Information Visualization

Intro

Tamar Munzner
Department of Computer Science
University of British Columbia
13 September 2015

http://www.cs.ubc.ca/~tmm/courses/547-15

Audience
• no prerequisites
– many areas helpful but not required
– human-computer interaction, computer graphics, cognitive psychology, graphic design, algorithms, machine learning, statistics
• open to non-CS people
– if no programming background, can do analysis or survey project
– open to advanced undergrads
– talk to me
• open to informal auditors
– some or all days of readings/discussion, as you like
– you get out of it what you put into it...

Waitlist
• currently 40 registered and 16 on waitlist
– wow!
• don’t panic, people are still shopping around for classes
• highly likely that all who want to take can be accommodated
– without sprinkling extra chairs each time :-)
• make sure to record your name on signup sheet today
– with probability of attendance including real vs audit
– update at end of class today and start of class
• structure plans thus slightly tentative
– might tweak depending on final enrollment

Marking
• 50% Project
– 2% Pitches
– 10% Proposal
– 6% Status Updates
– 12% Final Presentation
– 20% Final Exam
– 25% Participation
– 25% Readings

Class time
• week I
– I lecture
• weeks 2-9: Participation [30%]
– before class, you do chapter-paper, write questions/comments
– during class I lecture briefly, we discuss, in-class design exercises...
• week 2: 1
– guest lectures (Rabani, Kastor, Pax Brinner)
• week 8
– in-class exam (or if conferences)
• weeks 10-13: Presentations [20%]
– before one of the classes: you each read paper on topic of your choice
– during class you present it to everybody else (~10 min)

Course Goals
• twofold goal
– get you to do something useful
– to use existing tools
• feedback through detailed written comments on writing and presenting
– both content and style
– at level of paper review for your final project
– goal: within a week or so

Participation [30%]
• written questions on reading in advance (18% of total mark)
– due 1:30pm (30 min before class)
– 3 total, at least 1 for each reading
– bring printout or laptop with you, springboard for discussion
• discussion/participation in class (12% of total mark)
– attendance expected
– tell me in advance if you’ll miss class (and why)
– question credit still possible if submitted in advance
– tell when you recover if you were ill

Presentations [20%]
• last several weeks of class
• present, analyze, and critique one paper
– send me topic choices by Nov 21, will assign papers accordingly
• expectations
– slides required
– summary/description important, but also your own thoughts
– analysis according to book framework
– critique of strengths and weaknesses
– timing
– exact time TBD depending on enrollment
– likely around 10 minutes each
• topics at http://www.cs.ubc.ca/~tmm/courses/infovis/presentations.html

Find me
• email is the best way to reach me: tmm@cs.ubc.ca
• office hours Tue right after class (3:30-4:30pm)
• by appointment
• X661 (X-Wing of ICICS/CS bldg)
• course page is font of all information
– don’t forget to refresh frequent updates
– http://www.cs.ubc.ca/~tmm/courses/547-15

Chapters/Topics
– What’s Vis and Why Do It?
– Maps and Choropleth
– What Data Abstraction
– Rules of Thumb
– Analytic Four Levels for Validiation
– Arrage Tables
– Arrage Spatial Data
– Arrage Networks
– Map Color and Other Channels
– Multiple Views
– Reduce Items and Attributes
– Analyze Case Studies

Readings
• textbook

– library has multiple ebook copies
– to buy yourself, cheapest is amazon.com

– papers
– listed posted on course page
– if OL links, use library EZproxy from off campus
– readings posted by one week before class
– usually one chapter + one paper per class session

Projects
• solo, or group of 2, or group of 3
– groups highly encouraged; amount of work commensurate with group size

Projects: Design Studies
• BYOD (Bring Your Own Data)
– you have your own data to analyze
– your thesis/research topic (very common case)

Projects
• programming
– common case
– I will only consider supervising students who do programming projects
– three types
– problem-driven design studies (target specific task)
– technique-driven (explore design choice space for encoding or interaction idiom)
– algorithm implementation (as described in previous paper)

– analysis
– use existing tools on dataset
– detailed domain survey
– particularly suitable for non-CS students

– survey
– very detailed domain survey
– particularly suitable for non-CS students

Marking
• 50% Project
– 2% Pitches
– 10% Proposal
– 6% Status Updates
– 12% Final Presentation
– 20% Final Exam
– 25% Participation
– 25% Presentations

– 75% Content: Summary 50%, Analysis 25%, Critique 25%
– question credit still possible if submitted in advance

– tell me in advance if you’ll miss class (and why)
– question credit still possible if submitted in advance
– tell when you recover if you were ill

Projects
• technique/algorithm
• data visualization (problem-driven)
• systems
• evaluation
• model/theory

– programming
– common case
– I will only consider supervising students who do programming projects

– analyze
– use existing tools on dataset
– detailed domain survey
– particularly suitable for non-CS students

– survey
– very detailed domain survey
– particularly suitable for non-CS students

• reading assignment
– many existing datasets, see resource page to get started
– http://www.cs.ubc.ca/group/infovis/resources.shtml

– technical/descriptive analysis
– critique of strengths and weaknesses

– feedback through detailed written comments on writing and presenting
– both content and style
– at level of paper review for your final project
– goal: within a week or so

– fast marking for reading questions
– great/good/poor/zero
– tell me in advance if you’ll miss class (and why)
– question credit still possible if submitted in advance

– survey
– very detailed domain survey
– particularly suitable for non-CS students

Course Work
• textbook

– library has multiple ebook copies
– to buy yourself, cheapest is amazon.com

– papers
– listed posted on course page
– if OL links, use library EZproxy from off campus
– readings posted by one week before class
– usually one chapter + one paper per class session

Marking
• no prerequisites
– many areas helpful but not required
– human-computer interaction, computer graphics, cognitive psychology, graphic design, algorithms, machine learning, statistics
• open to non-CS people
– if no programming background, can do analysis or survey project
– open to advanced undergrads
– talk to me
• open to informal auditors
– some or all days of readings/discussion, as you like
– you get out of it what you put into it...

