IV Group Project: Multiview Visualisation

This assignment will be done in groups of 5 students. You can organise yourselves into groups, or wait for me to assign you. (You might note that some in the class are on an MSc IT programme, others on MSC CS, or Data etc. With this in mind, you might choose your teams so that all members are from the same cohort, but you don't have to). **By noon on January 31st**, one person in each self-made group should email the names and matriculation numbers of their group members to *matthew.chalmers@glasgow.ac.uk*. After that week, I will assign the remaining students into groups. Do not send in a group of names without asking the other members! I will post the teams on Moodle, and remind students to check that they are in the correct groups.

Your project can be implemented using either Python/Altair, or JavaScript/Vega-Lite, as you wish. You can mix this, with one or two systems using Altair, and one or two systems using Vega-Lite. Don't worry too much if you don't have in-depth knowledge of at least one of these languages: the project involves several aspects of work, ranging from preparation and analysis of data, design choice/justification, evaluation, and write-up, so you can make a good contribution to a group, in many ways.

This coursework is worth 30% of the IV course mark. It has two major parts: design and implementation (20%), and evaluation (10%), as described below.

The deadline for submission is **16:30, Wednesday 19th March 2025.**

The Data Set

Choose a data set that interests you, and that suits the instructions below. Repositories such as the UCI Machine Learning Repository (http://archive.ics.uci.edu/ml/index.php) and Kaggle (https://www.kaggle.com/datasets) provide a large range of data sets, but you may choose a publicly available data set from somewhere else, if you like.

No minimum size of data set is specified, but you will find that (for some aspects of this assignment) it will be difficult to demonstrate the extent of your understanding of the concepts if you choose a small or trivial data set, where small refers to both the number of data items, and the dimensionality of the data.

You are therefore advised to read the whole of the assignment specification carefully from beginning to end, before choosing your data set.

A. Design and Implementation (20%)

1. **The data** (0.1). Give a one-sentence title and description of your chosen data set, with a link to the source data. Also describe and categorise the data set, using the concepts and the terminology introduced in Lecture 1a. Use examples as necessary. Be both specific and comprehensive. *[The output of this part must be text comprising not more than 400 words.]*

2. **The tasks** (0.1). Identify the actions that users might want to do when exploring this data set, and then use the concepts and the terminology introduced in Lecture 1b to describe them in more detail. Note that one of these actions must be the selection of a subset of the data. (This is also a prerequisite for (4), below.) Use examples as necessary. Be clear, unambiguous and comprehensive *[The output of this part must be text comprising not more than 400 words.]*

3. **The core systems** (0.2). Implement three different visualisation systems—call them A, B and C. Each system should allow users to perform **all** the tasks of (2). In addition, each system should be a multi-view composition of at least two views, supporting what the *Visualisation Pipeline* lecture calls *brushing and linking.* Note that the instruction is to implement three different visualisation *systems;* that is, each must be a clearly separate integrated system that can be used by other people, and that will allow them to perform these tasks. *[The output of this part must be three clearly labelled zipped folders, A, B & C, each containing program code for one system.]*

4. **Generalised selection** (0.2). This is a 'stretch goal', extending the core system functionality of (3) by implementing an advanced technique. As per the lecture on Data Selection, determine a suitable *semantic structure* for your data, that describes the data at different levels of abstraction, and a *traversal policy* for that semantic structure. Extend each of your systems from (3), to allow the user to interactively do generalised selection of your data, using this policy. *[The output of this part should be a textual description of your semantic structure, traversal policy, and your implementation approach. This text should not exceed 400 words. Note: extended code should be in the same folders mentioned in (3), i.e., you should replace the code for (3).]*

5. **Demo videos.** *(penalty applied if not submitted)* Produce a video, demonstrating all three systems and explaining the basics of their design and implementation. The maximum length of this video is 5 minutes. *[The output of this part must be a clearly labelled YouTube link, specified at the top of your document.]*

