 Nested model: Four levels of visualization design
- domain situation
  - who are the target users?
- abstraction
  - translate from specifics of domain to vocabulary of visualization
- what is shown? data abstraction
  - why is the user looking at it? task abstraction
    - often must transform data guided by task
- idiom
  - how is it shown?
  - visual encoding: idiom; how to draw
  - interaction: idiom; how to manipulate
- algorithm
  - efficient computation

Domain characterization
- details of an application domain
  - group of users, target domain, their questions, & their data
  - varies wildly by domain
  - must be specific enough to get traction
- domain questions/problems
  - break down into simpler abstract tasks

Task abstraction: Actions and targets
- very high-level pattern
  - {action, target} pairs
  - very high-level choices
  - search
  - find a known/unknown item
  - query
  - find out about characteristics of item

Analysis example: Derive one attribute
- smaller number
  - centrality metric for social networks
  - derived quantitative
  - draw top 5% of 500K for good skeleton

Means and ends
- actions: Search
  - what does user know?
  - target, location
  - look up
  - ex: word in dictionary
  - ex: node in network
  - ex: books in bookstore
  - explore: ex: a cool neighborhood in New York
- consume: 
  - discover vs present
  - aka explore vs explain
  - enjoy
  - newcomer
  - aka casual, social
  - produce
  - annotate, record, derive
  - creative design choice

Actions: Analyze
- consume
  - discover vs present
  - ex: classical split
  - aka explore vs explain
  - enjoy
  - newcomer
  - aka casual, social
  - produce
  - annotate, record, derive
  - creative design choice

Actions: Design Process
- Characterize Domain Situation
- Map Domain-Language Data Description to Data Abstraction
- Identify/Create Suitable Idiom/Technique
- Identify/Create Suitable Algorithm

Example: Find good movies
- one possible choice for data and tasks, in domain language
  - combine audience ratings and critic ratings
  - task: find high-scoring movies for specific genre
  - abstraction?
  - attribute: audience & critic ratings
  - similar to movies I liked?
  - matches specific genre
  - data: (is it available?)
    - yes! data sources IMDB, Rotten Tomatoes...

Domain characterization
- details of an application domain
  - group of users, target domain, their questions, & their data
  - varies wildly by domain
  - must be specific enough to get traction
- domain questions/problems
  - break down into simpler abstract tasks

Task abstraction: Actions and targets
- very high-level pattern
  - {action, target} pairs
  - very high-level choices
  - search
  - find a known/unknown item
  - query
  - find out about characteristics of item

Analysis example: Derive one attribute
- smaller number
  - centrality metric for social networks
  - derived quantitative
  - draw top 5% of 500K for good skeleton

Means and ends
- actions: Search
  - what does user know?
  - target, location
  - look up
  - ex: word in dictionary
  - ex: node in network
  - ex: books in bookstore
  - explore: ex: a cool neighborhood in New York

Actions: Analyze
- consume
  - discover vs present
  - ex: classical split
  - aka explore vs explain
  - enjoy
  - newcomer
  - aka casual, social
  - produce
  - annotate, record, derive
  - creative design choice

Actions: Design Process
- Characterize Domain Situation
- Map Domain-Language Data Description to Data Abstraction
- Identify/Create Suitable Idiom/Technique
- Identify/Create Suitable Algorithm

Example: Find good movies
- one possible choice for data and tasks, in domain language
  - combine audience ratings and critic ratings
  - task: find high-scoring movies for specific genre
  - abstraction?
  - attribute: audience & critic ratings
  - similar to movies I liked?
  - matches specific genre
  - data: (is it available?)
    - yes! data sources IMDB, Rotten Tomatoes...

Domain characterization
- details of an application domain
  - group of users, target domain, their questions, & their data
  - varies wildly by domain
  - must be specific enough to get traction
- domain questions/problems
  - break down into simpler abstract tasks

Task abstraction: Actions and targets
- very high-level pattern
  - {action, target} pairs
  - very high-level choices
  - search
  - find a known/unknown item
  - query
  - find out about characteristics of item

Analysis example: Derive one attribute
- smaller number
  - centrality metric for social networks
  - derived quantitative
  - draw top 5% of 500K for good skeleton

Means and ends
- actions: Search
  - what does user know?
  - target, location
  - look up
  - ex: word in dictionary
  - ex: node in network
  - ex: books in bookstore
  - explore: ex: a cool neighborhood in New York

Actions: Analyze
- consume
  - discover vs present
  - ex: classical split
  - aka explore vs explain
  - enjoy
  - newcomer
  - aka casual, social
  - produce
  - annotate, record, derive
  - creative design choice

Actions: Design Process
- Characterize Domain Situation
- Map Domain-Language Data Description to Data Abstraction
- Identify/Create Suitable Idiom/Technique
- Identify/Create Suitable Algorithm

Example: Find good movies
- one possible choice for data and tasks, in domain language
  - combine audience ratings and critic ratings
  - task: find high-scoring movies for specific genre
  - abstraction?
  - attribute: audience & critic ratings
  - similar to movies I liked?
  - matches specific genre
  - data: (is it available?)
    - yes! data sources IMDB, Rotten Tomatoes...

