Information Visualization
Tables, Manipulate, TACO
In Class: Pitches

Tamara Munzner
Department of Computer Science
University of British Columbia

Week 4, 25 Sep 2025

https://www.cs.ubc.ca/~tmm/courses/547-25

Plan for today

- Pitches
  - 60-80 min pitches
    - 4 min each, up to 4 min Q&A after each
- break
- this week's reading
  - Tables mini-lecture, Q&A / discussion
  - Manipulate mini-lecture, Q&A / discussion
  - TACO mini-lecture, Q&A / discussion

Next week

- to read & discuss (async, before next class)
  - VAD book, Ch 12: Facet into Multiple Views
  - VAD book, Ch 13: Reduce Items and Attributes
  - paper: Scalable Insets [technique]
- project groups
  - finalized by Fri Oct 3 noon

Pitches

Break: 3:30-3:45

Mini-Lecture: Tables

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

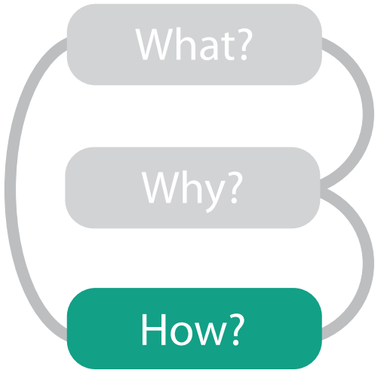
- Change
- Select
- Navigate

Facet

- Juxtapose
- Partition
- Superimpose

Reduce

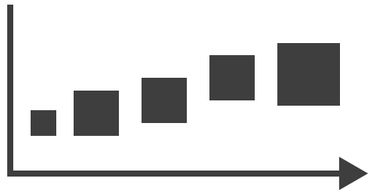
- Filter
- Aggregate
- Embed



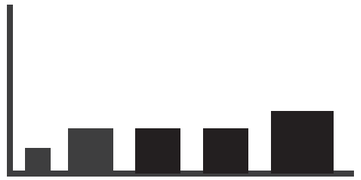
Some keys: Categorical regions

- Separate

- Order



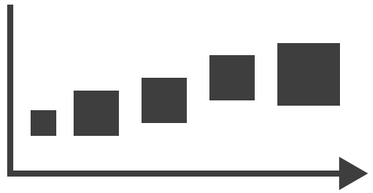
- Align



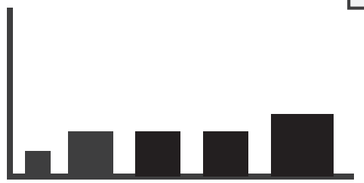
Regions: Separate, order, align

➔ Separate

➔ Order

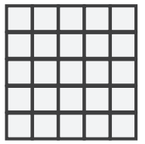


➔ Align



➔ *1 Key*
List

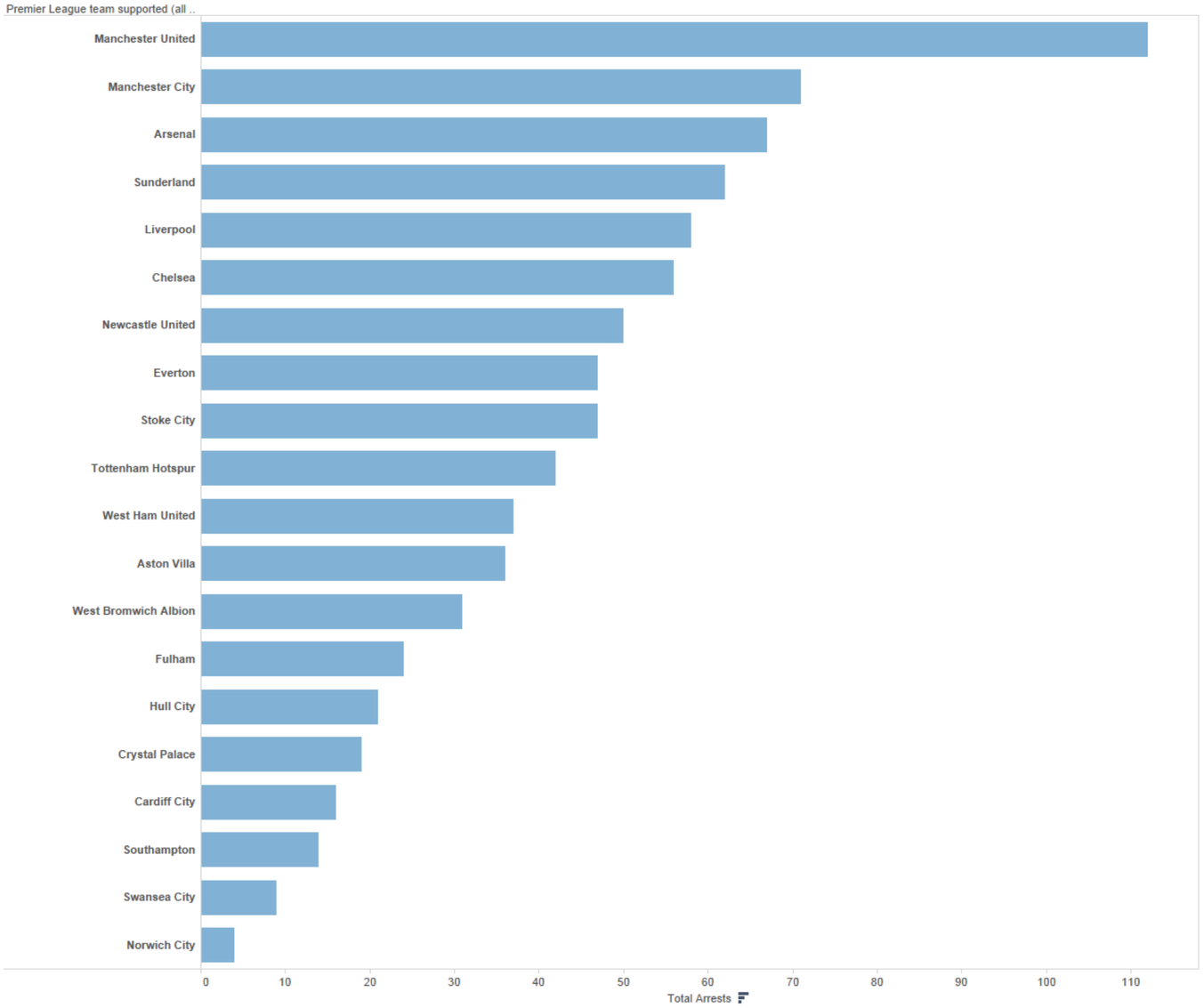
➔ *2 Keys*
Matrix



- separate into regions by any attribute: categorical or ordered
  - **regions**: contiguous bounded areas distinct from each other
  - no conflict with expressiveness principle for categorical attributes
  - one glyph/mark per region
- then can use ordered attribute to order and align regions
  - order in one direction (eg horiz), then align in other direction (eg vert)
  - align: impose shared coordinate frame so absolute position has meaning
    - difference between value and 0 point along axis

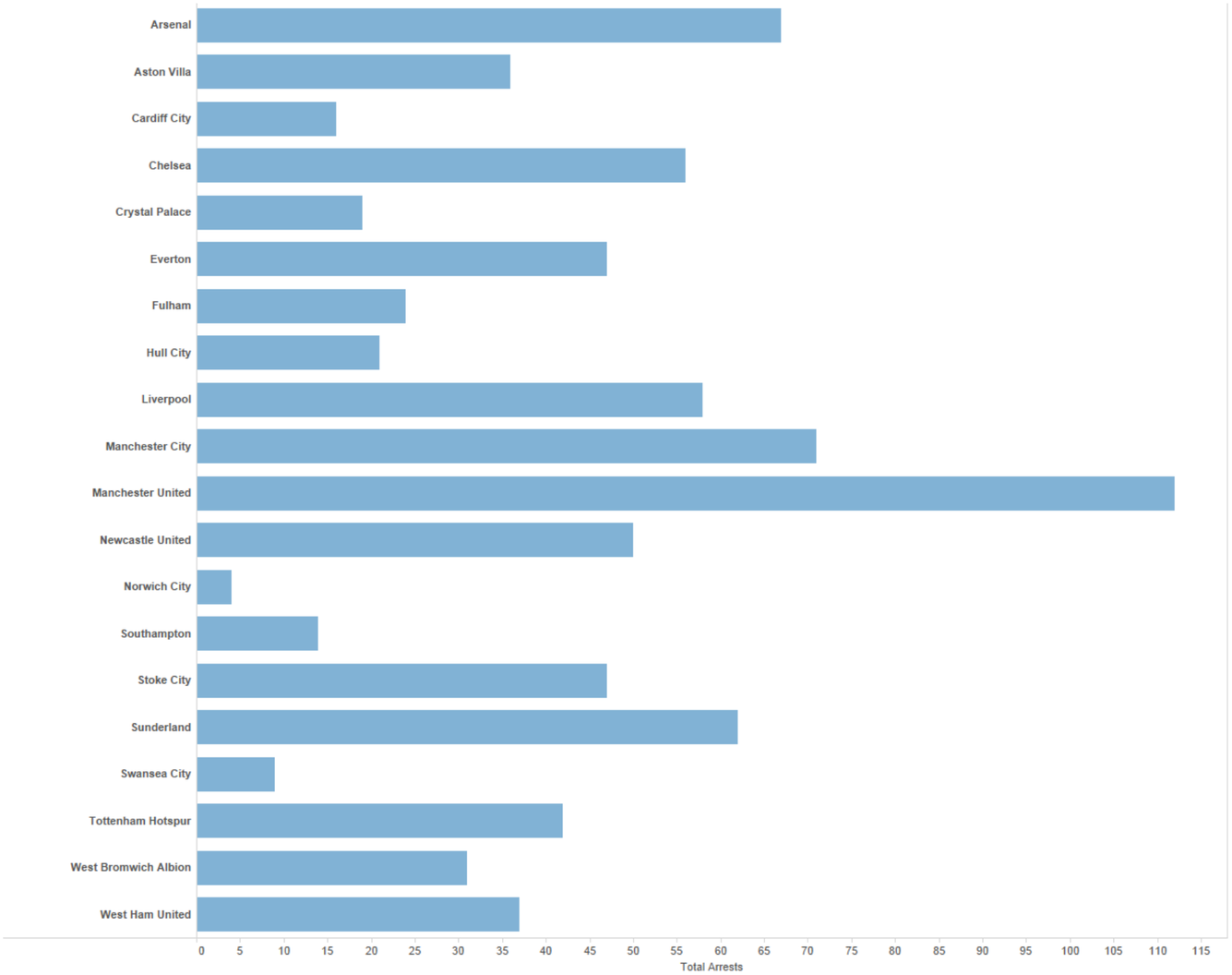
Separated and aligned and ordered

- best case



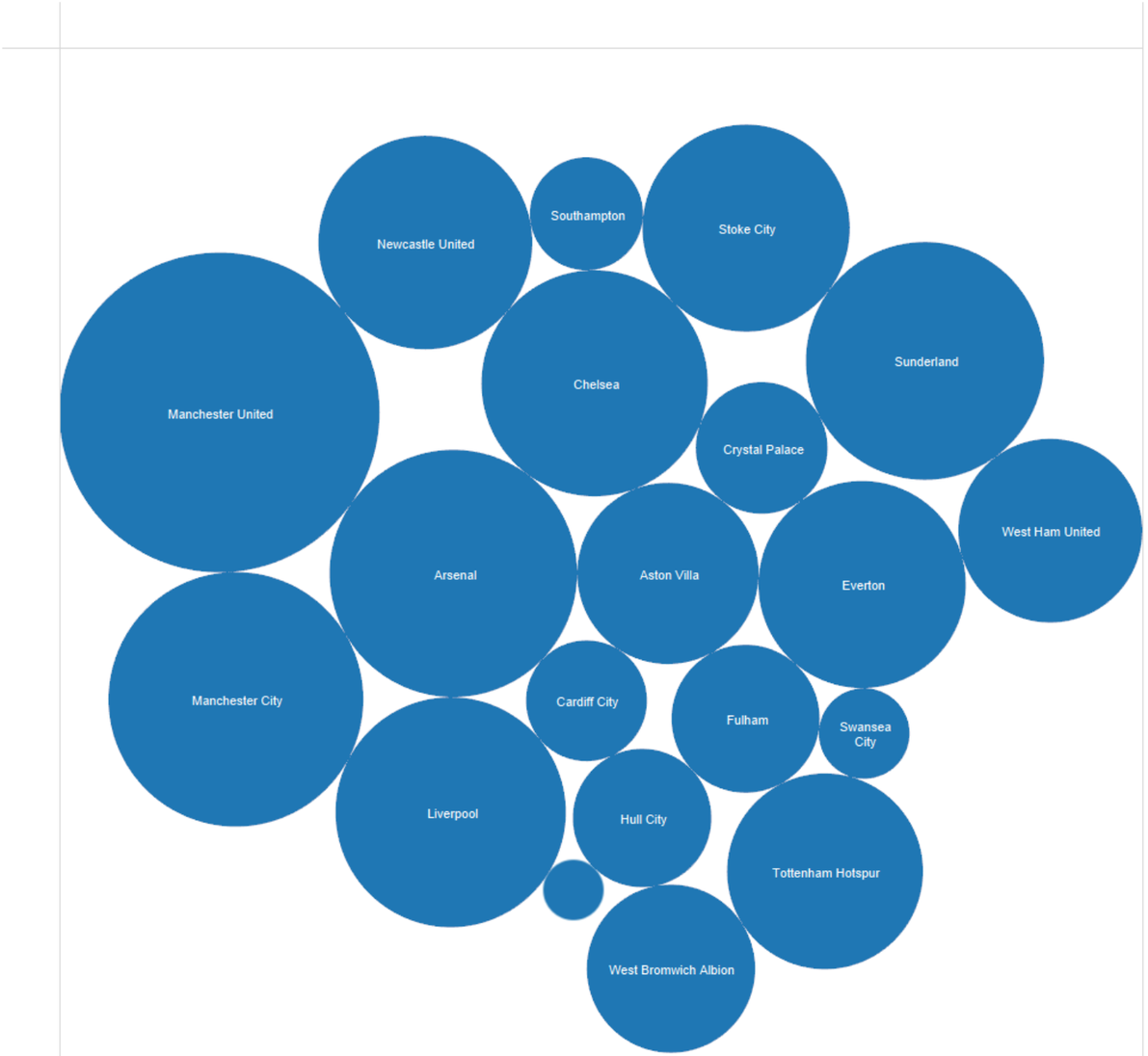
Separated and aligned but not ordered

- limitation: hard to know rank. what's 4th? what's 7th?



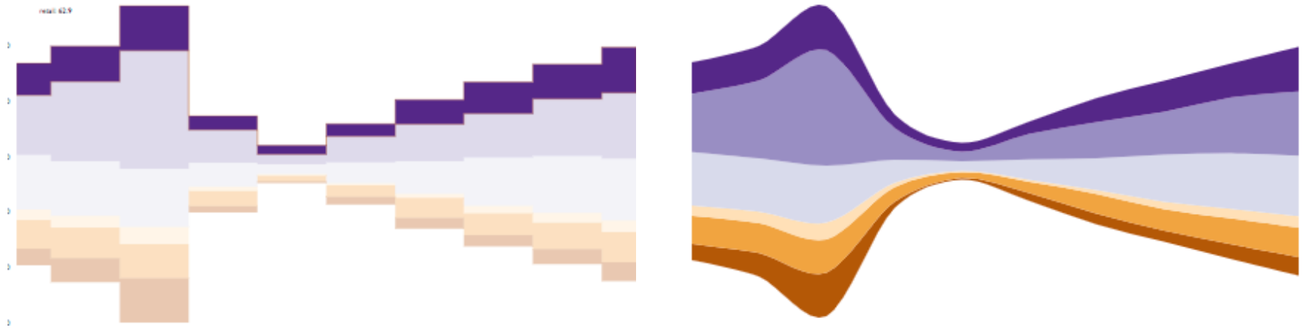
Separated but not aligned or ordered

- limitation: hard to make comparisons with 2D size (vs aligned 1D position)

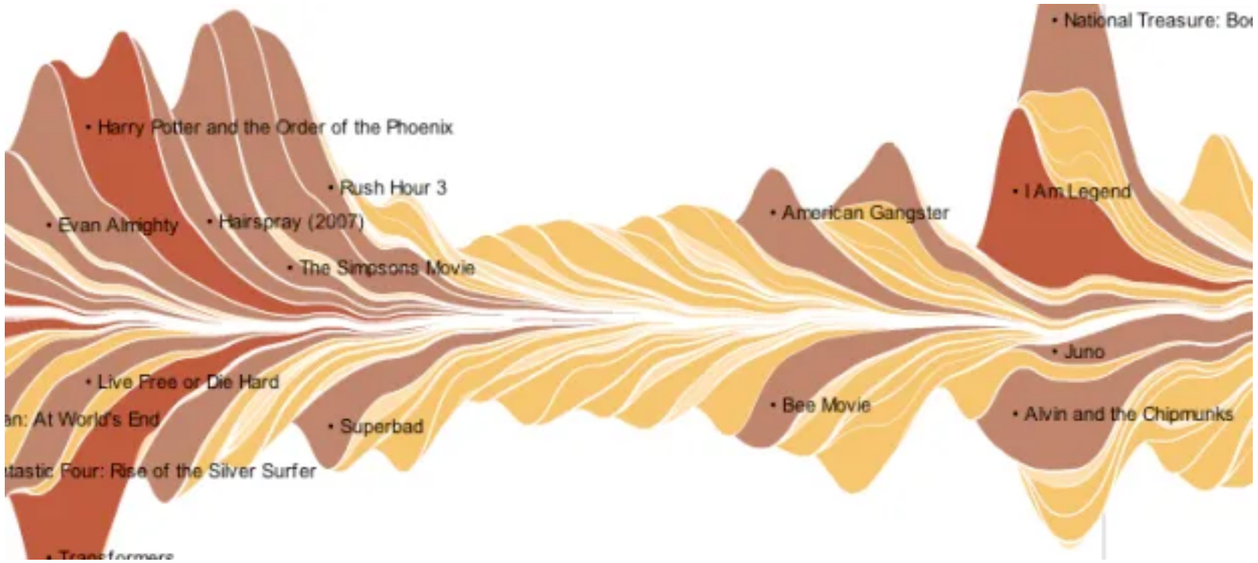


Idiom: **streamgraph**

- generalized stacked graph
  - emphasize horizontal continuity of "stacked" attribute across time
    - not lookup for quant attribute at discrete time point
  - data: 2 keys
    - 1 categ key attrib (movies)
    - 1 ordered key attrib (time)
    - 1 quant value attrib (counts)
  - derived data/marks: polys, 1 mark per movie
    - vert 1D size (height): count for movie at time
    - vert pos: unavailable
    - vert order: derived by algorithm
    - horiz pos/order: time
    - horiz size: unavailable
  - scalability
    - dozens to hundreds of time keys (like stacked bar chart)
    - dozens to hundreds of movies keys (much more!)
      - most layers don't extend across whole chart
      - but very few have readable labels (big ones static, others hover only)



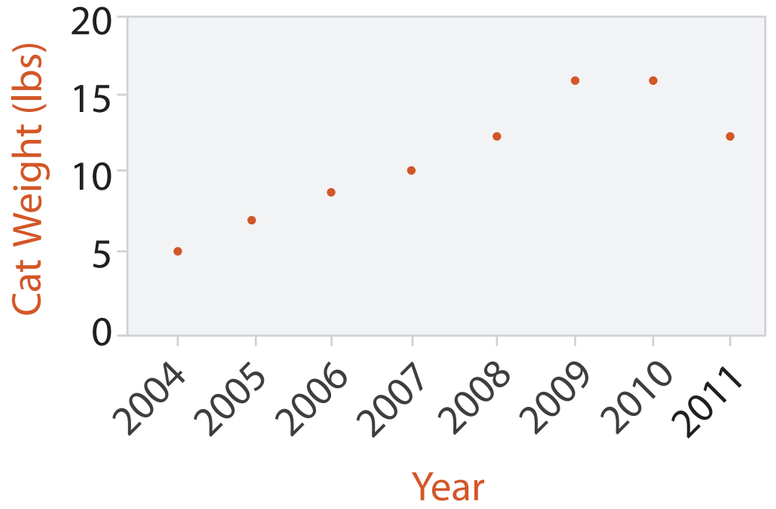
[Stacked Graphs Geometry & Aesthetics. Byron and Wattenberg. IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis 2008) 14(6): 1245–1252, (2008).]



https://flowingdata.com/2008/02/25/ebb-and-flow-of-box-office-receipts-over-past-20-y

Idiom: **line chart**

- one key, one value
  - data: 2 quant attribs: 1 key, 1 value
  - mark:
    - path: connecting mark between item values at each key
      - could be piecewise-linear or smooth
    - (could also use point marks at value locations)
  - channels
    - express quant value: horiz & vert pos/order
    - vert 1D size available (linewidth)
      - paths: consider along the line as horizontal, other way as vert
    - horiz size of each segment not available
      - just shows interval between expressed values, often uniform
  - task
    - find trend
      - connection marks emphasize ordering of items along key axis by explicitly showing relationship between one item and the next
  - scalability
    - 1K levels for value and key



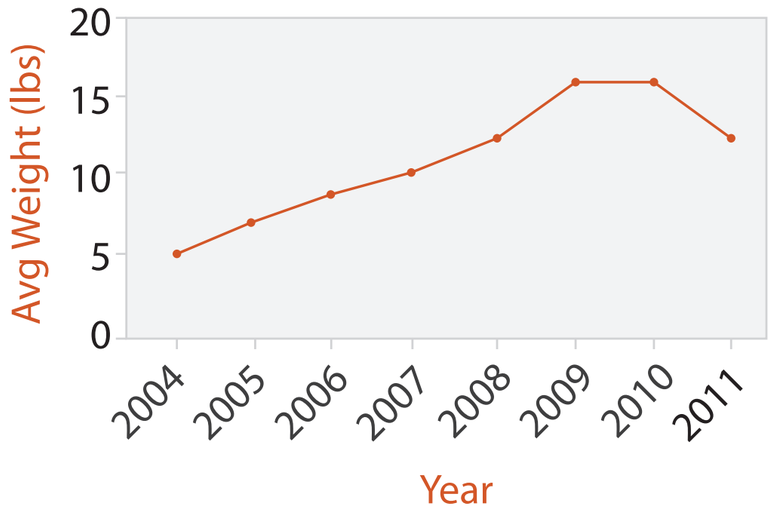
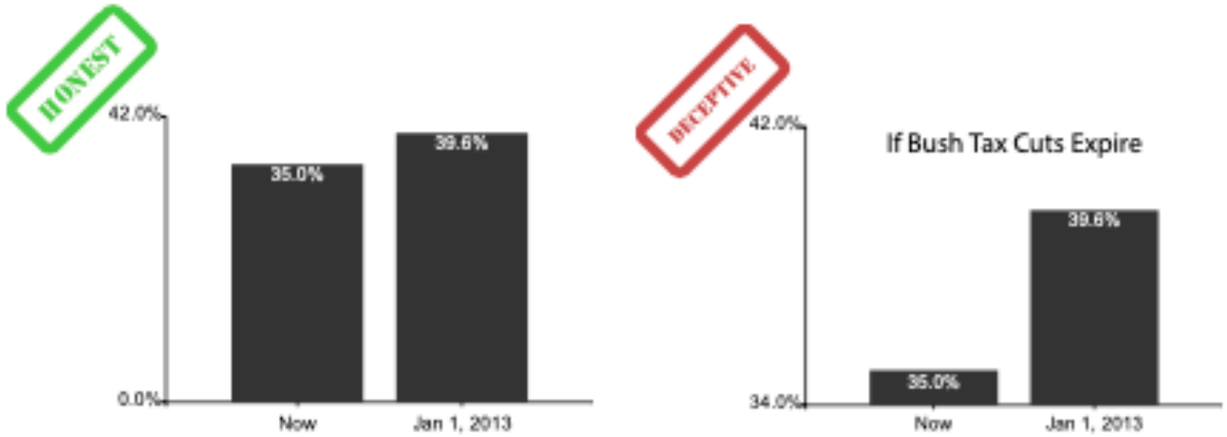


Chart axes: avoid cropping y axis

- include 0 at bottom left or slope misleads

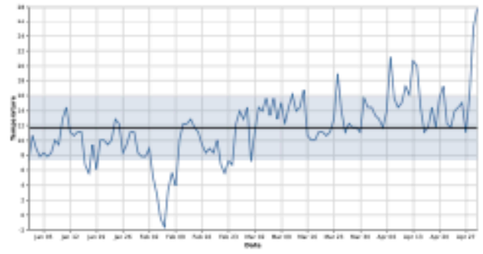


[Truncating the Y-Axis: Threat or Menace? Correll, Bertini, & Franconeri, CHI 2020.]

Chart axes: avoid cropping y axis

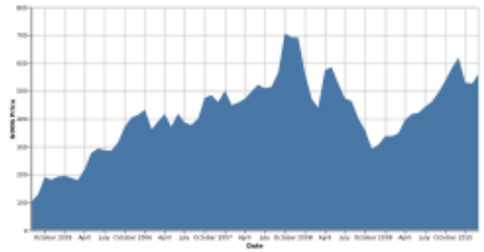
- include 0 at bottom left or slope misleads
  - some exceptions (arbitrary 0, small change matters)

[Truncating the Y-Axis: Threat or Menace? Correll, Bertini, & Franconeri, CHI 2020.]

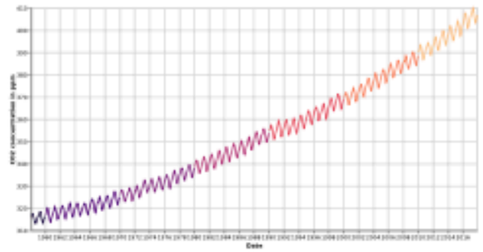


(a) Statistical process charts rely on comparison to an expected value, and so deviations from that value, not from zero, are important

(b) Index charts compare to an indexed value rather than zero.



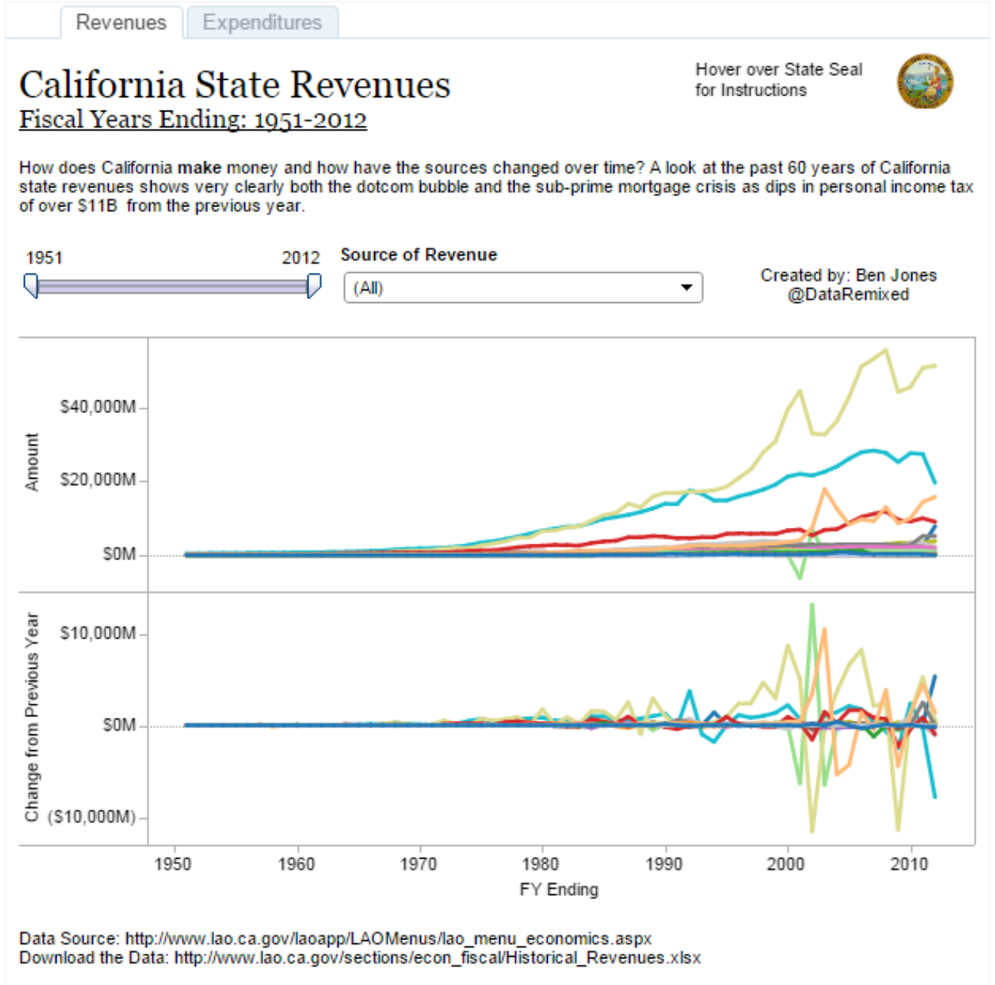
(c) Stock charts must show small differences in stock value, as these can translate to enormous monetary gains or losses.



(d) Climate Anomaly charts rely on both highlighting deviation from a non-zero expected value but also emphasize the potentially disastrous impact of even minute changes in climate.

Idiom: **Indexed line charts**

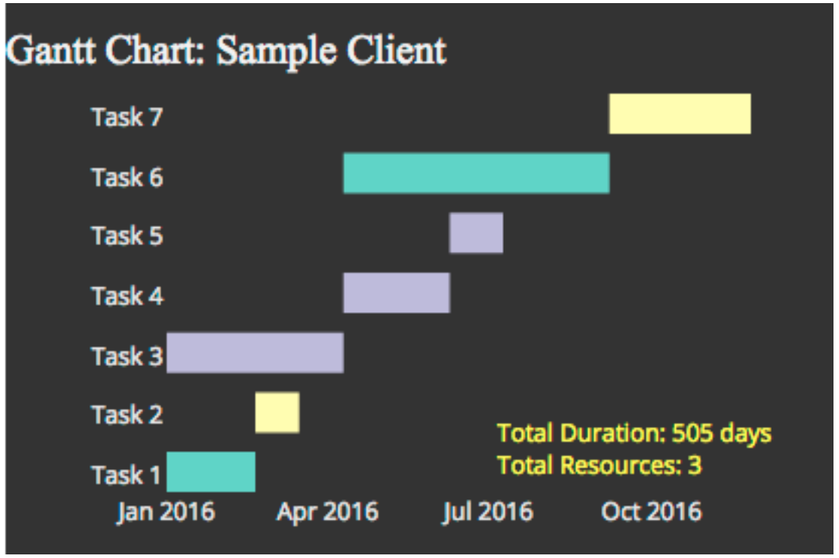
- data: 2 quant attribs
  - 1 key + 1 value
- derived data: new quant value attrib
  - index
  - plot instead of original value
- task: show change over time
  - principle: normalized, not absolute
- scalability
  - same as standard line chart



https://public.tableau.com/profile/ben.jones#!/vizhome/CAStateRevenues/Revenues

Idiom: **Gantt charts**

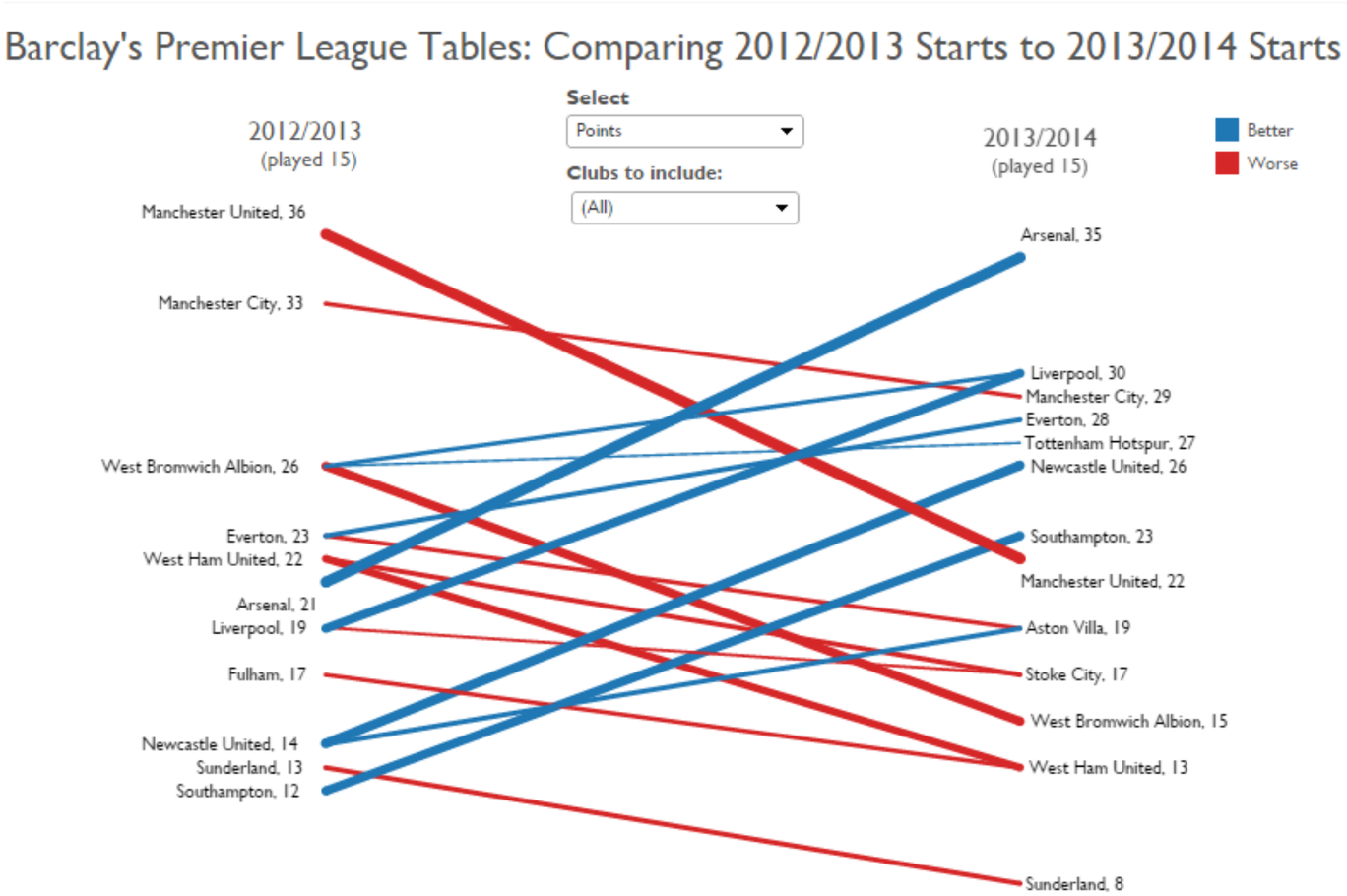
- one key, two (related) values
  - data
    - 1 categ attrib, 2 quant attribs
  - mark: segment
  - channels
    - horiz position/order: start time & end time
      - redundant: horiz length (1D size) is duration (end-start)
    - vert order: task name/number, vert position: task
  - task
    - emphasize temporal overlaps & start/end dependencies between items
  - scalability
    - dozens of key levels [bars]
    - 1K value levels [times, durations]



https://www.r-bloggers.com/gantt-charts-in-r-using-plotly/

Idiom: **Slopegraphs**

- express two values
  - data
    - 2 quant value attribs
    - (optional: 1 derived attrib: change magnitude)
  - mark: segment
    - connecting mark between 2 values
  - channels
    - vertical pos/order, 1 axis per attrib: express value
    - segment width (1D size, *vert*): change magnitude
    - segment length (1D size, *horiz*): **unavailable**
    - color
  - task
    - emphasize changes in rank/value
  - scalability
    - 1K value levels
    - dozens of items



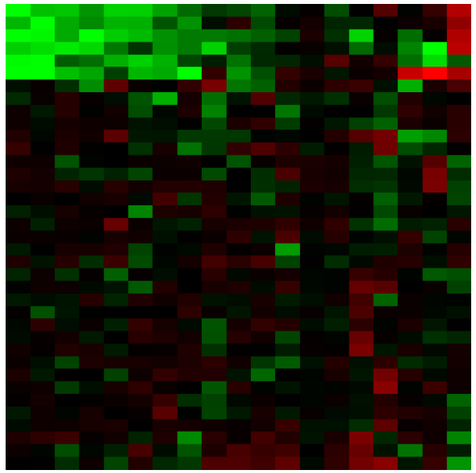
https://public.tableau.com/profile/ben.jones#!/vizhome/Slopegraphs/Slopegraphs

Idiom: **heatmap**

- two keys, one value
  - data
    - 2 categ attribs (gene, experimental condition)
    - 1 quant attrib (expression levels)
  - marks: point
    - separate into regions, order and align both directions
    - vertical & horizontal: position/order, **shared** boundaries, size **unavailable**
    - orientation & 2D size unavailable
    - indexed by 2 categorical key attributes
  - channels
    - color by quant attrib (ordered diverging colormap)
  - task: find outliers, find clusters (if ordered well)
  - scalability: asymmetric
    - 100s of categ levels, up to 1M items, but ~10 quant attrib levels

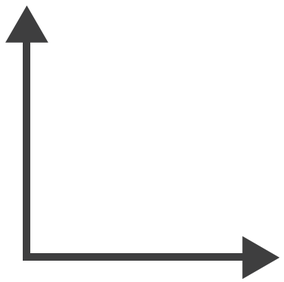
➔ *2 Keys*

Matrix

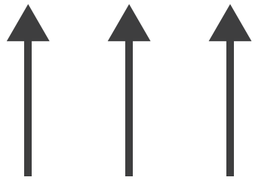


Axis Orientation

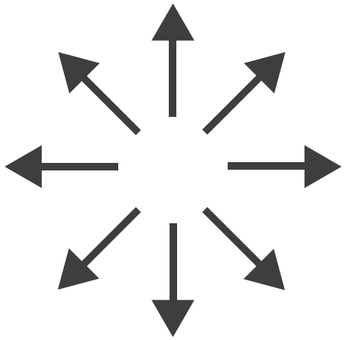
- Rectilinear



- Parallel

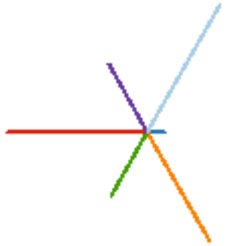


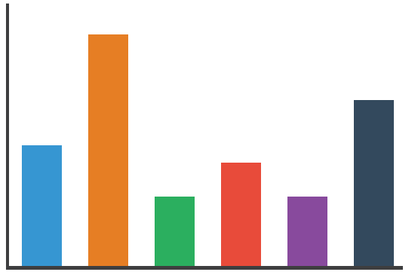
- Radial



Idioms: **radial bar chart, star plot**

- star plot
  - segment mark, axes meet at central point
- radial bar chart:
  - segment mark, axes meet at central ring
- both: radial (polar) axes
    - radial position/order/length for quant attrib
    - angular position/order for other attrib
- bar chart: rectilinear (cartesian) axes
    - vertical position/order/length for quant attrib
    - horizontal position/order for other attrib
- accuracy
  - high-precision rectilinear aligned position decoding does **not** apply to radial layouts!
    - lower-accuracy length judgement, not aligned position

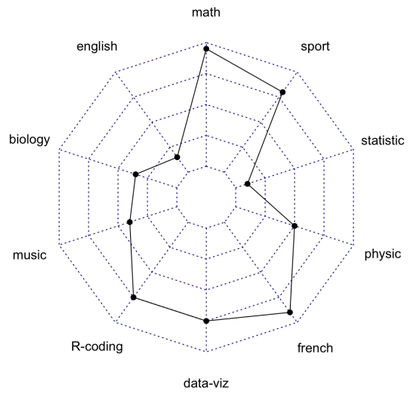


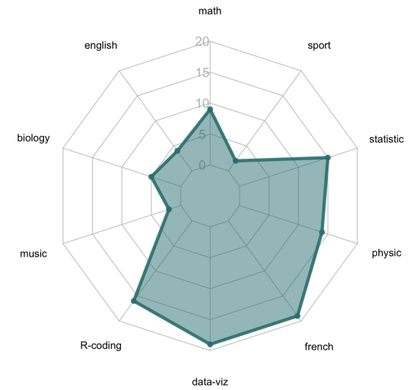


[Vismon: Facilitating Risk Assessment and Decision Making In Fisheries Management. Booshehrian, Möller, Peterman, and Munzner. Technical Report TR 2011-04, Simon Fraser University, School of Computing Science, 2011.]

