University of BRISTOL

Visual Analytics: Validation

Ian Nabney

ian.nabney@bristol.ac.uk

bristol.ac.uk

University of BRISTOL

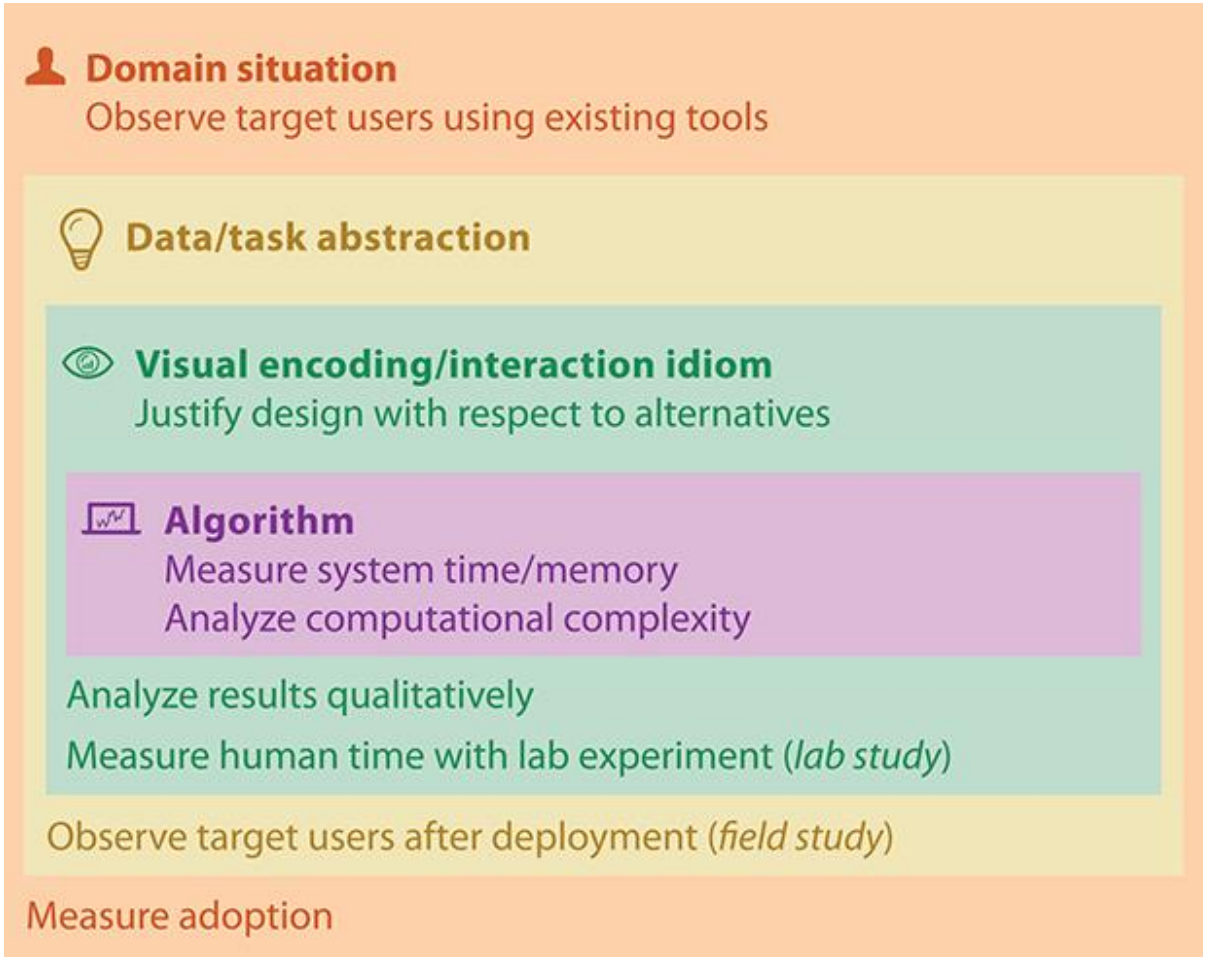
Overview

- Reading: Chapter 4 of Munzner (e-book that can be downloaded from UoB library website)
- Explain why validation is important for visual analytics
- Define the four levels of design and validation
- Able to define and use validation techniques
- Case study

Why validate?

- Validation is important because
  - there is no ‘best solution’ known in advance
  - the vis design space is huge, and most designs are ineffective.
- It’s valuable to think about how to validate your choices from the very beginning of the design process, rather than leaving these considerations for the end as an afterthought.
- Design in validation
- This lecture introduces two more levels of design to consider, one above the **why–what** abstraction level and one below the **how** idiom level.
- This unit focuses on the two middle levels, but considering all four is helpful when thinking about how to validate whether a given design has succeeded.

