Information Visualization

Intro, *Time Series Exercise*

Tamara Munzner
Department of Computer Science
University of British Columbia

10 September 2019

http://www.cs.ubc.ca/~tmm/courses/547-19
Visualization (vis) defined & motivated

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

Visualization is suitable when there is a need to augment human capabilities rather than replace people with computational decision-making methods.

• human in the loop needs the details
  – doesn't know exactly what questions to ask in advance
  – longterm exploratory analysis
  • speed up through human-in-the-loop visual data analysis
  – presentation of known results
  – stepping stone towards automation: refining, trustbuilding

• intended task, measurable definitions of effectiveness
Logistics
Finding me

• email is the best way to reach me: tmm@cs.ubc.ca
• office hours Tue right after class (5-6pm)
  – or by appointment
  – unlikely to catch me by dropping by, usually either in meeting or elsewhere
• 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/547-19
Audience

• no formal prerequisites
  – many areas helpful but not required
    • human-computer interaction (HCI), eg CPSC 544 this term or equivalent
    • computer graphics, cognitive psychology, machine learning, statistics, algorithms, <application domain>...
    – programming skills required for most project types

• open to non-CS people
  – if no programming background, can do analysis or survey project

• communication skills in English important for success
  – substantial reading, writing, discussion, presentations

• need strength in *at least* one of these 3: programming, English, HCI
  – **unsuccessful combination:** weak ESL, weak programming, no HCI background

• open to informal auditors
  – some or all days of readings/discussion/exercises, you’ll get out of it what you put into it...
Intros

• say your full name, program, year
• also sign up on paper sheet so I see who’s here vs who’s registered
Schedule, big picture

• once/week, 2-5pm Tuesdays, 12 sessions
• Sep 3, *no class: no CS grad classes, orientation events only*
• Sep 10, first class: today!
• Oct 22, no class: annual VIS conference
• Dec 3, last class: one week past usual time
• Dec 10, final presentations: afternoon, exact time TBD (1-5pm?)
• Dec 13, final reports due
Marking

• 50% Project
  – 15% Intermediate Milestones (pass/fail)
    – extensive feedback along the way
    – but formative not summative
      – goal: help you make projects the best they can be!
  – 15% Final Presentation
  – 20% Final Report
  – 50% Content

• 20% Presentations (*probably, depending on final enrolment*)
  – 75% Content: Summary 50%, Analysis 25%, Critique 25%
  – 25% Delivery: Presentation Style 50%, Slide Quality 50%

• 30% Participation
  – 60% Written Comments
  – 25% In-Class Work/Exercises (pass/fail)
  – 15% Discussion

• marking by buckets
  – great 100%
  – good 89%
  – ok 78%
  – poor 67%
  – zero 0%
Class sessions

• first part: read & participate [30%]
  – before class:
    • you do readings (~4, mix of chapters & papers)
    • you submit comments before class
    • you respond to at least two comments from classmates
  – during class:
    • sometimes I lecture (briefly) and we discuss
    • frequent in-class work/exercises/critique

• highly likely: presentations [20%]
  – before one of the classes: you read paper I assign on topic of your choice
  – during that class: you present it to everybody else (~10-15 min)
  – TBD depending on final enrollment
Readings

• textbook
    • http://www.cs.ubc.ca/~tmm/vadbook/
    – library has multiple free ebook copies
    – to buy yourself, cheapest is amazon.com
      • hardcover bundled with ebook

• papers
  – links posted on course page
  – if DL links, use library EZproxy from off campus

• readings posted by 6 days before class
• ~4 each session: mix of chapters & papers
Comments submission & marking

• written comments on reading in advance, in two rounds
  • round 1 due 10am (4 hrs before class), 90% of comment mark
    – 1 for each reading
    – bring printout or laptop with you, springboard for discussion
    – post to Canvas discussion group
  • round 2 due 1:30pm (30 min before class), 10% of comment mark
    – written responses to at least 2 comments per session/week
    – you can only read comments from others after you post your own
• start as pass/fail marking, see how it goes
  – switch to explicit marking if quality concerns
Comments content

• comments or questions
• fine to be less formal than written report
  – correct grammar and spelling still expected
  – be concise: one paragraph is good
• should be thoughtful, show you’ve read and reflected
  – poor to ask something trivial to look up
  – ok to ask for clarification of genuinely confusing section
  – good to show that you’re thinking carefully about what you read
  – great to point out something that I haven’t seen before
• examples on http://www.cs.ubc.ca/~tmm/courses/infovis/structure.html
Class participation

