Paul Rosen
paul.rosen@utah.edu
@paulrosenphd
https://cspaul.com

SCI
www.sci.utah.edu
THE UNIVERSITY OF UTAH

Visualization for Data Science
DS-4630 / CS-5630

Reducing Data by Filtering

Learning objectives

- Explain the tradeoffs of when to filter, de-emphasize, or aggregate
- Predict what a set of filters returns when combined with AND or OR
- Explain how brushing and linking filters coordinated views
- Recognize when filtering manufactures a finding, and guard against it

why reduce?

- Too many data items and/or too many attributes to focus on what is important in the data

Filter vs. de-emphasize vs. aggregate

- **Filter:** remove items; simple, but context is gone
- **De-emphasize:** grey out the rest (focus+context)
- **Aggregate:** keep everything, summarized



Reducing Items and Attributes

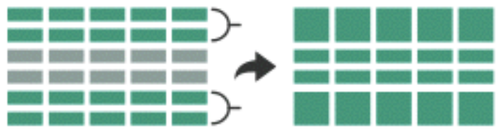
- **Filter**
  - Items



  - Attributes



- **Aggregate**
  - Items

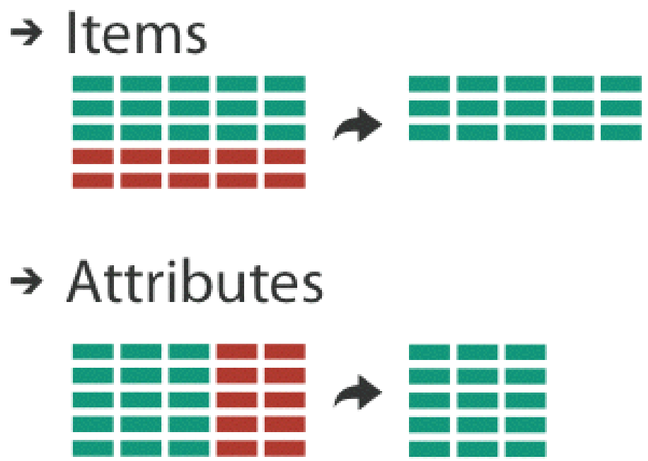


  - Attributes

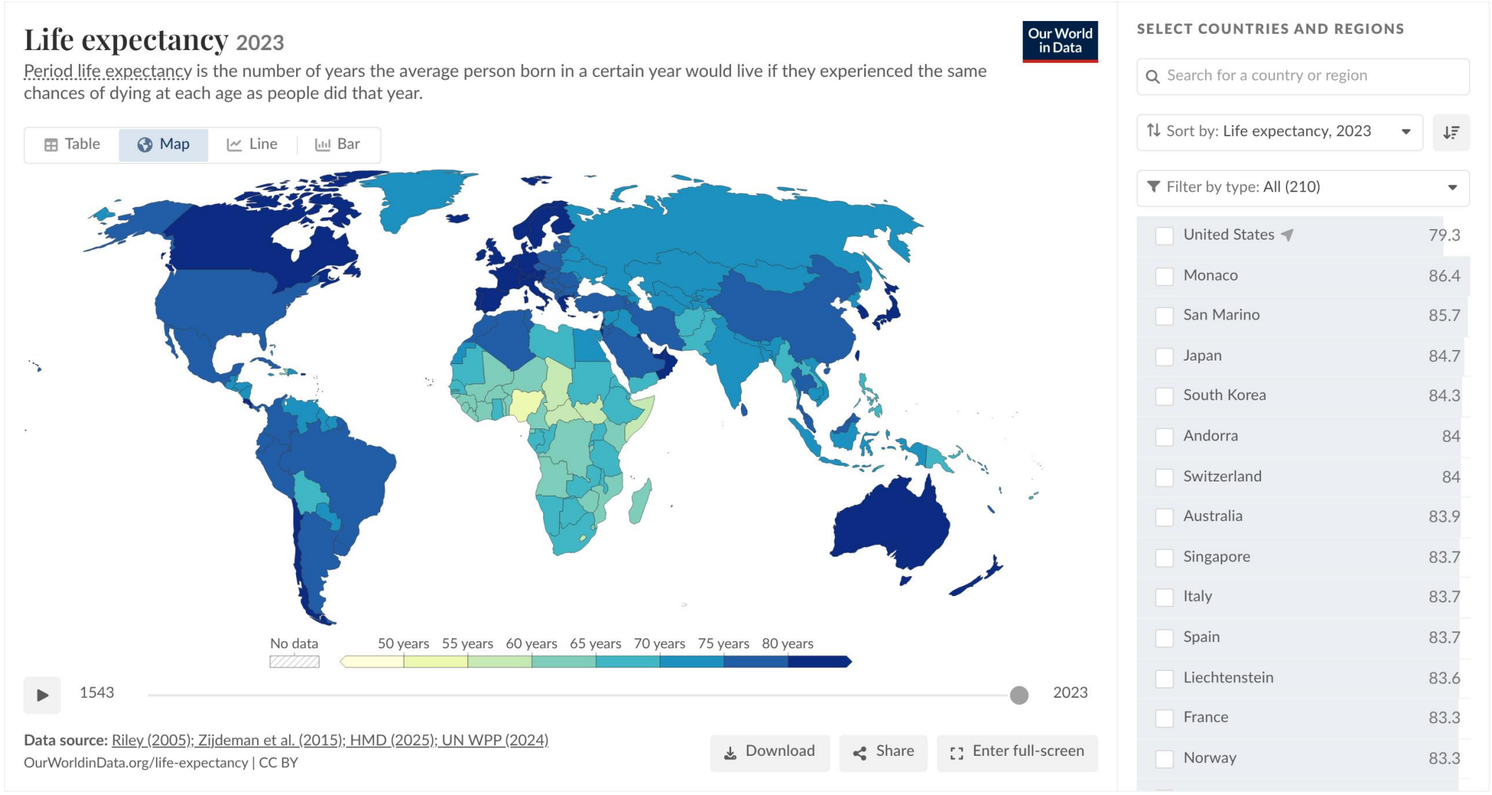


Filter

- Elements are eliminated to support dynamic queries
- Couple encoding and interaction, so that the user can immediately see the results of an action