Four levels of design



- **Situation** level concerned with application domain
- Next is **what-why** abstraction from specific domain
- Third is **how** - is the design of idioms that specify the approach to visual encoding and interaction
- Last level is the design of **algorithms** to instantiate those idioms computationally.

Domain situation

- Group of target users, their domain of interest, their questions, and their data
- Usually has its own vocabulary for describing data and problems, and existing workflow
- The public making medical decisions about their healthcare in the presence of uncertainty (sound familiar)
- Methods used by designers to identify domain situation blocks include interviews, observations, or careful research about target users
- Eliciting system requirements is not easy, even when you have unfettered access to target users fluent in the vocabulary
- of the domain and immersed in its workflow
- Asking users to simply introspect about their actions and needs is notoriously insufficient: what users say they do when reflecting on their past behaviour gives you an incomplete picture compared with what they actually do if you observe them.

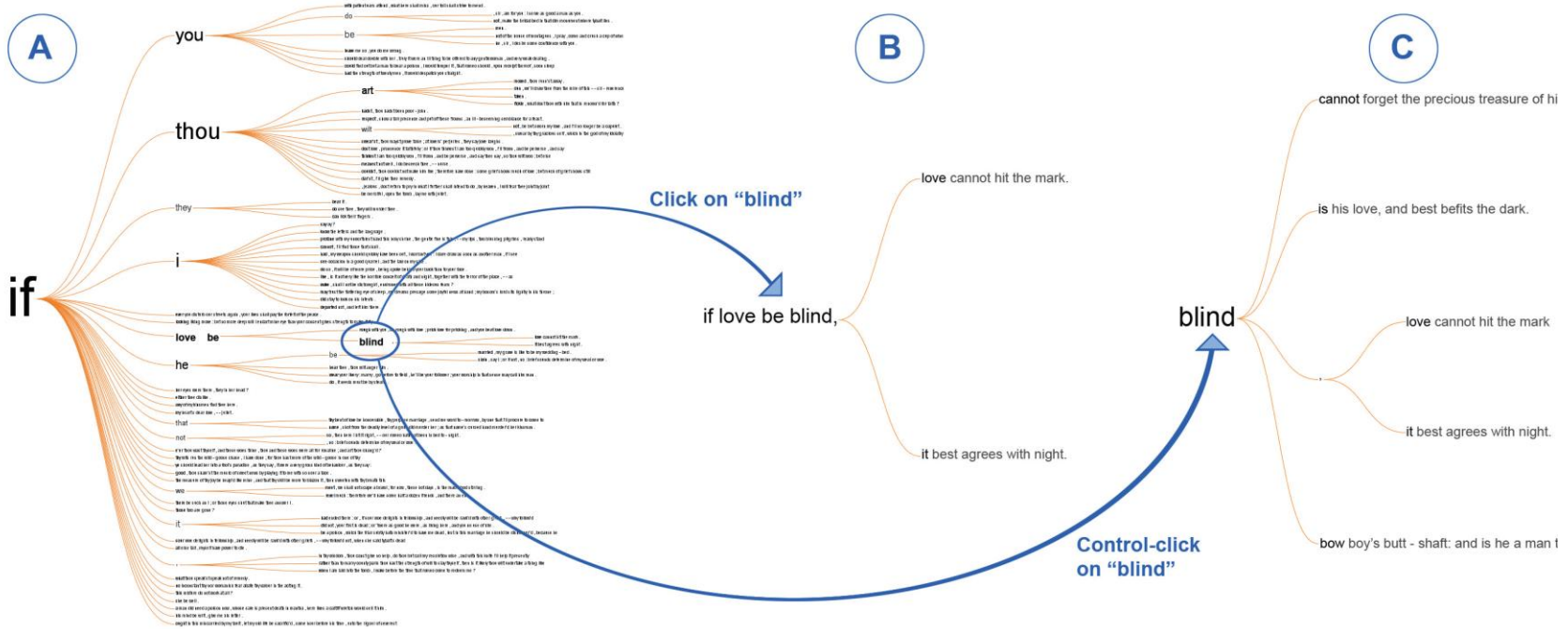
Task and data abstraction

- Abstract to a generic description
- Saw how this works for identifying tasks
- Selecting an abstract data block is a creative design step rather than simply an act of identification
- Your goal is to determine which data type would support a visual representation that addresses the user's problem
- Many early web vis papers posited that solving the "lost in hyperspace" problem should be done by showing the searcher a visual representation of the topological structure of the web's hyperlink connectivity graph
- In fact, people do not need an internal mental representation of this extremely complex structure to find a page of interest
- Thus, no matter how cleverly the information was visually encoded at the idiom design level, the resulting vis tools all incurred additional cognitive load for the user rather than reducing it