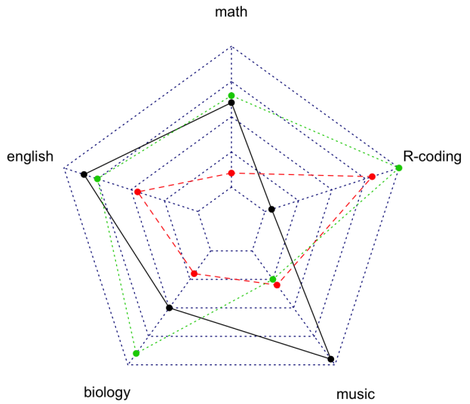
Idiom: **radar/spider chart**

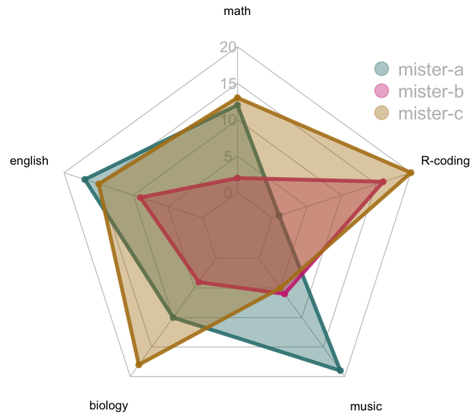
- like line chart, but radial:
  - connecting mark between multiple item values
- unlike rectilinear line chart
  - closed, so mark is poly boundary
  - vs rectilinear: open, so mark is path
- options
  - fill color for interior
    - makes shape even more salient
  - explicitly plotting point marks
- other names
  - Kiviat diagram, polar chart, ...
- guidance: avoid unless data is cyclic





https://r-graph-gallery.com/142-basic-radar-chart.html

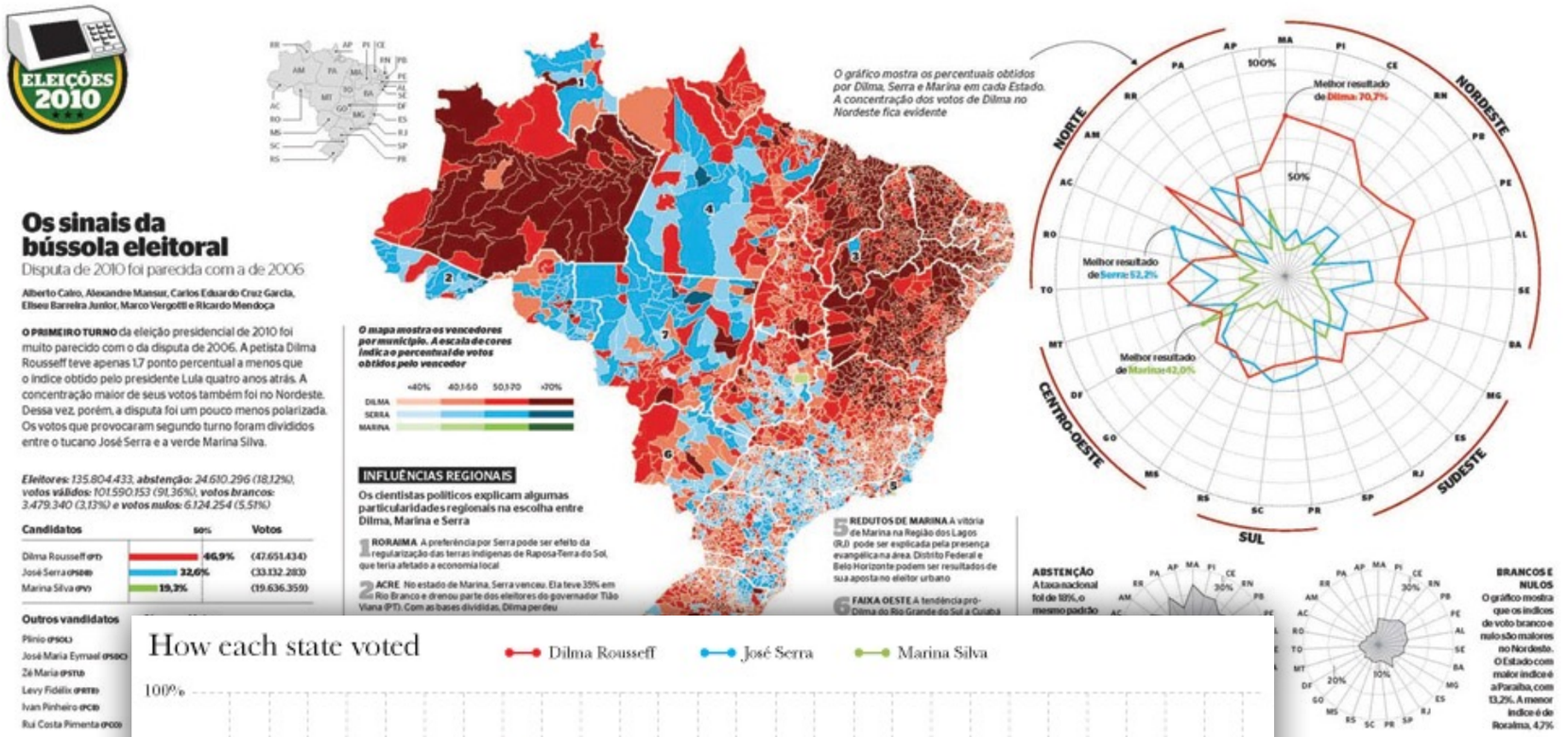




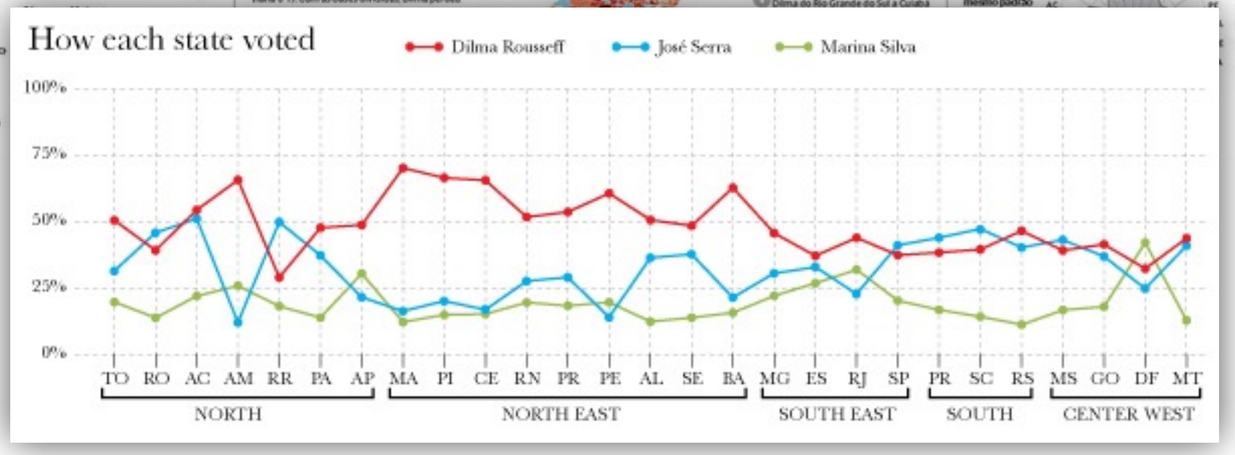
https://r-graph-gallery.com/143-spider-chart-with-saveral-individuals.html

https://www.chartjs.org/docs/latest/charts/radar.html

“Radar graphs: Avoid them (99.9% of the time)”



original
difficult to interpret

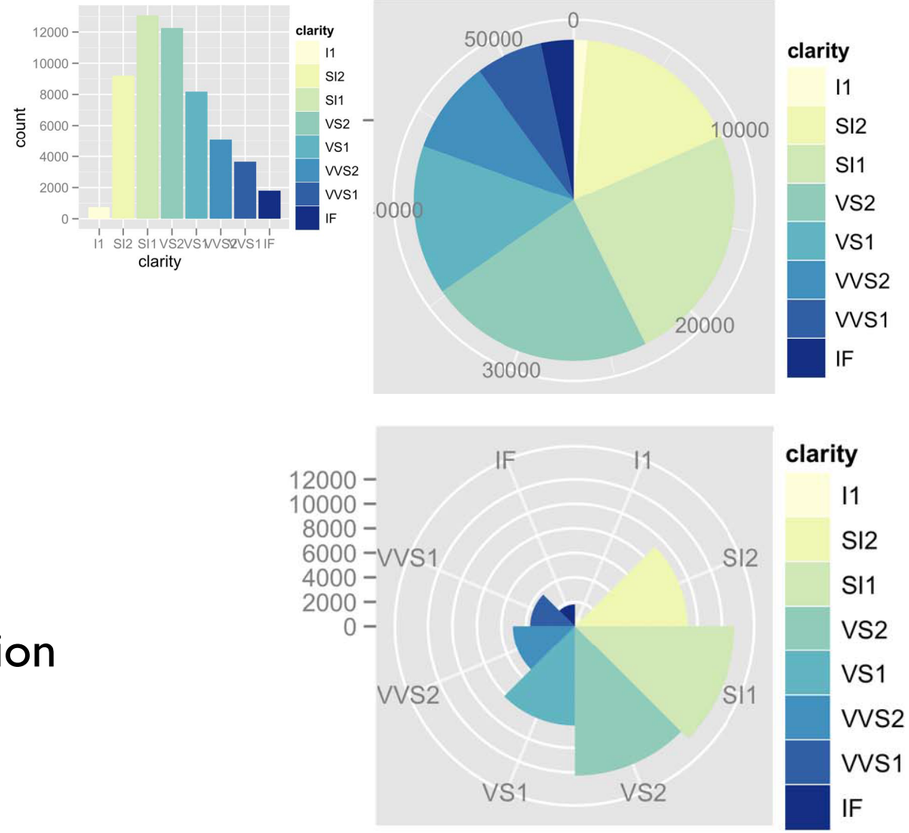


redesign for
rectilinear

http://www.thefunctionalart.com/2012/11/radar-graphs-avoid-them-999-of-time.html

Idioms: **pie chart, coxcomb chart**

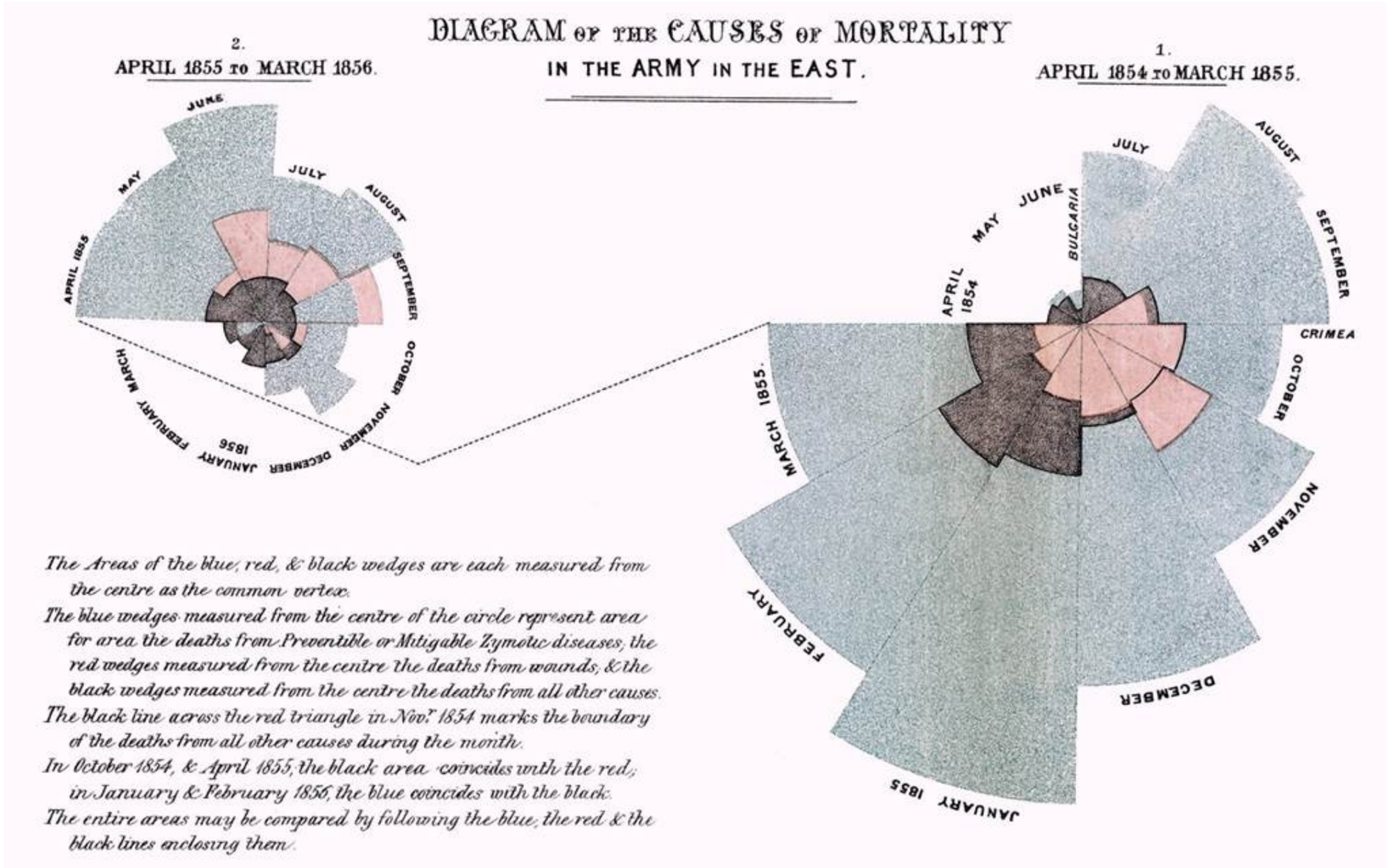
- data
  - 1 categ key attrib, 1 quant value attrib
- pie chart: vary angular position, fix radial distance
  - poly marks
    - angular position/order/size by attribute (clarity), **shared** boundary
    - radial position/order/size is uniform & **unavailable**
    - 2D size (area) **unavailable** (but varies linearly with angular 1D size)
  - accuracy: angular position less accurate than rectilinear aligned position
  - task: part-to-whole judgements
- coxcomb chart: vary radial distance, uniformly distribute angular position
  - direct analog to bar charts, just radial
  - poly marks
    - angular 1D size (width) is uniform & unavailable
    - angular position/order is uniform, **shared** boundary
    - radial position/order/size by attribute (clarity)
    - 2D size (area) **unavailable**



[A layered grammar of graphics. Wickham. Journ. Computational and Graphical Statistics 19:1 (2010), 3–28.]

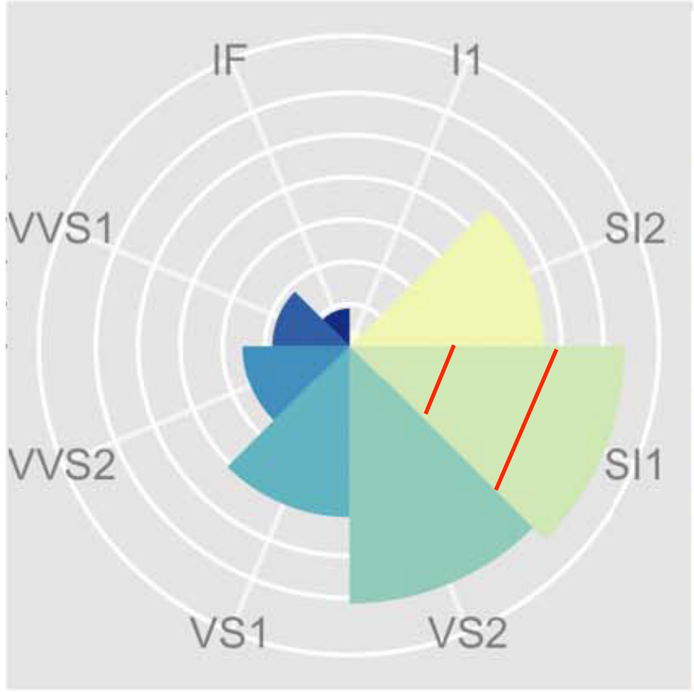
Coxcomb / nightingale rose / polar area chart

- invented by Florence Nightingale:
  Diagram of the Causes of Mortality in the Army in the East

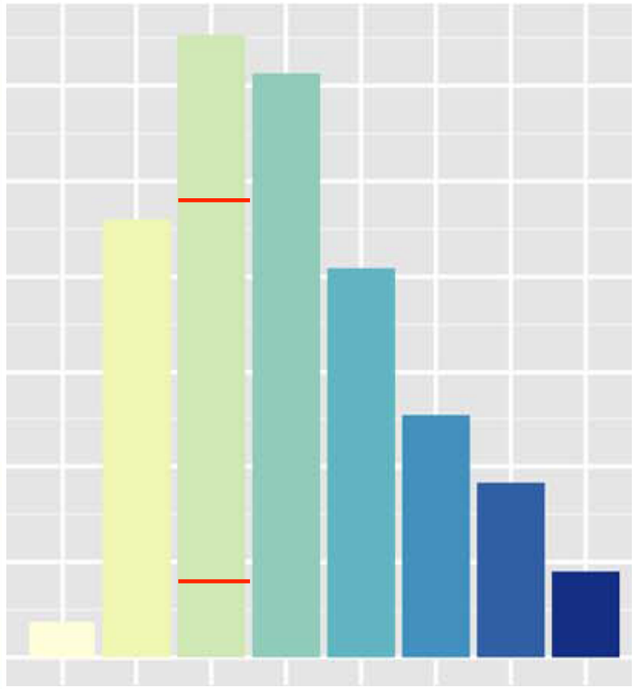


Coxcomb: perception

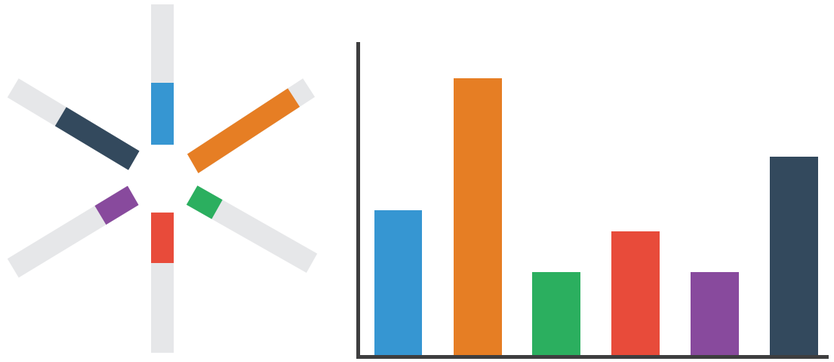
- nonuniform sector width as length increases
  - so area variation is nonlinear wrt mark length!
- encode: length
  - 1D angular size
- decode/perceive: area
  - 2D size
- bar chart safer: uniform width, so area is linear w/ mark length
  - both radial & rectilinear cases



nonuniform width as length increases



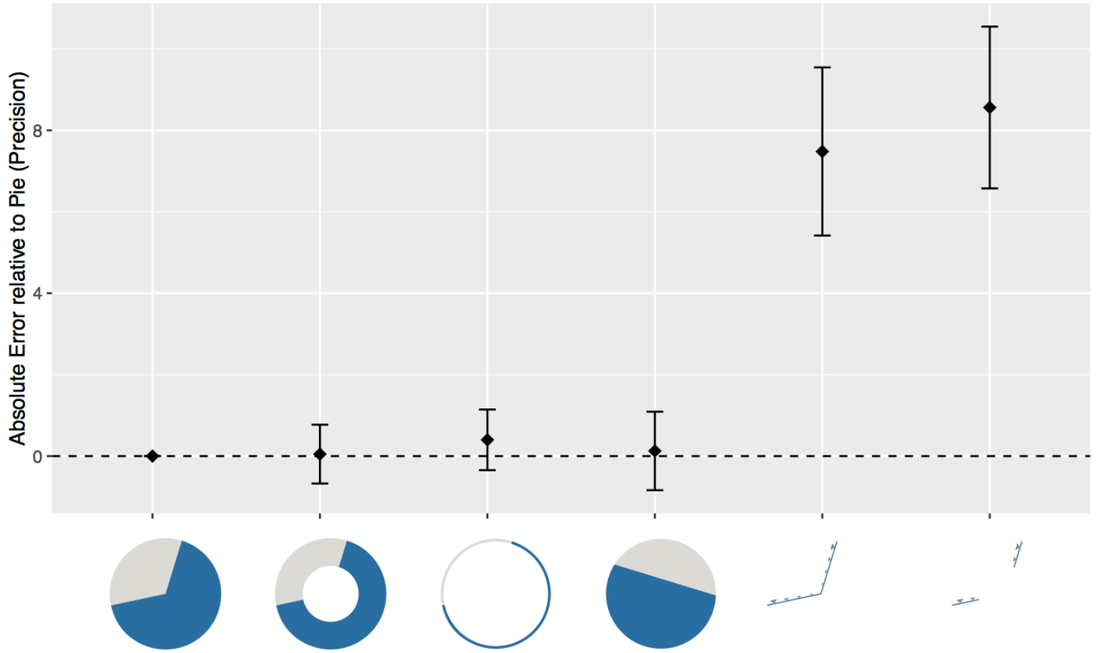
uniform width as length increases



radial & rectilinear bars: uniform width as length increases

Pie charts: perception

- what do viewers decode?
- some empirical evidence that people respond to arc length
  - decode/perceive: not angles
  - maybe also areas?…
- donut charts no worse than pie charts



[Arcs, Angles, or Areas: Individual Data Encodings in Pie and Donut Charts. Skau and Kosara. Proc. EuroVis 2016.]

https://eagereyes.org/blog/2016/an-illustrated-tour-of-the-pie-chart-study-results

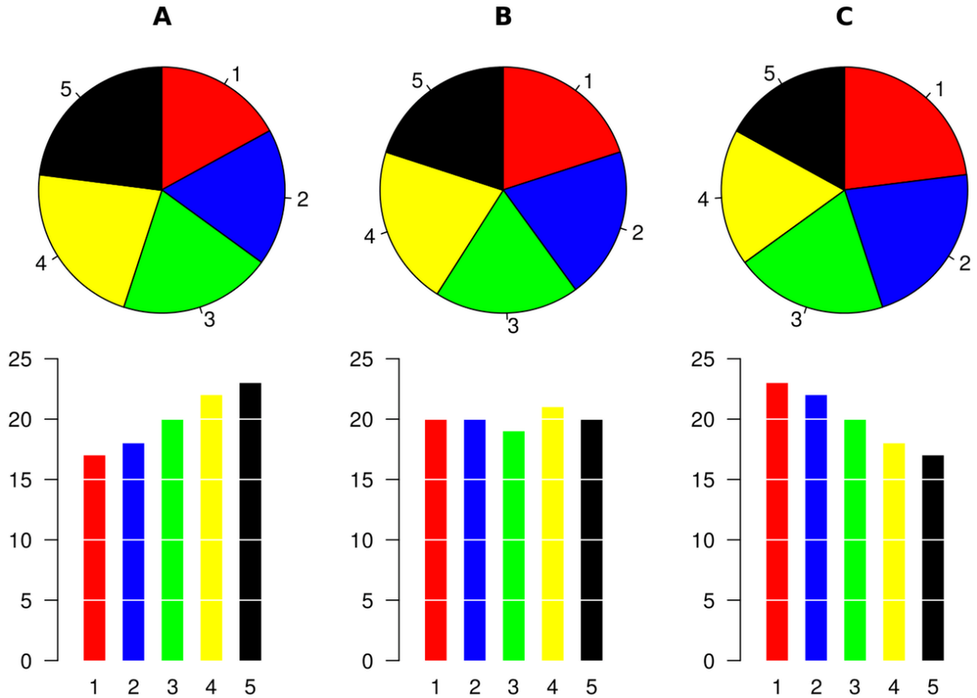
Pie charts: best practices

- not so bad for two (or few) levels, for part-to-whole task

https://eagereyes.org/pie-charts

Pie charts: best practices

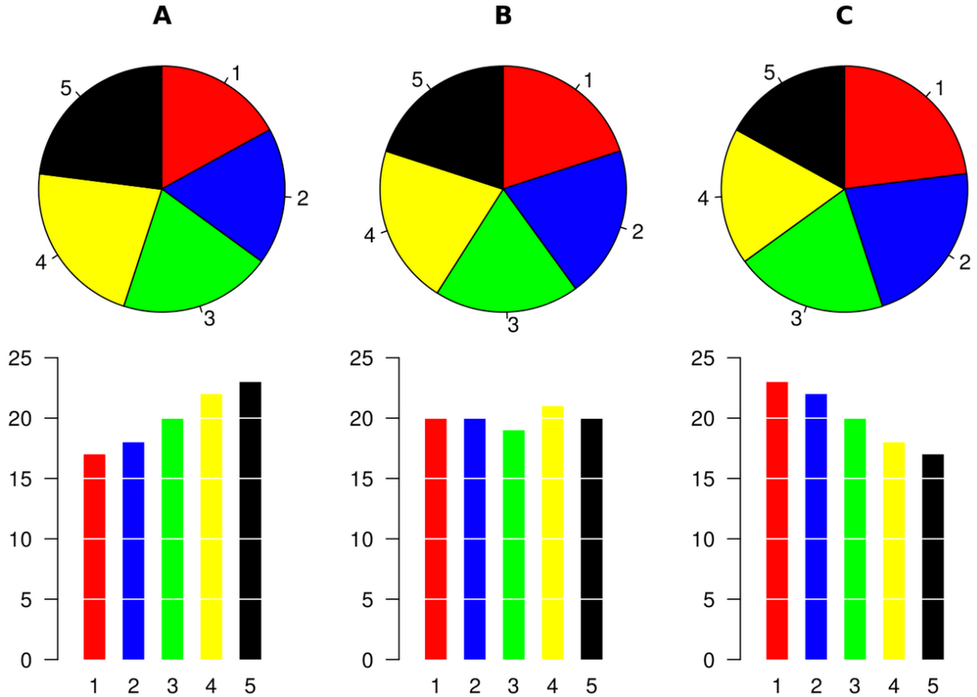
- not so bad for two (or few) levels, for part-to-whole task
- dubious for several levels if details matter

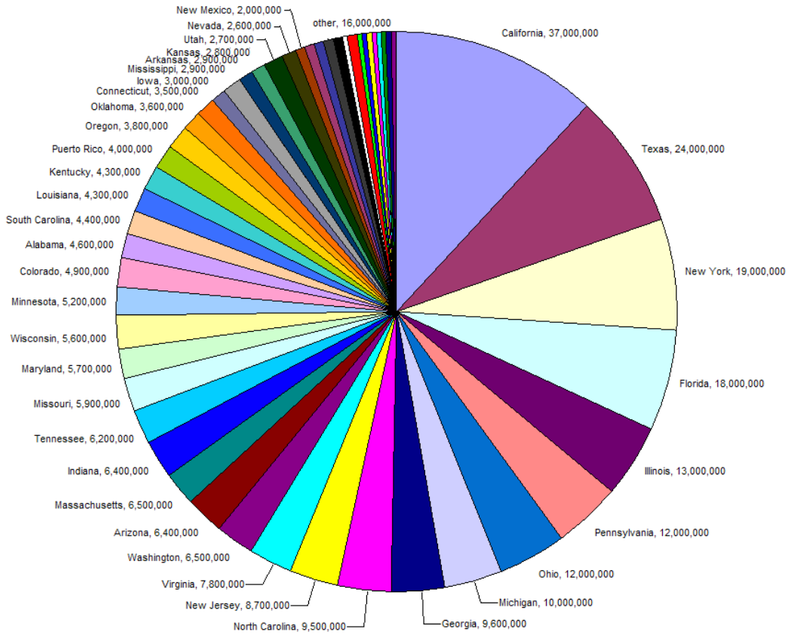


https://eagereyes.org/pie-charts

Pie charts: best practices

- not so bad for two (or few) levels, for part-to-whole task
- dubious for several levels if details matter
- terrible for many levels

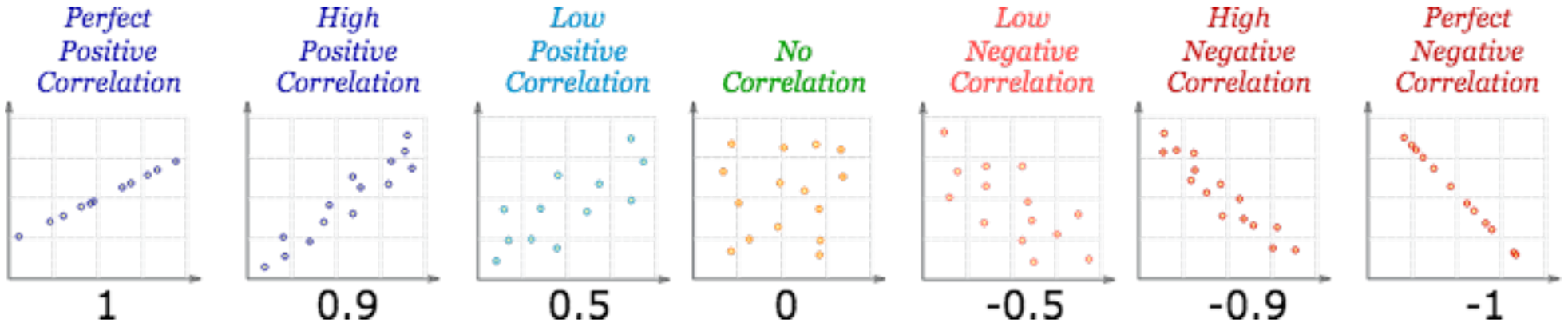




https://eagereyes.org/pie-charts

Scatterplot tasks

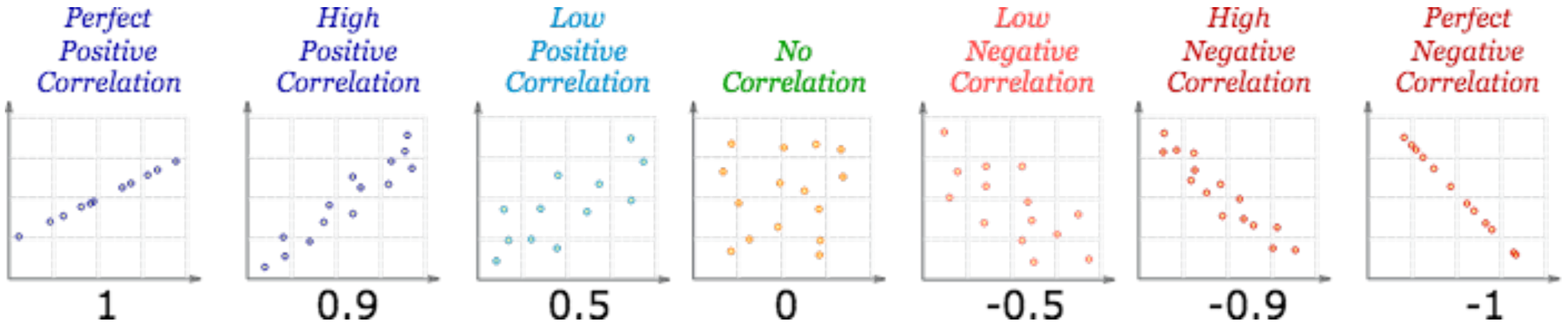
- correlation



https://www.mathsisfun.com/data/scatter-xy-plots.html

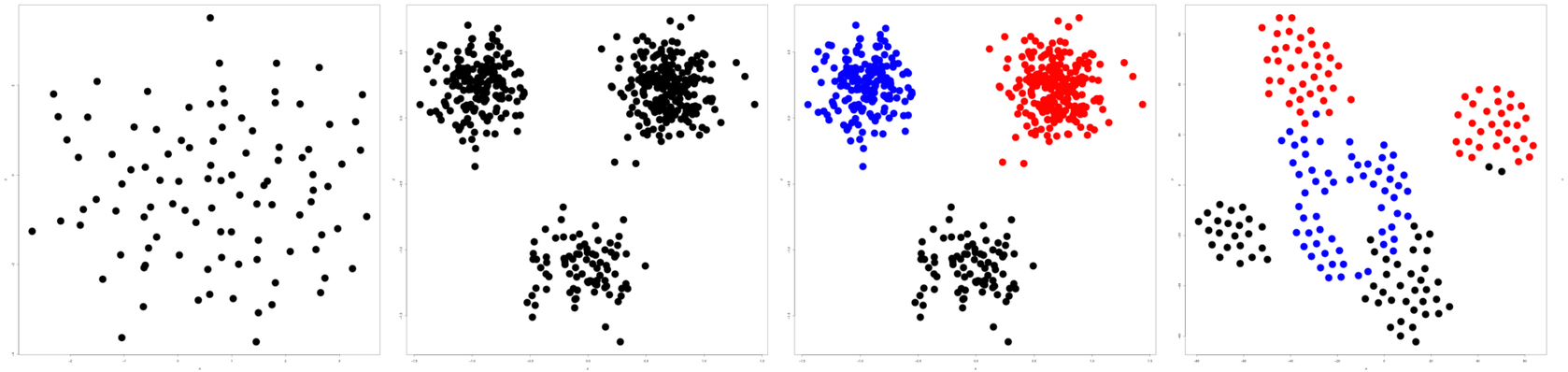
Scatterplot tasks

- correlation



https://www.mathsisfun.com/data/scatter-xy-plots.html

- clusters/groups, and clusters vs classes



https://www.cs.ubc.ca/labs/imager/tr/2014/DRVisTasks/

Idiom: **SPLOM**

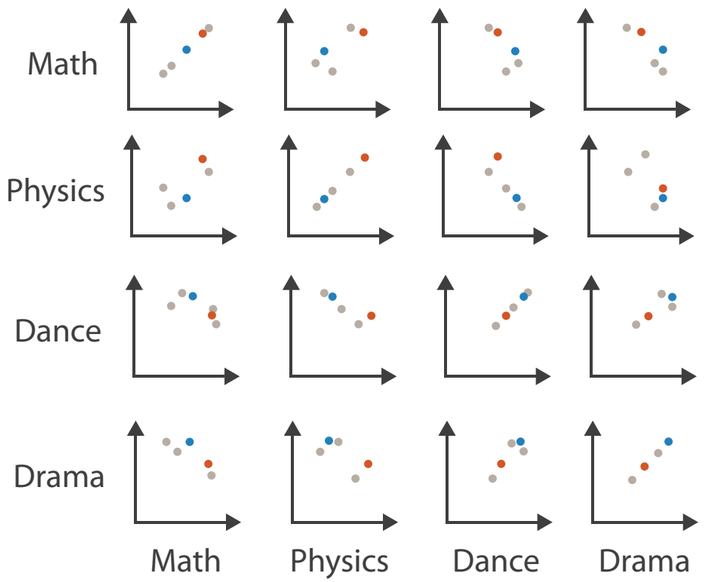
- scatterplot matrix (SPLOM)
  - rectilinear axes, point mark
  - all possible pairs of axes
  - scalability
    - one dozen attribs
    - dozens to hundreds of items

Wilkinson et al., 2005

Idioms: **parallel coordinates**

- scatterplot limitation
  - visual representation with orthogonal axes
  - can show only two attributes with vert & horiz position channels

Scatterplot Matrix



Table

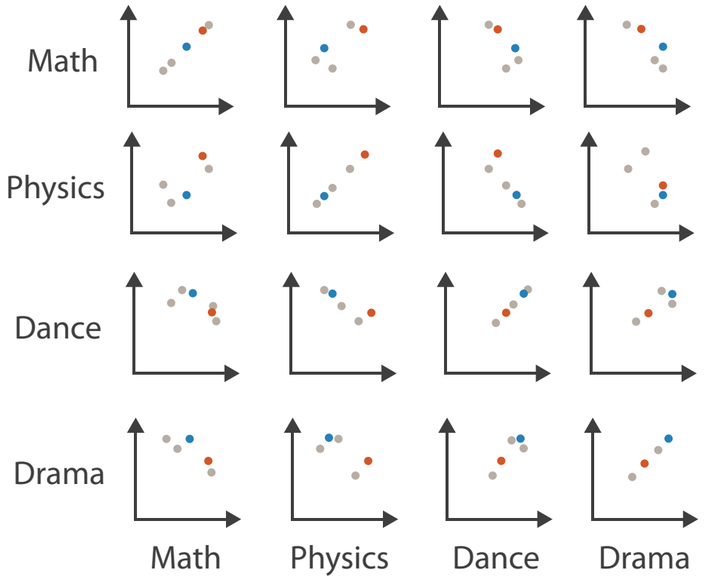
Math	Physics	Dance	Drama
85	95	70	65
90	80	60	50
65	50	90	90
50	40	95	80
40	60	80	90

after [Visualization Course Figures. McGuffin, 2014. http://www.michaelmcguffin.com/courses/vis/]

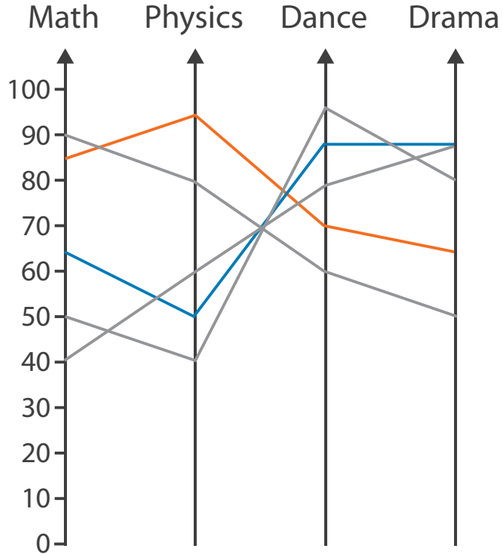
Idioms: **parallel coordinates**

- scatterplot limitation
  - visual representation with orthogonal axes
  - can show only 2 attributes with vert & horiz position channels
- alternative: line up n axes in parallel to show n attributes with (vert) position on each
  - vert position/order for each attribute value
  - item encoded with 1 connecting mark: path through value at each axis
- parallel coordinates
  - parallel axes, path for item
    - jagged, piecewise-linear by segment
    - vs rectilinear/Cartesian axes where item as point
  - scalability
    - dozens of attribs
    - hundreds of items