Filtering today: Our World in Data



Search box
text filter

Filter by type
categorical

Checkbox list
pick items; sorted, so also top-k

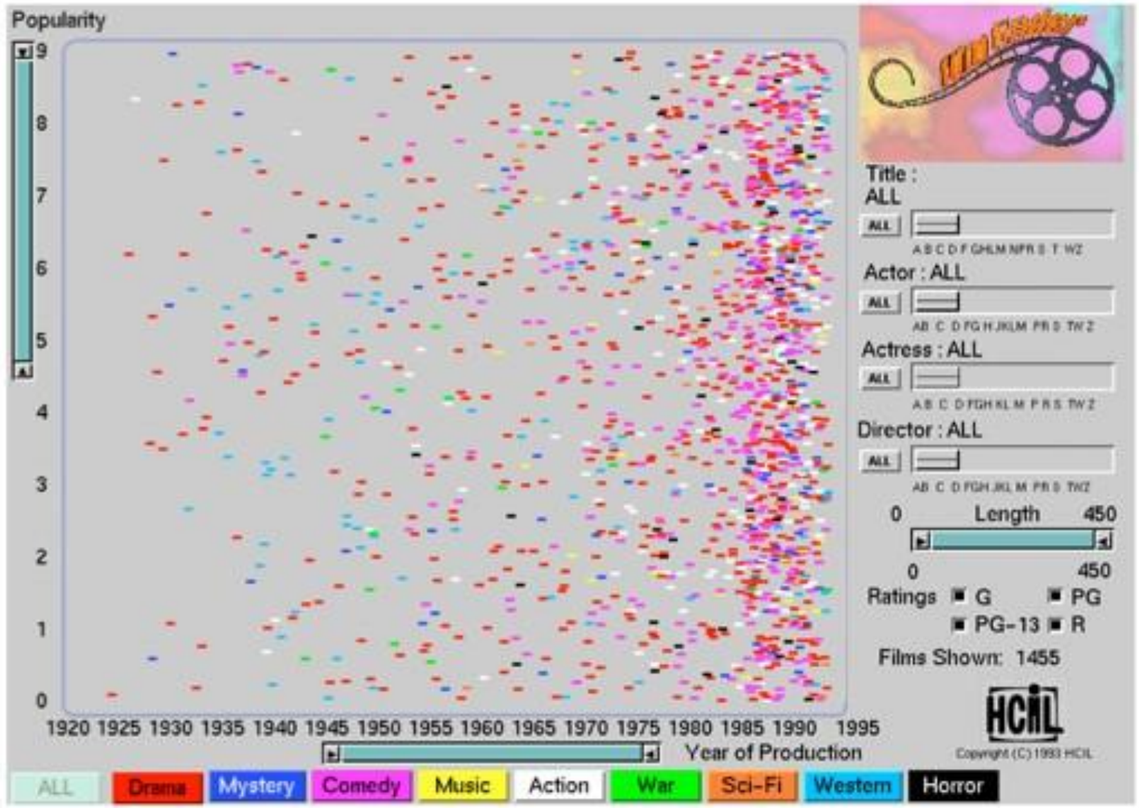
Time slider
range filter on year

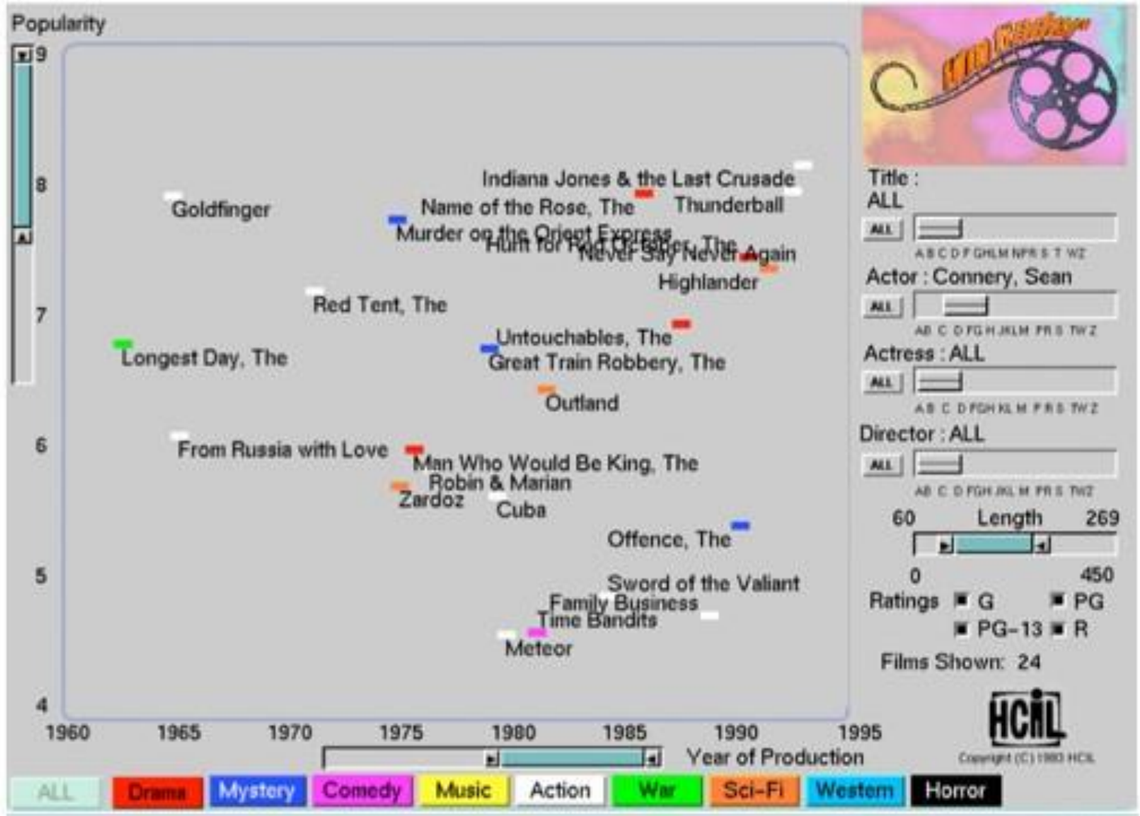
Map / Line / Bar
same filter across linked views