Visual encoding and interaction idiom

- How to create and manipulate the visual representation of the abstract data block guided by the abstract tasks
  - Visual encoding idiom defines the picture that users see
  - Interaction idiom defines how users change what they see: dynamic manipulation of representation – move away from print
- Sometimes possible to separate these two decisions, but often they are so intertwined that they are combined
- Should make these decisions on the basis of understanding human abilities in visual perception, cognition and memory (see later lecture)
- Sometimes useful to provide user choice over the idiom, if it can be done without overwhelming them

Word Tree



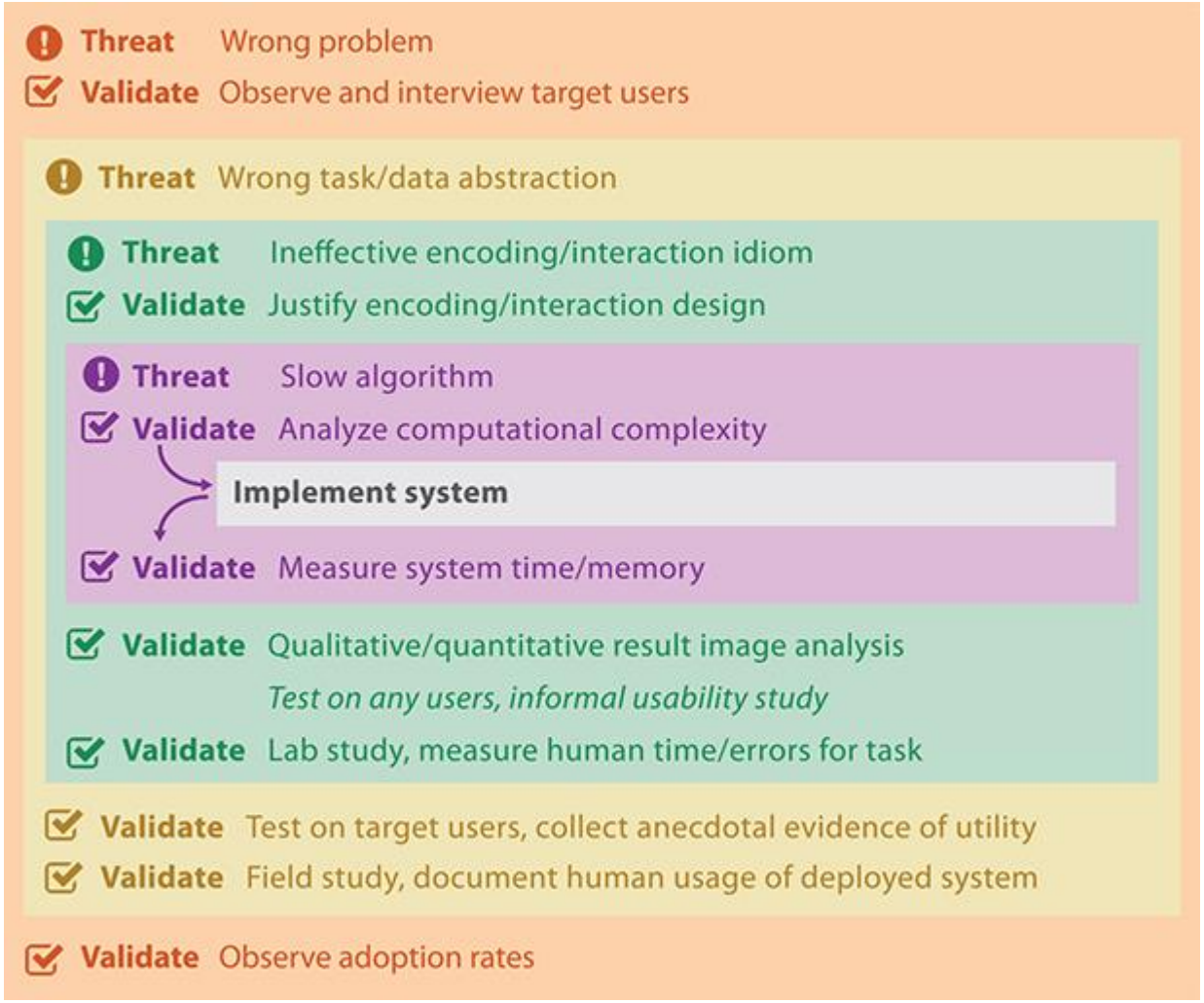
- Visual encoding idiom of a hierarchical tree representation of keywords laid out horizontally, preserving information about the context of their use within the original text
- Interaction idiom of navigation based on keyword selection.