Domain characterization
- details of an application domain
  - group of users, target domain, their questions, & their data
  - varies wildly by domain
  - must be specific enough to get traction
- domain questions/problems
  - break down into simpler abstract tasks

Task abstraction: Actions and targets
- very high-level pattern
  - {action, target} pairs
  - very high-level choices
  - search
  - find a known/unknown item
  - query
  - find out about characteristics of item

Analysis example: Derive one attribute
- smaller number
  - centrality metric for social networks
  - derived quantitative
  - draw top 5% of 500K for good skeleton

Means and ends
- actions: Search
  - what does user know?
  - target, location
  - look up
  - ex: word in dictionary
  - ex: node in network
  - ex: books in bookstore
  - explore: ex: a cool neighborhood in New York

Actions: Analyze
- consume
  - discover vs present
  - ex: classical split
  - aka explore vs explain
  - enjoy
  - newcomer
  - aka casual, social
  - produce
  - annotate, record, derive
  - creative design choice

Actions: Design Process
- Characterize Domain Situation
- Map Domain-Language Data Description to Data Abstraction
- Identify/Create Suitable Idiom/Technique
- Identify/Create Suitable Algorithm
Exercise: Rating Charts for Tasks
- task C: compare pair of attributes (Distributor vs OEM)
 25

Example: Horrified vs stacked bars
- horrified: browse/explore
- stacked bars: locate/lookup
- which is better?
  – ... 2nd-best rated movie of all time
  – Jeopardy call, < 10 seconds to respond!
 17
http://alhadaqa.com/2019/10/horrified/

Exercise: Rating Charts for Tasks
- task F: enjoy / engage
 28

Exercise: Task abstraction in genomics

You have been approached by a geneticists to help with a visualization problem. She has gene expression data (data that measures the activity of the genes) for 30 cancer tissue samples. She is applying an experimental drug to see whether the cancer tissue dies as she hopes, but she finds that only some samples show the desired effect. She believes that the difference between the samples is caused by differential expression of gene activity in a particular pathway. She would like you to
- locate the outlier in the network
- explore the topology

Example: Genomics II

- goal: control data quality for gene splicing data
  – judge magnitude of sample
  – compare samples, identify within-group variance & outliers
  – compare groups, identify between-group variance

Exercise: Task abstraction in genomics

- derive two groups of samples
- ... which genes are likely to cause the difference, and what role they play in that pathway

Exercise: Rating Charts for Tasks
- task A: sort attributes
 23

Example: Economics
- task: compare and derive
- data: derive change
 19
The Economist

Exercise: Rating Charts for Tasks
- task B: compare pair of attributes (Direct vs Distributor)

Exercise: Rating Charts for Tasks
- task D: present trends across all attributes

Exercise: Rating Charts for Tasks
- task E: spot outlier attributes

Exercise: Rating Charts for Tasks
- task G: enjoy / engage

Exercise: Rating Charts for Tasks
- task H: compare pair of attributes (Distributor vs OEM)

Exercise: Rating Charts for Tasks
- task I: compare pair of attributes (Distributor vs OEM)

Exercise: Rating Charts for Tasks
- task J: compare pair of attributes (Distributor vs OEM)

Exercise: Rating Charts for Tasks
- task K: compare pair of attributes (Distributor vs OEM)

Exercise: Rating Charts for Tasks
- task L: compare pair of attributes (Distributor vs OEM)

Exercise: Rating Charts for Tasks
- task M: compare pair of attributes (Distributor vs OEM)

Exercise: Rating Charts for Tasks
- task N: compare pair of attributes (Distributor vs OEM)

Exercise: Rating Charts for Tasks
- task O: compare pair of attributes (Distributor vs OEM)

Exercise: Rating Charts for Tasks
- task P: compare pair of attributes (Distributor vs OEM)

Exercise: Rating Charts for Tasks
- task Q: compare pair of attributes (Distributor vs OEM)

Exercise: Rating Charts for Tasks
- task R: compare pair of attributes (Distributor vs OEM)

Exercise: Rating Charts for Tasks
- task S: compare pair of attributes (Distributor vs OEM)

Exercise: Rating Charts for Tasks
- task T: compare pair of attributes (Distributor vs OEM)

Exercise: Rating Charts for Tasks
- task U: compare pair of attributes (Distributor vs OEM)

Exercise: Rating Charts for Tasks
- task V: compare pair of attributes (Distributor vs OEM)

Exercise: Rating Charts for Tasks
- task W: compare pair of attributes (Distributor vs OEM)

Exercise: Rating Charts for Tasks
- task X: compare pair of attributes (Distributor vs OEM)

Exercise: Rating Charts for Tasks
- task Y: compare pair of attributes (Distributor vs OEM)

Exercise: Rating Charts for Tasks
- task Z: compare pair of attributes (Distributor vs OEM)
Todo this week

- D3 videos to watch this week
  - Making a Bar Chart with D3 and SVG [30 min]
- Quiz 2 to do this week, due by Fri Jan 17, 8am
- Labs start this week!
  - Fri 7-8, 11-12, 4-5
  - TA office hours for individual consultation and help
- TAs will typically alternate weeks
  - if you can't register, try attending the one you want
- Strongly recommended but optional: we do not track attendance
- Seats for registered students first, but may be room
- Foundations Exercise 2 out next time (Thu Jan 16)
  - due Wed Jan 22
- Programming Exercise 1 out next time (Thu Jan 16)
  - due Wed Jan 29

Credits

- Visualization Analysis and Design (Ch 3)
- Alex Lex & Miriah Meyer, http://dataviscourse.net/
- Marti Hearst, exercise (tasks & charts)
  - Teaching as Coaching (VIS 2015 panel on Vis, The Next Generation)