Kinds of item filters

- **Range:** sliders on quantitative attributes (year, price, rating)
- **Categorical:** checkboxes, toggles, clicking legend entries
- **Text search:** match names or keywords
- **Top-k and threshold:** "the 10 largest", "only $p < 0.05$"
- **Derived attributes:** filter on something computed, e.g., "rated above its genre's median"
- **Visual selection:** brush a region of a plot or map

Combined Filters





Dynamic queries: combining filters

Title	Year	Genre
Jaws	1975	Thriller
Raiders of the Lost Ark	1981	Action
Ghostbusters	1984	Comedy
Die Hard	1988	Action
Groundhog Day	1993	Comedy
Speed	1994	Action
Titanic	1997	Drama
The Matrix	1999	Action

- Two filters:
  - Year slider: 1980 – 1995
  - Genre: Action
- How do they combine?

Dynamic queries: combining filters

Title	Year	Genre
Jaws	1975	Thriller
Raiders of the Lost Ark	**1981**	**Action**
Ghostbusters	1984	Comedy
Die Hard	**1988**	**Action**
Groundhog Day	1993	Comedy
Speed	**1994**	**Action**
Titanic	1997	Drama
The Matrix	1999	Action

- **AND** (conjunctive)
  - Year in [1980, 1995] AND Genre = Action
- 3 of 8 movies pass
- Each added filter can only shrink the result

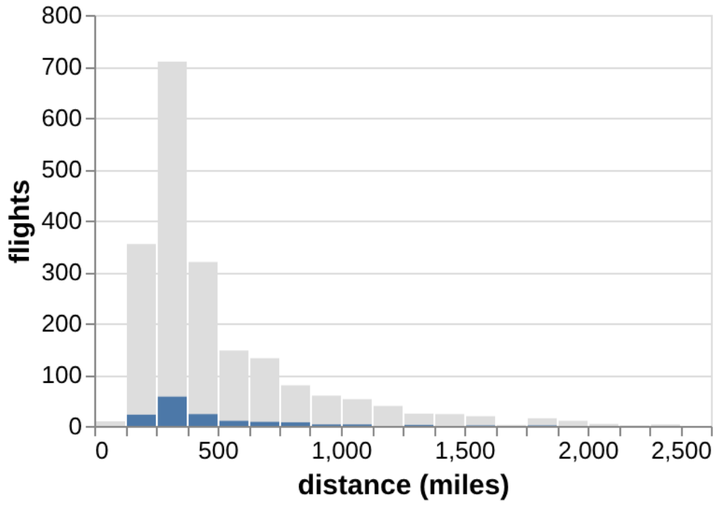
Dynamic queries: combining filters

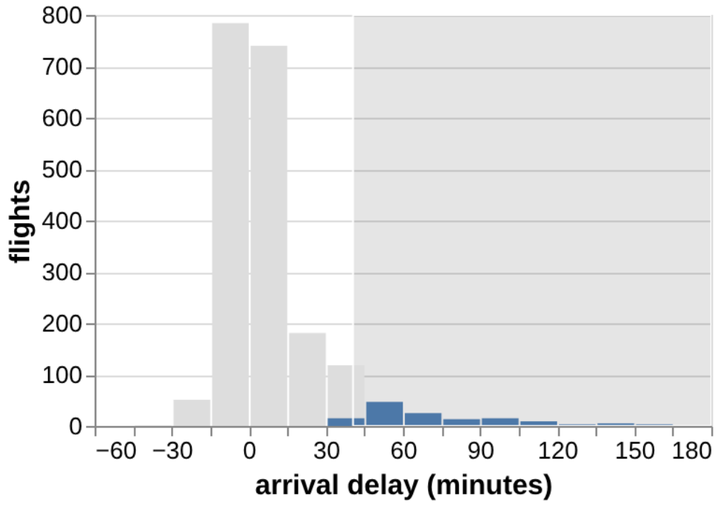
Title	Year	Genre
Jaws	1975	Thriller
Raiders of the Lost Ark	**1981**	**Action**
Ghostbusters	**1984**	Comedy
Die Hard	**1988**	**Action**
Groundhog Day	**1993**	Comedy
Speed	**1994**	**Action**
Titanic	1997	Drama
The Matrix	1999	**Action**