Algorithm

- The innermost level involves all of the design choices involved in choosing or creating an algorithm
- The goal is to efficiently handle the visual encoding and interaction idioms that you chose in the previous level
- In this course we will primarily use existing algorithms
- For example, one visual encoding idiom for creating images from a three-dimensional field of measurements, such as MRI scans, is direct volume rendering.
- Many different algorithms have been proposed: including ray casting, splatting, and texture mapping.
- Evaluate based on measures such as the speed of the computation, how much computer memory is required, and whether the resulting image is an exact match with the specified visual encoding idiom or just an approximation

Threats to validity

Validating against threats



Distinguishes between pre- and post-implementation approaches

Munzner uses the terms ‘immediate’ and ‘downstream’

Domain validation

- Primary threat is that the problem is mischaracterized: the target users do not have these problems.
- An immediate form of validation is to interview and observe the target audience to verify the characterization. A common approach for this case is a **field study**, where the investigator observes how people act in real-world settings.
- The researcher observes users working in their real-world context and interrupts to ask questions when clarification is needed, is better suited than silent observation because of the complex cognitive tasks that are targeted.
- One downstream form of validation is to measure the **adoption rate**. Of course, adoption rates do not tell the whole story: many well-designed tools fail to be adopted, and some poorly designed tools win in the marketplace.
- Nevertheless, this reports what the target users do of their own accord.

Abstraction validation

- Threat is that the task/data abstraction blocks do not solve the users' problems
- Validation against this threat must have users testing on their own work, rather than an abstract task specified by the designers
- A common downstream form of validation is to have a user try the tool, to collect anecdotal evidence that the tool is useful
- A more rigorous validation approach to conduct a field study to observe and document how the target audience uses the deployed system as part of their real-world workflow.
- Observe how their behaviour changes after deployment of a vis tool, as opposed to documenting their existing work practices