6. **Design comparison** (0.4). Compare the design of the systems, by describing **six** different Information Visualisation design decisions you made. That is, choose six design decisions where you made one choice for A, a different choice for B, and another choice for C. Remember that you would have made many more than simply six design decisions, but you are only required to discuss six. Our advice is to choose ones that help you demonstrate the extent of your understanding of Information Visualisation. For each decision you should:
   - clearly state what the design decision is (for example, a system design that supports the first task declared in section 2, earlier);
   - state which implementation choice you made for this design decision for A, which choice you made for B, and which for C (for example, a bar chart for A, a scatterplot for B, and a stacked bar chart for C… with each supporting the first task you declared in section 2);
   - clearly state which choice you think is best for this design decision, and explain why it is the best – remembering that it may be a choice that is not present in all your systems;
   - describe alternative designs you could have chosen, but did not use (for example, a scatterplot matrix);
   - support your discussion with clearly labelled and captioned diagrams and/or screenshots. *[The output of this part will be text comprising not more than 1200 words (max 200 words for each design decision), with associated relevant diagrams and figures]*

B. Evaluation (10%)

7. **User evaluation comparison.** (0.8) Compare the systems by describing the results of a user evaluation of each of them, where each system should be evaluated by at least five people[2]. That is, for each system you must collect and present data from *at least* five people. You can find your own friends/family/etc. to be evaluation participants. However, in case you want me to find you such participants, I will assign each group one other student group as potential

2

participants. You can then ask that group to help, as evaluation participants. If that group does not help after being asked, let me know. They will get a penalty applied to their marks.

As part of this evaluation comparison:

- describe and justify your evaluation methodology (that is, what you asked the participants to do, what data you collected, and why);
- describe the data you collected, and how you got it;
- explain and justify the way(s) that you analysed the data.

Your comparison should clearly identify which aspects of each system were revealed to be 'best', as a result of the comparative evaluation, remembering that it might be the case that A, B and C may each be 'best' in different ways. For example, system A might be best for one task, but system B might be best for another task. Another example: system A might let people work most quickly, but another system was the one that they liked the best. These differences may or may not be related to the decisions you discussed in section 6 above.

[The output of this part will be text comprising not more than 1000 words, with associated relevant diagrams and figures.

You must also include all your raw evaluation data, from both systems and for all participants in an Appendix. The Appendix does not contribute towards the word count.]

8. **Future work.** (0.2) Describe, in detail, the changes you would make to both systems if you were to improve them, *based on the result of your evaluations*. Importantly: you must not actually make these changes – the code you submit must be exactly the same as the versions you evaluated. Be specific and comprehensive. *[The output of this part will be text comprising not more than 400 words.]*

Submission

You must submit a PDF document that includes all the text for parts 1–8 above, with each part clearly numbered and labelled, followed by a reference list. Note that explanatory figures can also be added, to make the text clearer. The references (and any text in figures) will not be included in the word count. References must conform to the ACM referencing style. At the top of the document (not the middle, not the bottom), you must include YouTube links for your demo videos for your three systems. Your document must have an Appendix that contains all your raw user evaluation data, clearly labelled. Finally, include an appendix that lists each team member and details what they contributed to the project. This could be specific pieces of code, sections of documents written, organisation, planning etc. (This will be used for the *deltas* used to calculate individual student marks – see the Teamwork section at the end of this document.)

You must also submit the three zipped folders containing your program code, separate and clearly labelled, as mentioned under points (3) and (4).

Only one member of the group should submit your group's work to Moodle.

Penalties

Penalties are necessary to ensure that all students take the submission instructions for this assignment seriously. Two-band penalties will be applied to your final grade if:

- any of your word counts are significantly[3] longer than the amounts specified;
- your demo video is significantly longer than the duration specified;
- failure to submit your demo video;
- failure to submit any of your program code folders;
- failure to include an appendix containing your evaluation data, or failure to have five sets of user data for each of the systems;
- failure to include a log of each team member's contribution
- your references do not conform to ACM style;
- your document does not clearly follow the numbered and labelled structure of tasks specified above.