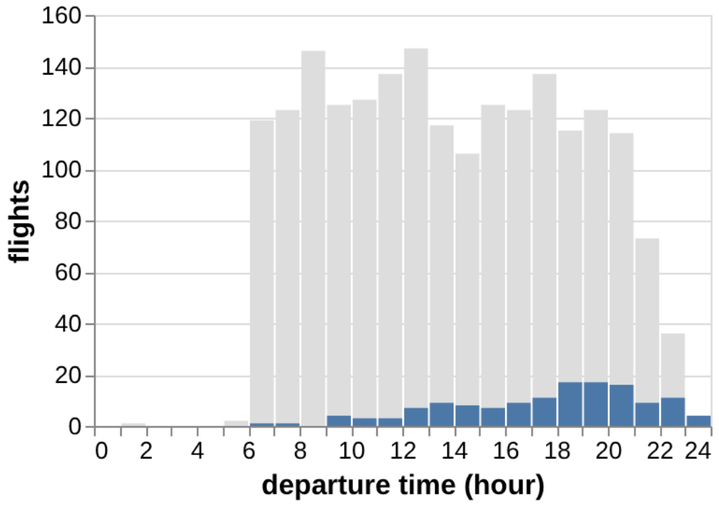
- **OR** (disjunctive)
  - Year in [1980, 1995] OR Genre = Action
- 6 of 8 movies pass
- Each added filter can only grow the result
- Common convention: **AND across widgets, OR within a widget** (checking two genres shows both)

Interaction via brushing and linking

- Brushing a range in one view filters every other view
- Grey bars keep the unfiltered distribution as context
- Try it: Vega-Lite crossfilter example (brush any chart)