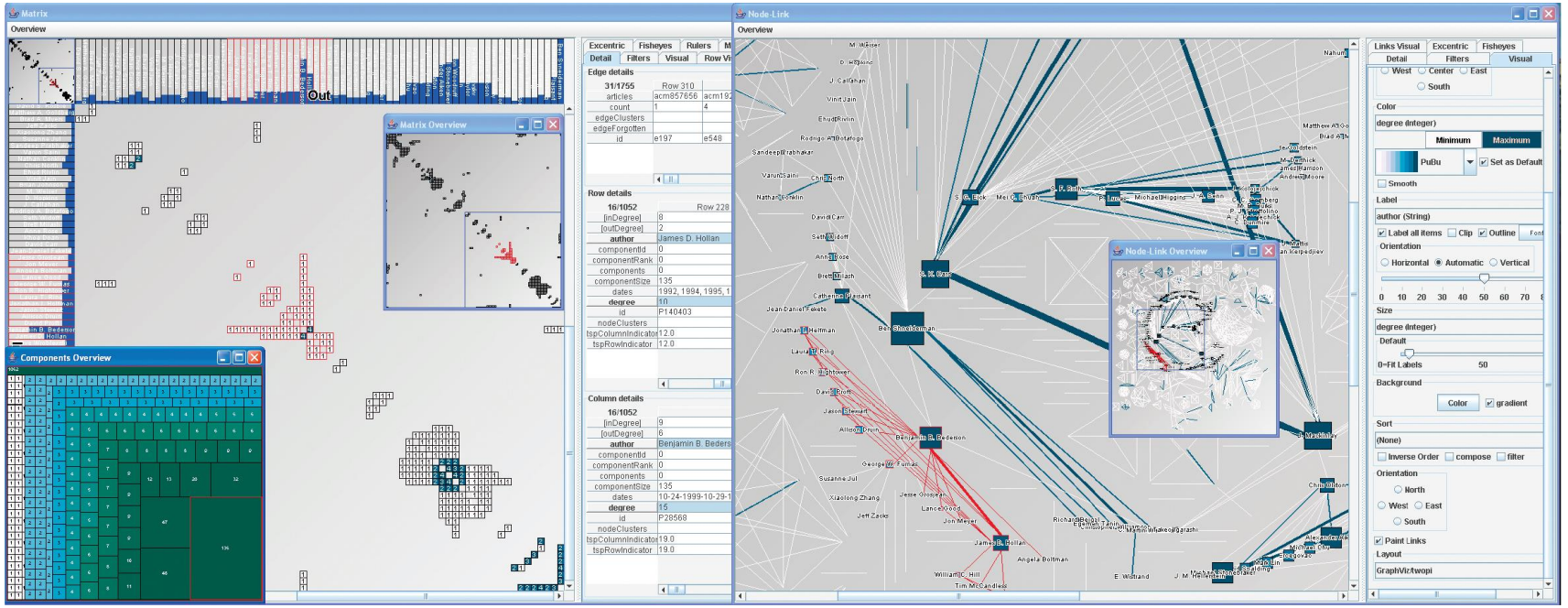
Idiom validation

- Threat is that the chosen idioms are not effective at communicating the desired abstraction to users
- Immediate validation approach is to carefully justify the design of the idiom with respect to known perceptual and cognitive principles
- Downstream validation approach is to carry out a lab study: a controlled experiment in a laboratory setting.
- Appropriate for understand the impact of specific idiom design choices by measuring human performance on abstract tasks
- Often collect objective measurements of time spent and errors made by the study participants
- Other kinds of quantitative data that are sometimes gathered include logging actions such as mouse moves and clicks, tracking the eye movements of the participants
- Qualitative data gathering often includes asking participants to reflect about their strategies through questionnaires.

Algorithm validation

- Threat is that the algorithm is suboptimal in terms of time or memory performance
- Immediate form of validation is to analyze the computational complexity of the algorithm
- Downstream form of validation is to measure the wall-clock time and memory performance of the implemented algorithm. The primary consideration is typically scalability in terms of how dataset size affects algorithm speed (from the point of view of a vis researcher)

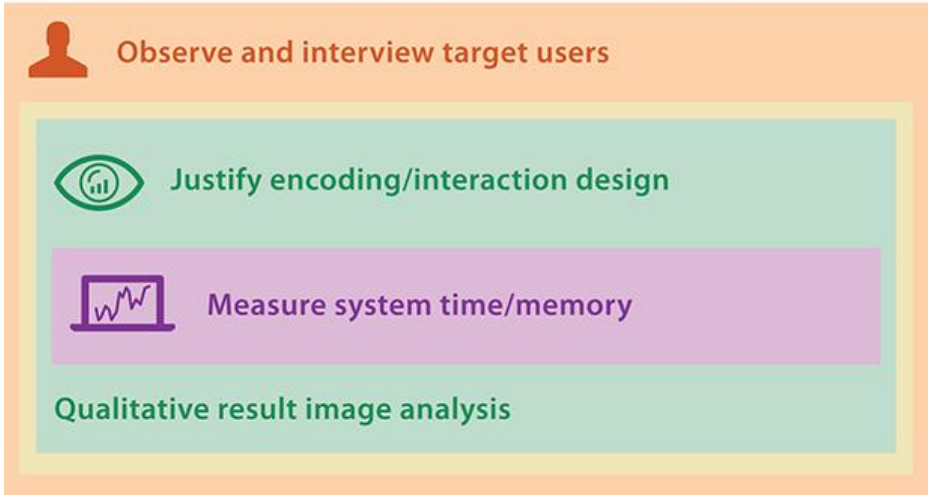
Case study: MatrixExplorer



- Designed for sociologists and historians to explore social networks
- Uses both matrix representations to minimize clutter for large and dense graphs and the more intuitive node–link representations of graphs for smaller networks

N. Henry and J-D Fekete, "MatrixExplorer: a Dual-Representation System to Explore Social Networks", IEEE T on Computer Graphics and Visualization, 13:6, 2007, 1302-1309

Validation



All four levels of the model are addressed, with validation at three of the levels

- Explicit characterization of the social network analysis **domain**, validated by interviews and an exploratory study using participatory design methods with researchers who use social network data
- Detailed list of requirements of the user needs in terms of abstract tasks and data
- Idiom design decision to use both node–link and matrix views to show the data, and also of many secondary encoding issues.
- Downstream validation of this level using qualitative discussion of result images.
- At the algorithm level, the focus is on the reordering algorithm. Downstream benchmark timings are mentioned very briefly.

University of BRISTOL

Summary

- Explain why validation is important for visual analytics
- Define the four levels of design and validation
- Able to define and use validation techniques
- Case study