Waitlist
• currently 40 registered and 16 on waitlist
– wow!
• don’t panic, people are still shopping around for classes
• highly likely that all who want to take can be accommodated
– without sprinkling extra chairs each time :-)
• make sure to record your name on signup sheet today
– with probability of attendance including real vs audit
– update at end of class today and start of class

• structure plans thus slightly tentative
– might tweak depending on final enrollment

Marking
• 50% Project
– 2% Pitches
– 10% Proposal
– 6% Status Updates
– 12% Final Presentation
– 20% Final Exam
– 25% Participation
– 25% Presentations

– 75% Content: Summary 50%, Analysis 25%, Critique 25%
– question credit still possible if submitted in advance
**Topics Preview**

**Defining visualization (vis)**

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

**Why?...**

**Why have a human in the loop?**

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

- Vision is not reliable when there is a need to augment human capabilities rather than replace people with computational decision-making methods.
- don't need vis when fully automatic solution exists and is trusted
- many analysis problems ill-specified
- don't know exactly what questions to ask in advance
- possibilities
  - long-term use for end users (e.g. exploratory analysis of scientific data)
  - presentation of known results
  - stepping stone to better understanding of requirements before developing models
- help developers of automatic solution refine/debug, determine parameters
- help end users of automatic solutions verify/build trust

**Why show the data in detail?**

- summaries lose information
- confirm expected and find unexpected patterns
- assess validity of statistical models
- development of automatic solution refinement, determine parameters
- what-why-how analysis framework as scaffold to think systematically

**Why focus on tasks and effectiveness?**

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

- tasks serve as constraint on design (as does data)
- vis designers must take into account three very different kinds of resource limitations: those of computers, of humans, and of displays.

**Resource limitations**

Vis designers must take into account three very different kinds of resource limitations:

- computational limits
  - processing time
  - system memory
- human limits
  - human attention and memory
- display limits
  - pixels are precious resource, the most constrained resource
  - information density: ratio of space used to encode info vs unused whitespace
  - traded between clutter and wasted space
- data abstraction
  - what is shown?
  - what is it shown?
  - why is the user looking at it?
  - task vs abstraction

**Why use an external representation?**

Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.

- external representation: replace cognition with perception

**Idiom design space**

The design space of possible Vis idioms is huge, and includes the considerations of both how to create and how to interact with visual representations.

- **idiom**: distinct approach to creating or manipulating visual representation
  - how to draw in visual encoding idioms
  - many possibilities for how to create
  - how to manipulate: interaction idioms
    - even more possibilities
  - visual idioms: single idiom dynamic
  - link multiple idioms together through interaction

**Why depend on vision?**

- human visual system is high-bandwidth channel to brain
  - overview possible due to background processing
  - subjective experiences of seeing everything simultaneously
  - sound: lower bandwidth and different semantics
  - overview not supported
  - subjective experiences of sequential streams
  - touch/haptics: impoverished record/replay capacity
  - only very low-bandwidth communication thus far
  - taste, smell, no viable record/replay devices

**Why have a computer in the loop?**

- beyond human patience: scale to large datasets, support interactivity

**Why have a computer in the loop?**

- beyond human patience: scale to large datasets, support interactivity

**Encoding**

- **encode**: distinct approach to creating or manipulating visual representation
  - how to draw in visual encoding idioms
  - many possibilities for how to create
  - how to manipulate: interaction idioms
  - even more possibilities
  - visual idioms: single idiom dynamic
  - link multiple idioms together through interaction

**Mark types**

- **Marks**: point, line, area
- **channels**: position, color, size, angle, curvature, density

**Anscorn's Quartet**

- x mean 9
- x variance 10
- y mean 8
- y variance 4
- x/y correlation 1

**Vis designers must take into account three very different kinds of resource limitations: those of computers, of humans, and of displays.**
**Facet**
- Juxtapose
- Partition
- Superimpose

**Juxtapose and coordinate views**
- Share Encoding: Same/Different
- Linked highlighting
- Share Data: All/Subset/None
- Share Navigation

**Reduce items and attributes**
- Reduce/increase: inverses
- Filter
  - pro: straightforward and intuitive
  - con: out of sight, out of mind
- Aggregation
  - pro: inform about whole set
  - con: difficult to avoid losing signal
- Not mutually exclusive
  - combine filter, aggregate
  - combine reduce, change, facet

**Embed: Focus+Context**
- Combine information within single view
- Elide
  - selectively filter and aggregate
- Superimpose layer
  - local lens
- Distortion design choices
  - region shape: radial, rectilinear, complex
  - how many regions: one, many
  - region extent: local, global
  - interaction metaphor

**Next Time**
- to read
  - Book: Marks and Channels (Ch 5)
  - Paper: Polaris
- Academic paper, Tableau is the spinoff company
- Guest lecture by Robert Kosara on Tableau