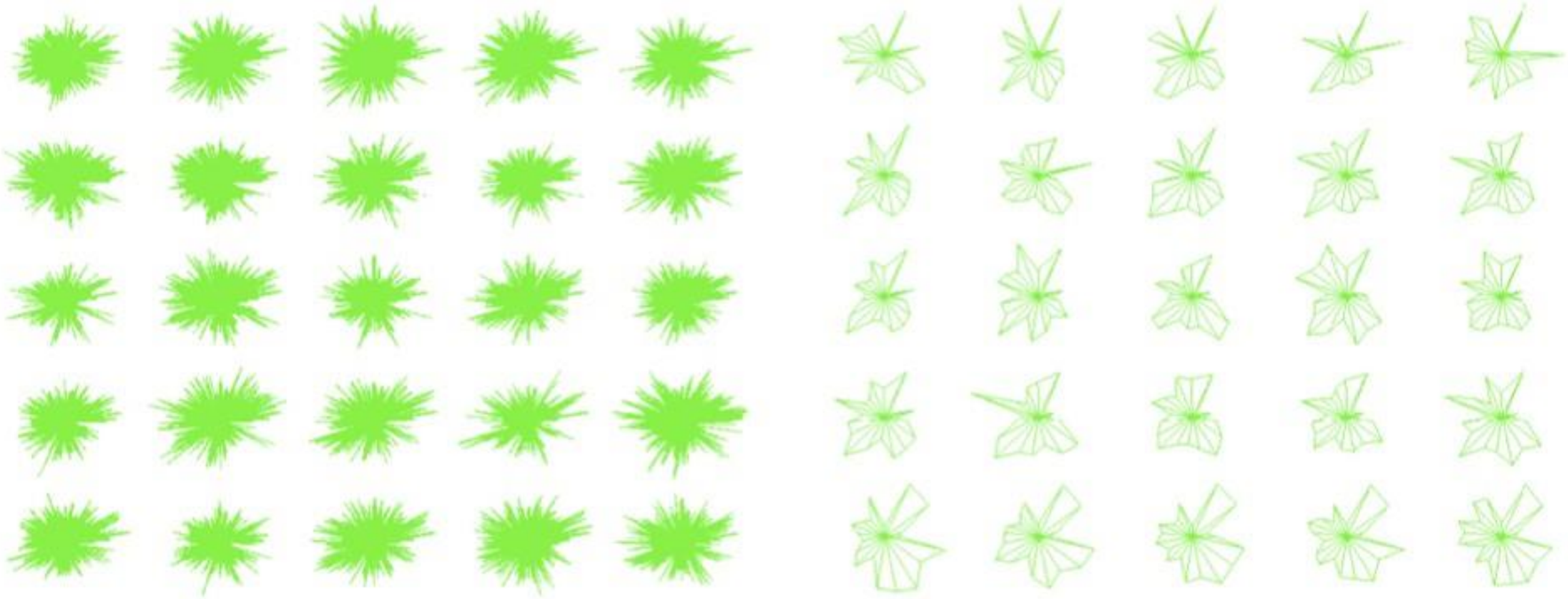


Becker & Cleveland 1987; crossfilter (Square, 2012); Vega-Lite

Filtering at scale: latency matters

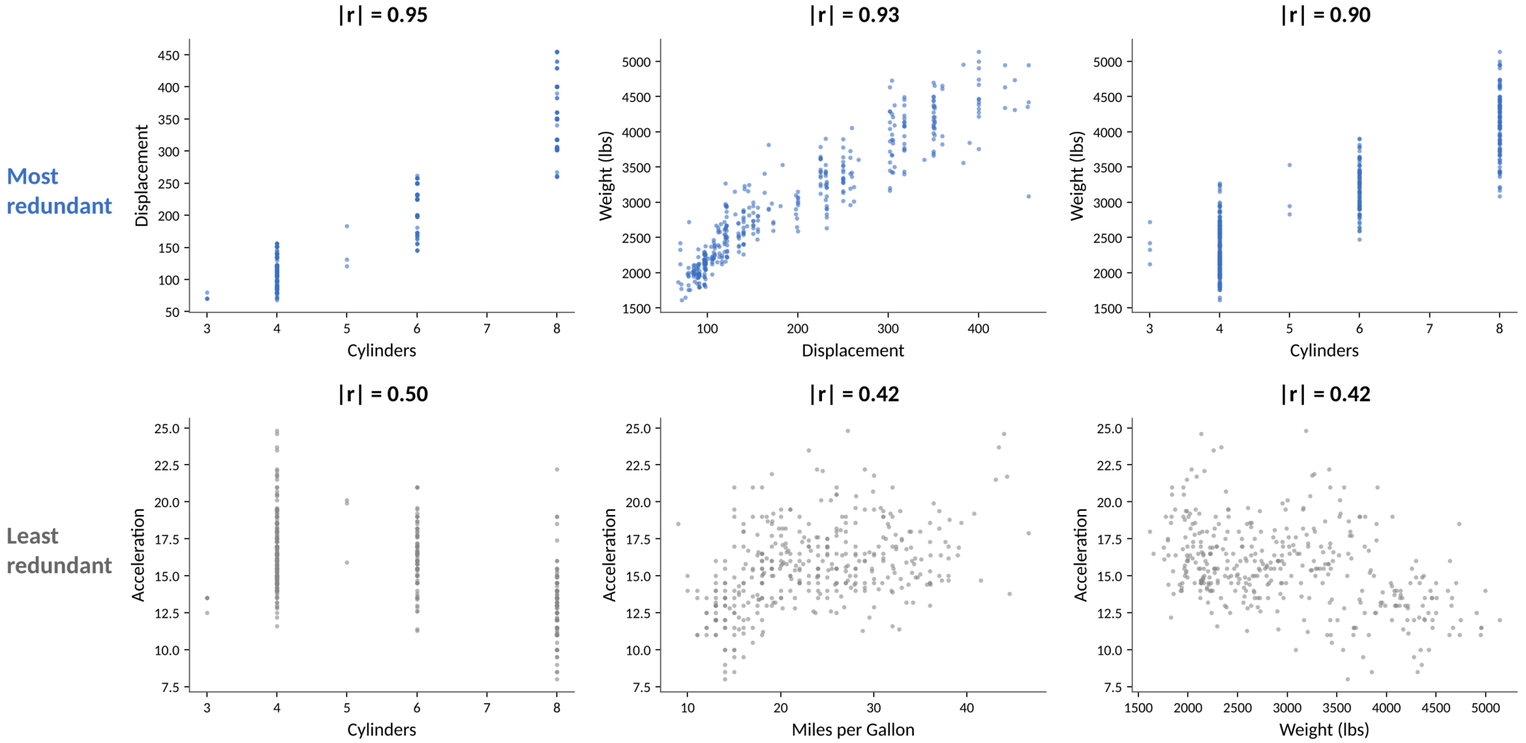
- Dynamic queries only work if feedback feels immediate
- Prior work shows that adding 500 ms of delay made people explore less
- **Performance is part of the design, not an afterthought**
  - Linked brushing over millions of rows needs precomputation

Attribute Filtering



Choosing which attributes to keep

- Attribute filtering is feature selection for the eye
- Drop
  - Near-constant attributes (low variance)
  - Redundant ones: highly correlated pairs say the same thing
- Rank candidate views by a quality measure
  - Rank-by-feature: score every 1D/2D view (correlation, outliers, …)
  - Scagnostics features score scatterplots by shape (clumpy, striated, outlying)