These penalties are cumulative – this means that a maximum of fourteen bands can be lost! *So: take care, not risks.*

Assessment Criteria

Design and implementation (20% of the IV course mark):

1. The data (0.1): Correctness and completeness of the categorisation of the elements of your data set. Submissions that correctly and appropriately use examples and demonstrate significant understanding of IV concepts are more likely to get higher marks.
2. The tasks (0.1): Validity and extent of the tasks identified. Marks are likely to be lost if your description of the tasks are ambiguous. Submissions that correctly and appropriately use examples and demonstrate significant understanding of IV concepts are more likely to get higher marks.
3. The systems (0.2): The systems will be assessed based on both utility (especially with regard to the tasks of (2)), and elegance of design/implementation/functionality.
4. Generalised selection (0.2): In the same way as (3), the systems will be assessed based on both utility (especially with regard to the tasks of (2)), and elegance of design/implementation/functionality
5. Demo videos (0.0): Required, to assist with other aspects of the marking (especially 3 and 4), but not adding to the mark.
6. Design comparison (0.4): For each of six decisions, clarity in the definition of the design decision and its alternatives, and justification for choice. Submissions that demonstrate a rational process for making the choice (with use of appropriate figures as necessary) and demonstrate significant understanding of IV concepts are more likely to get higher marks

Evaluation (10%):

7. User evaluation comparison (0.8): Completeness of the description of the evaluation process, the data collected and its analysis. Appropriateness of the evaluation process chosen. Clarity of your identification of 'best', the rationale of the choice, and the extent to which it clearly relates to your actual data will be taken into account when determining your mark
8. Future work (0.2): The extent to which the future work suggestions relate to the results of your evaluation. Submissions that make appropriate reference to IV concepts and the literature are more likely to get higher marks.

[3] Note that 'significantly' is deliberately not quantitatively defined here: don't take chances, stick to the limits.

Advice

Read this project specification through, beginning to end, several times before starting the assignment.

- Start small. Do not be too ambitious to start off with. Start with a small (sub)set of data and a small set of tasks first, and then grow the list and make the scope of your projects more complex as you get more confident, keeping an eye on the amount of time you have left before submission. More complex systems are likely to give more scope for demonstrating your understanding of Information Visualisation.
- Use good software engineering approaches. For example, the three systems may share a lot of their core modelling and infrastructural code, assuming they handle the same data and conform to the same general design pattern.
- Start early. Submit early if you can.
- Give good attention to the design decisions. If you run out of time, it will be better to complete all six parts partially, rather than only complete some of the parts.
- Don't exceed the word limits. You will be penalised if you do. Use bullet points as necessary – this is not an essay, it is a report, so adding structural elements that make your document easier to navigate is acceptable.
- Use the headings and the numbered structure provided above to make sure that you cover all aspects of the assignment.
- Aim to submit your report to Moodle at least 30 minutes before the deadline, so that you are not affected by any technical problems. **No last-minute excuses will be accepted if you have not taken this advice.**
- Do not lose track of the importance of the report by focussing too much on the implementation.

Teamwork

Teamwork is an important part of software development: in industry, large software development tasks are constantly being undertaken in teams that can vary in size from two to several hundred. In any team it is recognised that people will contribute in different ways. It is important to ensure that you are always aware of your role and that you have an opportunity to make a meaningful contribution to the project at all times.

However, for some projects it is the case that some people contribute more than others, and with this in mind **we will be using *deltas* as a way of adjusting the team mark**, in order to arrive at an individual's mark for the team-based components of the course. A delta typically adjusts the team mark up or down, by a number of bands, for a given individual according to their contribution. The computation of these deltas will be informed by the percentage scores that each member of the team will provide, which gives a numerical estimate of the proportion of the overall effort undertaken by each person (including themselves), backed up by each individual's description of their contribution.

Example: team T has four members, namely W, X, Y and Z. Each person states that W and X contributed 25%, whilst Y contributed 30% and Z contributed 20%. The overall team mark was B1. The deltas for W, X, Y and Z are 0, 0, +1 and -1 respectively. Thus, W and X each obtain B1, Y obtains A5, and Z obtains B2. Note: this is just for illustrative purposes, and does not reflect any particular rule that might be used to construct deltas from points scores in practice.

Your submission should contain a breakdown of each member's contribution, as explained above.

More details about this delta process, including deadlines and where to submit deltas, will be announced closer to the deadline.