Scatterplot Matrix



Parallel Coordinates



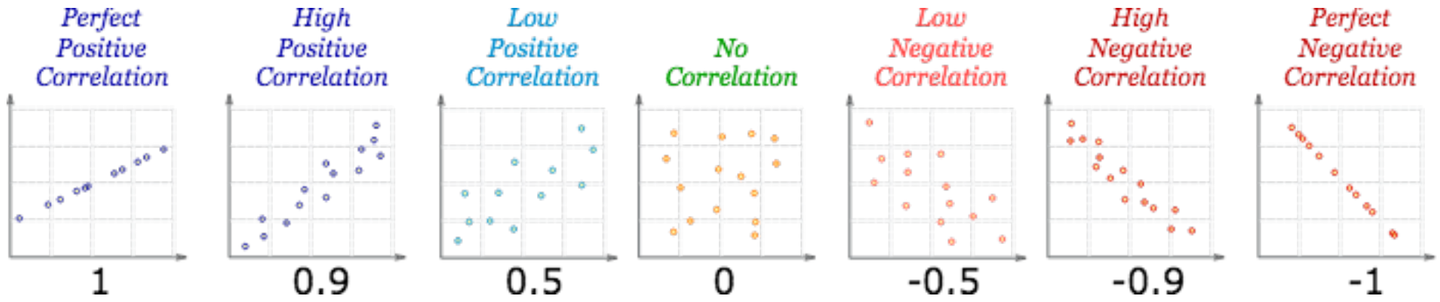
Table

Math	Physics	Dance	Drama
85	95	70	65
90	80	60	50
65	50	90	90
50	40	95	80
40	60	80	90

after [Visualization Course Figures. McGuffin, 2014. http://www.michaelmcguffin.com/courses/vis/]

Task: Correlation

- scatterplot matrix
  - positive correlation
    - diagonal low-to-high
  - negative correlation
    - diagonal high-to-low
  - uncorrelated: spread out
- parallel coordinates
  - positive correlation
    - parallel line segments
  - negative correlation
    - all segments cross at halfway point
  - uncorrelated
    - scattered crossings



https://www.mathsisfun.com/data/scatter-xy-plots.html

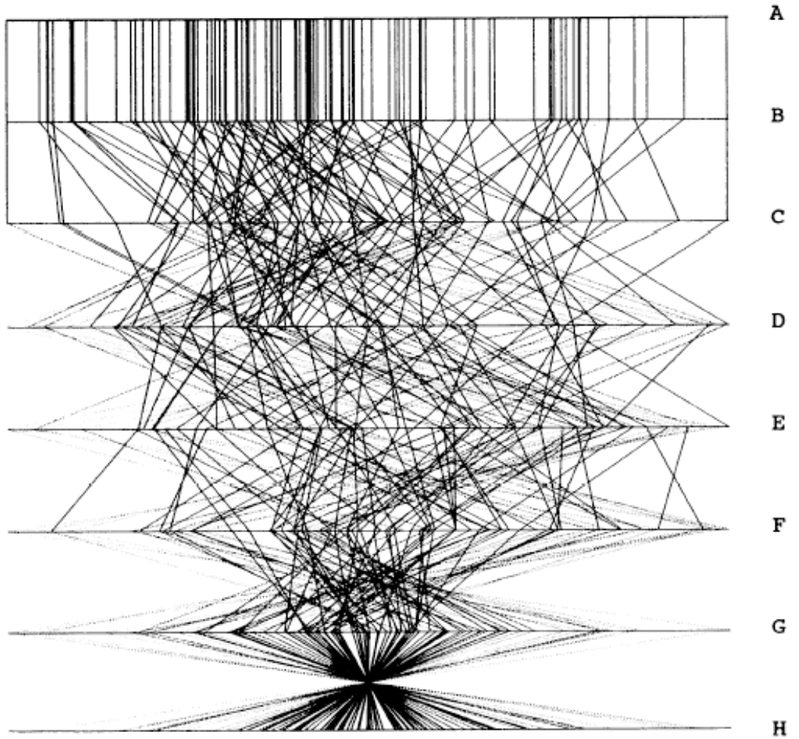
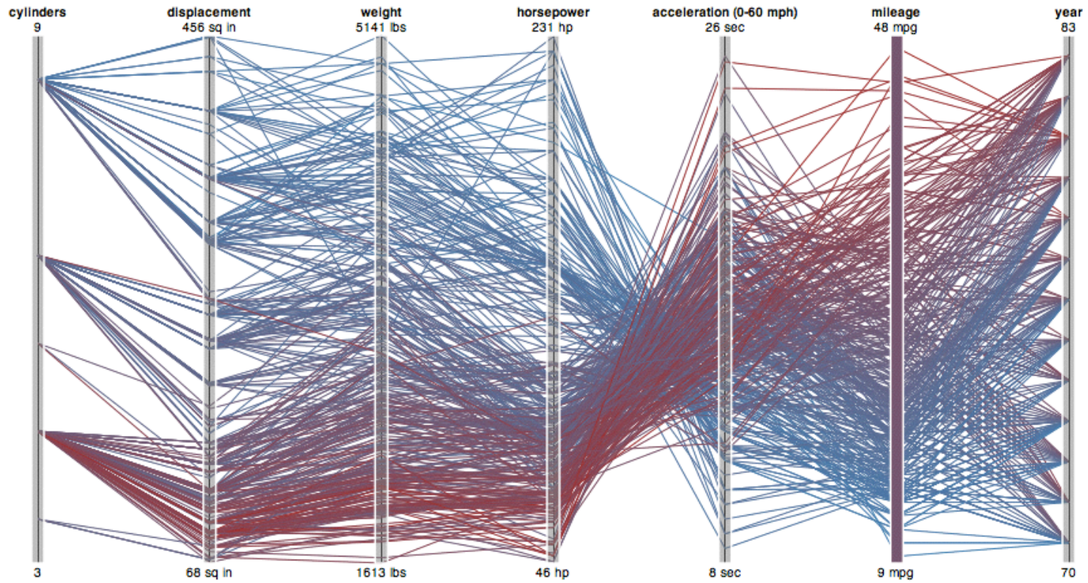


Figure 3. Parallel Coordinate Plot of Six-Dimensional Data Illustrating Correlations of $\rho = 1, .8, .2, 0, -.2, -.8,$ and -1.

[Hyperdimensional Data Analysis Using Parallel Coordinates. Wegman. Journ. American Statistical Association 85:411 (1990), 664–675.]

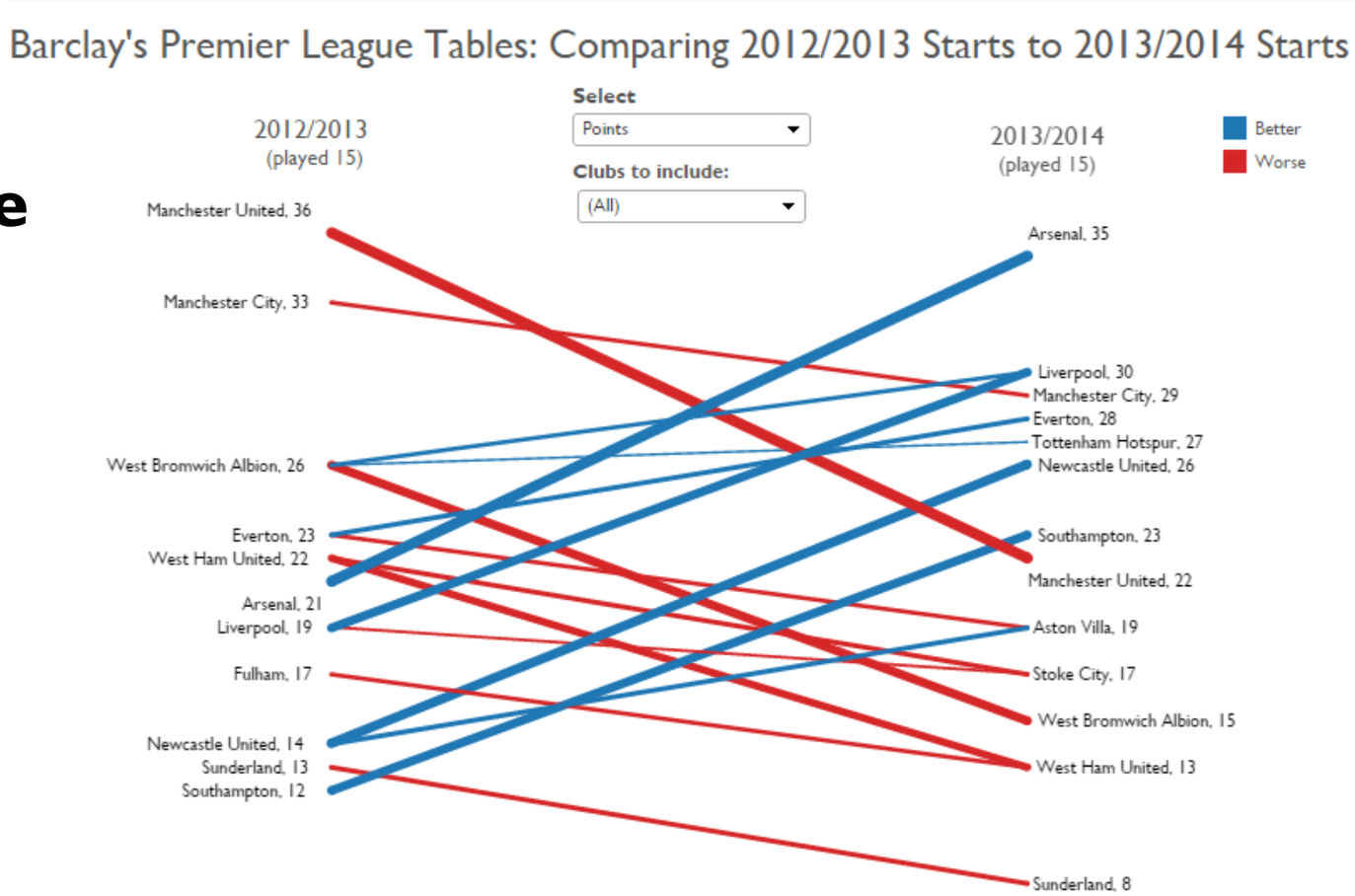
Parallel coordinates, limitations

- visible patterns only between neighboring axis pairs
- how to pick axis order?
  - usual solution: reorderable axes, interactive exploration
  - same weakness as many other techniques
    - downside of interaction: human-powered search
  - some algorithms proposed, none fully solve



Idiom: **Slopegraphs**

- **parallel coordinates special case** with only 2 axes
  - mark: segment
    - connecting mark between 2 values
  - channels
    - vertical pos/order, 1 axis per attrib: express value
    - segment length (1D size, *horiz*): unavailable
    - segment width (1D size, *vert*): change magnitude
    - color
  - task
    - emphasize changes in rank/value

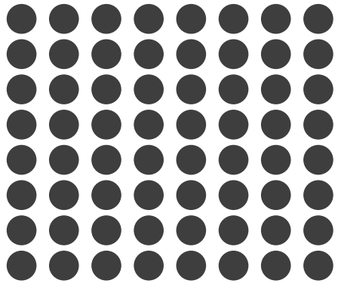


https://public.tableau.com/profile/ben.jones#!/vizhome/Slopegraphs/Slopegraphs

Layout density

Layout Density

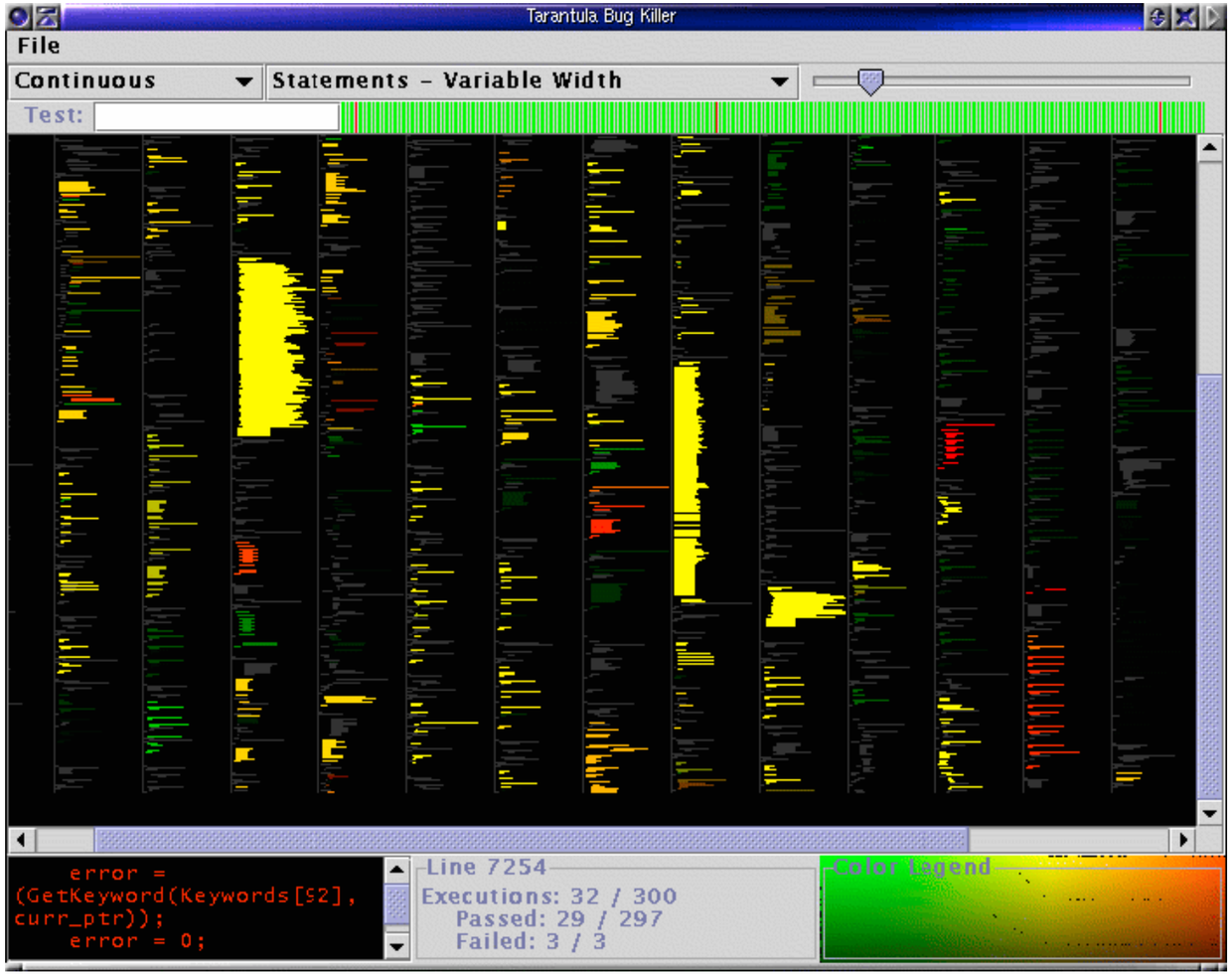
- Dense



- Space-Filling

Idiom: Dense software overviews

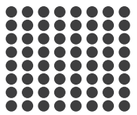
- data: text (LOC)
  - 1 quant attrib: test results
- derived data:
  - 2 quant attribs: min & max char pos
- mark: segment
- channels:
  - color by test results
  - within column, level 1
    - vert pos/order, 1 value: LOC order in program
    - vert shared: yes
    - vert size: **unavailable** (1 pixel high, by idiom design)
    - horiz pos/order, 2 values: min/max char
    - horiz length (1D size): extent
      - **not aligned**, extent != max
  - across columns (global), level 2
    - scanline wrap to fit long vertical line into square
- scalability: 10K+ lines



[Visualization of test information to assist fault localization. Jones, Harrold, Stasko. Proc. ICSE 2002, p 467-477.]

Layout Density

→ Dense



Mini-Lecture: Manipulate

How to handle complexity: 1 previous strategy

➔ *Derive*

- derive new data to show within view

How to handle complexity: 1 previous strategy + 1 more

➔ *Derive*

- derive new data to show within view
- change view over time

Manipulate

➔ **Change**

➔ **Select**

➔ **Navigate**

Manipulate View

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

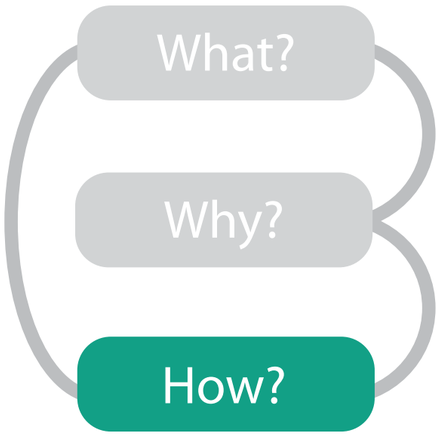
- Change
- Select
- Navigate

Facet

- Juxtapose
- Partition
- Superimpose

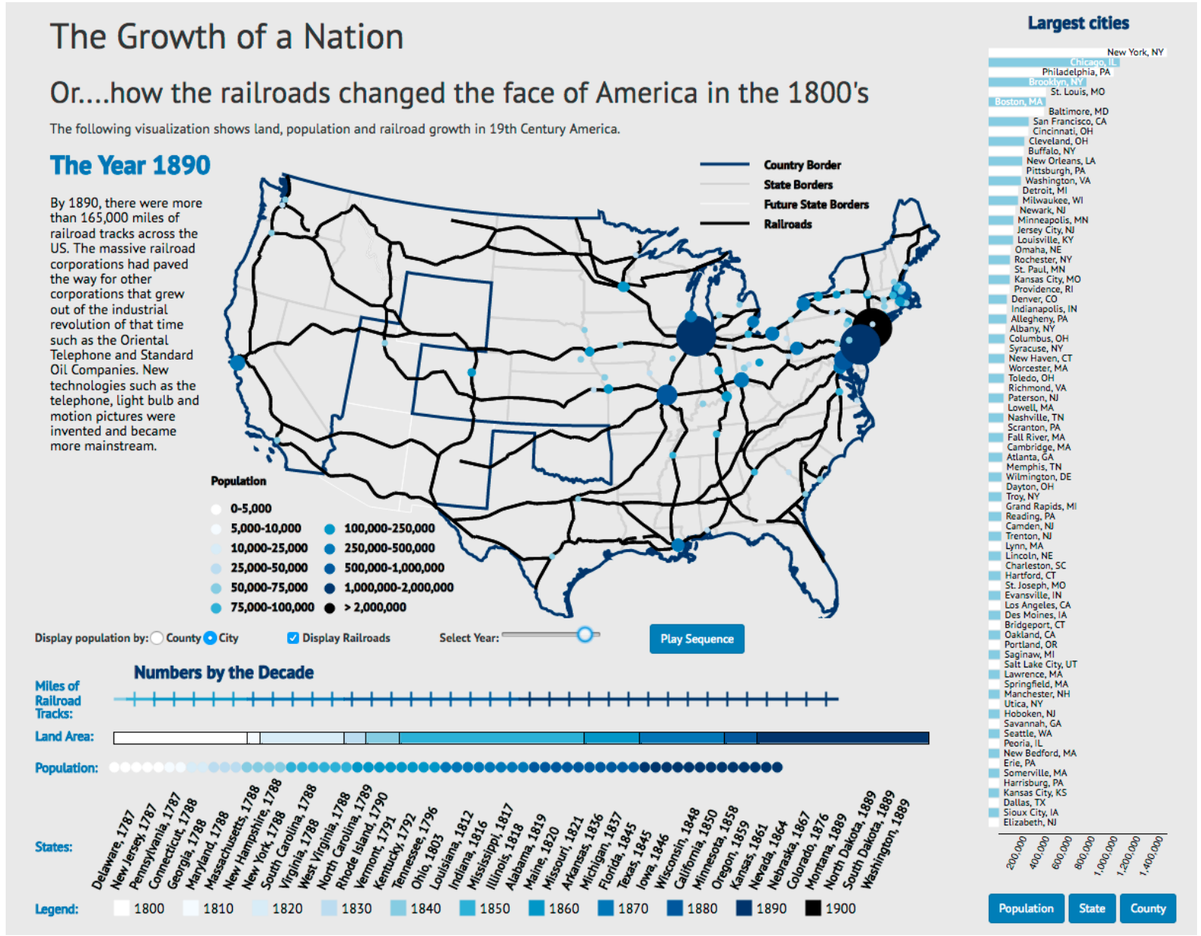
Reduce

- Filter
- Aggregate
- Embed



Idiom: **Change parameters**

- widgets and controls
  - sliders, buttons, radio buttons, checkboxes, dropdowns/comboboxes
- pros
  - clear affordances, self-documenting (with labels)
- cons
  - uses screen space
- design choices
  - separated vs interleaved
    - controls & canvas

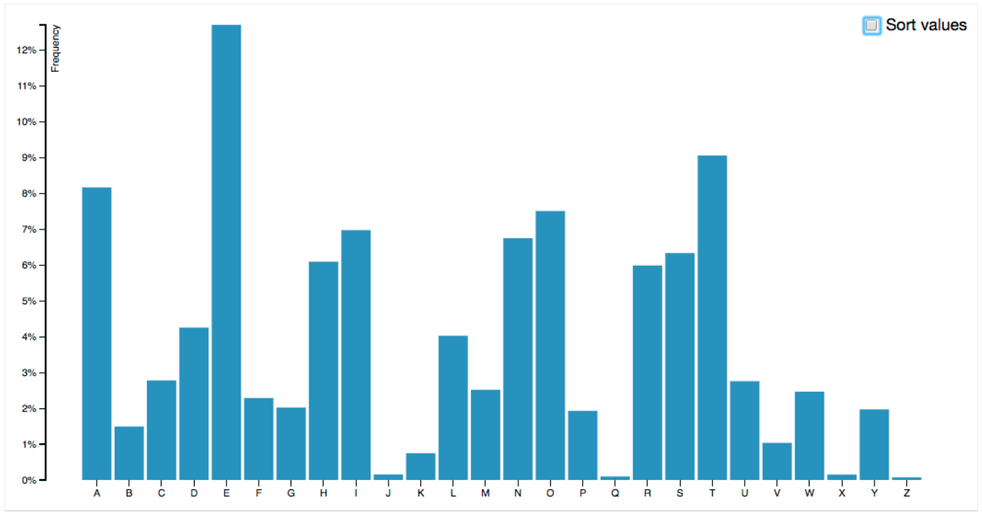


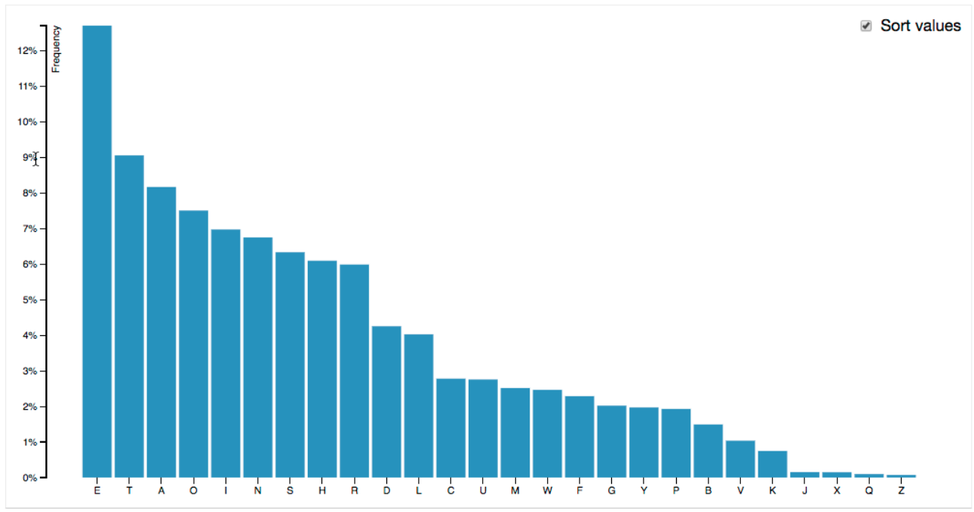
[Growth of a Nation](http://laurenwood.github.io/)

made with D3

Idiom: **Change order/arrangement**

- what: simple table
- how: data-driven reordering
- why: find extreme values, trends





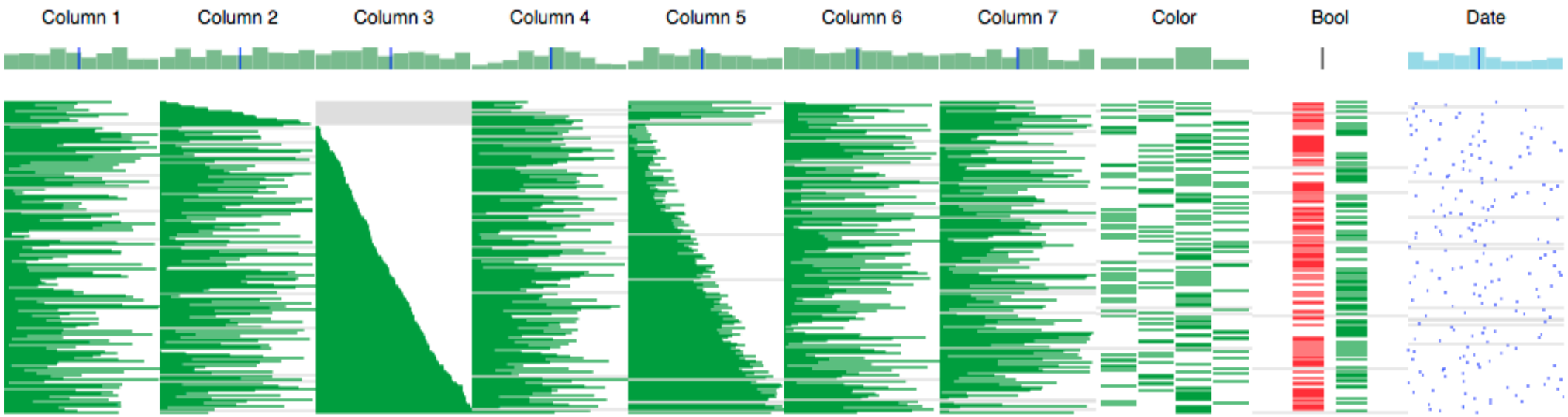
[Sortable Bar Chart] https://observablehq.com/@d3/sortable-bar-chart

made with D3

Idiom: **Reorder**

System: **DataStripes**

- what: table with many attributes
- how: data-driven reordering by selecting column
- why: find correlations between attributes

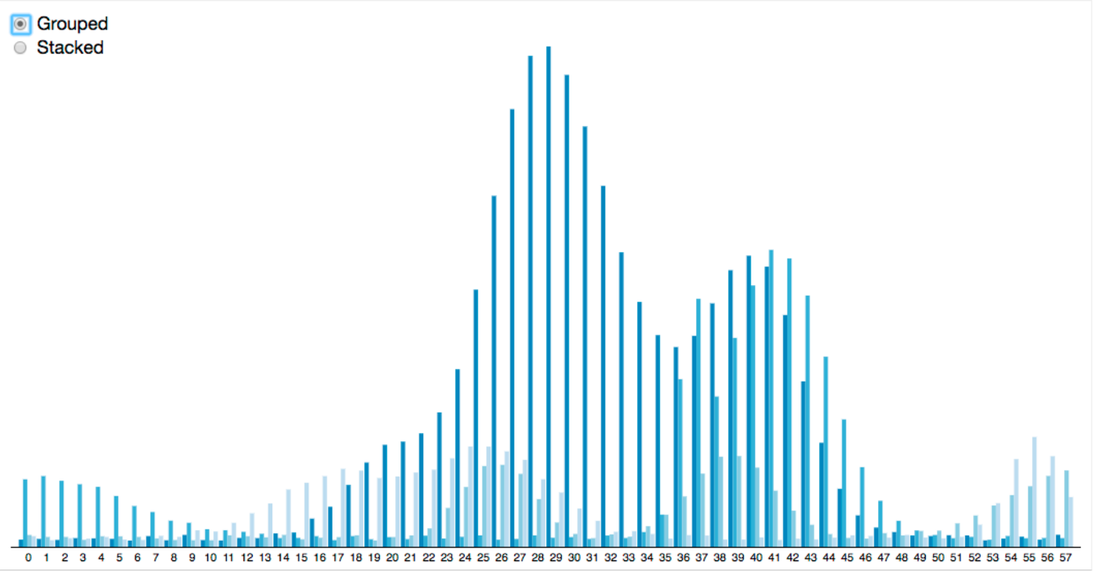


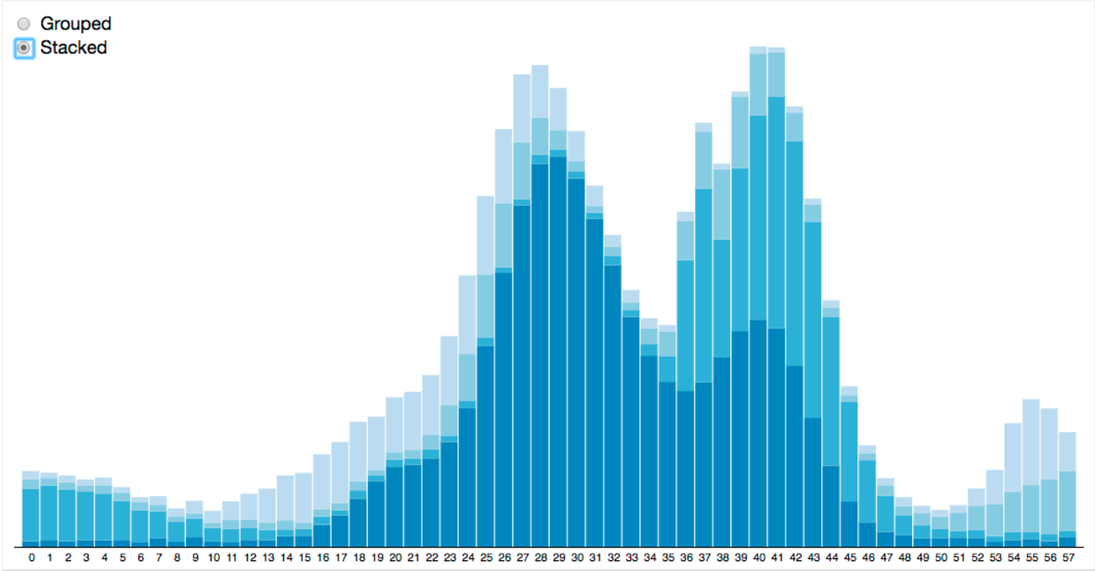
[http://carlmanaster.github.io/datastripes/]

made with D3

Idiom: **Animated transitions - visual encoding**

- smooth transition from one state to another
  - alternative to jump cuts, supports item tracking
    - best case for animation
  - staging to reduce cognitive load

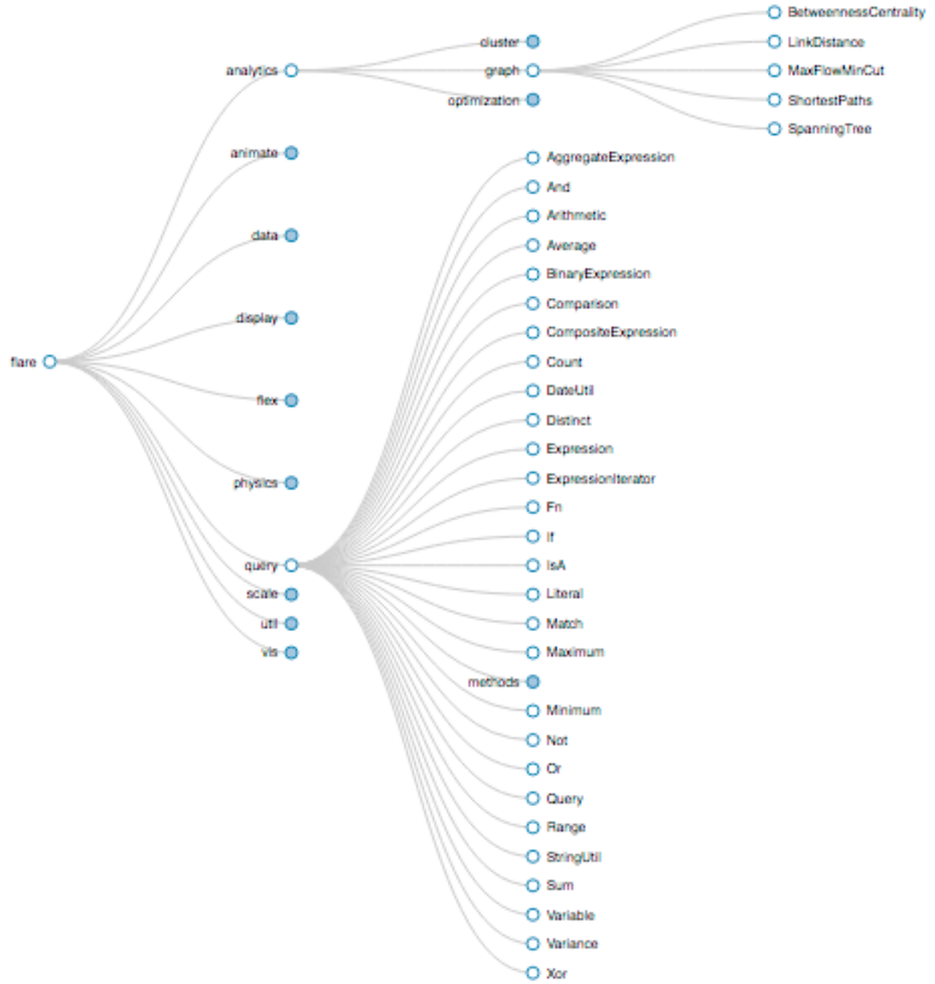




[Stacked to Grouped Bars] https://observablehq.com/@d3/stacked-to-grouped-bars

Idiom: **Animated transition - tree detail**

- animated transition
  - network drilldown/rollup

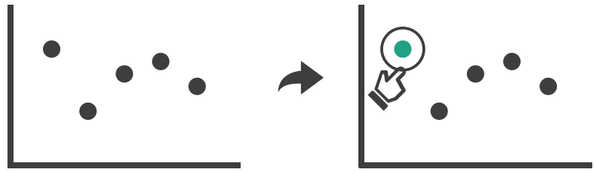


[Collapsible Tree] https://observablehq.com/@d3/collapsible-tree

Manipulate

- Change over Time

- Select



Interaction technology

- what do you design for?
  - mouse & keyboard on desktop?
    - large screens, hover, multiple clicks
  - touch interaction on mobile?
    - small screens, no hover, just tap
  - gestures from video / sensors?
    - ergonomic reality vs movie bombast
  - eye tracking?