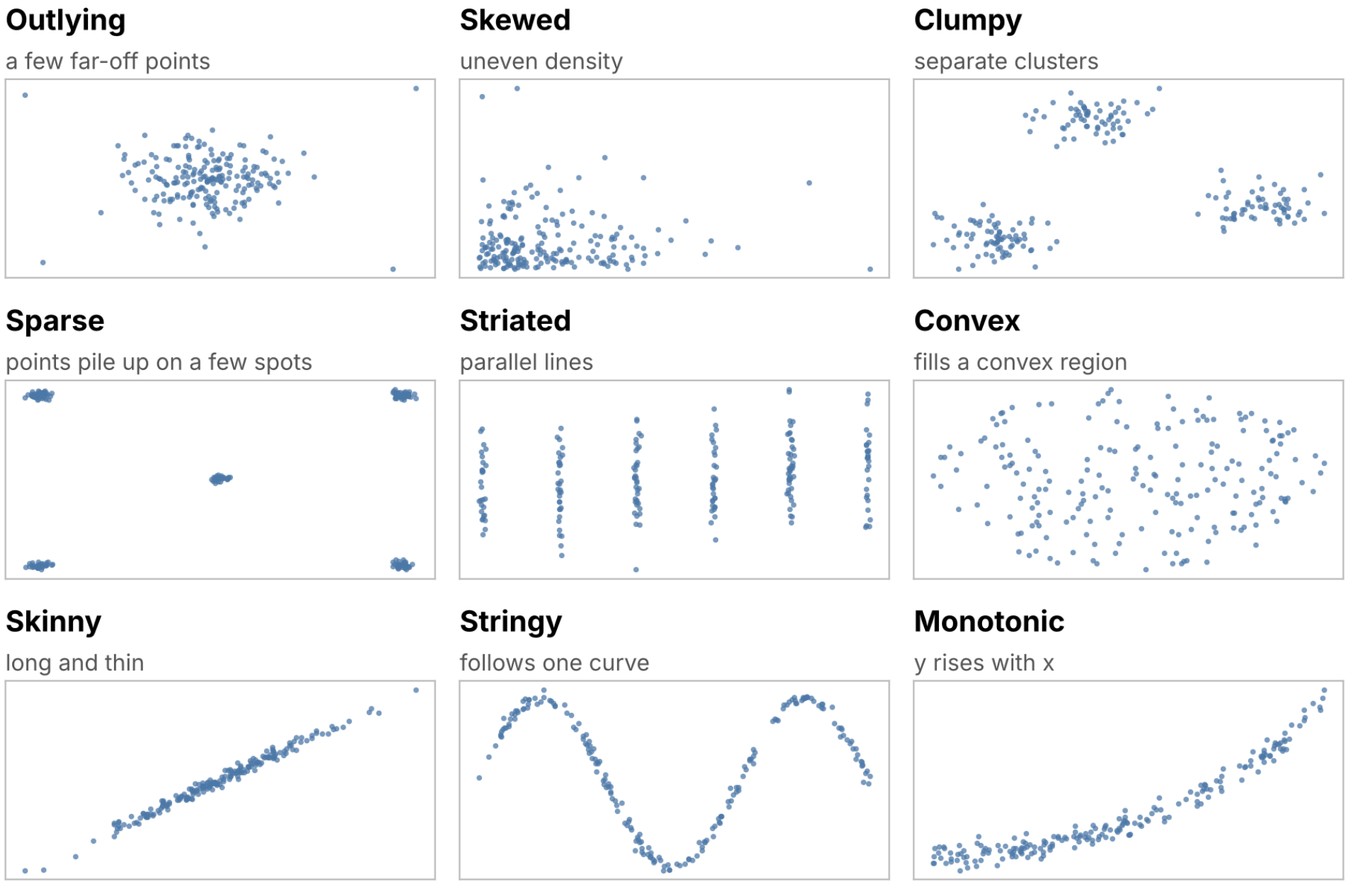


Cars: all 15 attribute pairs ranked by |r|. Top row is redundant (cylinders, displacement, weight move together); bottom row carries independent information.

Seo & Shneiderman 2005; Wilkinson et al. 2005

Scagnostics: ranking plots by shape

- Nine measures of shape, computed for every scatterplot
- Built from graphs over the points: minimum spanning tree, convex hull, alpha hull



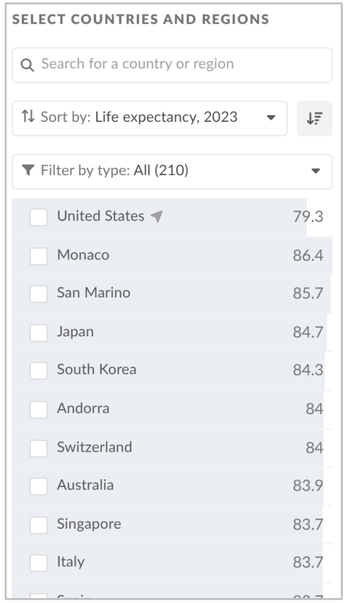
Tukey & Tukey 1985; Wilkinson et al. 2005