• in-class group/individual exercises
• workshopping/critique for projects
• crucial part of course, attendance expected
  – tell me in advance if you’ll miss class (and why)
  – tell me when you recover if you were ill
  – (written comments credit still possible if submitted in advance)
Projects [50%]

• groups of 2, 3, or 4
  – amount of work commensurate with group size
  – permission for solo project granted in exceptional circumstances, by petition

• stages
  – milestones along the way, mix of written & in-class
    • new last few years: formative feedback only
    • pitches (data/task), proposals, peer project reviews
  – final versions
    • final presentations (oral): Tue Dec 10, afternoon
      – whole dept invited, refreshments served
    • final reports (written): Fri Dec 13, 11:59pm
    • summative written feedback for both

• resources
  – more on datasets and tools later
Projects

• programming
  – common case (I will only consider supervising students who do these)
  – four types
    • problem-driven design studies (target specific task/data)
    • technique-driven (explore design choice space for encoding or interaction idiom)
    • algorithm implementation (as described in previous paper)
    • interactive explainer (like distill articles)

• 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
Projects: Design studies

• BYOD (Bring Your Own Data)
  – you (or your teammates) have your own data to analyze
    • thesis/research topic
    • personal interest
    • dovetail with another course (sometimes works, but timing may be tricky)

• FDOI (Find Data Of Interest)
  – many existing datasets, see resource page to get started
    • http://www.cs.ubc.ca/group/infovis/resources.shtml
  – can be tricky to determine reasonable task
Project examples

• http://www.cs.ubc.ca/~tmm/courses/547-17F/projectdesc.html#examp
Presentations [20%]

• probably - *depends on final enrollment!* TBD
• present, analyze, and critique one paper
  – send me topic choices, I 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 times TBD depending on enrollment
  – likely around 15 +/- 5 minutes each
• topics at [http://www.cs.ubc.ca/~tmm/courses/infovis/presentations.html](http://www.cs.ubc.ca/~tmm/courses/infovis/presentations.html)
Enrollment

• don't worry if you're not registered yet, just attend and keep up
  – major churn is normal the first few weeks
  – spaces will definitely open up
• do make sure you've signed up on paper sheet going around!
• registered students currently have fob access to FSC 2330 door
  – will update after final enrollment settles (after Sep 17)
  – you can work in this room when not otherwise in use
  • other classes: Tue/Thu 9:30-11am, Tue 5-8pm, Fri 1-4pm
  • DFP Seminar: 2nd & 4th Wed 12-1pm
Break
Now: In-class design exercise, in small groups

• Five time-series scenarios
  – A: every 5 min, duration 1 year, 1 thing: building occupancy rates
  – B: every 5 min, 1 year, 2 things: currency values (exchange rate)
  – C: several years and several things: every 5 min, 5 years, 10 currencies
  – D: many things: every 5 min, 1 year, CPU load across 1000 machines
  – E: several parameters, many things: every 5 min, 1 year, 10 params on 1000 machines

• Small-group exercise: 15-20 min
  – one group per table (4 people/group)
  – discuss/sketch possible visual encodings appropriate for your assigned scenario

• Reportback: 30-40 min
  – 3 min from each group

• Design space examples/discussion: 20-30 min
Case A: 3D Approach (Not Recommended)

- extruded curves: detailed comparisons impossible

[Cluster and Calendar based Visualization of Time Series Data. van Wijk and van Selow, Proc. InfoVis 99.]
Case A: Cluster-Calendar Solution

- derived data: cluster hierarchy
- juxtapose multiple views: calendar, superimposed 2D curves

[Cluster and Calendar based Visualization of Time Series Data. van Wijk and van Selow, Proc. InfoVis 99.]
Case B: Stack Zooming

Case C: ChronoLenses

Case D: RankExplorer


https://youtu.be/rdgnlqcZ2A4
Case E: LiveRAC video

http://youtu.be/ld0c3H0VSkw

Next Time

• to read
  – VAD book, Ch 1: What's Vis, and Why Do It?
  – VAD book, Ch 2: What: Data Abstraction
  – VAD book, Ch 3: Why: Task Abstraction
  – paper: Design Study Methodology