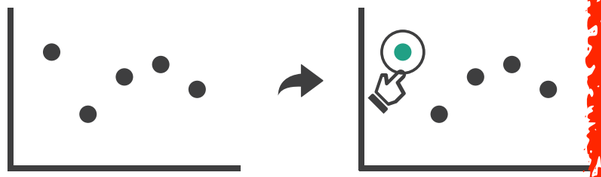
Data visualization and the news - Gregor Aisch (37 min)
vimeo.com/182590214

I Hate Tom Cruise - Alex Kauffmann (5 min)
www.youtube.com/watch?v=QXLfT9sFcbc

Manipulate

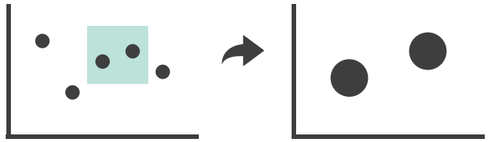
Change over Time

Select



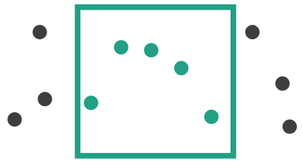
Navigate

- *Zoom*
  Geometric



- *Pan/Translate*

- *Constrained*



Navigate: Changing viewpoint/visibility

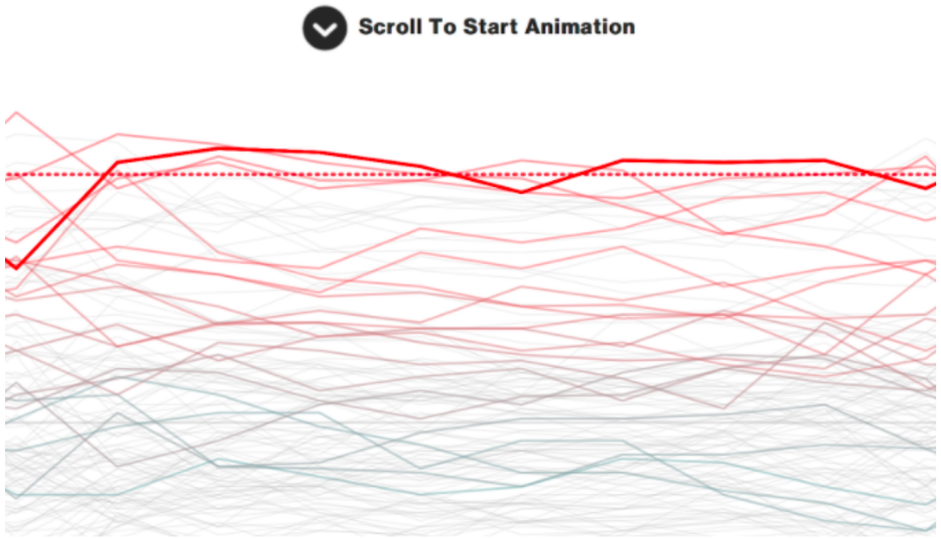
- change viewpoint
  - changes which items are visible within view
- camera metaphor
  - pan/translate/scroll
    - move up/down/sideways

Navigate

→ *Pan/Translate*

Idiom: **Scrollytelling**

- how: navigate page by scrolling (panning down)
- pros:
  - familiar & intuitive, from standard web browsing
  - linear (only up & down) vs possible overload of click-based interface choices
- cons:
  - full-screen mode may lack affordances
  - scrolljacking, no direct access
  - unexpected behaviour
  - continuous control for discrete steps



[How to Scroll, Bostock](https://bost.ocks.org/mike/scroll/)

https://eagereyes.org/blog/2016/the-scrollytelling-scourge

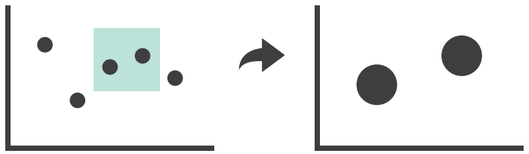
Navigate: Unconstrained vs constrained

- unconstrained navigation
  - easy to implement for designer
  - hard to control for user
    - easy to overshoot/undershoot
- constrained navigation
  - typically uses animated transitions
  - trajectory automatically computed based on selection
    - just click; selection ends up framed nicely in final viewport

Navigate

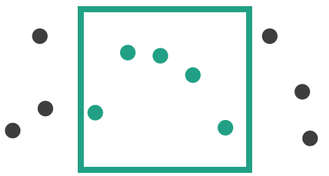
→ Item Reduction

→ *Zoom*
Geometric or *Semantic*



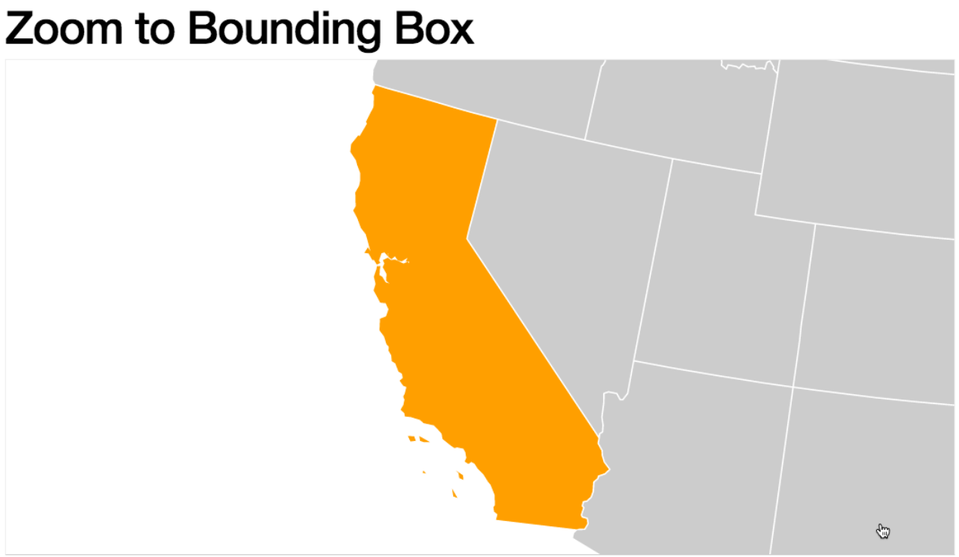
→ *Pan/Translate*

→ *Constrained*



Idiom: **Animated transition + constrained navigation**

- example: geographic map
  - simple zoom, only viewport changes, shapes preserved



[Zoom to Bounding Box] https://observablehq.com/@d3/zoom-to-bounding-box

Interaction benefits

- interaction pros
  - major advantage of computer-based vs paper-based visualization
  - flexible, powerful, intuitive
    - exploratory data analysis: change as you go during analysis process
    - fluid task switching: different visual encodings support different tasks
  - animated transitions provide excellent support
    - empirical evidence that animated transitions help people stay oriented

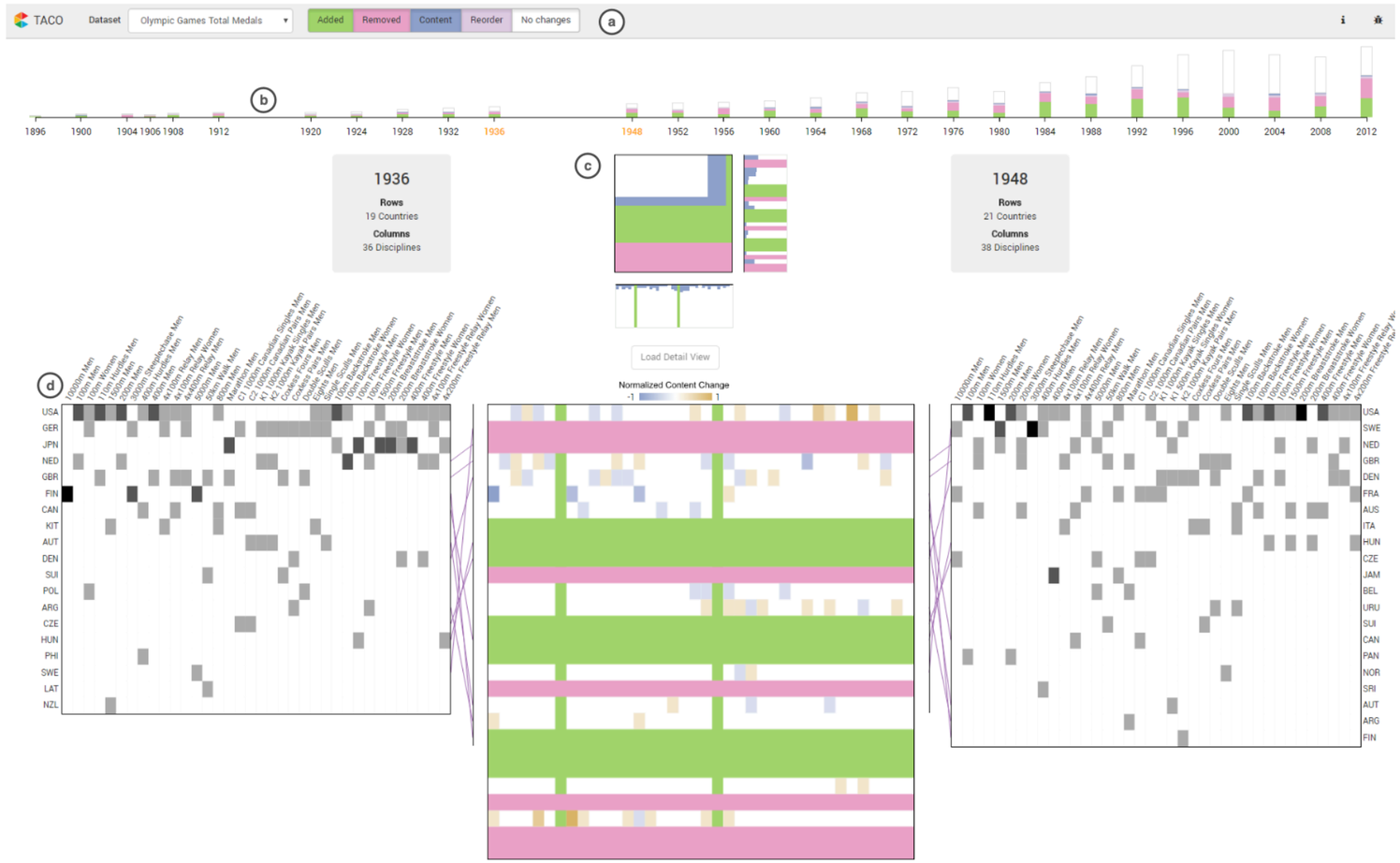
Interaction limitations

- interaction has a time cost
  - sometimes minor, sometimes significant
  - degenerates to human-powered search in worst case
- remembering previous state imposes cognitive load
- controls may take screen real estate
  - or invisible functionality may be difficult to discover (lack of affordances)
- users may not interact as planned by designer
  - NYTimes logs show ~90% don't interact beyond scrollytelling - Aisch, 2016

TACO

TACO: Visualizing Changes in Tables over Time

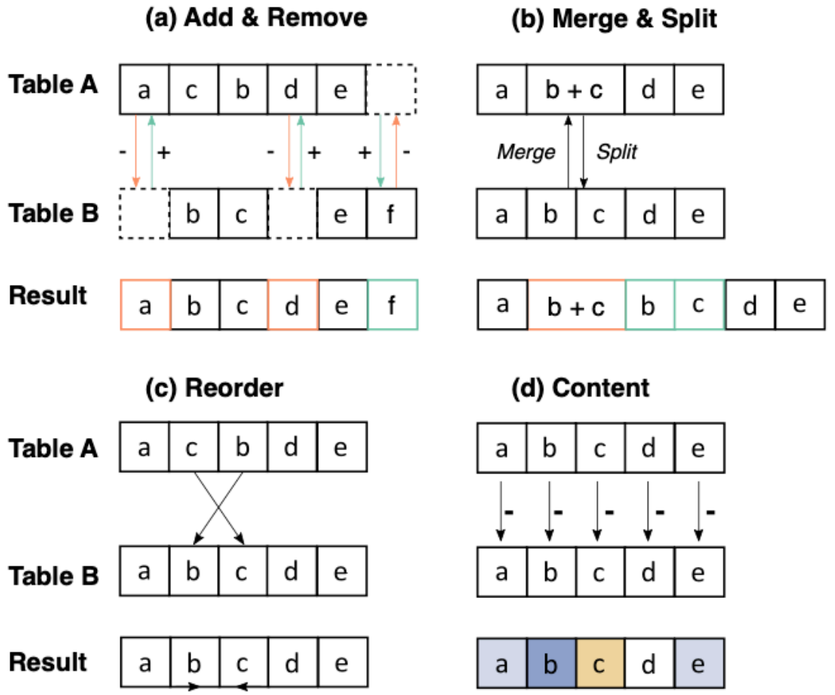
- TAble COmparison
  - technique paper
- change types
  - add, remove, change, move
- overview: timeline with stacked bars, pick 2
- high-level: aggregated changes in ratio chart + histos
- low-level: raw & diff heat maps



[TACO: Visualizing Changes in Tables Over Time. Niederer, Stitz, Hourieh, Grassinger, Aigner, Streit. IEEE TVCG (InfoVis '17), 24(1): 677-686]

TACO tasks & data

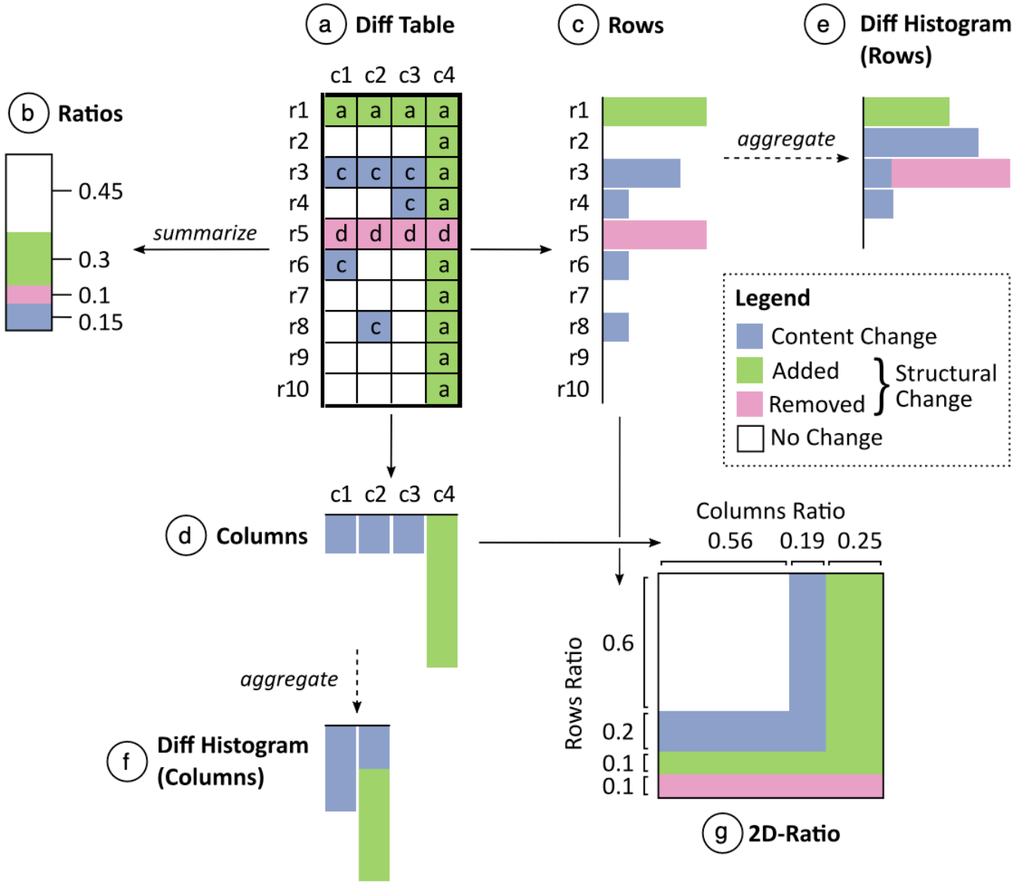
- tasks (explicitly identified, even though technique paper)
  - T1: identify types of changes
    - structural, content, reorder, merge/split
  - T2: compare multiple table versions over time
  - T3: compare one table at two specific time points
  - T4: present raw data tables and meta-data
- data
  - homogeneous table (matrix)
  - union of all rows/columns

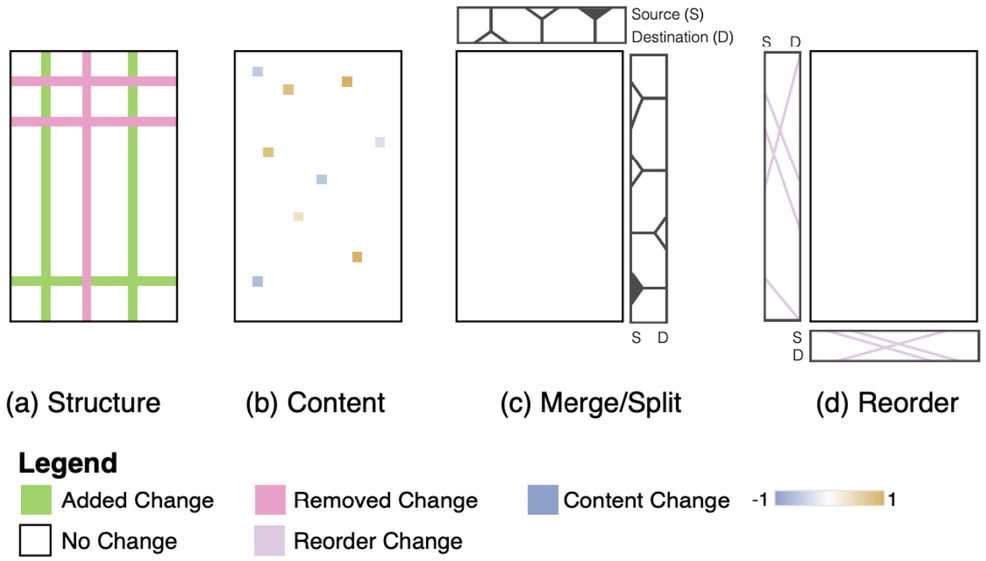


[TACO: Visualizing Changes in Tables Over Time. Niederer, Stitz, Hourieh, Grassinger, Aigner, Streit. IEEE TVCG (InfoVis '17), 24(1): 677-686]

Visual encoding

- 2D ratio chart
  - derived data
    - diff table
  - aggregation (Ch 12 next week)
- diff heatmap
  - tie visual results to support for specific tasks





[TACO: Visualizing Changes in Tables Over Time. Niederer, Stitz, Hourieh, Grassinger, Aigner, Streit. IEEE TVCG (InfoVis '17), 24(1): 677-686]

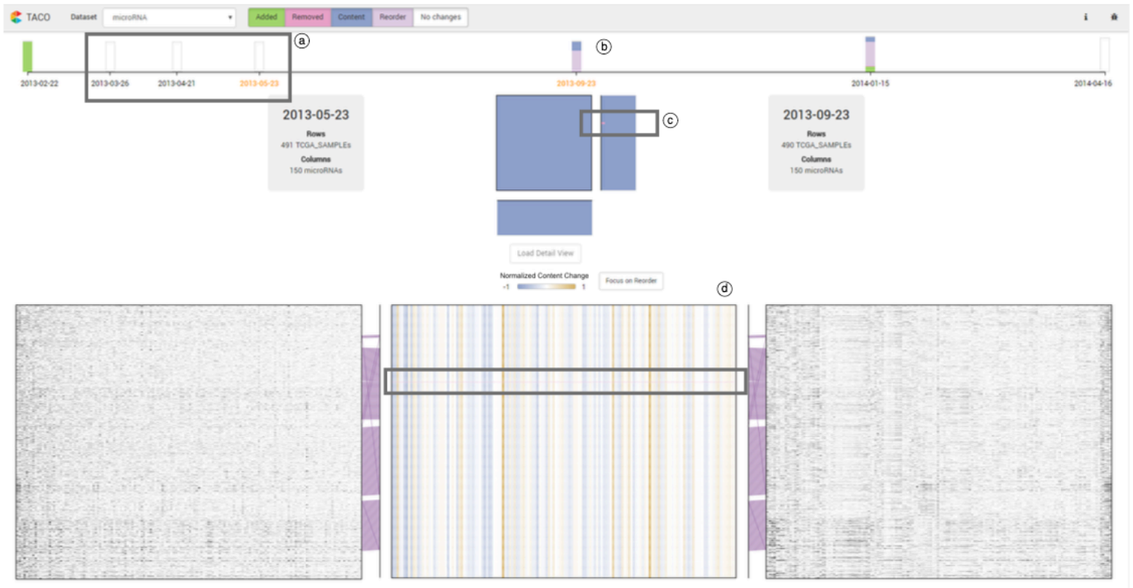
Visual encoding: explanation / justification / rationale

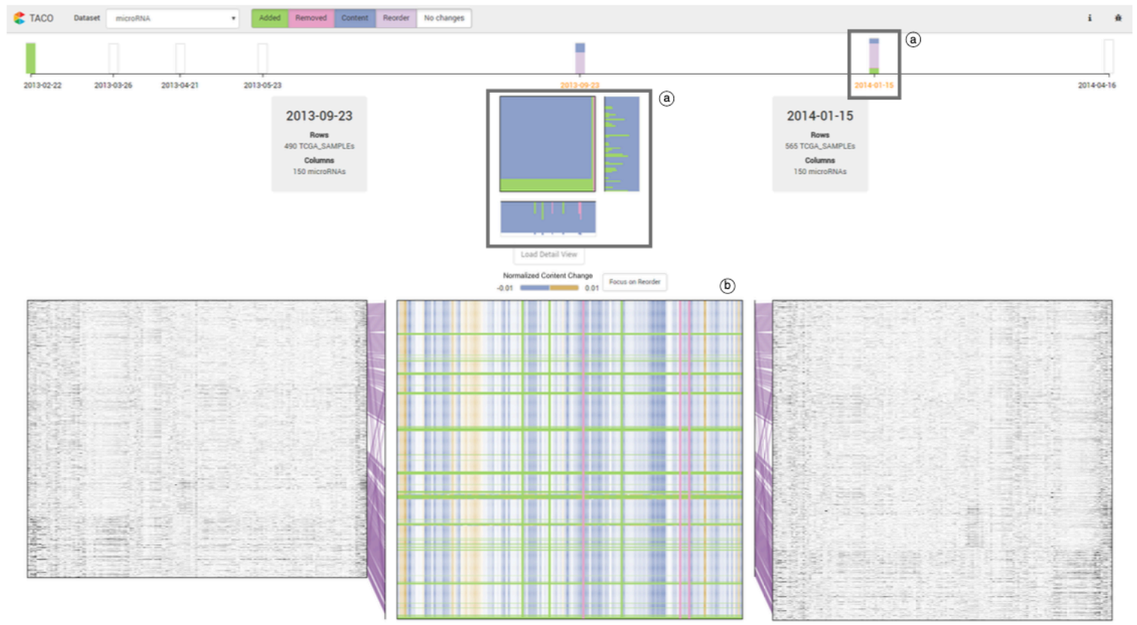
- explain low-level design decisions & justify choices
- explicit discussion of alternatives wrt tasks
  - vertical stacked bars (vs star plots or horizontal bars) to show change over time (T2)
  - color for quant differences (vs size/length: unavailable given idiom choices)
  - slope graphs (vs color: already many in use) for merge

- you will need to do this kind of justification in your projects

Validation

- usage scenarios (created by authors)
  - vs case study (created by target users)
    - claim that draws on feedback session from target user
    - different levels of evidence of utility / validity
  - vs formal comparative evaluation
    - argument: previous tools do not scale, so pointless to show
- detailed walkthrough of how visual encoding choices yield results
  - you will need to do this for your projects





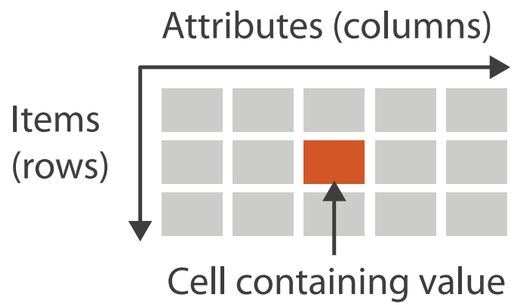
Backup Slides

Tables

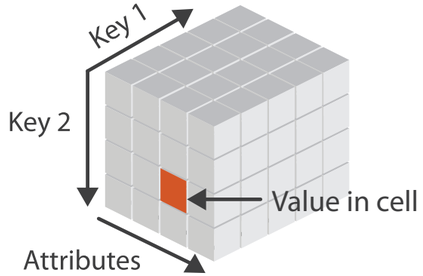
Focus on Tables

Dataset Types

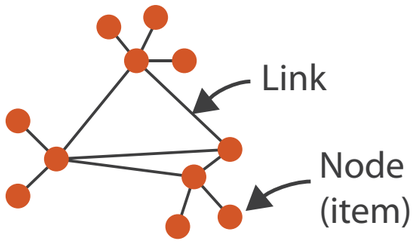
Tables



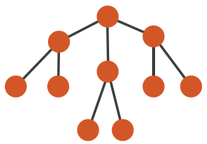
Multidimensional Table



Networks

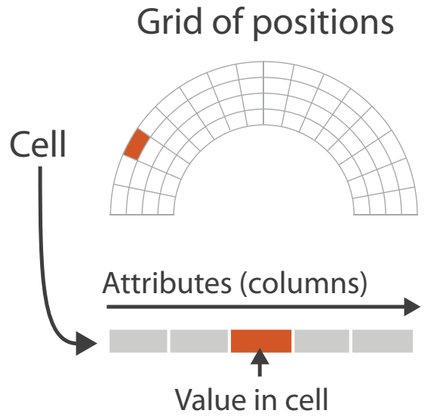


Trees

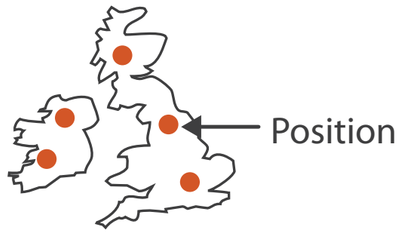


Spatial

Fields (Continuous)



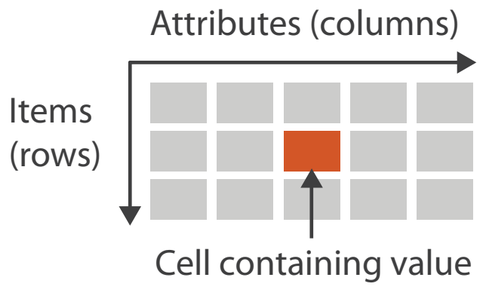
Geometry (Spatial)



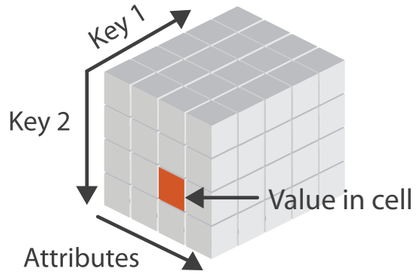
Keys and values

- key
  - independent attribute
  - used as unique index to look up items
  - simple tables: 1 key
  - multidimensional tables: multiple keys
- value
  - dependent attribute, value of cell

➔ Tables



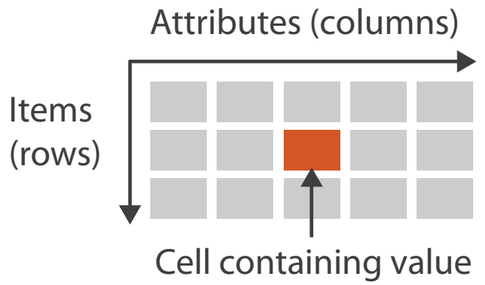
➔ *Multidimensional Table*



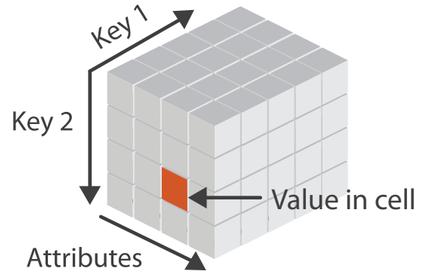
Keys and values

- key
  - independent attribute
  - used as unique index to look up items
  - simple tables: 1 key
  - multidimensional tables: multiple keys
- value
  - dependent attribute, value of cell
- classify arrangements by keys used
  - 0, 1, 2, ...

➔ Tables



➔ *Multidimensional Table*



➔ *0 Keys*

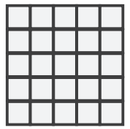
➔ **Express Values**

➔ *1 Key*

List

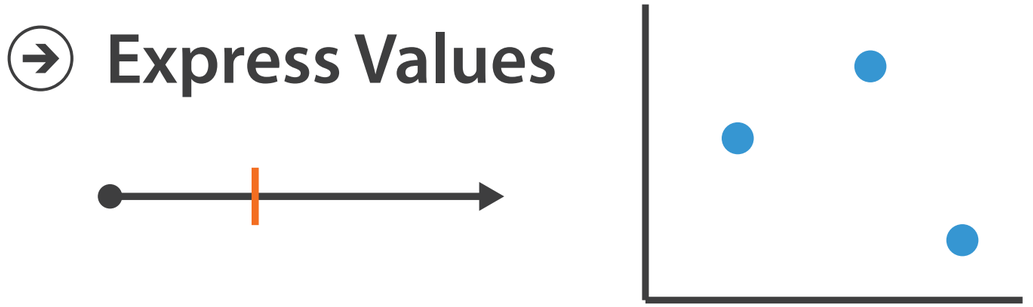
➔ *2 Keys*

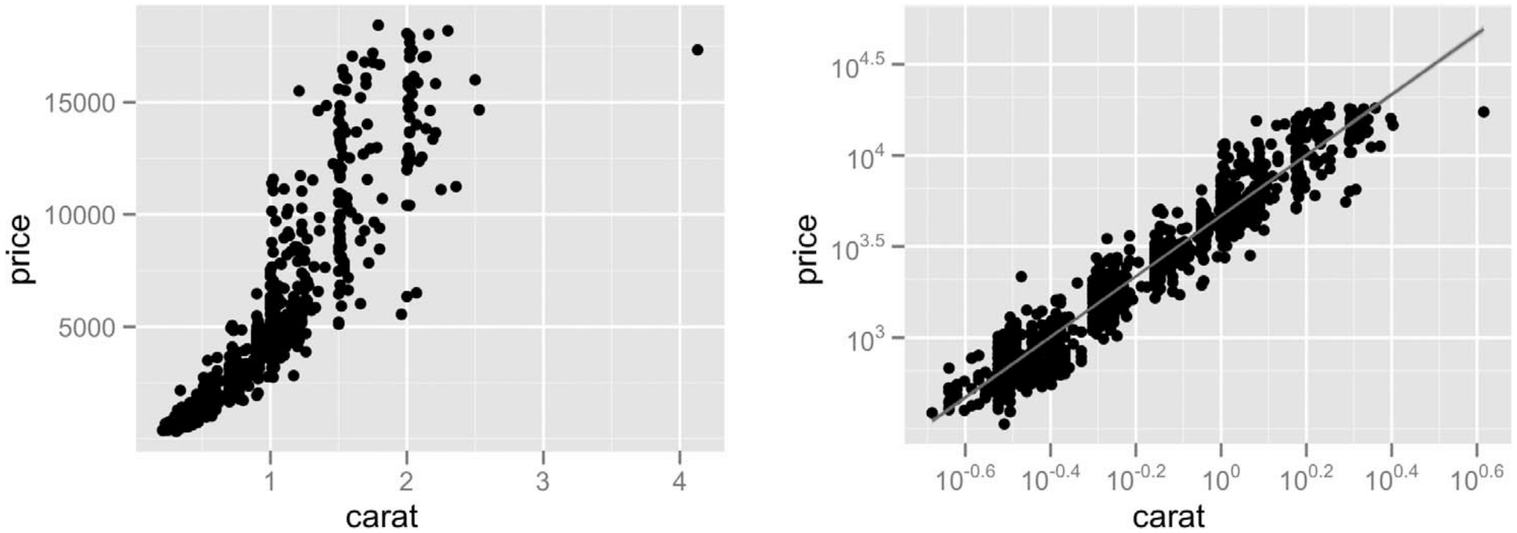
Matrix



Idiom: **scatterplot**

- **express** values (magnitudes)
  - quantitative attributes
- no keys, only values

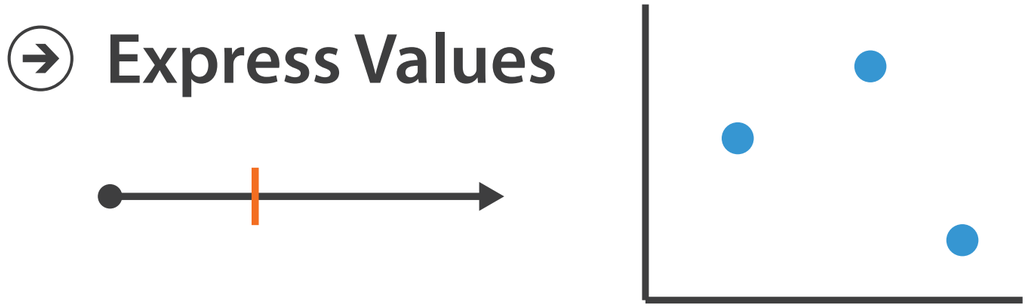


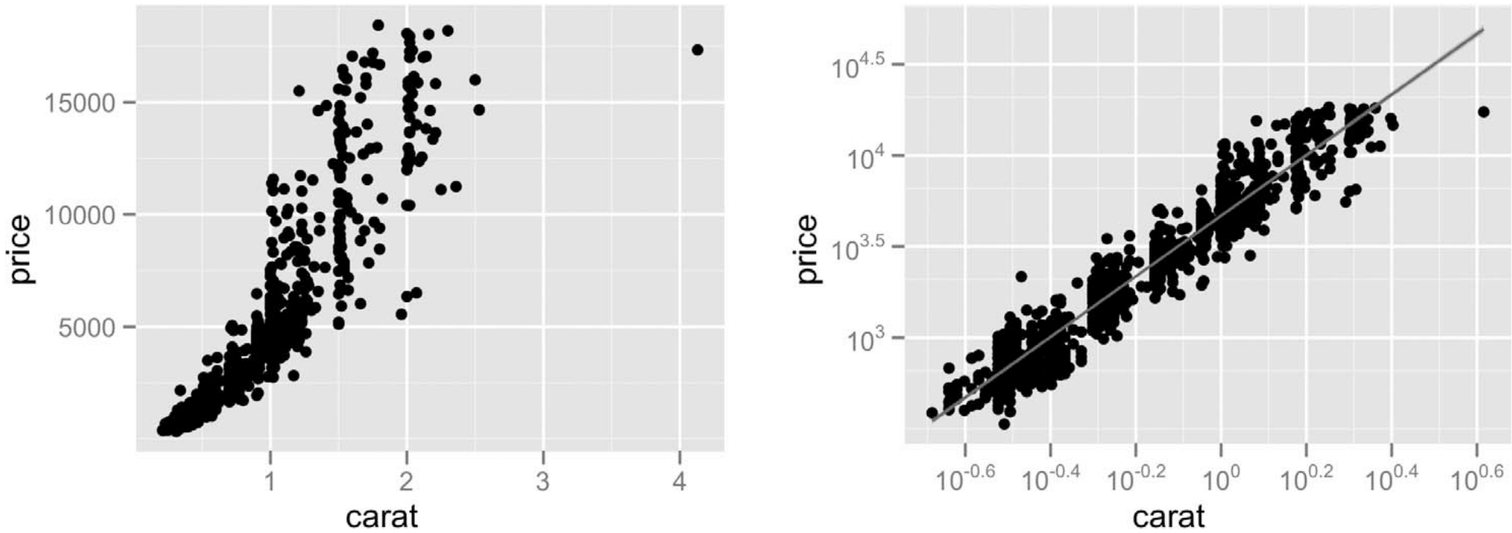


[A layered grammar of graphics. Wickham. Journ. Computational and Graphical Statistics 19:1 (2010), 3–28.]

Idiom: **scatterplot**

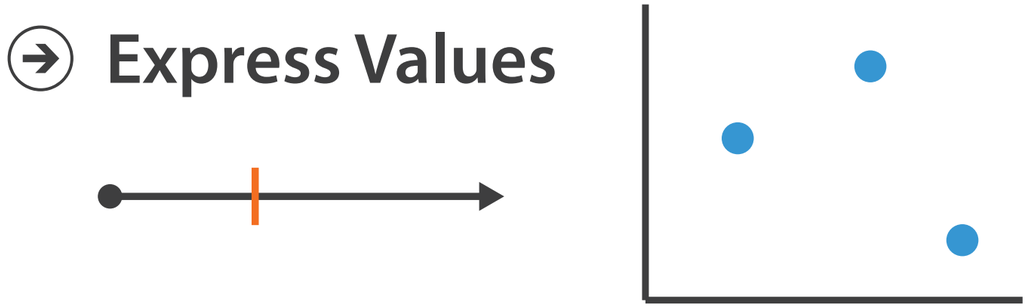
- **express** values (magnitudes)
  - quantitative attributes
- no keys, only values
  - data
    - 2 quant attribs
  - mark: points
  - channels
    - horiz position/order
    - vert position/order



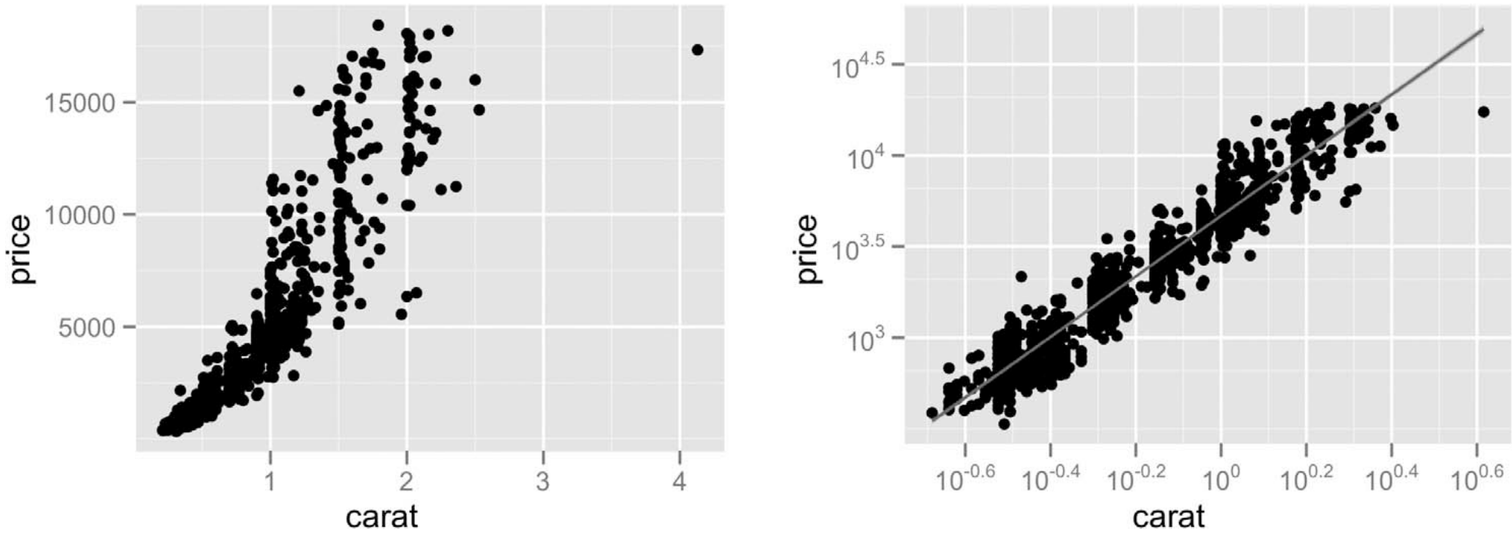


[A layered grammar of graphics. Wickham. Journ. Computational and Graphical Statistics 19:1 (2010), 3–28.]

Idiom: **scatterplot**



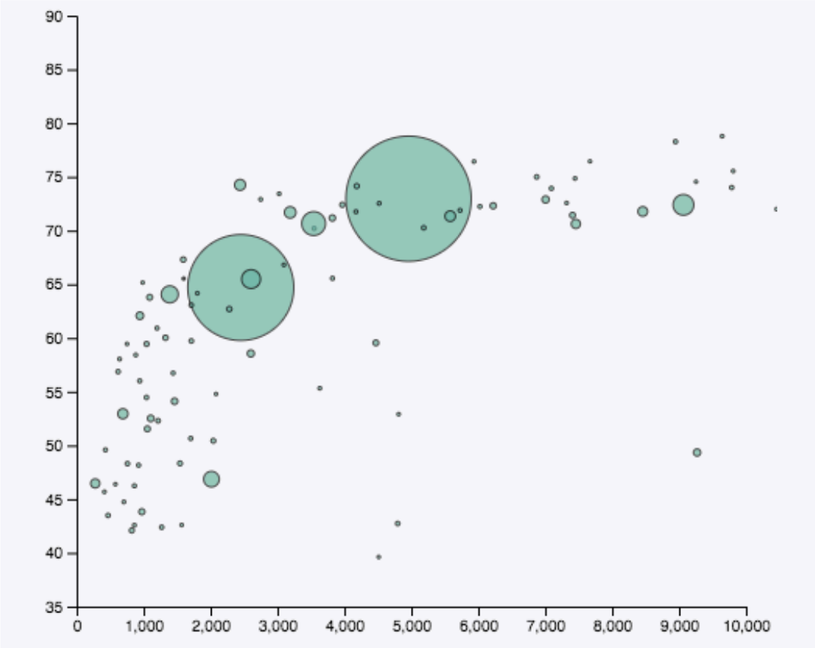
- **express** values (magnitudes)
  - quantitative attributes
- no keys, only values
  - data
    - 2 quant attribs
  - mark: points
  - channels
    - horiz position/order
    - vert position/order
  - tasks
    - find trends, outliers, distribution, correlation, clusters
  - scalability
    - 1K items

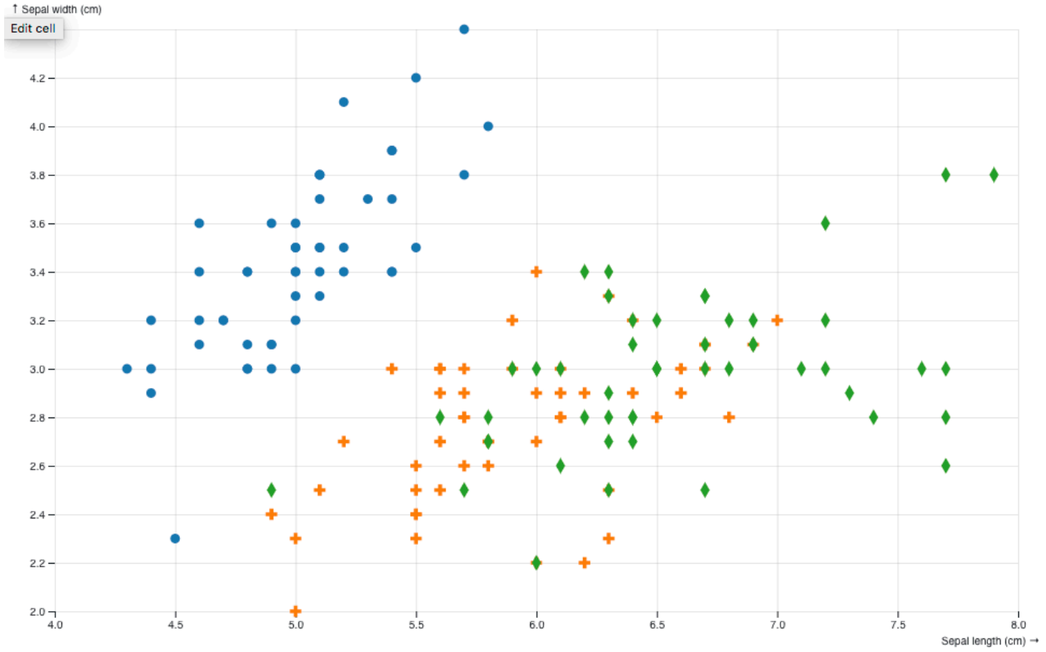


[A layered grammar of graphics. Wickham. Journ. Computational and Graphical Statistics 19:1 (2010), 3–28.]