Controlling filtering: 2 approaches

Widget-based filtering

search, type filter, checkboxes, time slider

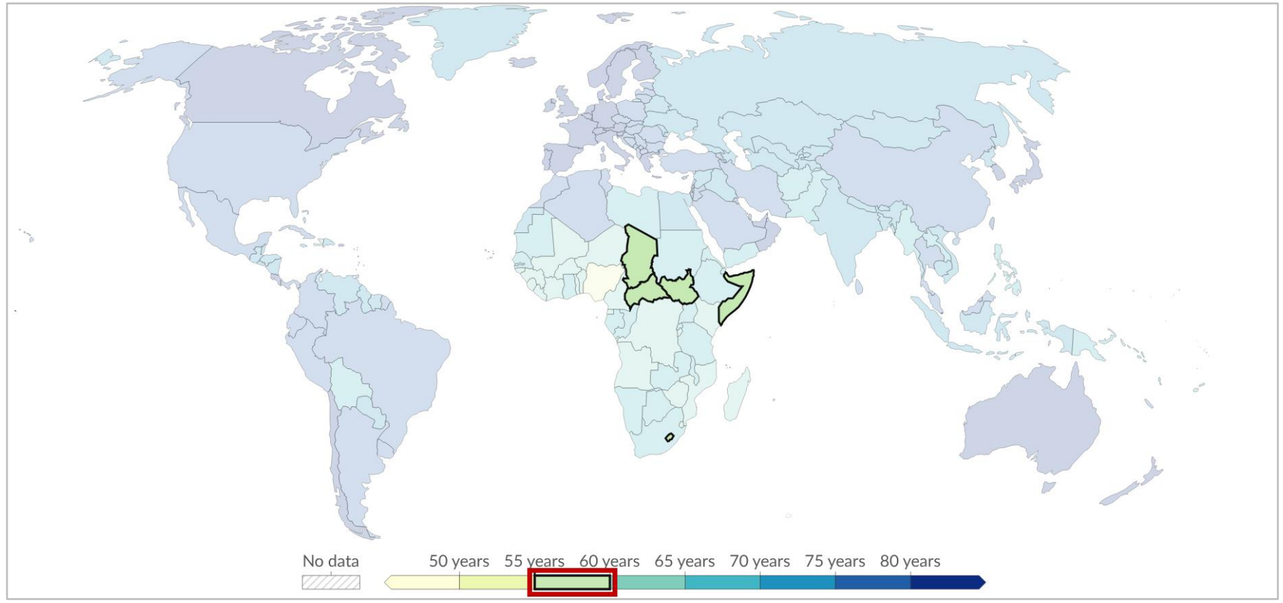
+ precise and discoverable



Visualization-based filtering

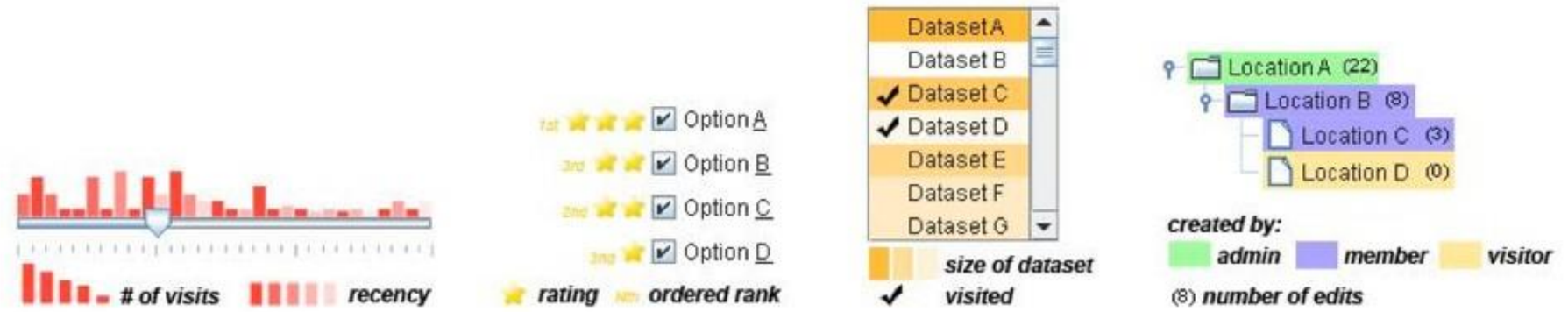
hover a legend bin to keep only that range

+ faster; your eyes stay on the data



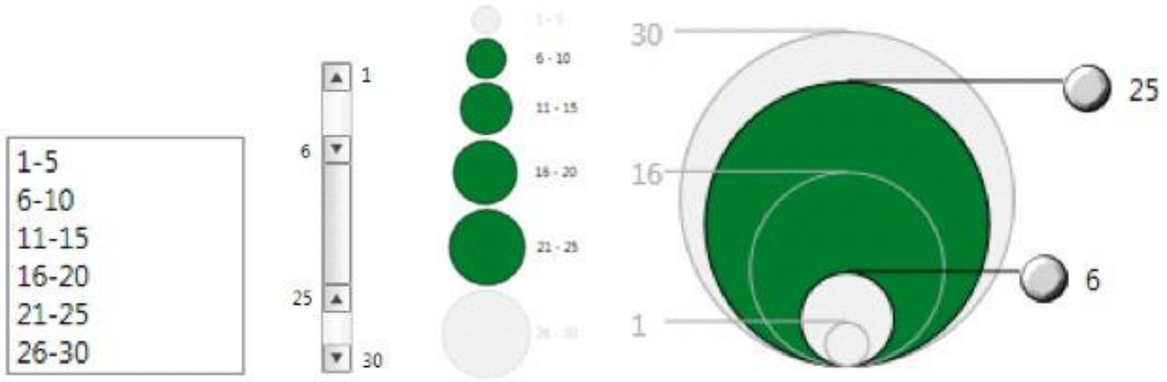
Controlling Filtering: scented widgets

- information scent: user gets sense of data
- GOAL: lower the cost of information foraging through better cues



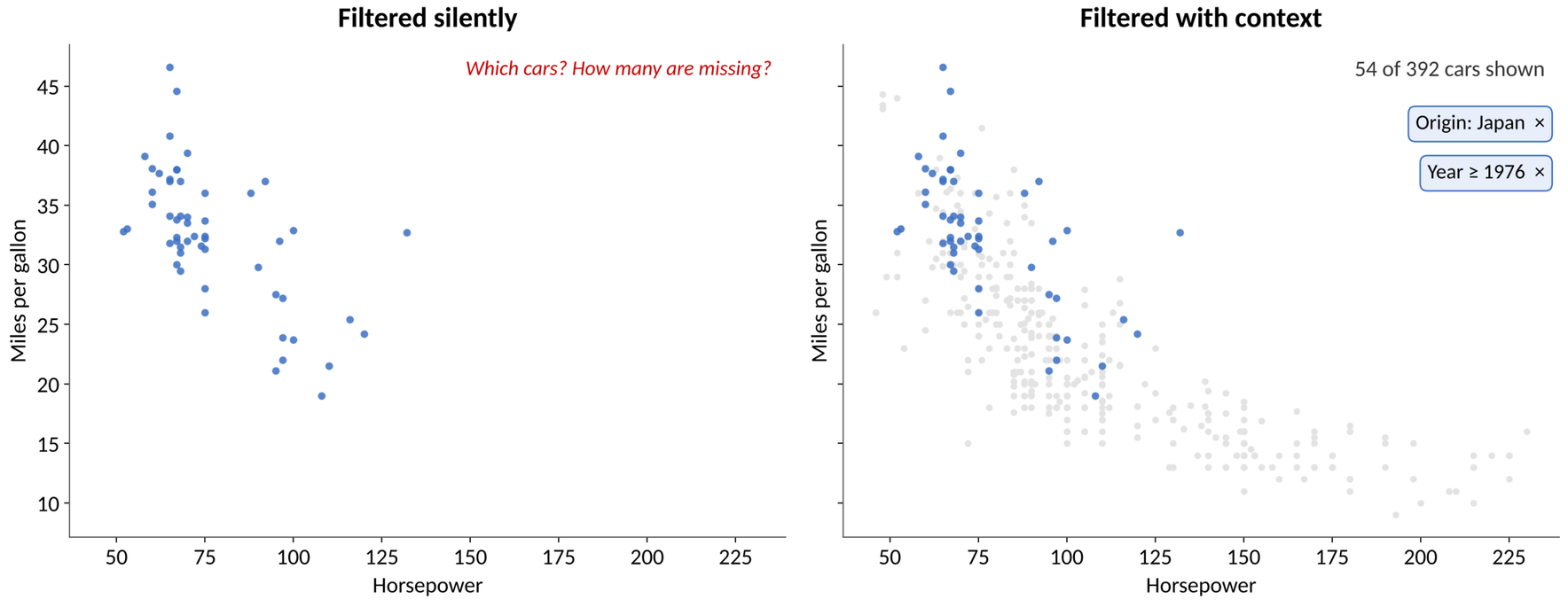
Controlling Filtering: interactive legends

