Information Visualization
Aggregate & Filter 1

Tamara Munzner
Department of Computer Science
University of British Columbia

Lect 17, 10 Mar 2020

https://www.cs.ubc.ca/~tmm/courses/436V-20
Upcoming

• Foundations 5: out Thu Mar 12, due Wed Mar 18 11:59pm
• Milestone 2: due Wed Mar 25 11:59pm
  – (with update announce last week, schedule status component)
Correction
Idiom: Small multiples

- encoding: same
- data: none shared
  - different attributes
    different items
    (different condition keys, same gene keys), same attributes: expression values for node colors
  - (same network layout for nodes=genes)
- navigation: shared

System: Cerebral

Reminder
Beyond slides: Textbook for further reading (optional)

- Intro
  - Ch 1. What's Vis, and Why Do It?
- Data Abstraction
  - Ch 2. What: Data Abstraction
  - Ch 4. Analysis: Four Levels for Validation
- Task Abstraction
  - Ch 3. Why: Task Abstraction
- Marks & Channels
  - Ch 5. Marks and Channels
- Multivariate Tables
  - Ch 7. Arrange Tables
- Interactive Views
  - Ch 11. Manipulate View
  - Ch 12. Facet into Multiple Views
- Maps
  - Ch 8. Arrange Spatial Data (only 8.1-8.3)
- Color
  - Ch 10. Map Color and Other Channels
- Networks & Trees
  - Ch 9. Arrange Networks and Trees
- Aggregation
  - Ch 13. Reduce Items and Attributes
  - Ch 14. Embed: Focus+Context
- Rules of Thumb (upcoming)
  - Ch 6. Rules of Thumb

Visualization Analysis & Design, free through library: catalog page  EZProxy direct link
Filter & Aggregate
Exercise: Too much stuff

• Cars dataset: 7 attributes
  – MPG quantitative
  – Cylinders ordinal
  – Horsepower quantitative
  – Weight quantitative
  – Acceleration quantitative
  – Model Year ordinal
  – Origin categorical

• This table has 100 million items

• Pair up, discuss how to have scalable approach, create sketch to illustrate
  – [8 min]
  – Socrative: true when done
How to handle complexity: 1 previous strategy + 3 more

- Derive
  - derive new data to show within view
  - change view over time
  - facet across multiple views
  - reduce items/attributes within single view

### Manipulate
- Change
- Select
- Navigate

### Facet
- Juxtapose
- Partition
- Superimpose

### Reduce
- Filter
- Aggregate
- Embed
How?

Encode

- Arrange
  - Express
  - Separate
- Order
  - Align
- Use

Map

- from categorical and ordered attributes
- Color
  - Hue
  - Saturation
  - Luminance
- Size, Angle, Curvature, ...
- Shape
  - + ● ■ ▲
- Motion
  - Direction, Rate, Frequency, ...

Manipulate

- Change
- Select
- Navigate

Facet

- Juxtapose
- Partition
- Superimpose

Reduce

- Filter
- Aggregate
- Embed

What?

Why?

How?
Reducing Items and Attributes

Filter

- Items

- Attributes

Aggregate

- Items

- Attributes
Reduce items and attributes

• reduce/increase: inverses

• filter
  – pro: straightforward and intuitive
    • to understand and compute
  – 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
Filter

• eliminate some elements
  – either items or attributes

• according to what?
  – any possible function that partitions dataset into two sets
    • attribute values bigger/smaller than x
    • noise/signal

• filters vs queries
  – query: start with nothing, add in elements
  – filters: start with everything, remove elements
  – best approach depends on dataset size
Idiom: **FilmFinder**

- dynamic queries/filters for items
  - tightly coupled interaction and visual encoding idioms, so user can immediately see results of action

Idiom: **cross filtering**

- item filtering
- coordinated views/controls combined
  - all scented histogram bisliders update when any ranges change

System: **Crossfilter**

[http://square.github.io/crossfilter/]
Idiom: **cross filtering**

[https://www.nytimes.com/interactive/2014/upshot/buy-rent-calculator.html?_r=0]
Aggregate

• a group of elements is represented by a smaller number of derived elements

⇒ Aggregate

⇒ Items

⇒ Attributes
**Idiom: histogram**

- static item aggregation
- task: find distribution
- data: table
- derived data
  - new table: keys are bins, values are counts
- bin size crucial
  - pattern can change dramatically depending on discretization
  - opportunity for interaction: control bin size on the fly
Histograms explained

- also great example of scrollytelling!

http://tinlizzie.org/histograms/
Histogram bins

• good # bins hard to predict
  – make it interactive when possible

• rules of thumb
  – # bins = sqrt(n)
  – # bins = log2(n)+1
Idiom: **scented widgets**

- augmented widgets show *information scent*
  - better cues for *information foraging*: show whether value in drilling down further vs looking elsewhere
- concise use of space: histogram on slider


Scented histogram bisliders: detailed

Example: Keshif

- interactive item filtering with scented widgets
- also: interaction speed w/ scatterplot vs list view

https://keshif.me/gallery/olympics
Interactive legends

• controls combining
  – visual representation of static legends w/
  – interaction mechanisms of widgets

• define & control visual display together

Riche 2010
Idiom: boxplot

- static item aggregation
- task: find distribution
- data: table
- derived data
  - 5 quant attrs
    - median: central line
    - lower and upper quartile: boxes
    - lower upper fences: whiskers
      - values beyond which items are outliers
    - outliers beyond fence cutoffs explicitly shown
- scalability
  - unlimited number of items!

[40 years of boxplots. Wickham and Stryjewski. 2012. had.co.nz]
Boxplots

• aka box-and-whisker plots
  – show outliers as points
• bad for non-normal distributions
• really bad for bimodal or multimodal distributions
Boxplots: Drawbacks

• four distributions with same boxplot

http://stat.mq.edu.au/wp-content/uploads/2014/05/Can_the_Box_Plot_be_Improved.pdf
Violin plots

• boxplot + probability density function

https://towardsdatascience.com/violin-plots-explained-fb1d115e023d
Density plots

• aka kernel density plots, kernel density estimation (KDE)
  – smoothed, continuous version of a histogram estimated from data
  – continuous curve (the kernel, usually Gaussian bell curve) drawn at each data point
  – add curves together for single smooth density estimation
    • bandwidth influences estimate

KDE wikipedia

https://towardsdatascience.com/histograms-and-density-plots-in-python-f6bda88f5ac0
KDE in D3: Interactive bandwidth controls

Idiom: **Continuous scatterplot**

- static item aggregation
- data: table
- derived data: table
  - key attrs x,y for pixels
  - quant attr: overplot density
- dense space-filling 2D matrix
- color: sequential categorical hue + ordered luminance colormap
- scalability
  - no limits on overplotting: millions of items

Credits

• Visualization Analysis and Design (Ch 13, 14)
• Alex Lex & Miriah Meyer, http://dataviscourse.net/