Scatterplots: Encoding more channels

- additional channels viable since using point marks
  - color
  - 2D size (1 quant attribute, used to control 2D area)
    - note radius would mislead, take square root since area grows quadratically
  - symbol



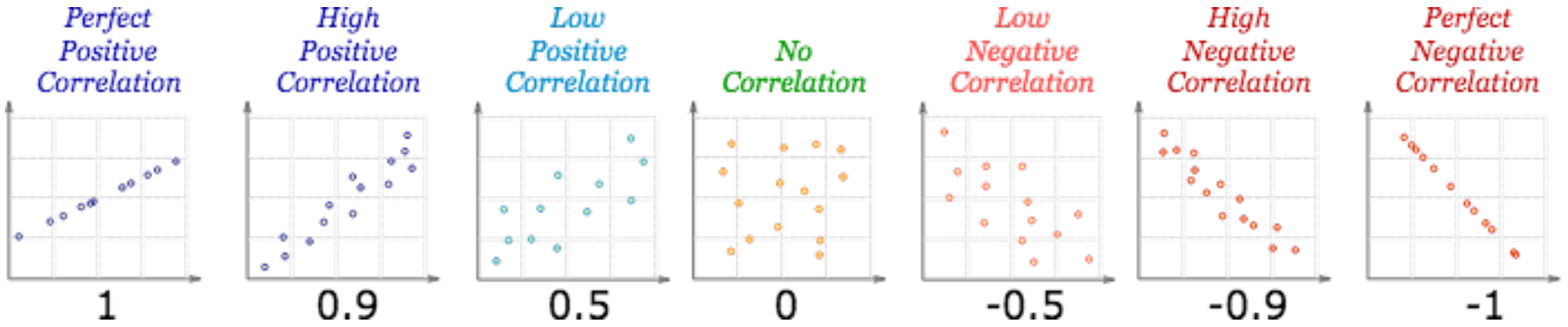


https://www.d3-graph-gallery.com/graph/bubble_basic.html https://observablehq.com/@d3/scatterplot-with-shapes

Scatterplot tasks

Scatterplot tasks

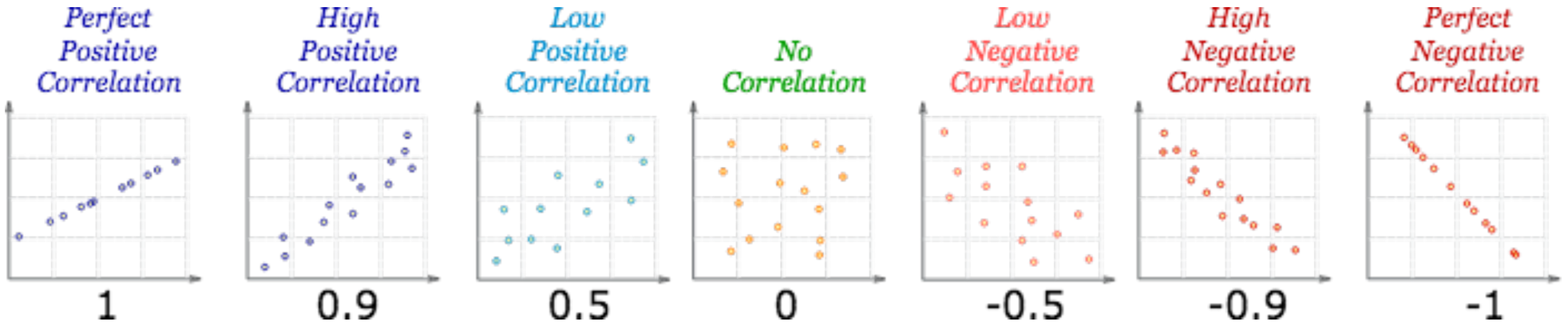
- correlation



https://www.mathsisfun.com/data/scatter-xy-plots.html

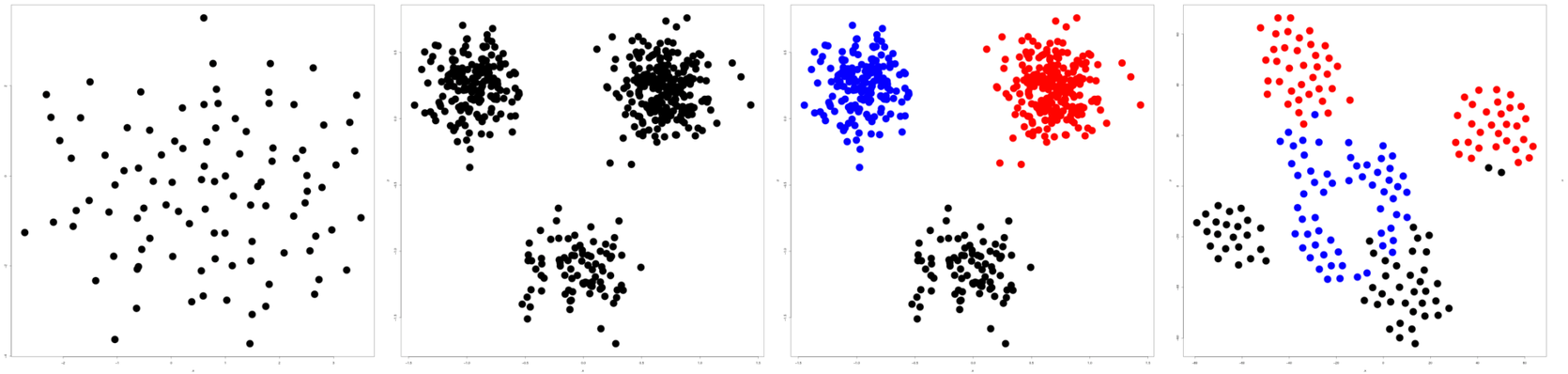
Scatterplot tasks

- correlation



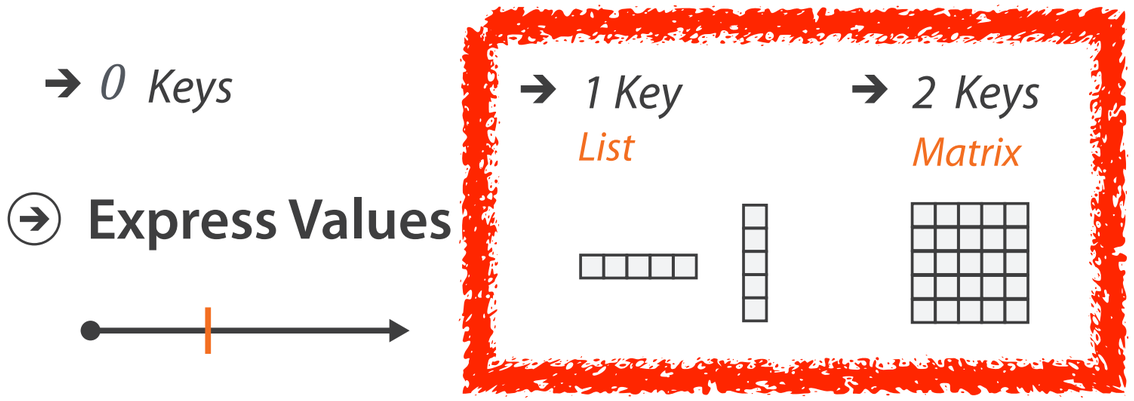
https://www.mathsisfun.com/data/scatter-xy-plots.html

- clusters/groups, and clusters vs classes



https://www.cs.ubc.ca/labs/imager/tr/2014/DRVisTasks/

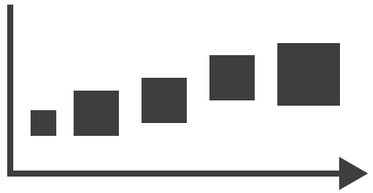
Some keys



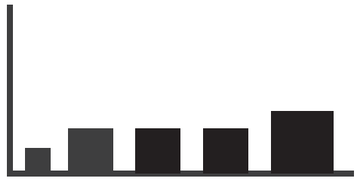
Some keys: Categorical regions

- Separate

- Order



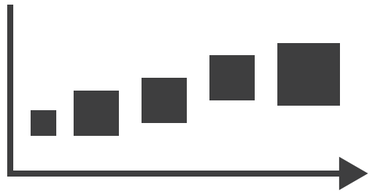
- Align



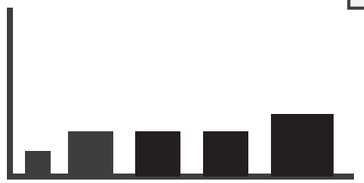
Regions: Separate, order, align

➔ Separate

➔ Order

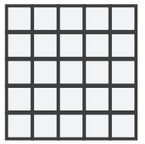


➔ Align



➔ *1 Key*
List

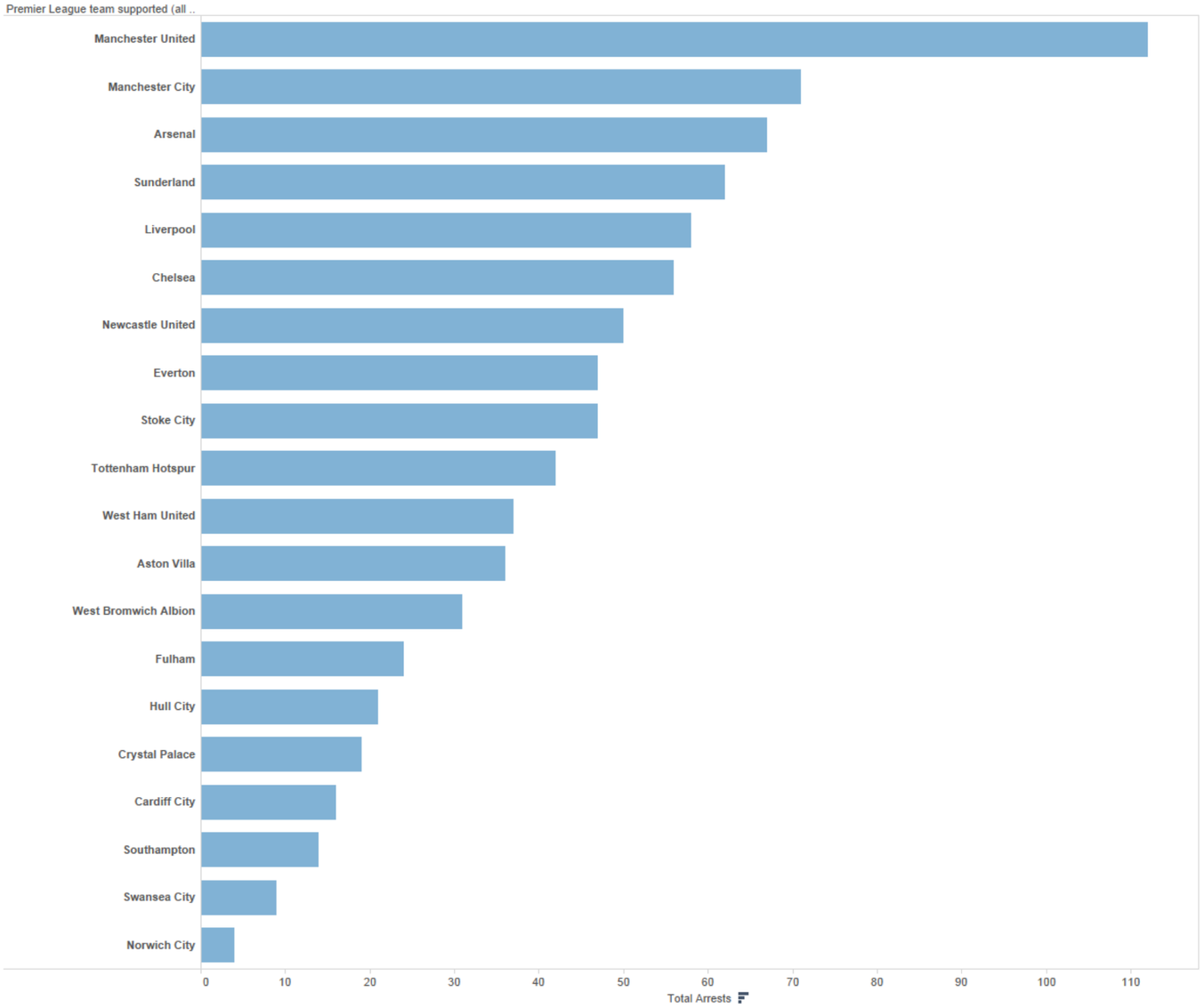
➔ *2 Keys*
Matrix



- separate into regions by any attribute: categorical or ordered
  - **regions**: contiguous bounded areas distinct from each other
  - no conflict with expressiveness principle for categorical attributes
  - one glyph/mark per region
- then can use ordered attribute to order and align regions
  - order in one direction (eg horiz), then align in other direction (eg vert)
  - align: impose shared coordinate frame so absolute position has meaning
    - difference between value and 0 point along axis

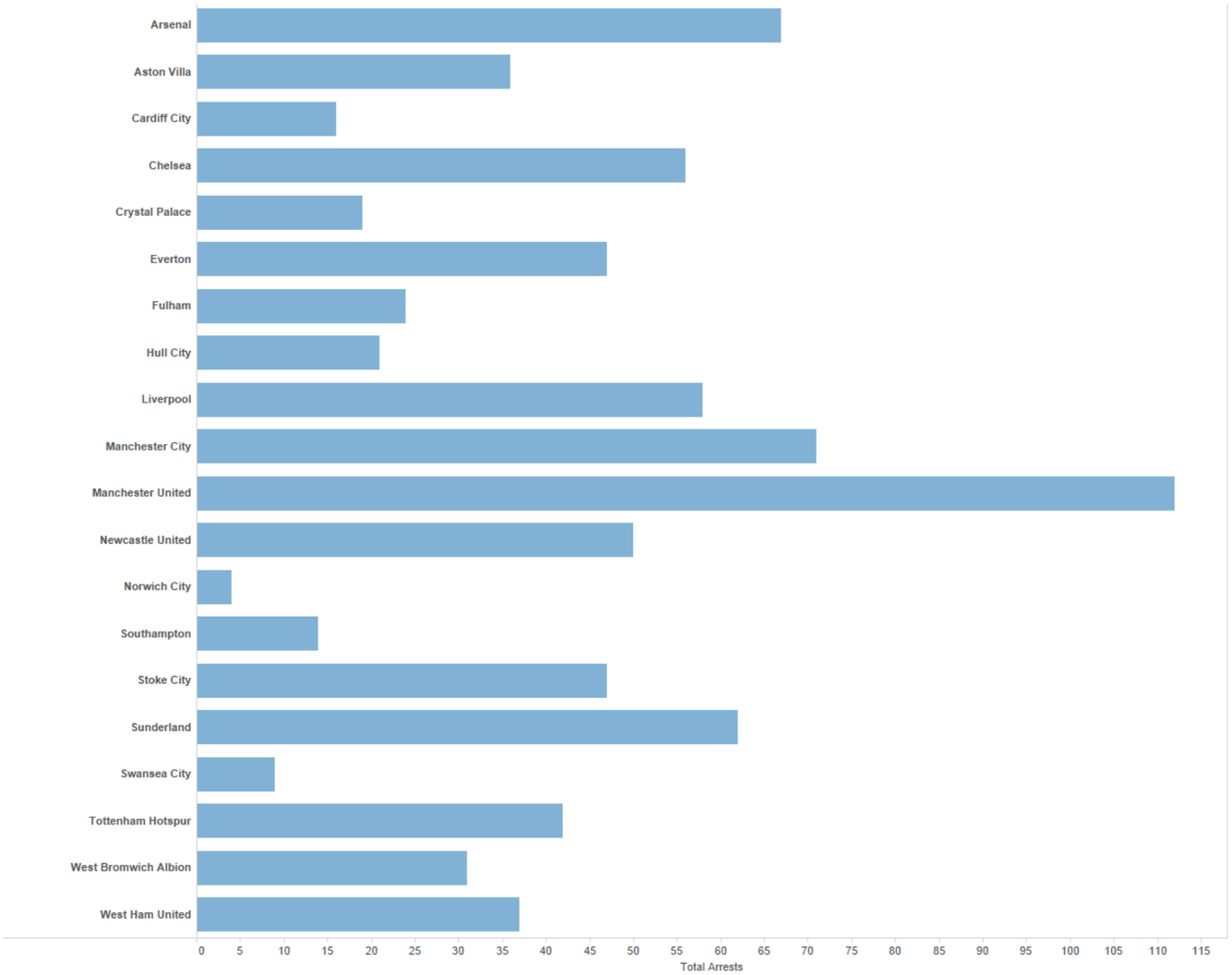
Separated and aligned and ordered

- best case



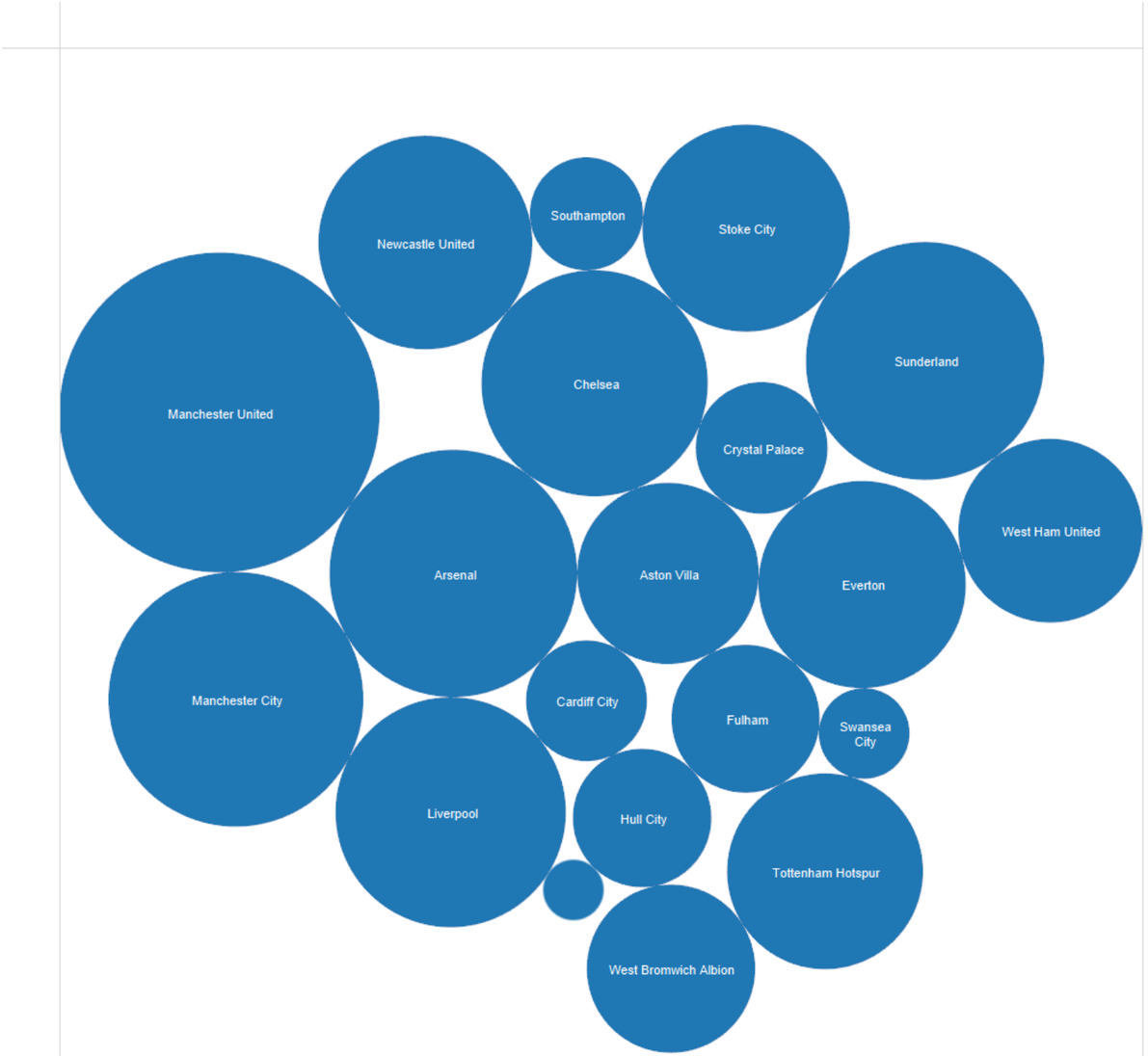
Separated and aligned but not ordered

- limitation: hard to know rank. what's 4th? what's 7th?



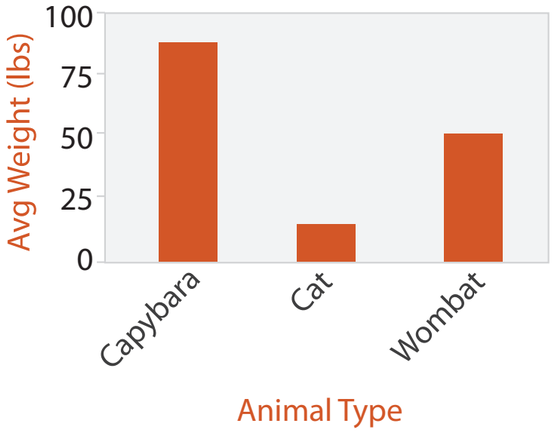
Separated but not aligned or ordered

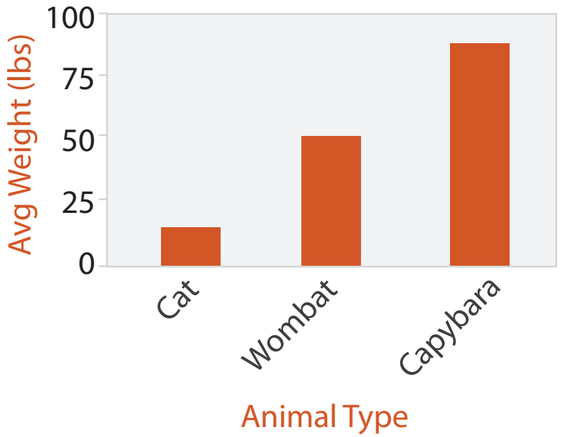
- limitation: hard to make comparisons with 2D size (vs aligned 1D position)



Idiom: **bar chart** (vertical)

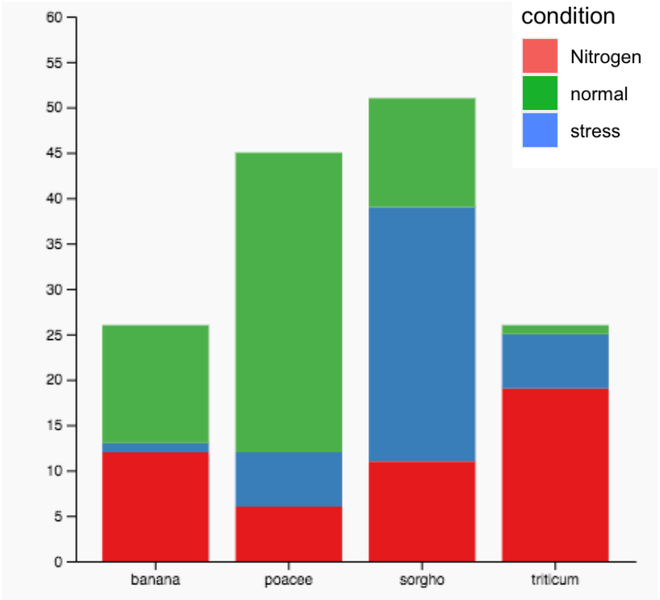
- one key, one value
  - data: 1 categ attrib, 1 quant attrib
  - mark: segments (or length-coded points)
  - channels
    - separate by type into one mark per region
      - order horizontally by some quant attrib
        - » alphabetical by attrib name (easy lookup)
        - » by same attrib as vert size (easy to see distributions, find extremes)
      - horiz: position according to order, distribute at equal intervals along axis
    - align vertically: express quant value with
      - vert position/order, vert size
      - segment: emphasize distance between value & 0 point on axis below
  - task: compare, lookup values
  - scalability
    - dozens to hundreds of levels for key [# bars], 1K values [lengths]





Idiom: **stacked bar chart**

- one more key
  - data: 2 categ attrib (both keys), 1 quant attrib
    - quant value indexed by categ pair {species, condition}
  - multi-level encoding: 2 levels
    - **glyph**: composite object, internal structure from multiple marks
    - entire-glyph level (full bar, by species)
    - within-glyph level (stacked components, by condition for each species)
  - channels
    - entire glyph (level 2): just like normal bar chart
      - separate by species, horiz order by species name, horiz position from order at equal intervals
      - vert position/order/size by value (for whole species)
    - within-glyph (level 1)
      - separate & color by condition, vert order alphabetical (by condition name)
      - vert 1D size (length) is value, vertical shared boundary
      - vert position: **unavailable** *except for lowest component*
      - mark: segment (or length-coded point mark)
  - task: + part-to-whole relationship [components]
  - scalability: asymmetric
    - for *stacked* key attrib, 10-12 levels [components]
    - for *main* key attrib, dozens to hundreds of levels [bars]

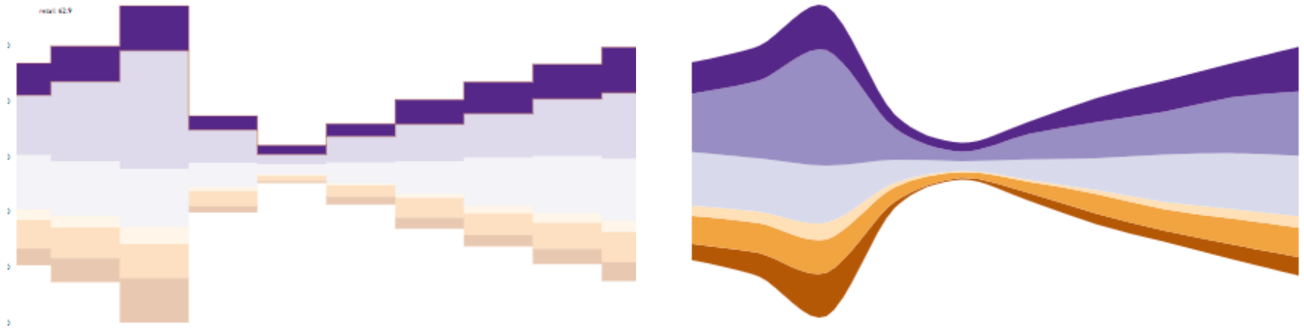


https://www.d3-graph-gallery.com/graph/barplot_stacked_basicWide.html

https://r-graph-gallery.com/48-grouped-barplot-with-ggplot2.html

Idiom: **streamgraph**

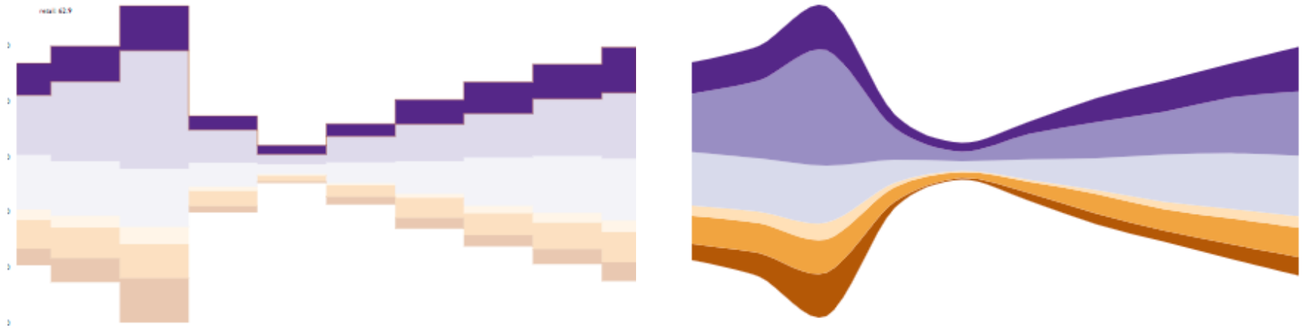
- generalized stacked graph
  - emphasize horizontal continuity of "stacked" attribute across time
    - not lookup for quant attribute at discrete time point
  - data: 2 keys
    - 1 categ key attrib (movies)
    - 1 ordered key attrib (time)
    - 1 quant value attrib (counts)
  - derived data/marks: polys, 1 mark per movie
    - vert 1D size (height): count for movie at time
    - vert pos: unavailable
    - vert order: derived by algorithm
    - horiz pos/order: time
    - horiz size: unavailable



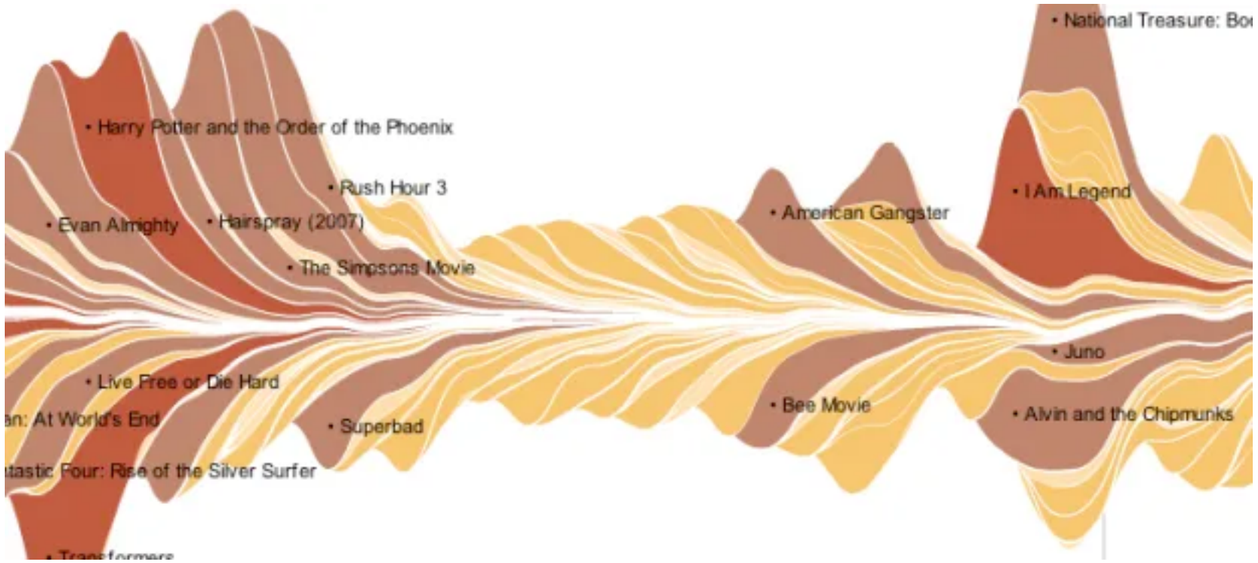
[Stacked Graphs Geometry & Aesthetics. Byron and Wattenberg. IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis 2008) 14(6): 1245–1252, (2008).]

Idiom: **streamgraph**

- generalized stacked graph
  - emphasize horizontal continuity of "stacked" attribute across time
    - not lookup for quant attribute at discrete time point
  - data: 2 keys
    - 1 categ key attrib (movies)
    - 1 ordered key attrib (time)
    - 1 quant value attrib (counts)
  - derived data/marks: polys, 1 mark per movie
    - vert 1D size (height): count for movie at time
    - vert pos: unavailable
    - vert order: derived by algorithm
    - horiz pos/order: time
    - horiz size: unavailable
  - scalability
    - dozens to hundreds of time keys (like stacked bar chart)
    - dozens to hundreds of movies keys (much more!)
      - most layers don't extend across whole chart
      - but very few have readable labels (big ones static, others hover only)



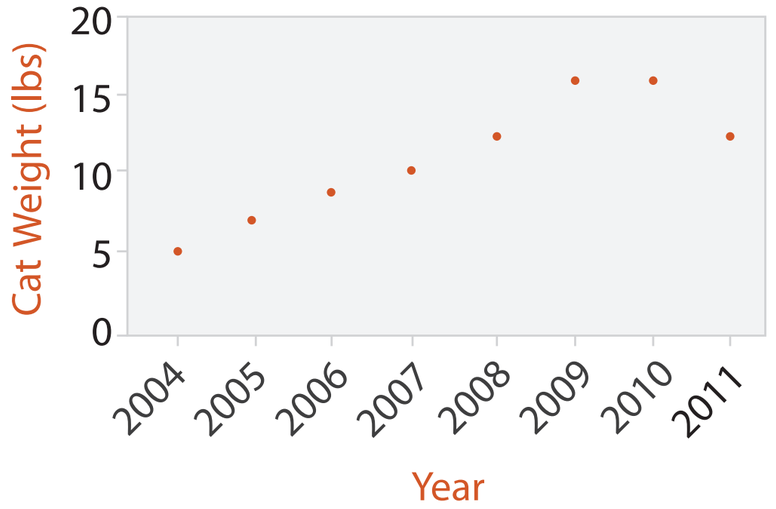
[Stacked Graphs Geometry & Aesthetics. Byron and Wattenberg. IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis 2008) 14(6): 1245–1252, (2008).]

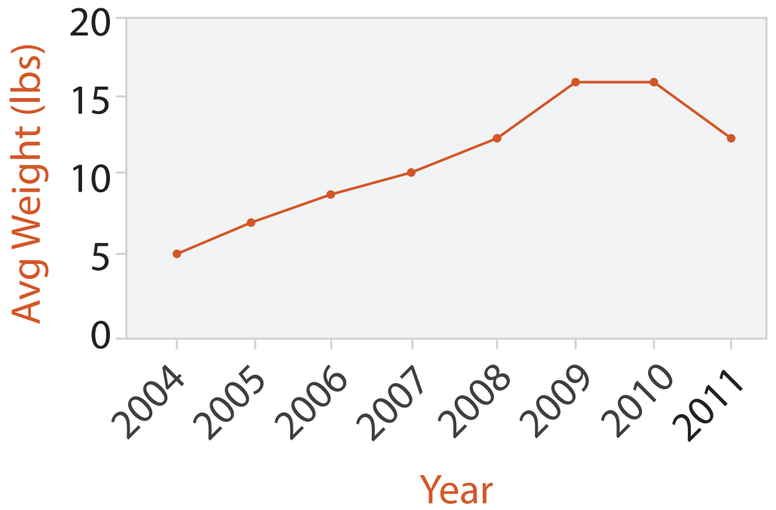


https://flowingdata.com/2008/02/25/ebb-and-flow-of-box-office-receipts-over-past-20-y

Idiom: **line chart**

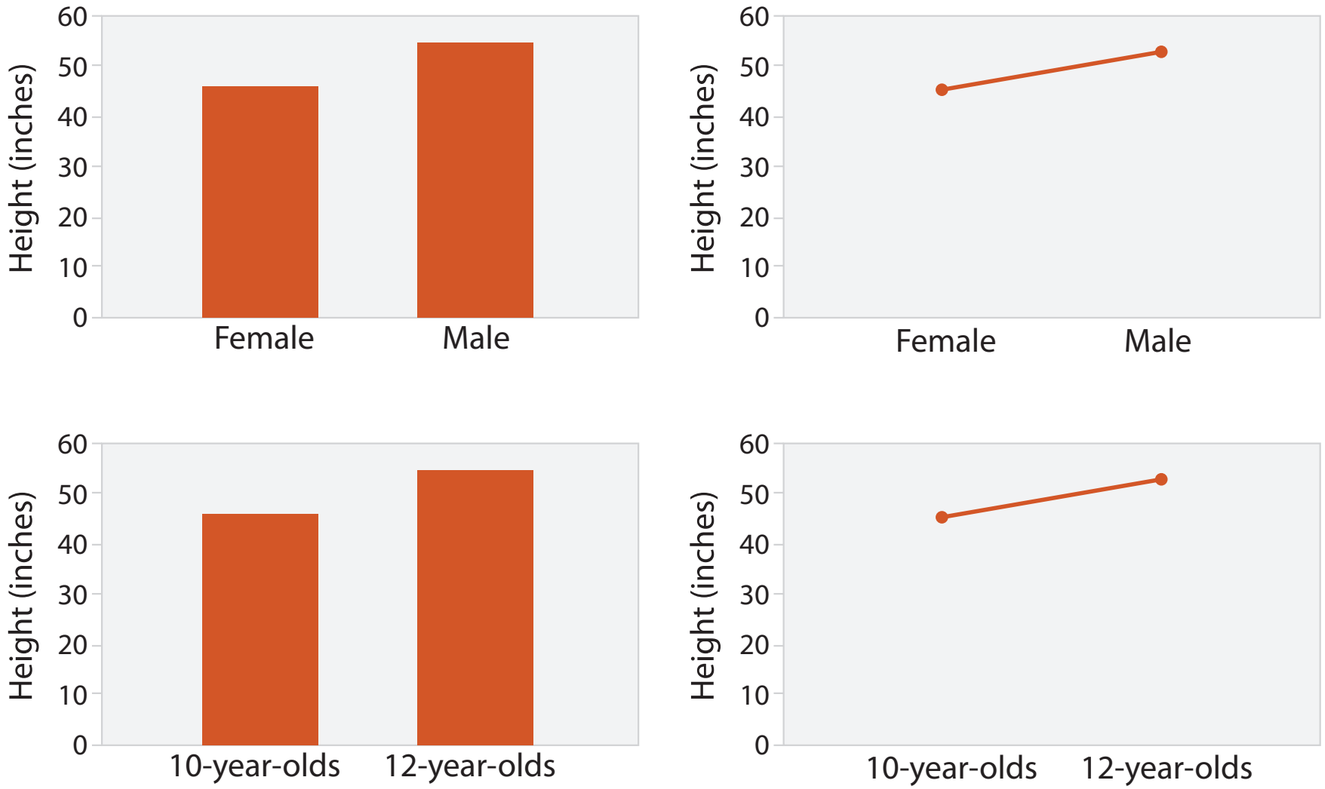
- one key, one value
  - data: 2 quant attribs: 1 key, 1 value
  - mark:
    - path: connecting mark between item values at each key
      - could be piecewise-linear or smooth
    - (could also use point marks at value locations)
  - channels
    - express quant value: horiz & vert pos/order
    - vert 1D size available (linewidth)
      - paths: consider along the line as horizontal, other way as vert
    - horiz size of each segment not available
      - just shows interval between expressed values, often uniform
  - task
    - find trend
      - connection marks emphasize ordering of items along key axis by explicitly showing relationship between one item and the next
  - scalability
    - 1K levels for value and key





Choosing bar vs line charts

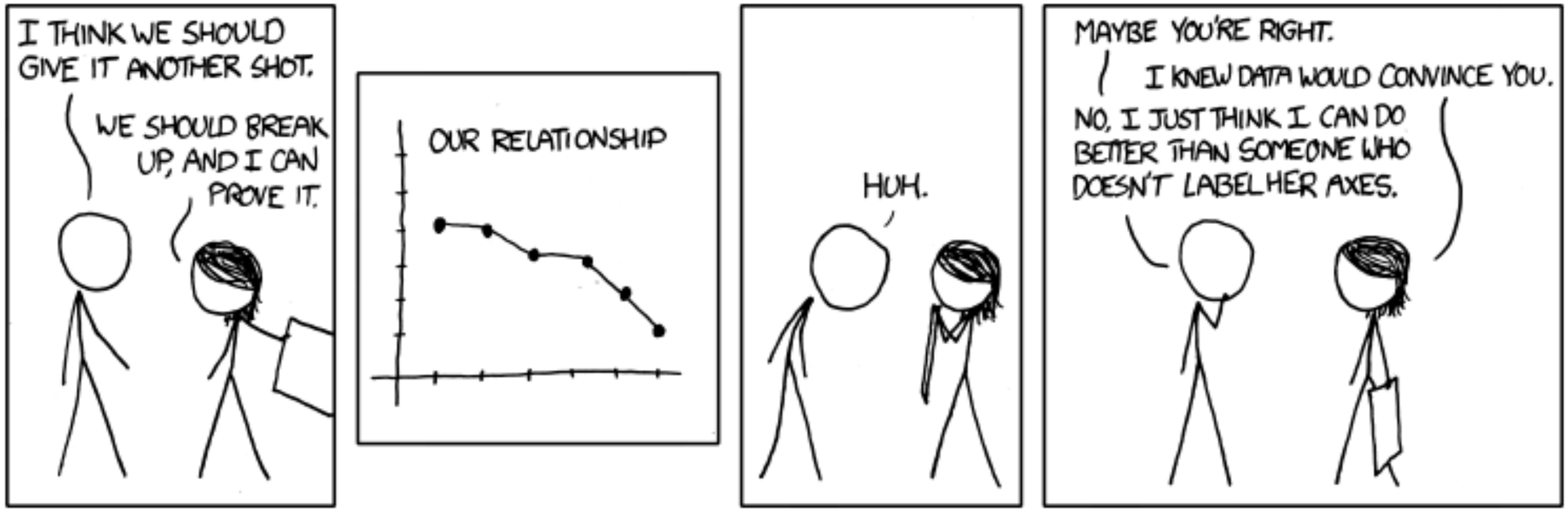
- depends on type of key attrib
  - bar charts if categorical
  - line charts if ordered
- do not use line charts for categorical key attribs
  - violates expressiveness principle
    - implication of trend so strong that it overrides semantics!
      - "The more male a person is, the taller he/she is"



after [Bars and Lines: A Study of Graphic Communication. Zacks and Tversky. Memory and Cognition 27:6 (1999), 1073–1079.]