- controls combining the visual representation of static legends with interaction mechanisms of widgets
- **define and control visual display together**



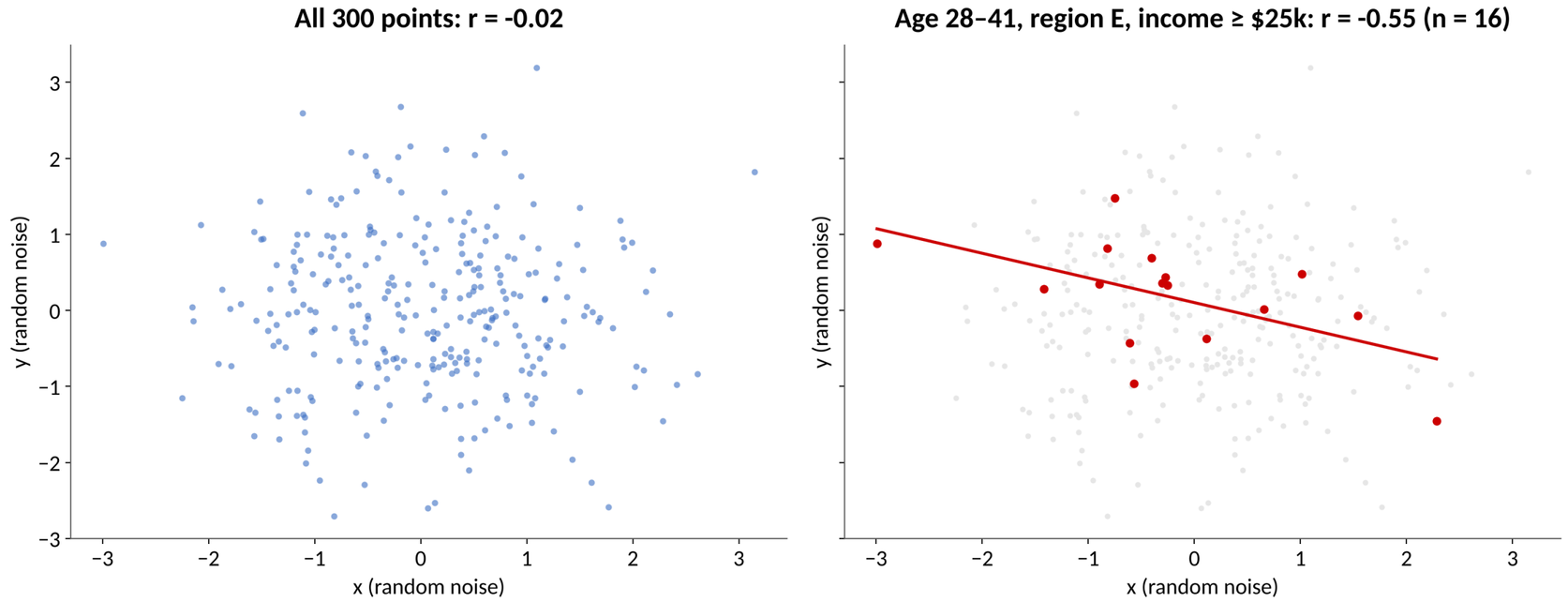
Careful to show what was filtered out

- Say how much is hidden: "54 of 392 cars shown"
- Keep filtered items as faint context when there is room
- Show active filters as removable chips



But filtering can manufacture findings…

- x and y are pure random noise
- Trying combinations of age, region, and income filters finds a subset with r = –0.55
- Every filter you try is an implicit hypothesis test: try enough and you will find a pattern



Zgraggen et al. 2018

Guarding against false findings

- In a user study, most "insights" reported during visual exploration were false discoveries
- **Remedies**
  - **Decide the question** before you start filtering
  - **Hold out data:** explore on one half, confirm on the other
  - **Count your looks:** report how many filters and views you tried
  - **Treat a filtered pattern as a hypothesis,** then test it

Takeaways

- Reduce by filtering (remove), de-emphasizing (keep as context), or aggregating (summarize)
- Filters combine: AND across widgets, OR within a widget
- Dynamic queries need immediate feedback
- Filter attributes like features: drop constant and redundant ones
- Widgets are precise; visualization-based filters are fast & in context
- Always show what was filtered out
- Every filter is a hypothesis: confirm before you report

- Next: aggregation