are a method) . However, your project does not need to be a completely new visualization concept, an adaptation is possible.

Your project might be a combination of variant 1 and variant 2. Implementing with D3 does not make the project automatically a TD project. It depends on the complexity of the implementation and the visualization methods used. If you use very simple visualizations like Bar Charts in for example D3, this does not make it a full technical project. You have to find a balance between the complexity of the analysis of what you implement and the data set analysis that you use.

Steps

- Choose data (see the end of this document for starting points. YOU CAN SEARCH A DATA SET FROM DAY ONE)
    - **Variant 1** the data has to have a certain level of complexity to need aggregation/clustering. Look for additional sources of data/information to enrich the dataset (use at least one, more is better). Originality is also relevant. Preprocessing will probably be unavoidable, so be prepared to do data cleansing, curation, filtering, etc.
    - **Variant 2** complexity of the data is less relevant for this variant. However the data set still should be complex enough to justify visualization you propose and the methods that you use for the visualization.
- Derive and document important aspects of the data that could be of interest to an analyst. Formulate a set of tasks that an analyst might want to perform with the data, and some specific questions where visualization can play a major role. For example, develop the analysis based on T. Munzner's method shown in class. Make sure that at least some of the tasks and questions require interaction and/or multiple linked views in order to be performed or solved. **Variant 1** should also show more complexity in this part than **variant 2**.
- Consider/design various interactive visualization techniques and combinations thereof (as linked views) that support these tasks, and that are suitable to analyze this data. Discuss marks and visual channels encodings (as seen in class). Discuss pros and cons.
    - **Variant 1** Choose an existing information visualization tool, for example the free GGobi (http://www.ggobi.org/), a scientific prototype like iVisDesigner (https://donghaoren.org/ivisdesigner/), or request a demo license from a commercial package such as Tableau (http://www.tableausoftware.com) or IBM Analytics (http://www.ibm.com/analytics/). Your choice should be based on the considerations that you made in the previous step.
    - **Variant 2** Build an interactive application in any language that you prefer to visualize the data with the techniques chosen, using D3 would be a good option (see below). You should have several techniques in your approach that are linked interactively and are shown simultaneously. One single image and view in the data with no interaction will not be considered positively.

- Results/evaluation: Go back to the tasks and questions you formulated and use the application you build to make interesting observations about the data. This part is especially relevant for **variant 1**. In **variant 1**, we would expect less trivial findings than in **variant 2** also as a result of using a more complex data set. Document how you came to these observations and how your design or visualization technique was beneficial (or not) to your discoveries. Explain why you think your approach is justified based on the lectures, for example, from a perceptual point of view, and how it takes design principles into account. If needed, provide some support by citing relevant literature.

Work is done in triplets (Create your groups in Brightspace)

Deliverables (submit in blackboard)

Deadline **December 22nd 23:59 UTC/GMT**

- **Report** a maximum of **8 pages A4 size** including imaging material where you document: your project problem, and solution, justification of your choices and present results and evaluation of your work. You can use the steps mentioned above as a basis to guide the report. The pages should have similar font size and margins as in this project description document. The text has to be to the point. So 4 well written clear pages are better than 8 pages with the extra redundant content. Do not forget to include images that demonstrate your approach.
- **Source code**, implementation or final outcome (e.g., D3 code, working website, tableau files), depending on your project. If you submit code 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.
- **Individual reports:** there should be a maximum of 300 words document for each of the group members, with extra images when necessary, 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. If you did everything together and there is no clear division of tasks then just say that. You can divide the tasks but all of you should be able to explain all aspects of the project, specially the visualization design and justification.
- **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.
  There are a number of tools available to capture either your whole screen or a part of it:

  - Linux: Simple Screen Recorder, http://www.maartenbaert.be/simplescreenrecorder/. It does unfortunately not provide video editing facilities.
  - Windows/OS X: Camtasia, http://www.techsmith.com/camtasia.html . A fully functional trial version can be used for 30 days.
  - OS X has built in screen casting capabilities using Quicktime, via file->new screen recording.

There are multiple video editing tools you should be able to 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.

Report short guidelines (based on Michel Westenberg 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.
- The report should not read like a manual the justification of what you did is very important so do not forget to document it. This is mentioned clearly in the steps shown above.

How we evaluate your work

We will evaluate your work based on:

- Problem definition and originality: Evaluate how adequate the problem chosen is to be solved with visualization. How well are defined the questions to be answered by the visualization, and how complex they are. Also the originality of what have been chosen to solve.
- Reasoning /justification of visualization design: Data/task abstraction and how well the choices made are justified. This is a very important aspect for all projects.
- Effectiveness of the visualization design (how well it works): evaluation of the result themselves. How are the results documented. How well it works and how effective the visualizations are to answer the original questions. Are there findings thanks to the visualizations?
- Technical ingenuity and contribution on the visualization (implementation): This is especially interesting for the TD type. How well the technical challenge of the project is solved. How "difficult" was the actual implementation of the ideas. How good is the code.
- The quality of your deliverables including all deliverables (report, screencast, source code, etc). How good is the report written. How clean and well commented is the code. How well done is the screencast.

Overall: the final mark will be a mix of the points above and the relevance of each point will depend on whether the project is rather DS or TD type. The overall effort will be considered here, too. The grades can be made individual, if the work of the team members differs significantly in quality and/or effort.

D3 and Assistance

The JavaScript library D3.js is a powerful library to generate your own tailored visualizations. It has a wealth of possibilities. Therefore, we consider that it is important that people who are working on Data Visualization have knowledge of its existence and are aware of its potential.

During the first two practical sessions you will get an introduction to D3 including exercises. These are meant to give you a good start in the use of D3. The exercises will NOT be evaluated, they are provided for you to learn. We recommend you to do the exercises to decide whether you want to use D3 in your InfoVis project, and such that you have an easier start.

It is NOT mandatory to do your InfoVis project in D3. If you decide to not use D3 in your project, you basically have learned something related to Data Visualization that can be useful for you in the future if you work with data visualization.

The **practical sessions (Thursdays)** until the **7th of December** are meant for you to also do your InfoVis project. Assisstance will be present most of the time.

If you have questions related to the project that you did not manage to get answered during the practical sessions, send an email to: datavis-ewi@graphics.tudelft.nl

Example datasets

The web is brimming over with juicy datasets. Here are some examples to get you started, but please do use google and your own network to find more. Pick something that you are passionate about!

- The Guardian Data Blog is chock full of examples and datasets in easily editable formats.
- Datasets listing of Berkeley's post-graduate visualization course.
- Google Public Data Explorer has a growing list of data sets.
- https://github.com/jdorfman/awesome-json-datasets datasets in json format.
- https://perso.telecom-paristech.fr/eagan/class/igr204/datasets collection of data sets in .csv format.
- http://www.vispubdata.org/site/vispubdata/ publication information in most important publication venues in visualization.
- http://hcil2.cs.umd.edu/newvarepository/benchmarks.php benchmark data sets in InfoVis.
- http://opendatanederland.org Dutch open data.
- https://data.overheid.nl open data from the Dutch government.
- 30 places to find Open Data on the web, http://blog.visual.ly/data-sources/.
- http://datacatalog.worldbank.org worldbank data indicators.