Chart axes: label them!

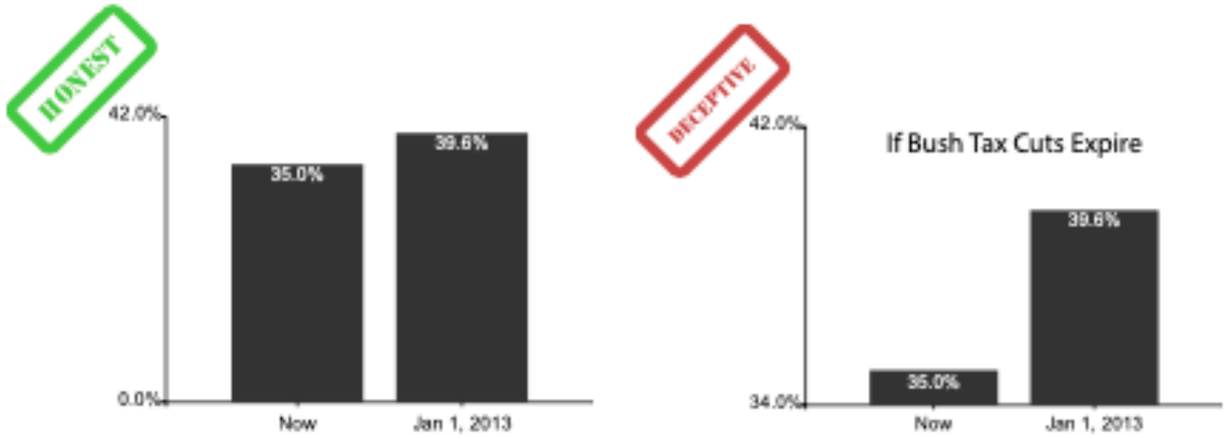
- best practice to label
  - few exceptions: individual small multiple views could share axis label



https://xkcd.com/833/

Chart axes: avoid cropping y axis

- include 0 at bottom left or slope misleads

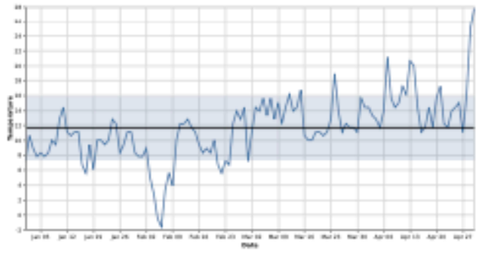


[Truncating the Y-Axis: Threat or Menace?
Correll, Bertini, & Franconeri, CHI 2020.]

Chart axes: avoid cropping y axis

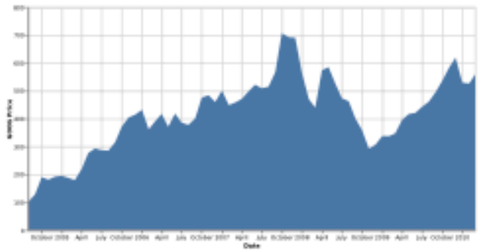
- include 0 at bottom left or slope misleads
  - some exceptions (arbitrary 0, small change matters)

[Truncating the Y-Axis: Threat or Menace? Correll, Bertini, & Franconeri, CHI 2020.]

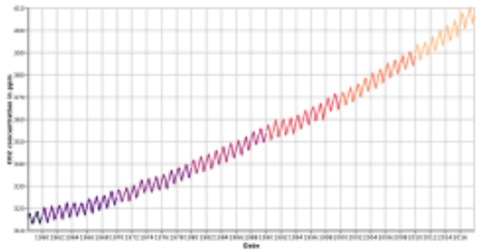


(a) Statistical process charts rely on comparison to an expected value, and so deviations from that value, not from zero, are important

(b) Index charts compare to an indexed value rather than zero.



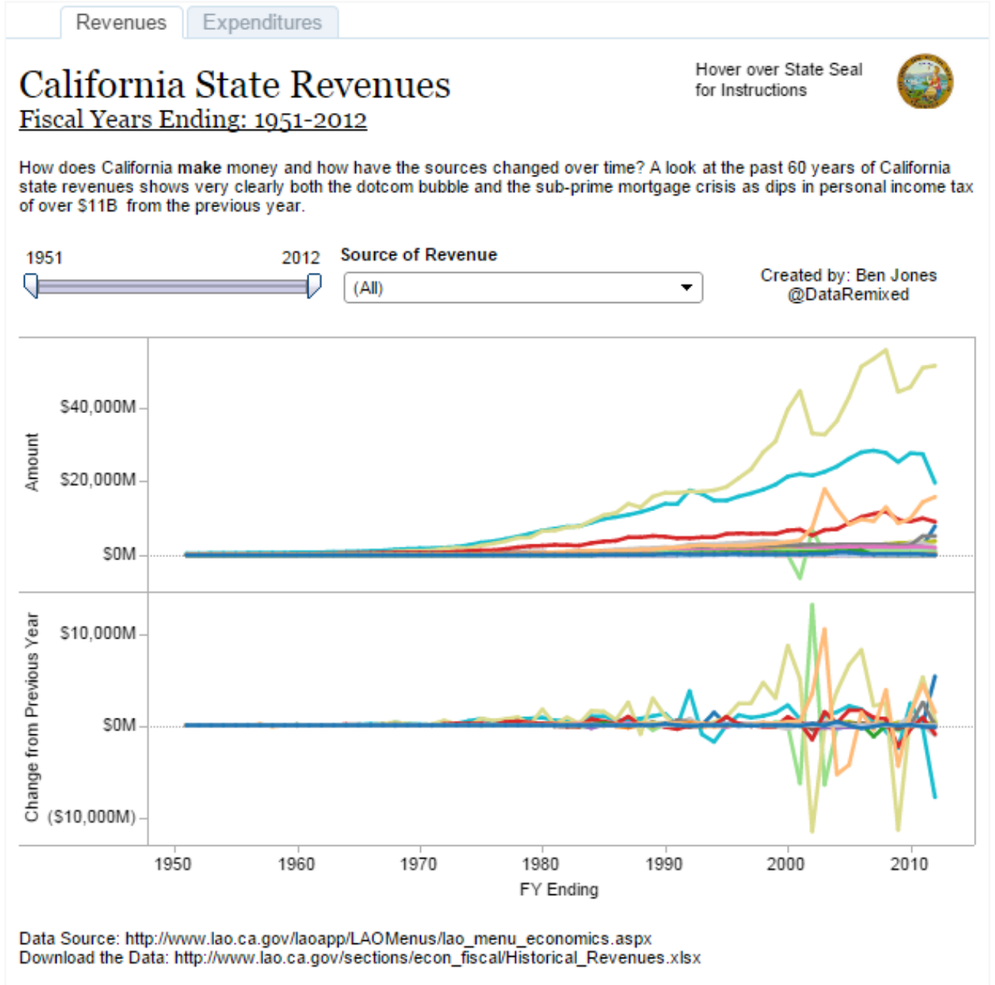
(c) Stock charts must show small differences in stock value, as these can translate to enormous monetary gains or losses.



(d) Climate Anomaly charts rely on both highlighting deviation from a non-zero expected value but also emphasize the potentially disastrous impact of even minute changes in climate.

Idiom: **Indexed line charts**

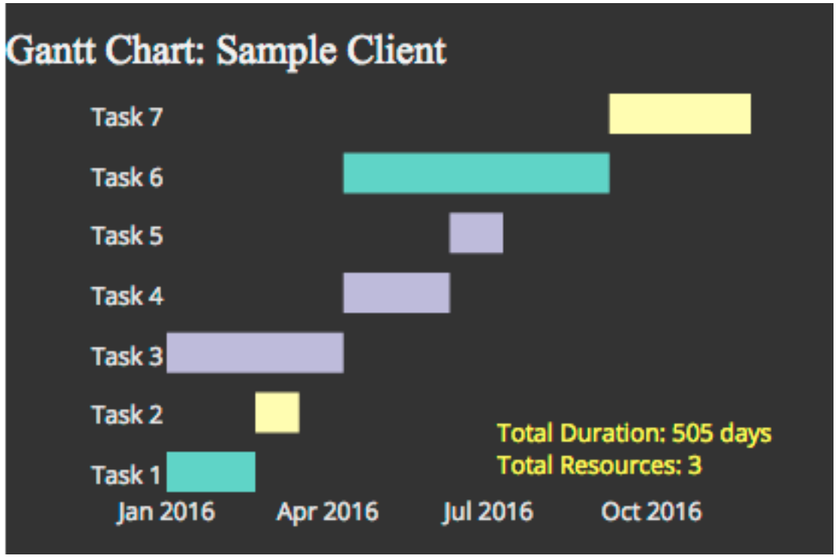
- data: 2 quant attribs
  - 1 key + 1 value
- derived data: new quant value attrib
  - index
  - plot instead of original value
- task: show change over time
  - principle: normalized, not absolute
- scalability
  - same as standard line chart



https://public.tableau.com/profile/ben.jones#!/vizhome/CAStateRevenues/Revenues

Idiom: **Gantt charts**

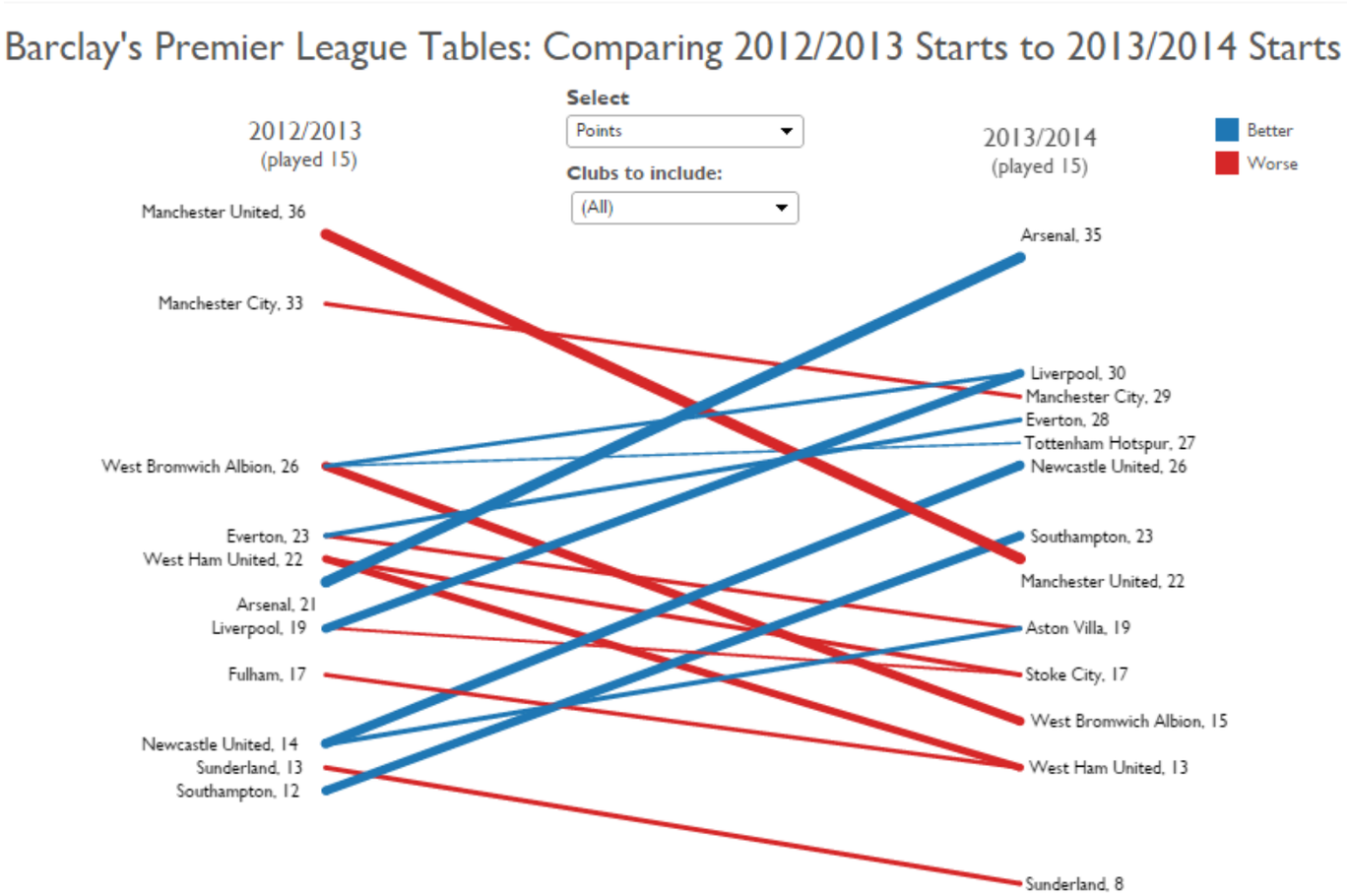
- one key, two (related) values
  - data
    - 1 categ attrib, 2 quant attribs
  - mark: segment
  - channels
    - horiz position/order: start time & end time
      - redundant: horiz length (1D size) is duration (end-start)
    - vert order: task name/number, vert position: task
  - task
    - emphasize temporal overlaps & start/end dependencies between items
  - scalability
    - dozens of key levels [bars]
    - 1K value levels [times, durations]



https://www.r-bloggers.com/gantt-charts-in-r-using-plotly/

Idiom: **Slopegraphs**

- express two values
  - data
    - 2 quant value attribs
    - (optional: 1 derived attrib: change magnitude)
  - mark: segment
    - connecting mark between 2 values
  - channels
    - vertical pos/order, 1 axis per attrib: express value
    - segment width (1D size, *vert*): change magnitude
    - segment length (1D size, *horiz*): **unavailable**
    - color
  - task
    - emphasize changes in rank/value
  - scalability
    - 1K value levels
    - dozens of items



https://public.tableau.com/profile/ben.jones#!/vizhome/Slopegraphs/Slopegraphs

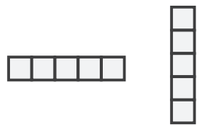
2 Keys

➔ *0 Keys*

➔ **Express Values**

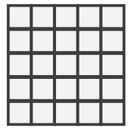
➔ *1 Key*

List



➔ *2 Keys*

Matrix

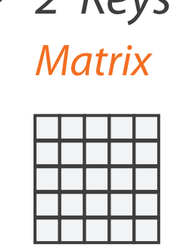


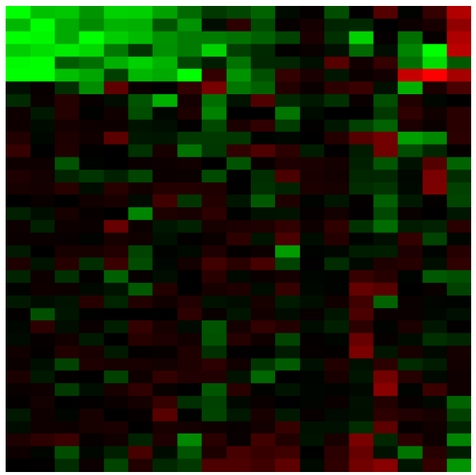
Idiom: **heatmap**

- two keys, one value
  - data
    - 2 categ attribs (gene, experimental condition)
    - 1 quant attrib (expression levels)
  - marks: point
    - separate into regions, order and align both directions
    - vertical & horizontal: position/order, **shared** boundaries, size **unavailable**
    - orientation & 2D size unavailable
    - indexed by 2 categorical key attributes
  - channels
    - color by quant attrib (ordered diverging colormap)
  - task: find outliers, find clusters (if ordered well)
  - scalability: asymmetric
    - 100s of categ levels, up to 1M items, but ~10 quant attrib levels

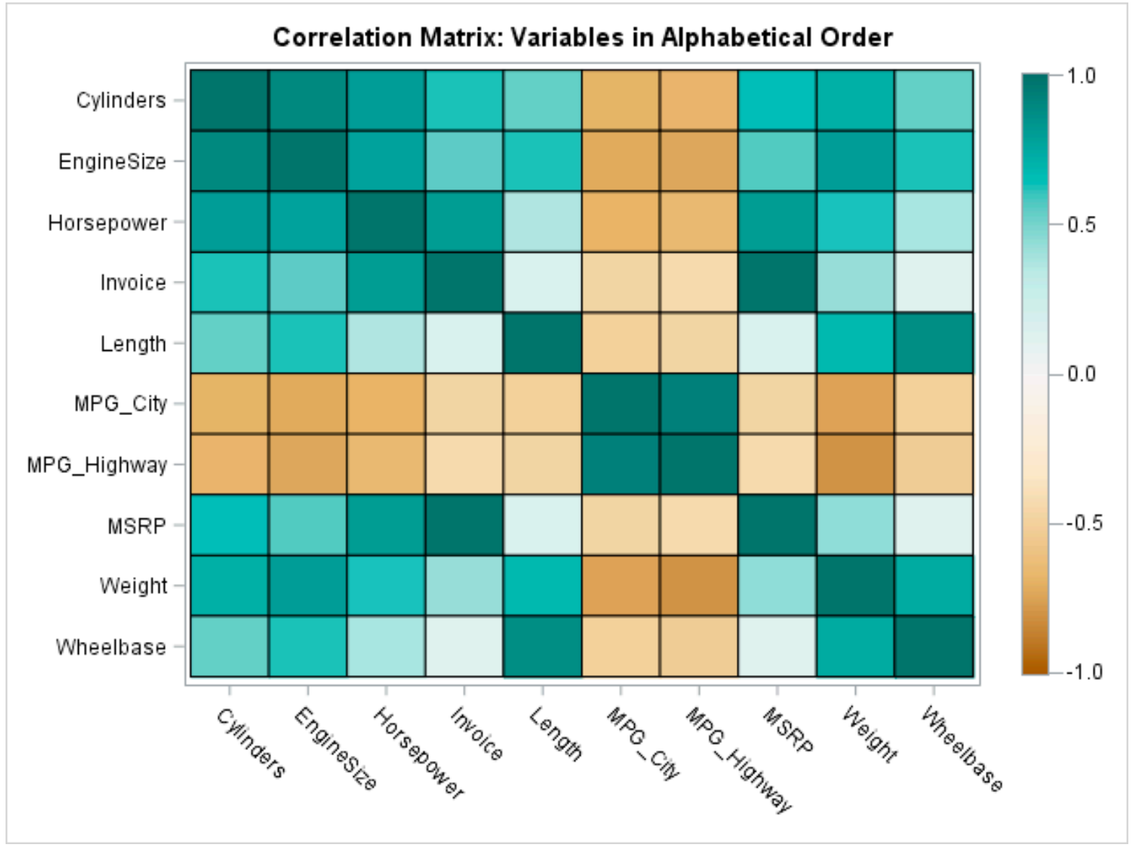
➔ *2 Keys*

Matrix

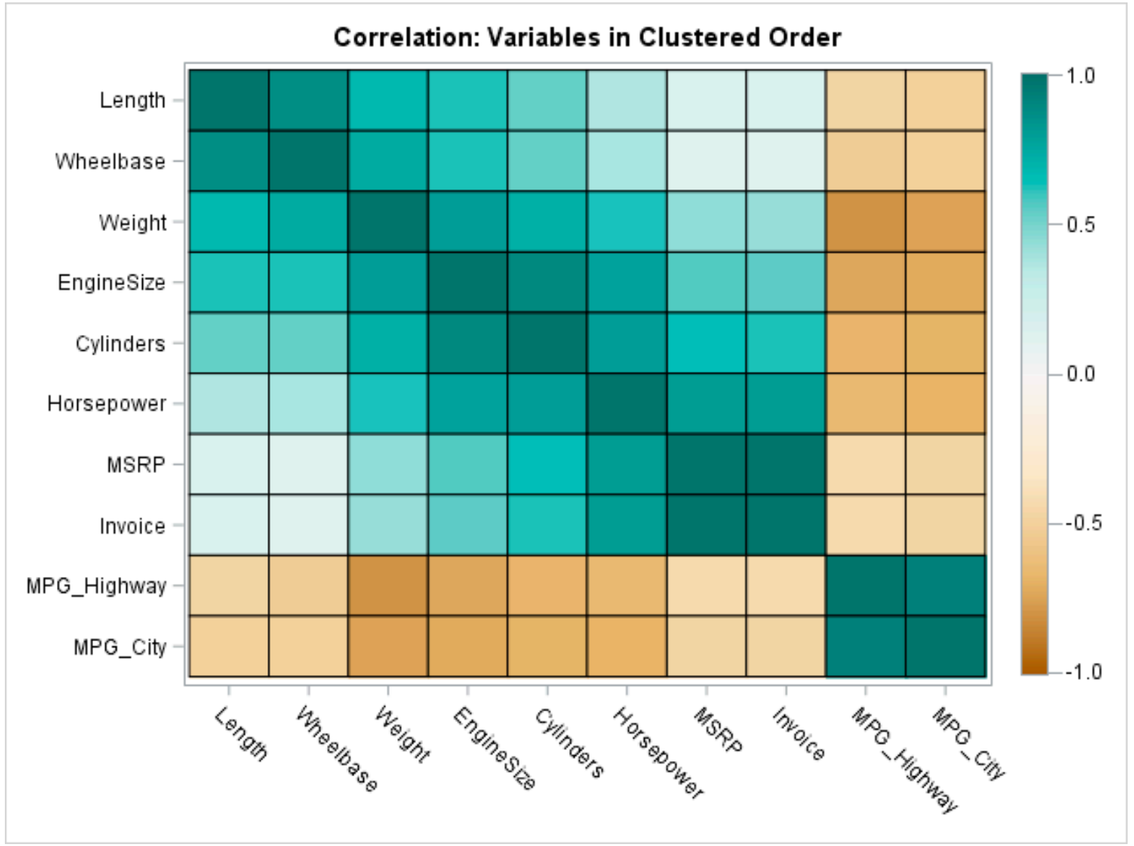




Heatmap reordering



order by attrib name (alphabetical)

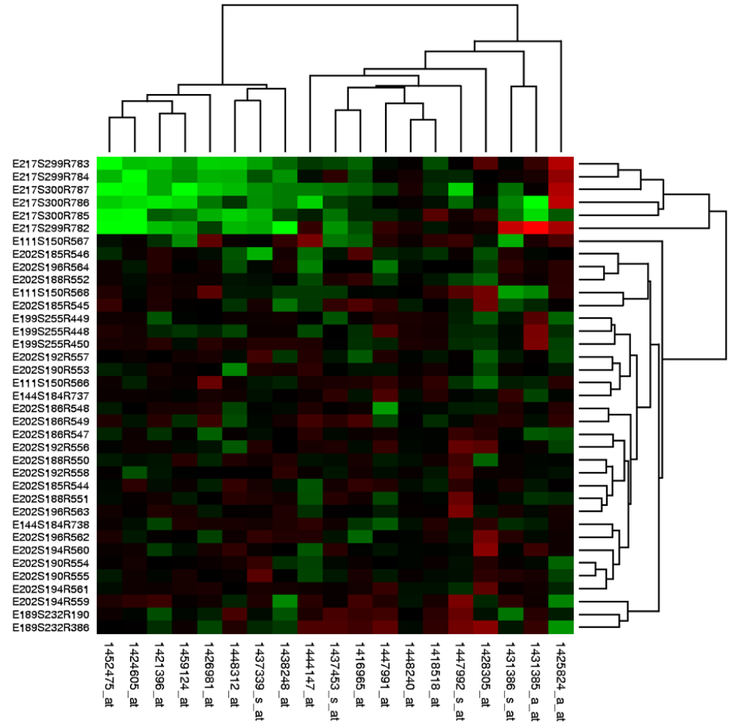


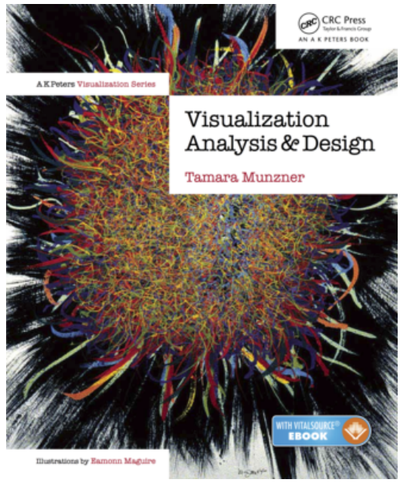
order by derived attribute (data-driven clustering)

https://blogs.sas.com/content/iml/2018/05/02/reorder-variables-correlation-heat-map.html

Idiom: **cluster heatmap**

- in addition
  - derived data
    - 2 cluster hierarchies
  - dendrogram
    - parent-child relationships in tree with connection marks
    - leaf aligned positions so interior branch heights easy to compare
  - heatmap
    - marks vert & horiz order & position by derived attrib
      - from cluster hierarchy traversal
    - task: assess quality of clusters found by automatic methods





Visualization Analysis & Design

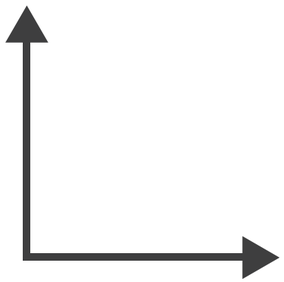
Tables (Ch 7 revised) II

Tamara Munzner
Department of Computer Science
University of British Columbia

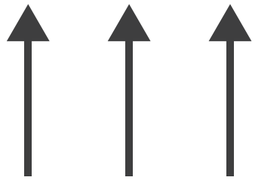
@tamara@vis.social
@tamaramunzner

Axis Orientation

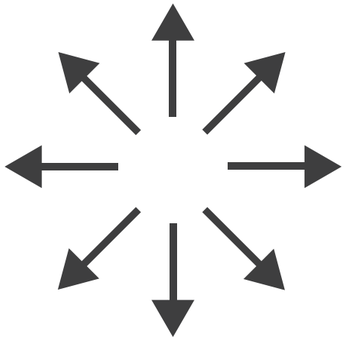
- Rectilinear



- Parallel

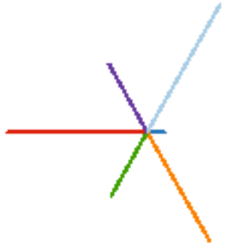


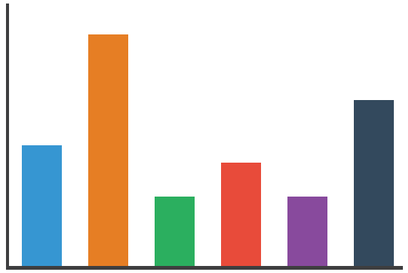
- Radial



Idioms: **radial bar chart, star plot**

- star plot
  - segment mark, axes meet at central point
- radial bar chart:
  - segment mark, axes meet at central ring
- both: radial (polar) axes
    - radial position/order/length for quant attrib
    - angular position/order for other attrib
- bar chart: rectilinear (cartesian) axes
    - vertical position/order/length for quant attrib
    - horizontal position/order for other attrib
- accuracy
  - high-precision rectilinear aligned position decoding does **not** apply to radial layouts!
    - lower-accuracy length judgement, not aligned position

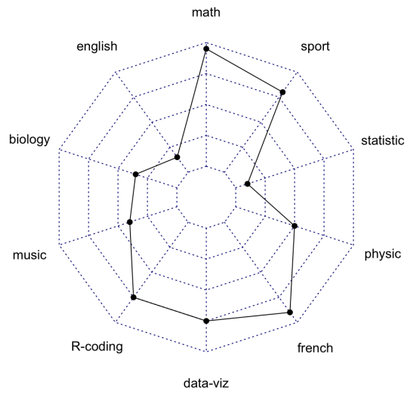


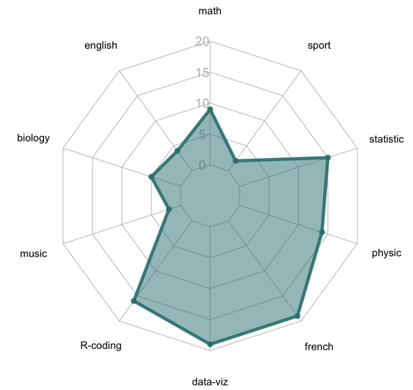


[Vismon: Facilitating Risk Assessment and Decision Making In Fisheries Management. Booshehrian, Möller, Peterman, and Munzner. Technical Report TR 2011-04, Simon Fraser University, School of Computing Science, 2011.]

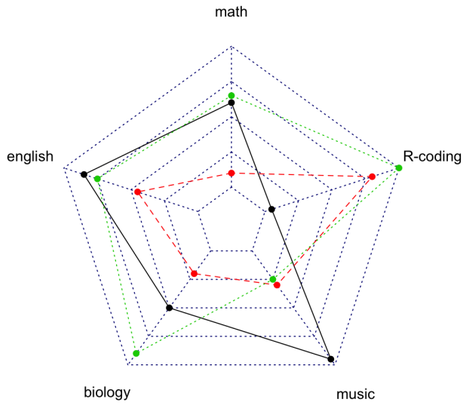
Idiom: **radar/spider chart**

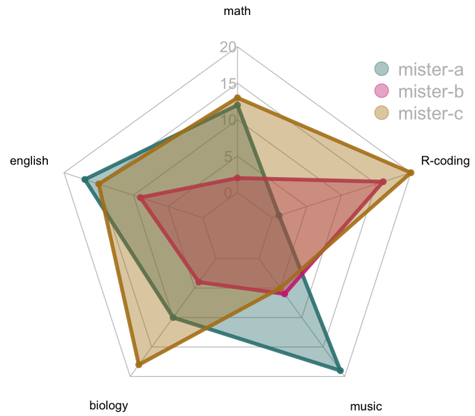
- like line chart, but radial:
  - connecting mark between multiple item values
- unlike rectilinear line chart
  - closed, so mark is poly boundary
  - vs rectilinear: open, so mark is path
- options
  - fill color for interior
    - makes shape even more salient
  - explicitly plotting point marks
- other names
  - Kiviat diagram, polar chart, ...
- guidance: avoid unless data is cyclic





https://r-graph-gallery.com/142-basic-radar-chart.html

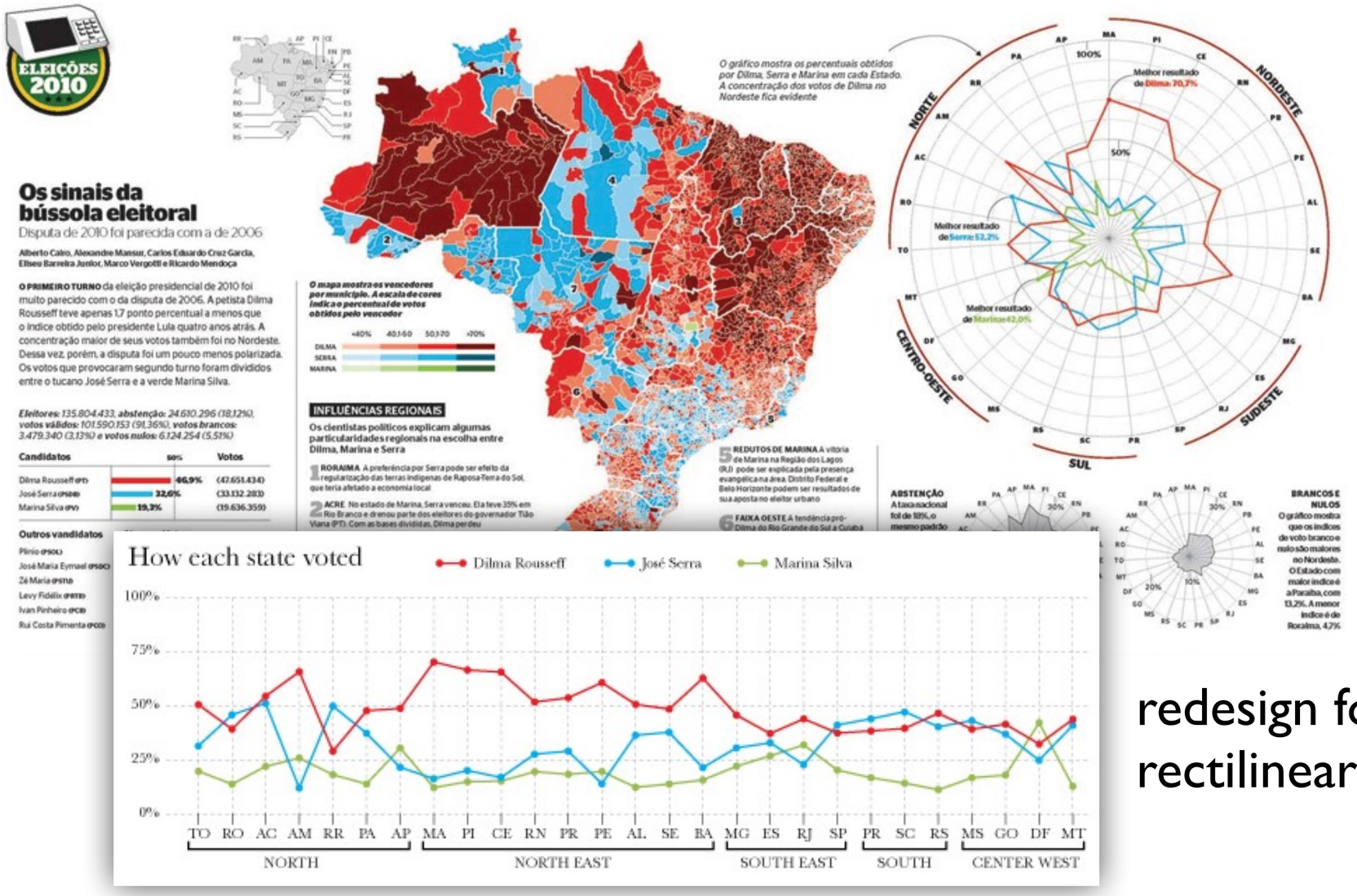




https://r-graph-gallery.com/143-spider-chart-with-saveral-individuals.html

https://www.chartjs.org/docs/latest/charts/radar.html

“Radar graphs: Avoid them (99.9% of the time)”



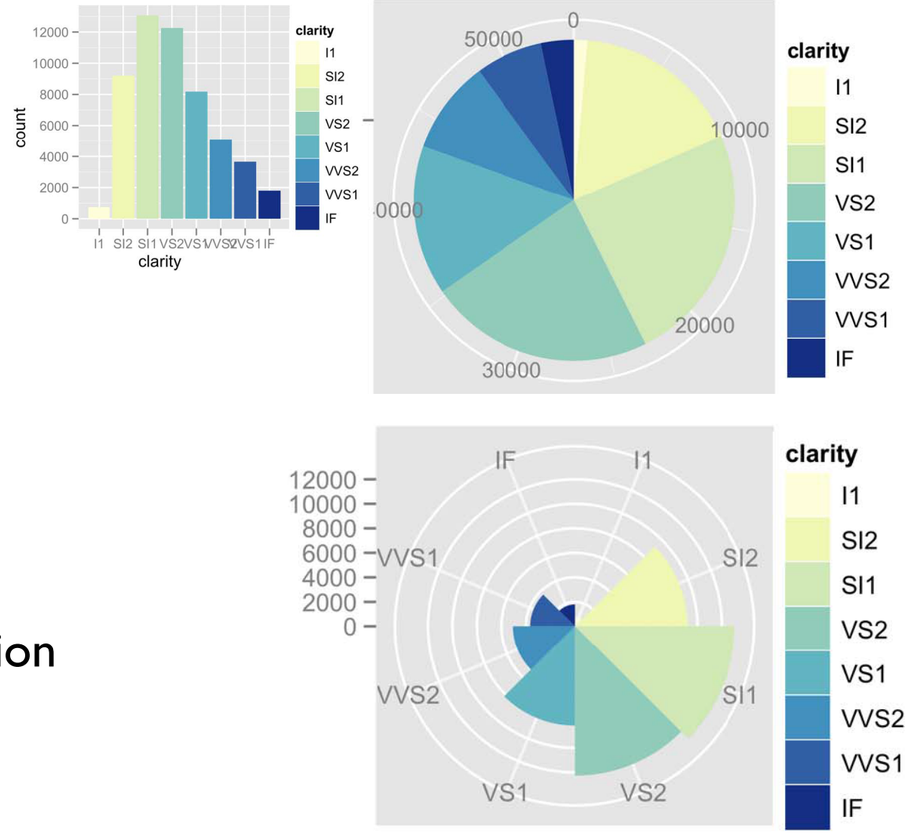
original
difficult to interpret

redesign for
rectilinear

http://www.thefunctionalart.com/2012/11/radar-graphs-avoid-them-999-of-time.html

Idioms: **pie chart, coxcomb chart**

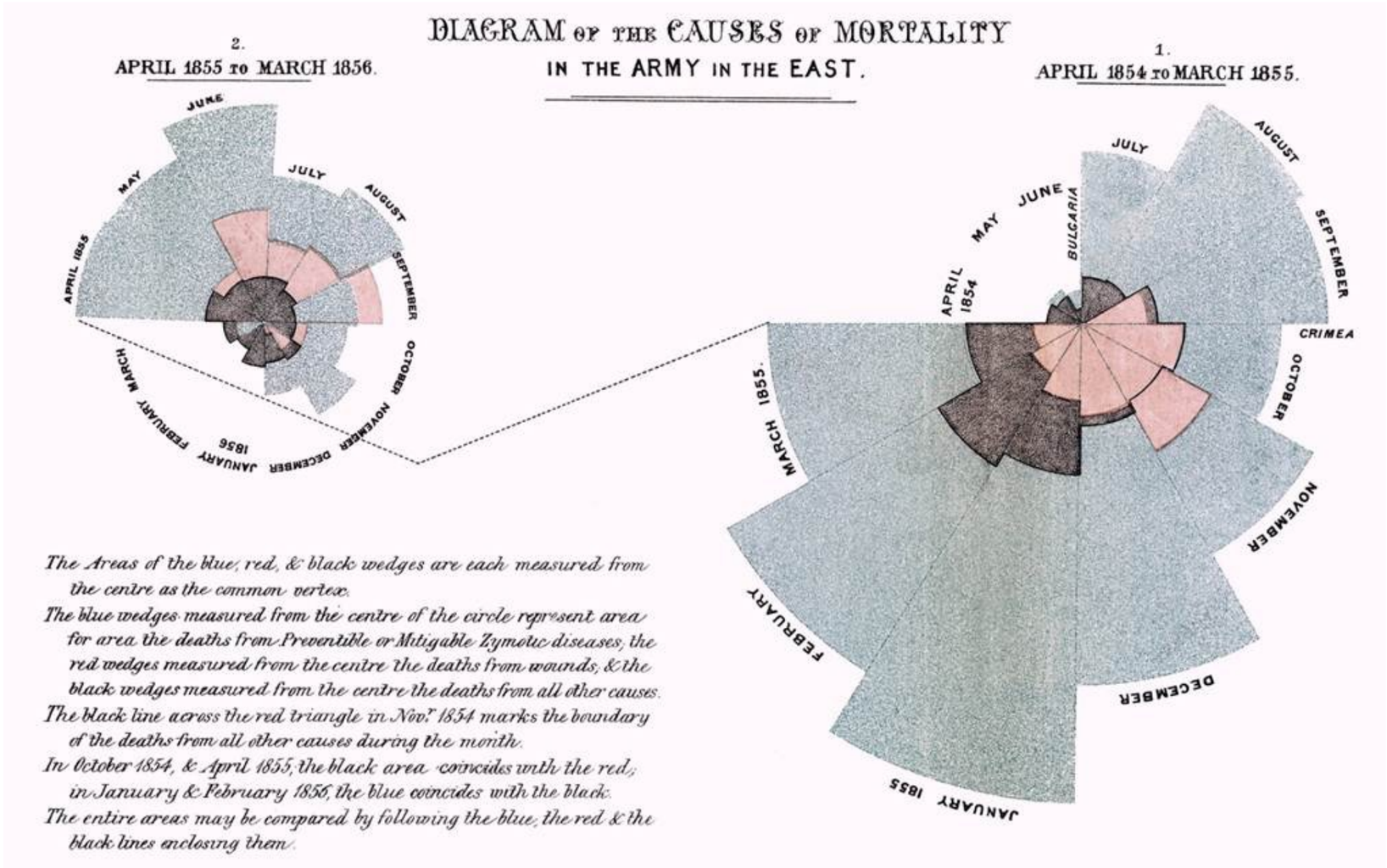
- data
  - 1 categ key attrib, 1 quant value attrib
- pie chart: vary angular position, fix radial distance
  - poly marks
    - angular position/order/size by attribute (clarity), **shared** boundary
    - radial position/order/size is uniform & **unavailable**
    - 2D size (area) **unavailable** (but varies linearly with angular 1D size)
  - accuracy: angular position less accurate than rectilinear aligned position
  - task: part-to-whole judgements
- coxcomb chart: vary radial distance, uniformly distribute angular position
  - direct analog to bar charts, just radial
  - poly marks
    - angular 1D size (width) is uniform & unavailable
    - angular position/order is uniform, **shared** boundary
    - radial position/order/size by attribute (clarity)
    - 2D size (area) **unavailable**



[A layered grammar of graphics. Wickham. Journ. Computational and Graphical Statistics 19:1 (2010), 3–28.]

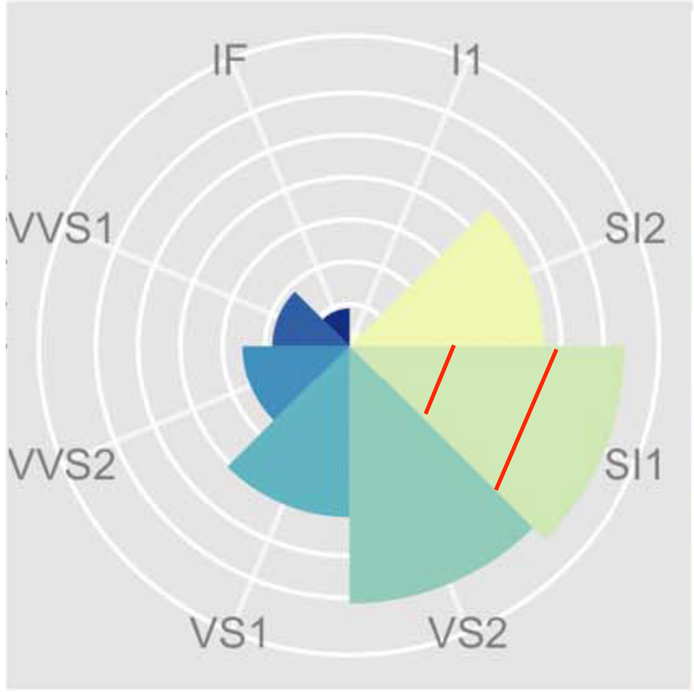
Coxcomb / nightingale rose / polar area chart

- invented by Florence Nightingale:
  Diagram of the Causes of Mortality in the Army in the East

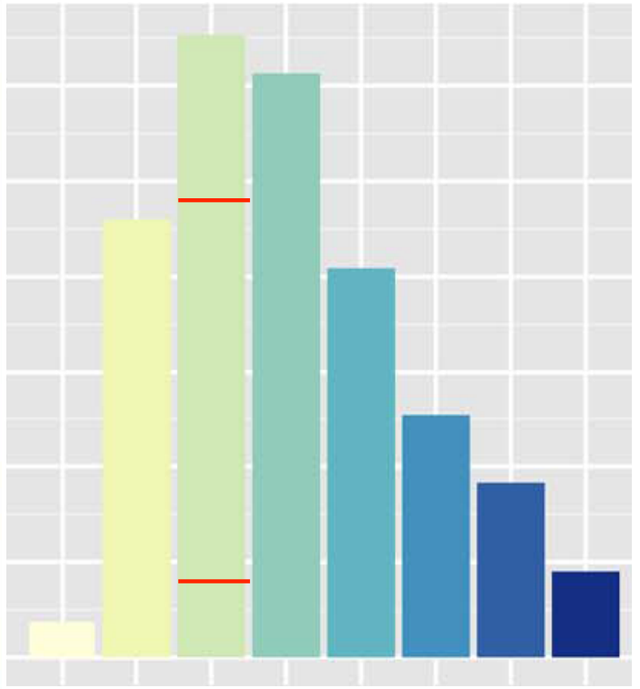


Coxcomb: perception

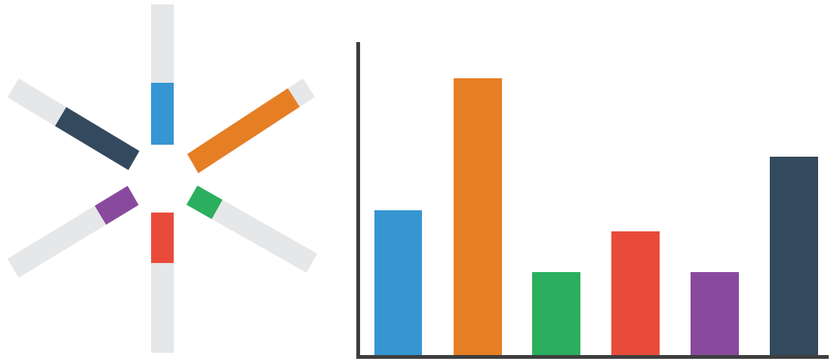
- nonuniform sector width as length increases
  - so area variation is nonlinear wrt mark length!
- encode: length
  - 1D angular size
- decode/perceive: area
  - 2D size
- bar chart safer: uniform width, so area is linear w/ mark length
  - both radial & rectilinear cases



nonuniform width as length increases



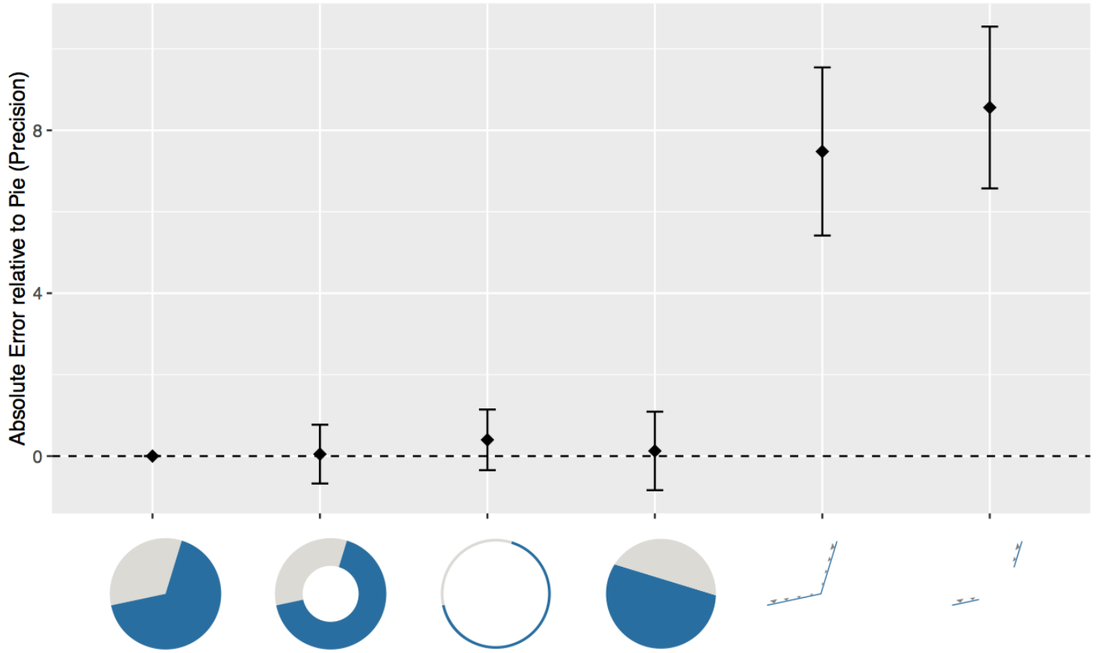
uniform width as length increases



radial & rectilinear bars: uniform width as length increases

Pie charts: perception

- what do viewers decode?
- some empirical evidence that people respond to arc length
  - decode/perceive: not angles
  - maybe also areas?…
- donut charts no worse than pie charts



[Arcs, Angles, or Areas: Individual Data Encodings in Pie and Donut Charts. Skau and Kosara. Proc. EuroVis 2016.]

https://eagereyes.org/blog/2016/an-illustrated-tour-of-the-pie-chart-study-results

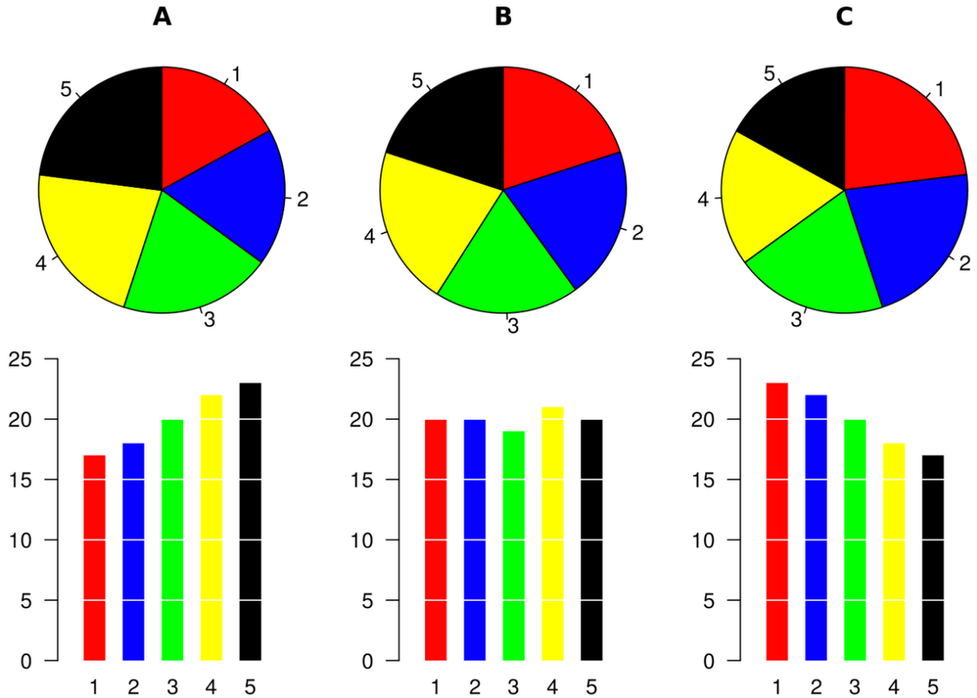
Pie charts: best practices

- not so bad for two (or few) levels, for part-to-whole task

https://eagereyes.org/pie-charts

Pie charts: best practices

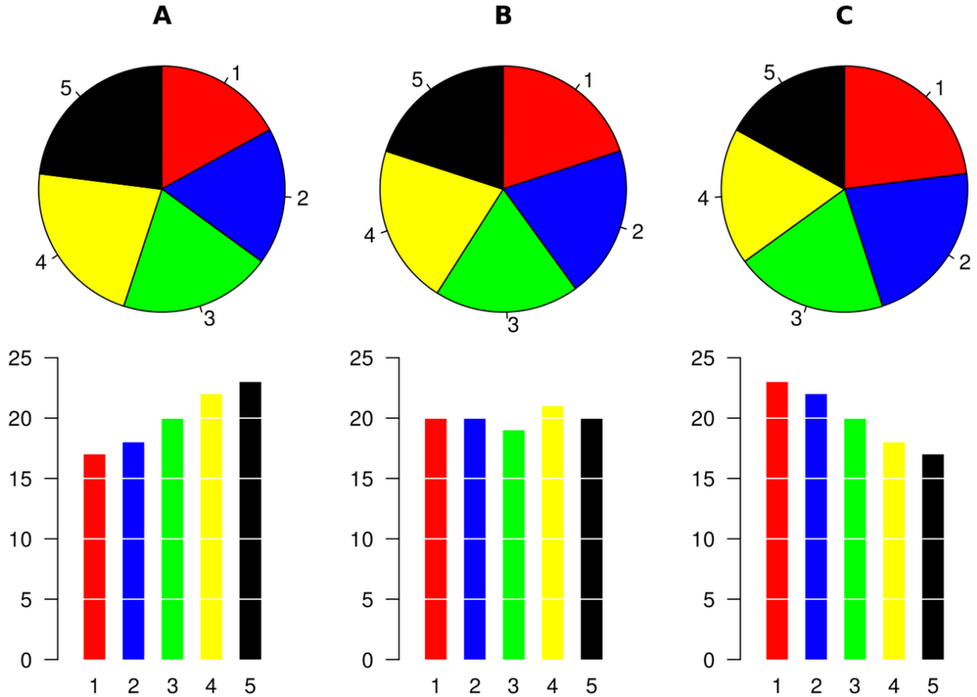
- not so bad for two (or few) levels, for part-to-whole task
- dubious for several levels if details matter

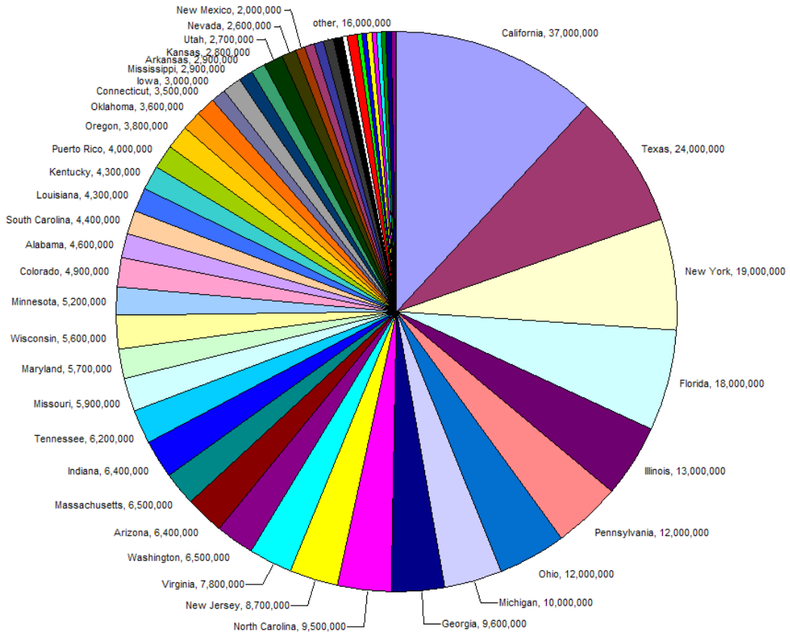


https://eagereyes.org/pie-charts

Pie charts: best practices

- not so bad for two (or few) levels, for part-to-whole task
- dubious for several levels if details matter
- terrible for many levels

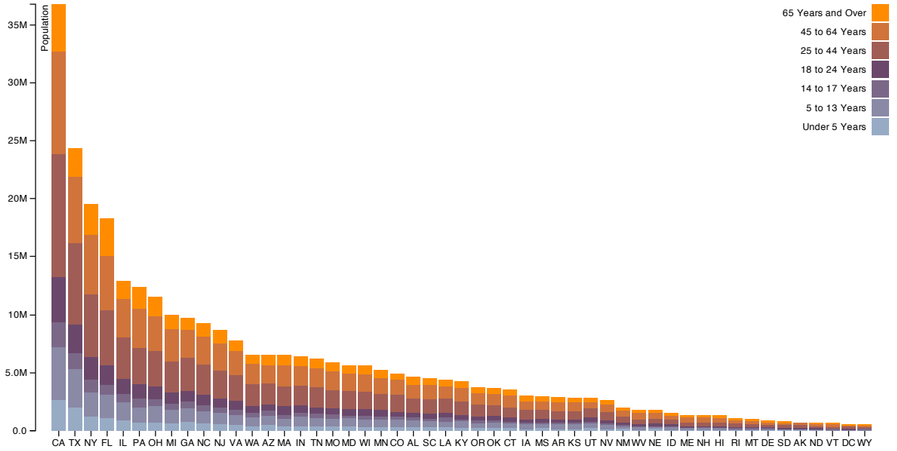


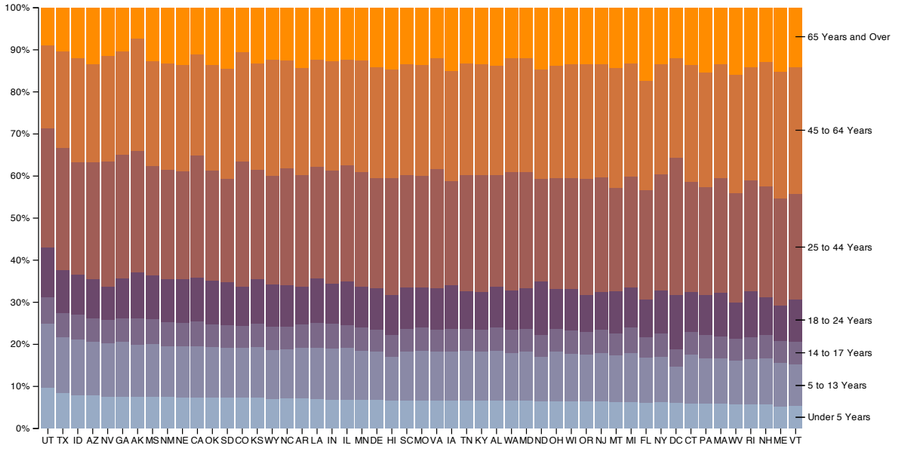


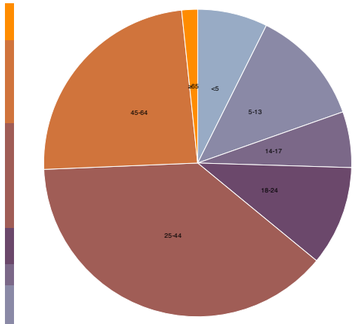
https://eagereyes.org/pie-charts

Idioms: **normalized stacked bar chart**

- task: part-to-whole judgements
- normalized stacked bar chart
  - stacked bar chart, normalized to full vert height
    - relative sizes show proportions
  - analysis of differences
    - level 2 (bar): vert 1D pos/size uniform & **unavailable**
    - vert pos meaningful for different 2 numbers
      - top & bottom components
      - vs bottom component & whole bar for unnormalized
- information density
  - single stacked bar equivalent to full pie
    - high info density: narrow rectangle vs large circle





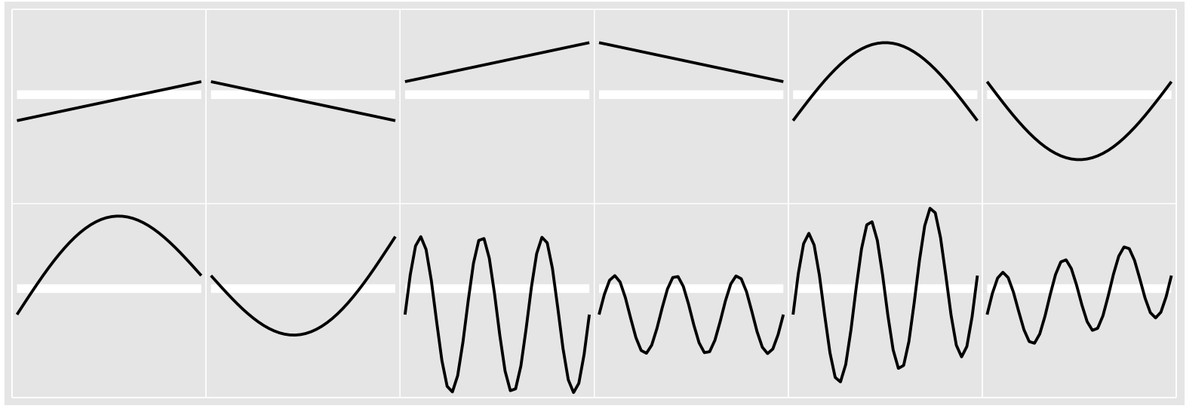


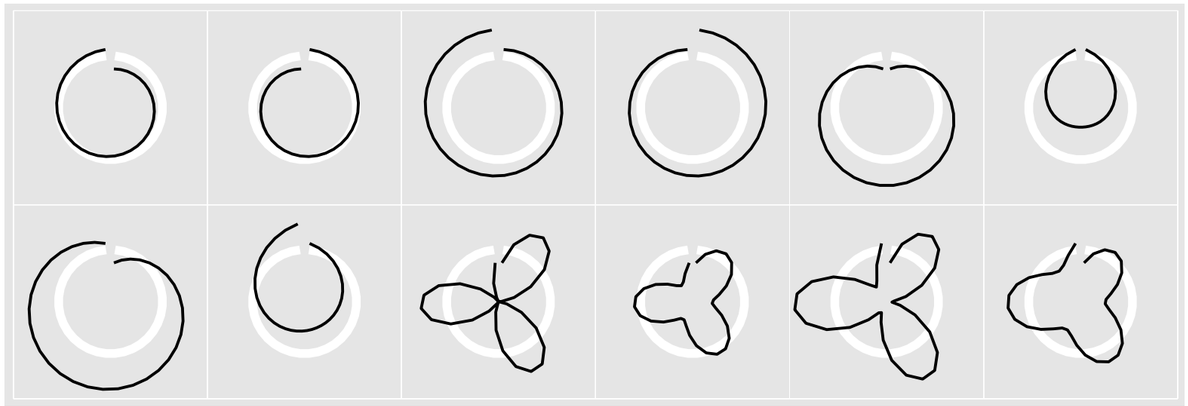
http://bl.ocks.org/mbostock/3886208, *http://bl.ocks.org/mbostock/3887235*, *http://bl.ocks.org/mbostock/3886394*

Idiom: **glyphmaps**

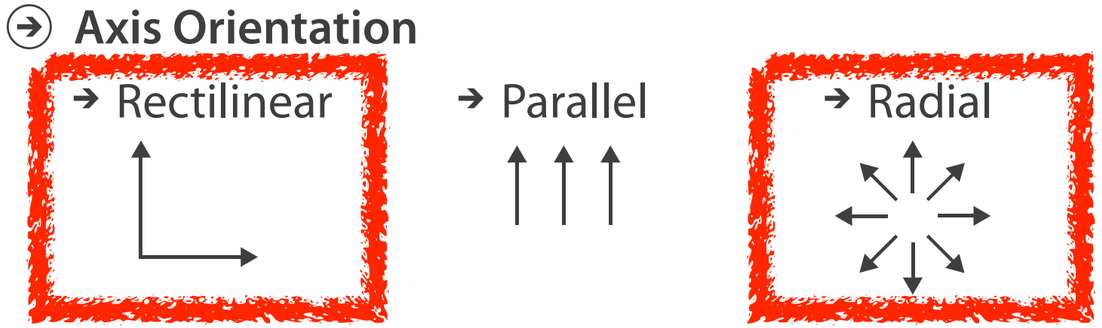
- rectilinear good for linear vs nonlinear trends

- radial good for cyclic patterns
  - evaluating periodicity



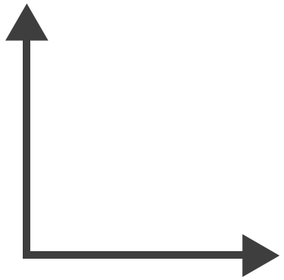


[Glyph-maps for Visually Exploring Temporal Patterns in Climate Data and Models. Wickham, Hofmann, Wickham, and Cook. Environmetrics 23:5 (2012), 382–393.]

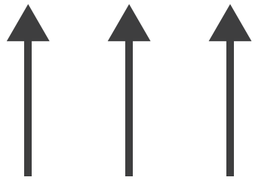


Axis Orientation

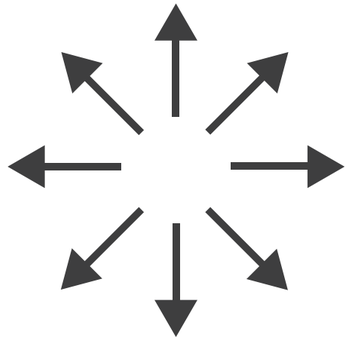
- Rectilinear



- Parallel



- Radial



Idiom: **SPLOM**

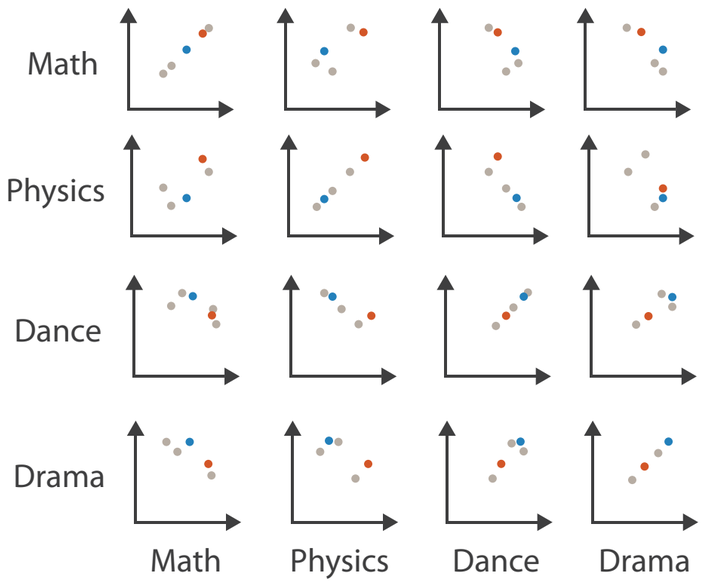
- scatterplot matrix (SPLOM)
  - rectilinear axes, point mark
  - all possible pairs of axes
  - scalability
    - one dozen attribs
    - dozens to hundreds of items

Wilkinson et al., 2005

Idioms: **parallel coordinates**

- scatterplot limitation
  - visual representation with orthogonal axes
  - can show only two attributes with vert & horiz position channels

Scatterplot Matrix



Table

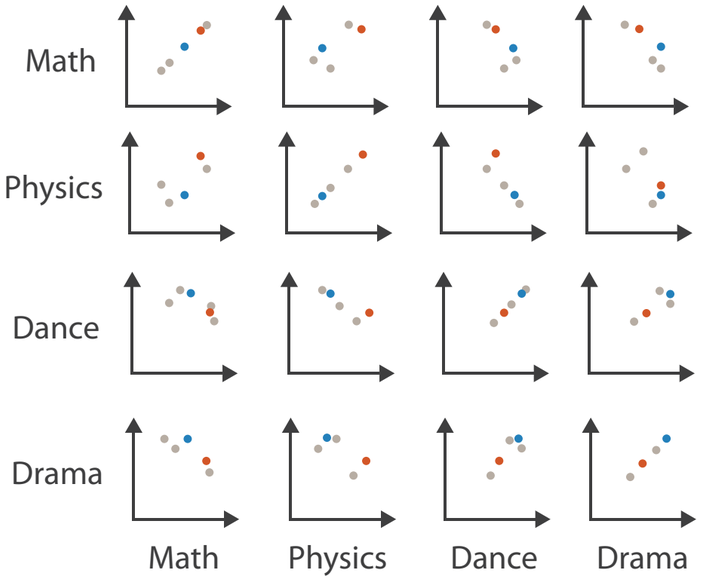
Math	Physics	Dance	Drama
85	95	70	65
90	80	60	50
65	50	90	90
50	40	95	80
40	60	80	90

after [Visualization Course Figures. McGuffin, 2014. http://www.michaelmcguffin.com/courses/vis/]

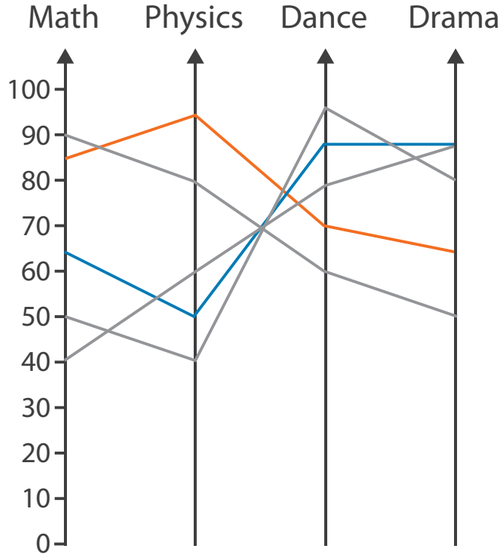
Idioms: **parallel coordinates**

- scatterplot limitation
  - visual representation with orthogonal axes
  - can show only 2 attributes with vert & horiz position channels
- alternative: line up n axes in parallel to show n attributes with (vert) position on each
  - vert position/order for each attribute value
  - item encoded with 1 connecting mark: path through value at each axis
- parallel coordinates
  - parallel axes, path for item
    - jagged, piecewise-linear by segment
    - vs rectilinear/Cartesian axes where item as point
  - scalability
    - dozens of attribs
    - hundreds of items

Scatterplot Matrix



Parallel Coordinates



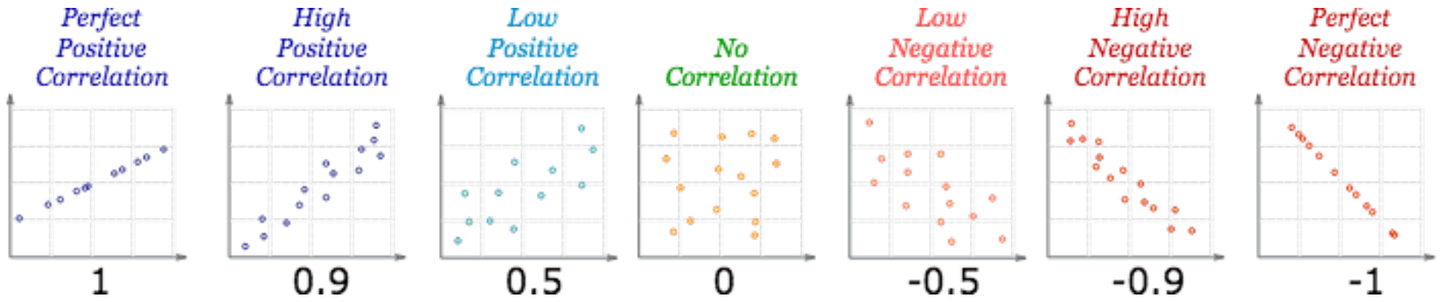
Table

Math	Physics	Dance	Drama
85	95	70	65
90	80	60	50
65	50	90	90
50	40	95	80
40	60	80	90

after [Visualization Course Figures. McGuffin, 2014. http://www.michaelmcguffin.com/courses/vis/]

Task: Correlation

- scatterplot matrix
  - positive correlation
    - diagonal low-to-high
  - negative correlation
    - diagonal high-to-low
  - uncorrelated: spread out
- parallel coordinates
  - positive correlation
    - parallel line segments
  - negative correlation
    - all segments cross at halfway point
  - uncorrelated
    - scattered crossings



https://www.mathsisfun.com/data/scatter-xy-plots.html

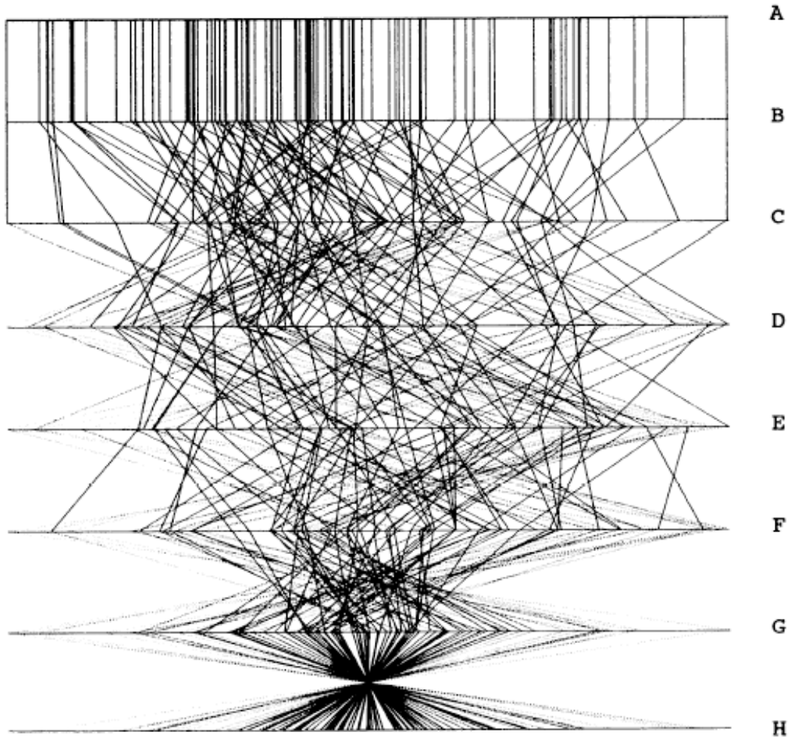
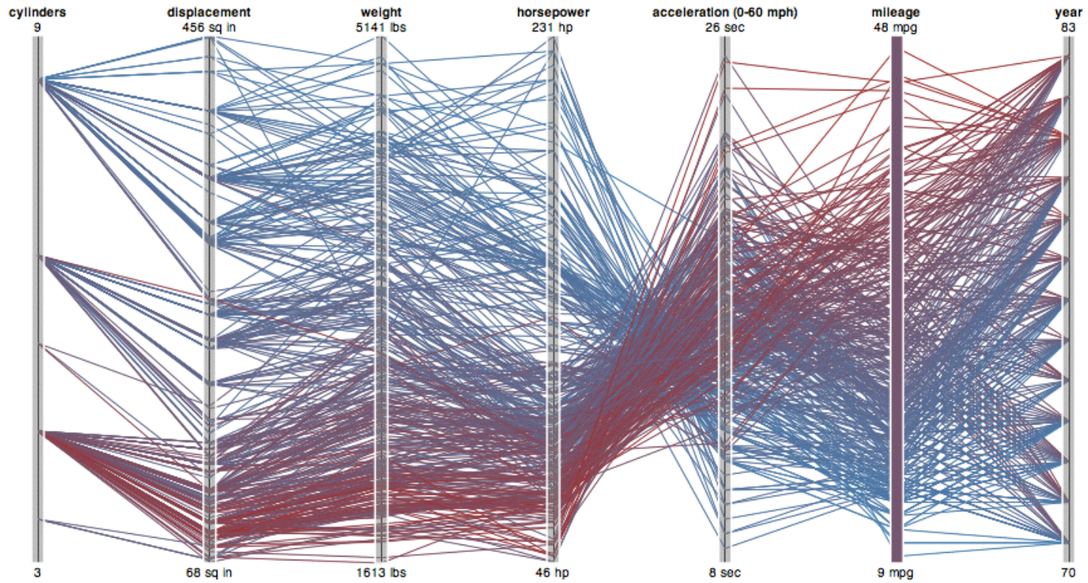


Figure 3. Parallel Coordinate Plot of Six-Dimensional Data Illustrating Correlations of $\rho = 1, .8, .2, 0, -.2, -.8,$ and -1.

[Hyperdimensional Data Analysis Using Parallel Coordinates. Wegman. Journ. American Statistical Association 85:411 (1990), 664–675.]

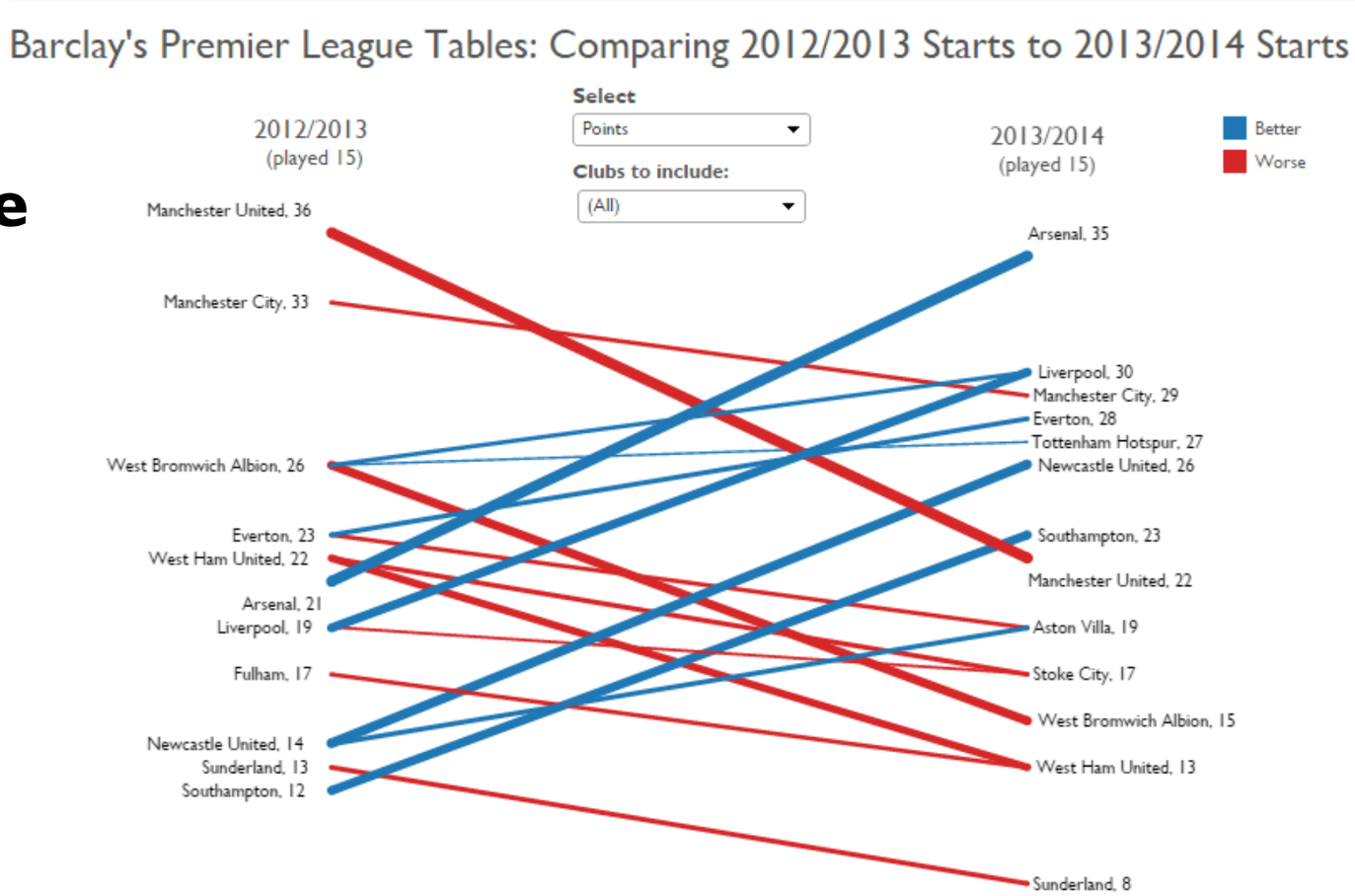
Parallel coordinates, limitations

- visible patterns only between neighboring axis pairs
- how to pick axis order?
  - usual solution: reorderable axes, interactive exploration
  - same weakness as many other techniques
    - downside of interaction: human-powered search
  - some algorithms proposed, none fully solve



Idiom: **Slopegraphs**

- **parallel coordinates special case** with only 2 axes
  - mark: segment
    - connecting mark between 2 values
  - channels
    - vertical pos/order, 1 axis per attrib: express value
    - segment length (1D size, *horiz*): unavailable
    - segment width (1D size, *vert*): change magnitude
    - color
  - task
    - emphasize changes in rank/value



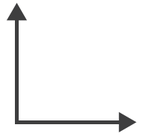
https://public.tableau.com/profile/ben.jones#!/vizhome/Slopegraphs/Slopegraphs

Orientation limitations

- rectilinear: scalability wrt #axes
  - 2 axes best, 3 problematic, 4+ impossible

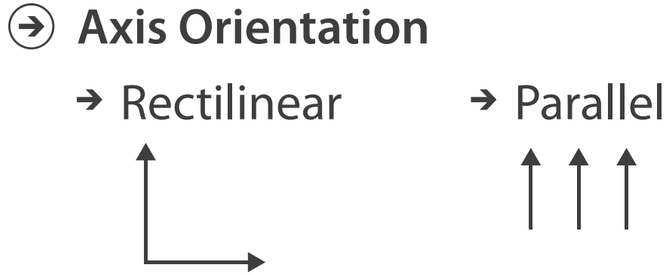
Axis Orientation

→ Rectilinear



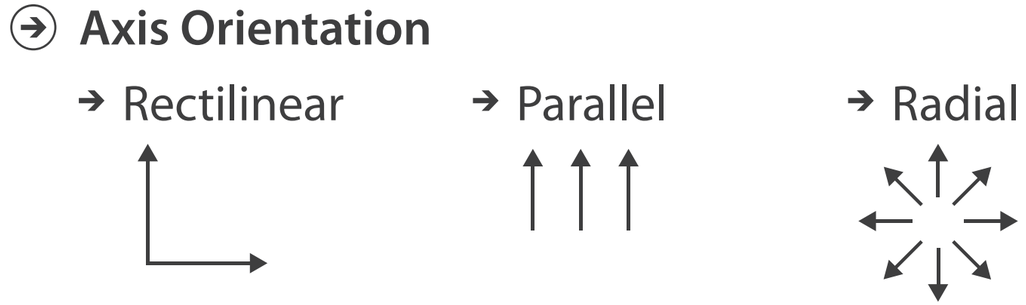
Orientation limitations

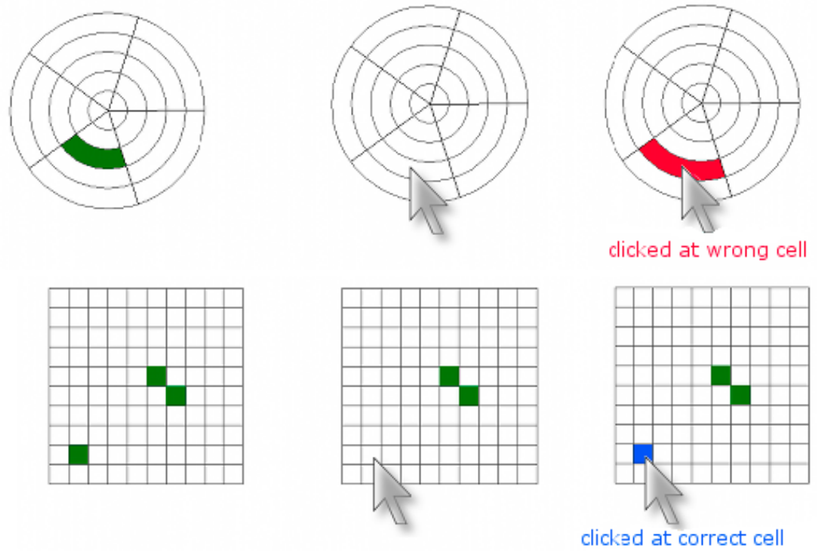
- rectilinear: scalability wrt #axes
  - 2 axes best, 3 problematic, 4+ impossible
- parallel: unfamiliarity, training time



Orientation limitations

- rectilinear: scalability wrt #axes
  - 2 axes best, 3 problematic, 4+ impossible
- parallel: unfamiliarity, training time
- radial: perceptual limits
  - polar coordinate asymmetry
    - angular position lower precision than rectilinear position
    - radial distance (length) lower precision than aligned rectilinear position
    - nonuniform sector size (width & area), depending on radial distance
    - frequently problematic -- but sometimes can exploit on purpose!
      - for 2 attribs of unequal importance
  - strength: cyclic representation (when matching data)



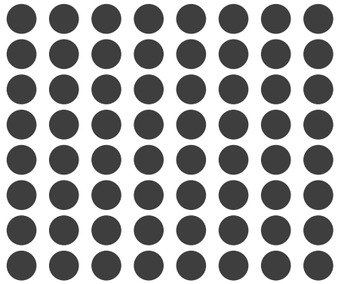


[Uncovering Strengths and Weaknesses of Radial Visualizations - an Empirical Approach. Diehl, Beck and Burch. IEEE TVCG (Proc. InfoVis) 16(6):935–942, 2010.]

Layout density

Layout Density

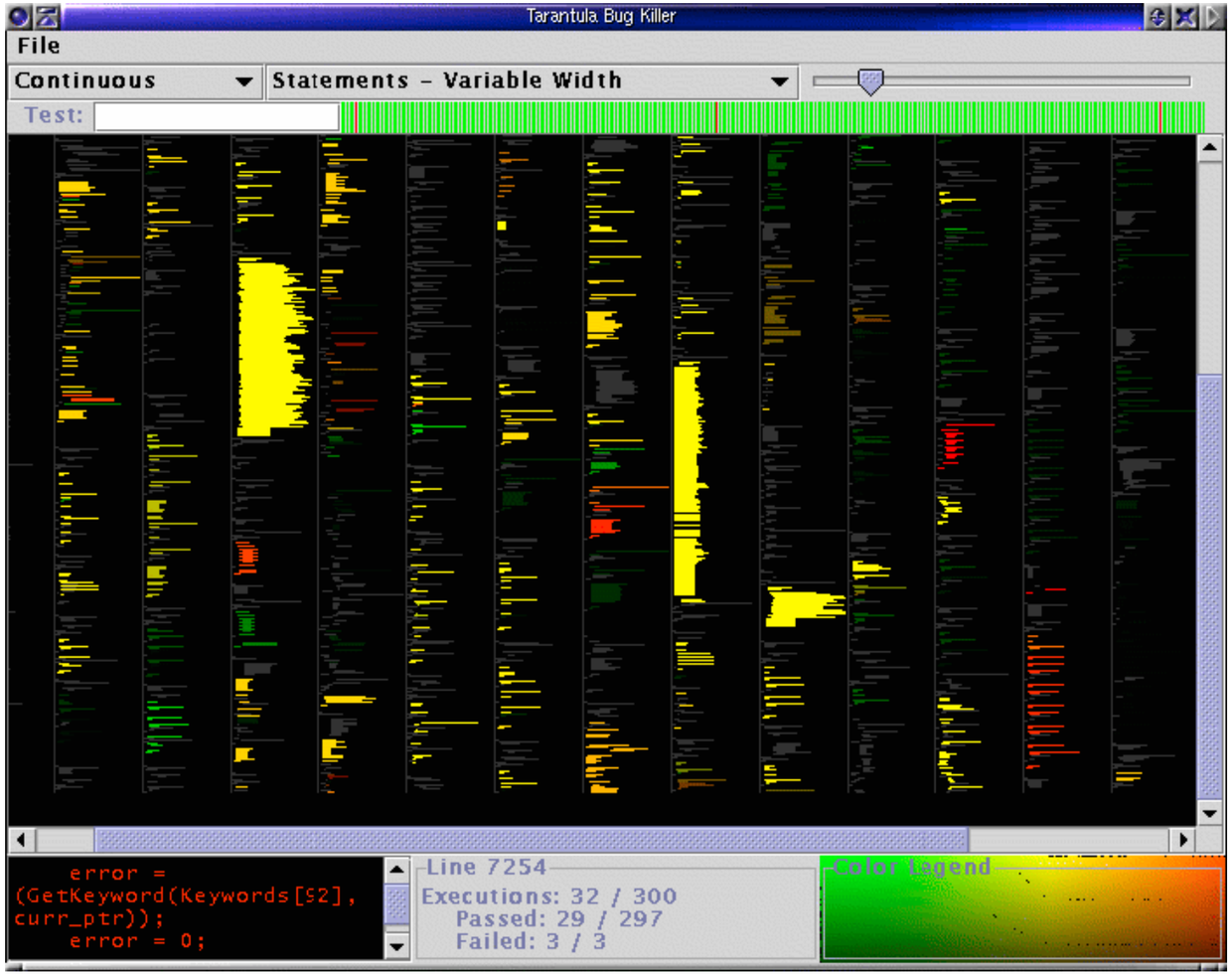
- Dense



- Space-Filling

Idiom: Dense software overviews

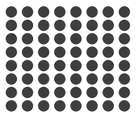
- data: text (LOC)
  - 1 quant attrib: test results
- derived data:
  - 2 quant attribs: min & max char pos
- mark: segment
- channels:
  - color by test results
  - within column, level 1
    - vert pos/order, 1 value: LOC order in program
    - vert shared: yes
    - vert size: **unavailable** (1 pixel high, by idiom design)
    - horiz pos/order, 2 values: min/max char
    - horiz length (1D size): extent
      - **not aligned**, extent != max
  - across columns (global), level 2
    - scanline wrap to fit long vertical line into square
- scalability: 10K+ lines



[Visualization of test information to assist fault localization. Jones, Harrold, Stasko. Proc. ICSE 2002, p 467-477.]

Layout Density

→ Dense

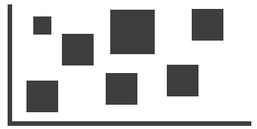


Arrange tables

Express Values

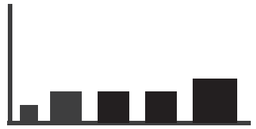
Separate, Order, Align Regions

Separate



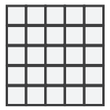
Order

Align



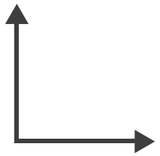
1 Key
List

2 Keys
Matrix



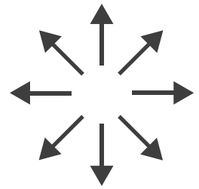
Axis Orientation

Rectilinear



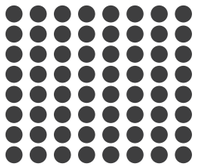
Parallel

Radial



Layout Density

Dense



Manipulate

How to handle complexity: 1 previous strategy

➔ *Derive*

- derive new data to show within view

How to handle complexity: 1 previous strategy + 2 more

→ *Derive*

- derive new data to show within view
- change view over time
- facet across multiple views

Manipulate

→ Change

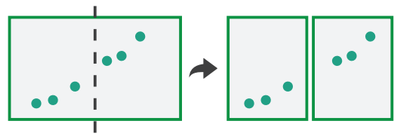
→ Select

→ Navigate

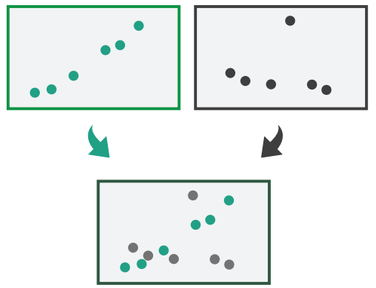
Facet

→ Juxtapose

→ Partition



→ Superimpose



Manipulate View

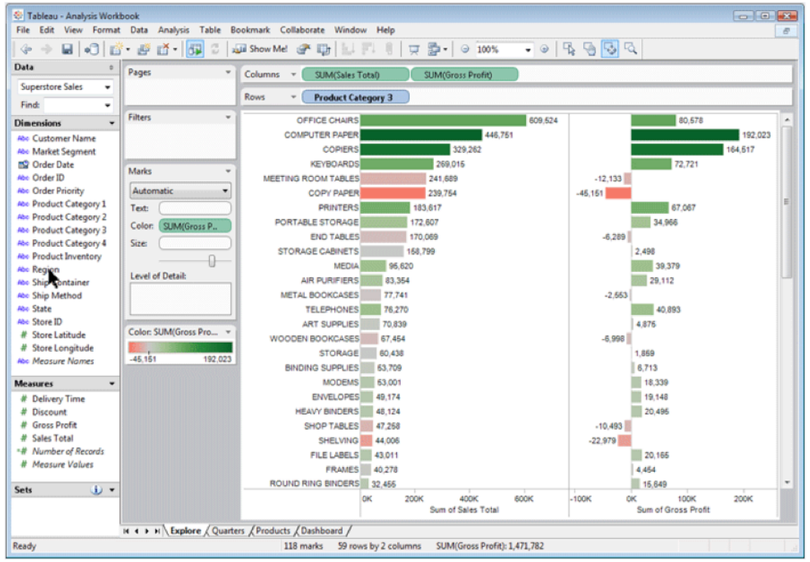
Manipulate

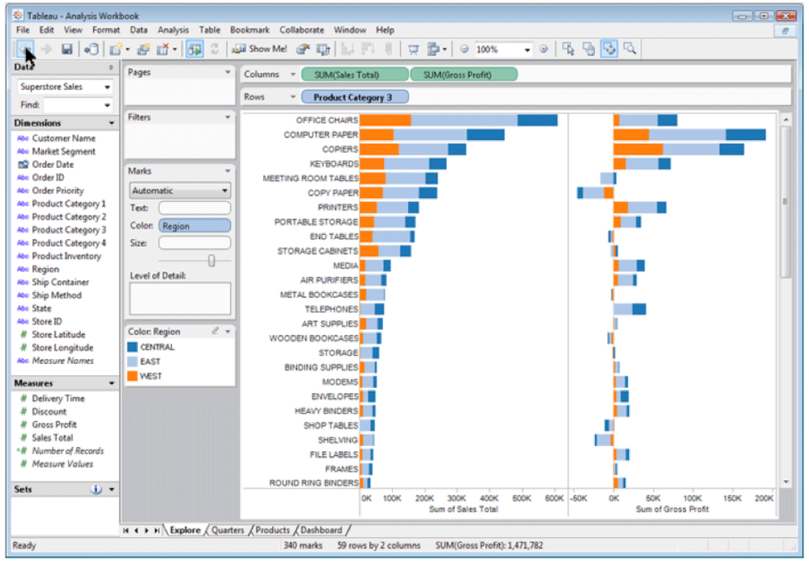
- Change over Time

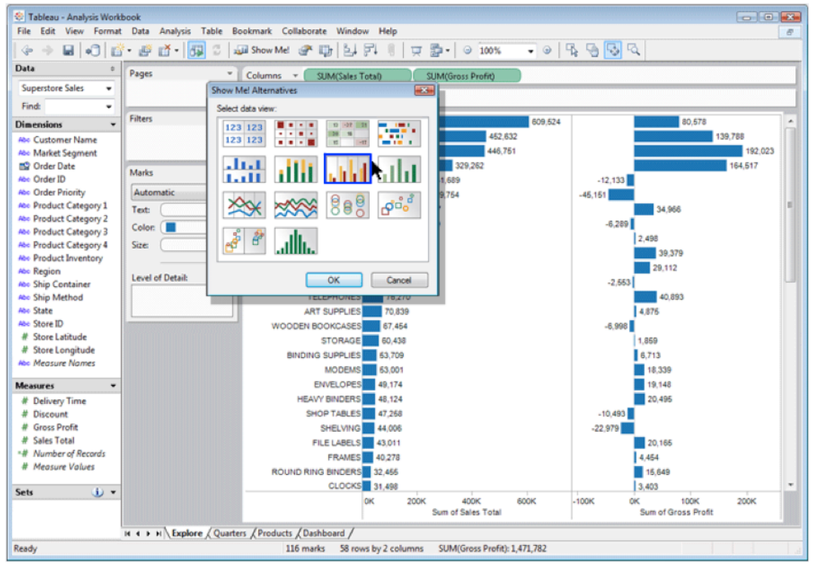
Change over time

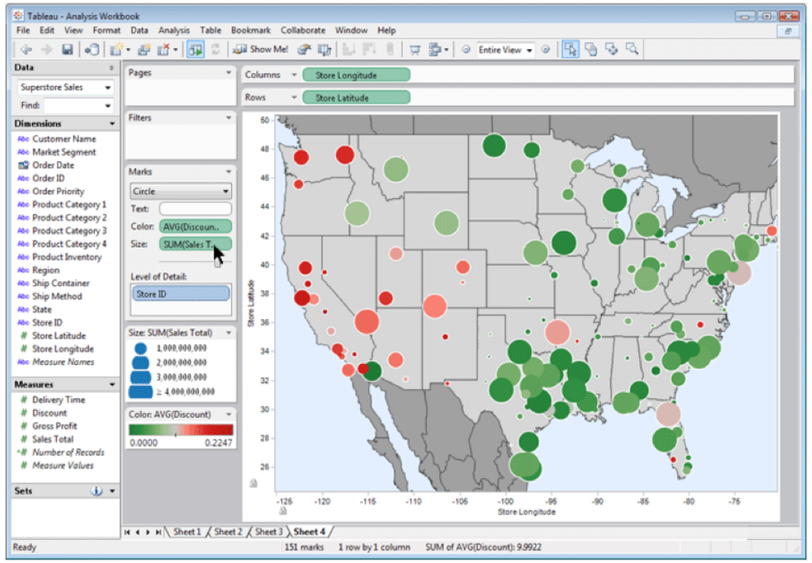
- change any of the other choices
  - encoding itself
  - parameters
  - arrange: rearrange, reorder
  - aggregation level, what is filtered...
  - interaction entails change
- powerful & flexible

Idiom: **Re-encode**





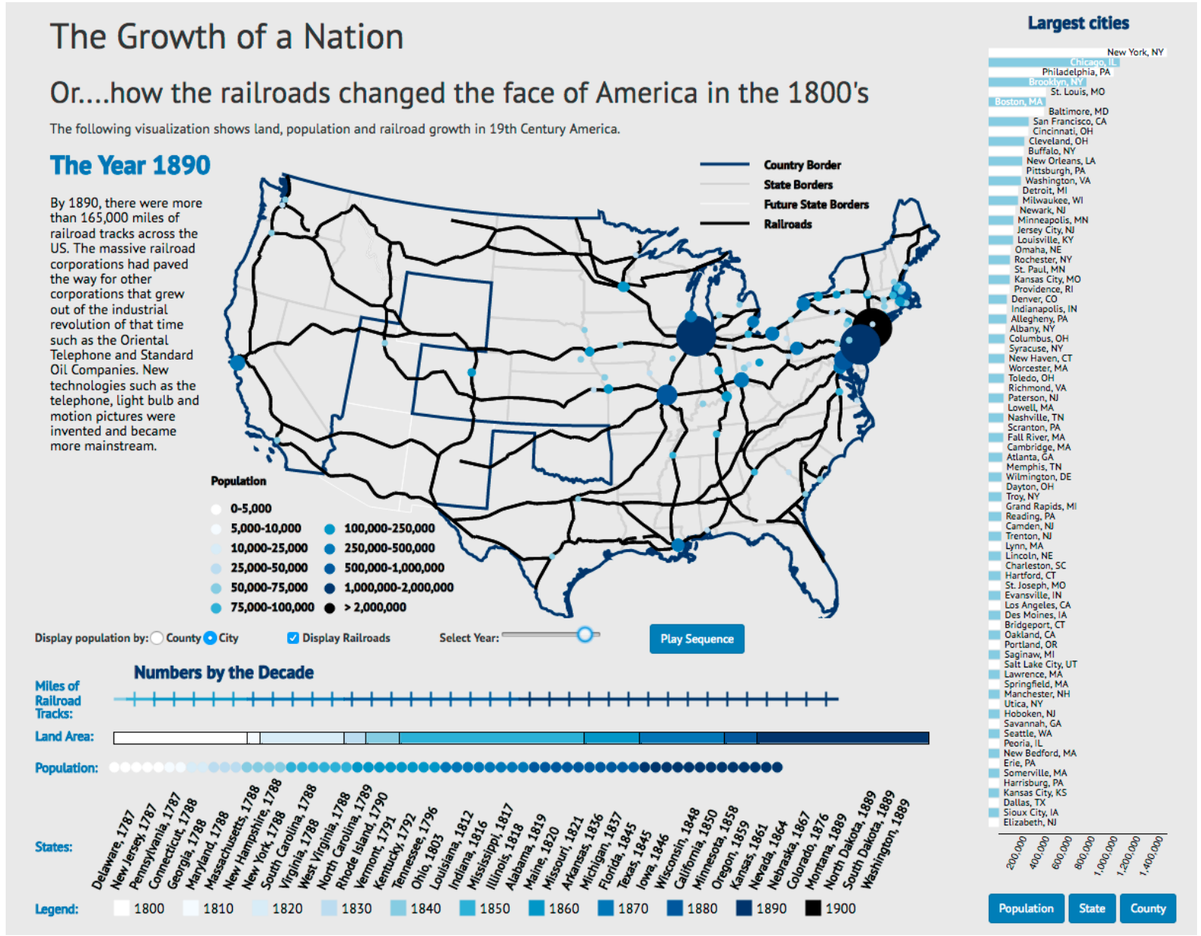




made with Tableau, http://tableausoftware.com

Idiom: **Change parameters**

- widgets and controls
  - sliders, buttons, radio buttons, checkboxes, dropdowns/comboboxes
- pros
  - clear affordances, self-documenting (with labels)
- cons
  - uses screen space
- design choices
  - separated vs interleaved
    - controls & canvas

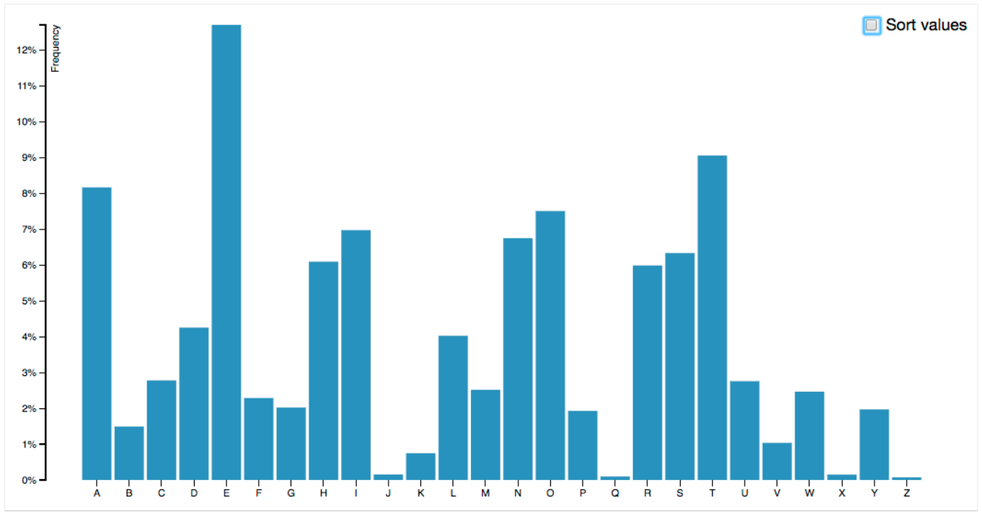


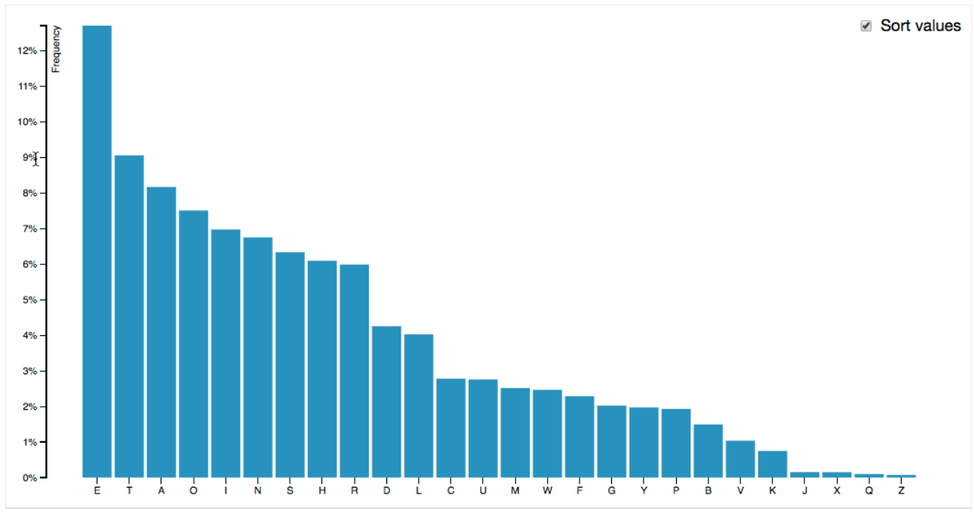
[Growth of a Nation](http://laurenwood.github.io/)

made with D3

Idiom: **Change order/arrangement**

- what: simple table
- how: data-driven reordering
- why: find extreme values, trends





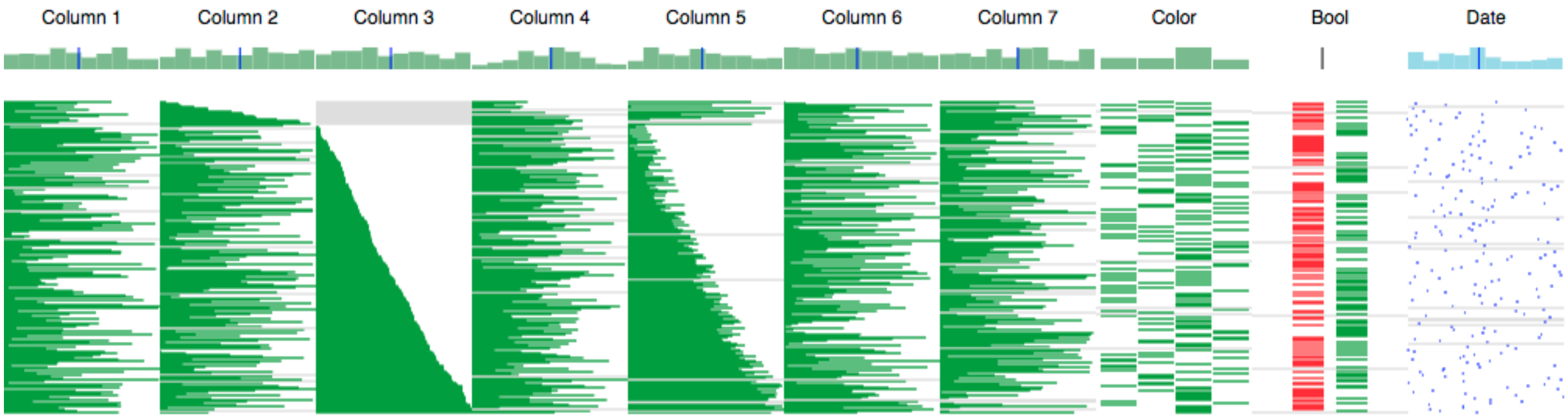
[Sortable Bar Chart] https://observablehq.com/@d3/sortable-bar-chart

made with D3

Idiom: **Reorder**

System: **DataStripes**

- what: table with many attributes
- how: data-driven reordering by selecting column
- why: find correlations between attributes



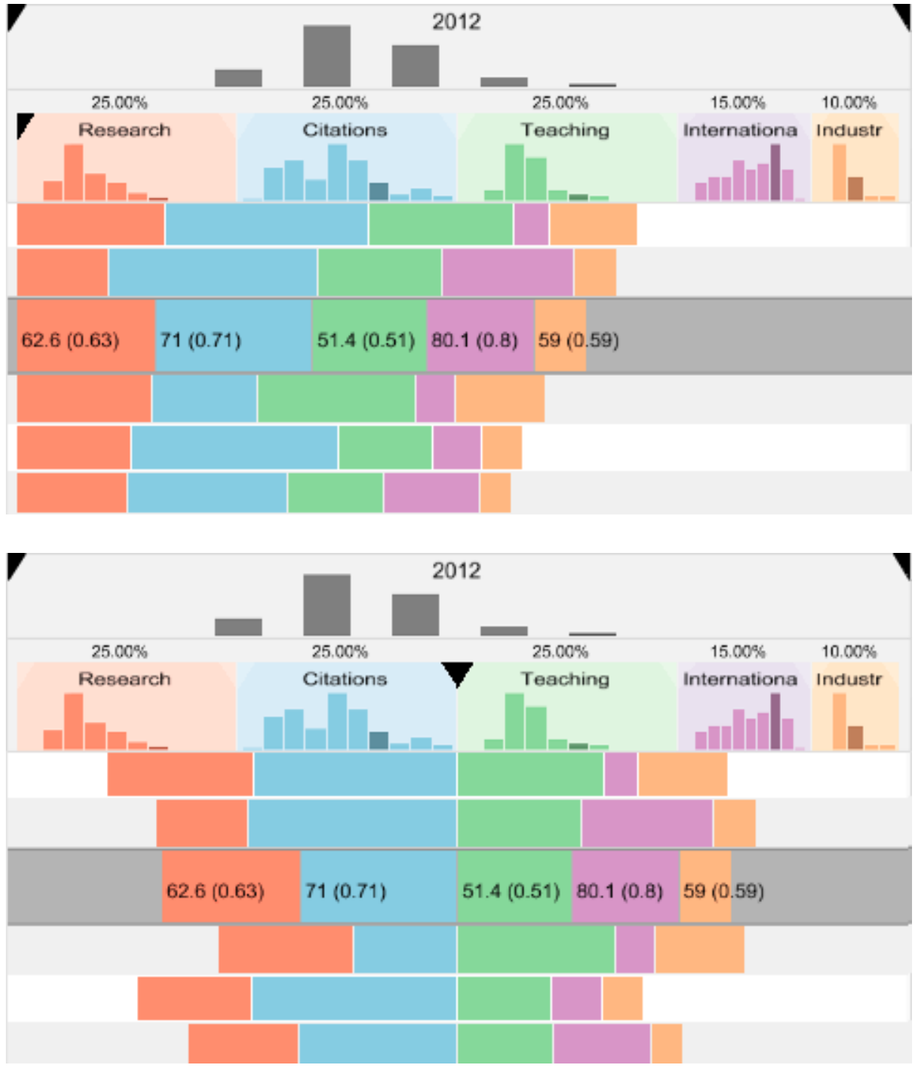
[http://carlmanaster.github.io/datastripes/]

made with D3

Idiom: **Change alignment**

System: **LineUp**

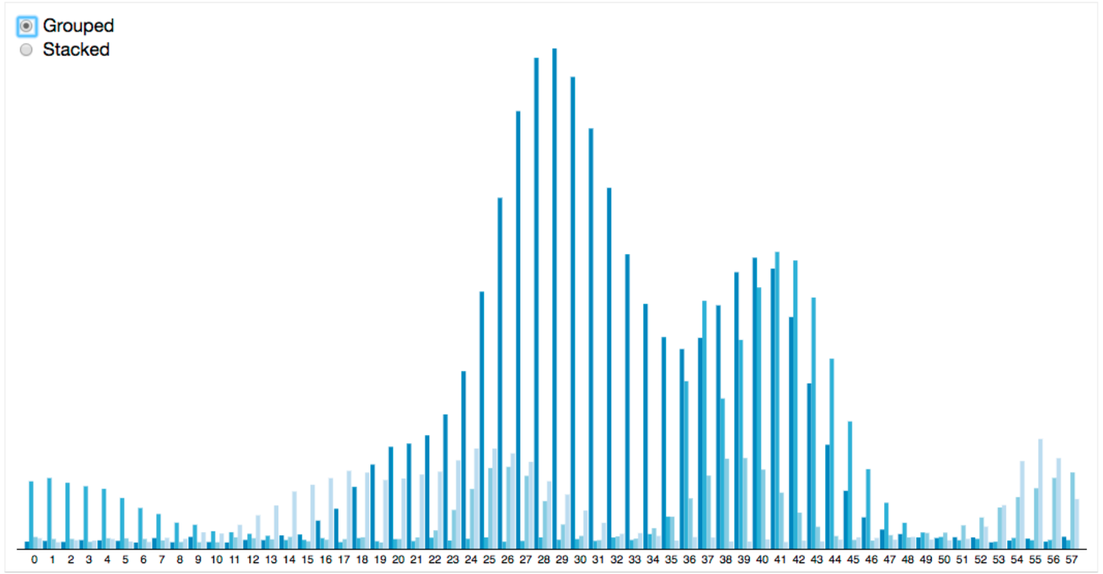
- stacked bars
  - easy to compare
    - first segment
    - total bar
- align to different segment
  - supports flexible comparison

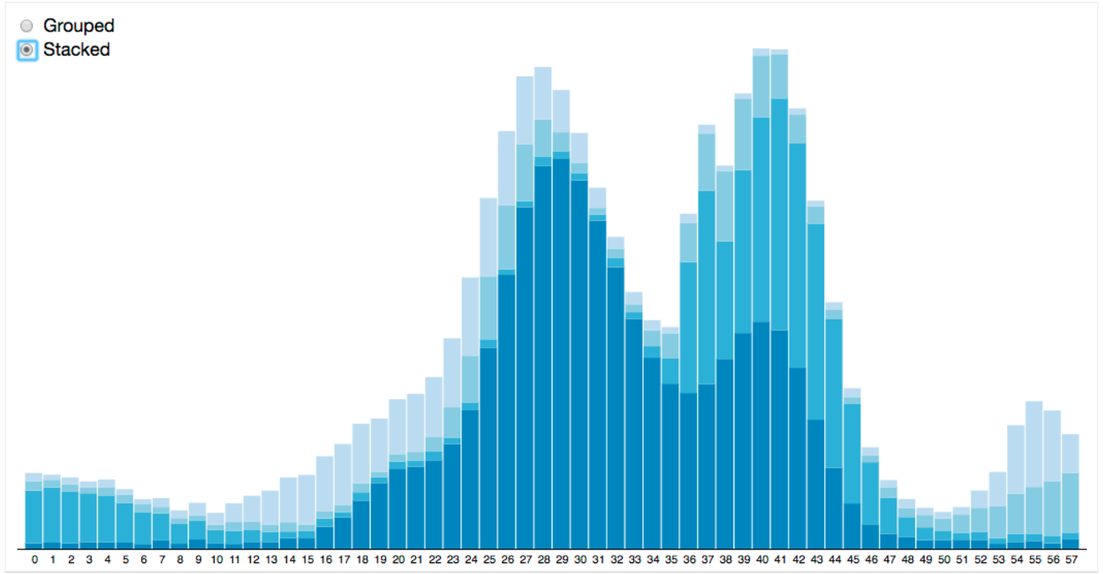


[LineUp: Visual Analysis of Multi-Attribute Rankings. Gratzl, Lex, Gehlenborg, Pfister, and Streit. IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis 2013) 19:12 (2013), 2277–2286.]

Idiom: **Animated transitions - visual encoding**

- smooth transition from one state to another
  - alternative to jump cuts, supports item tracking
    - best case for animation
  - staging to reduce cognitive load

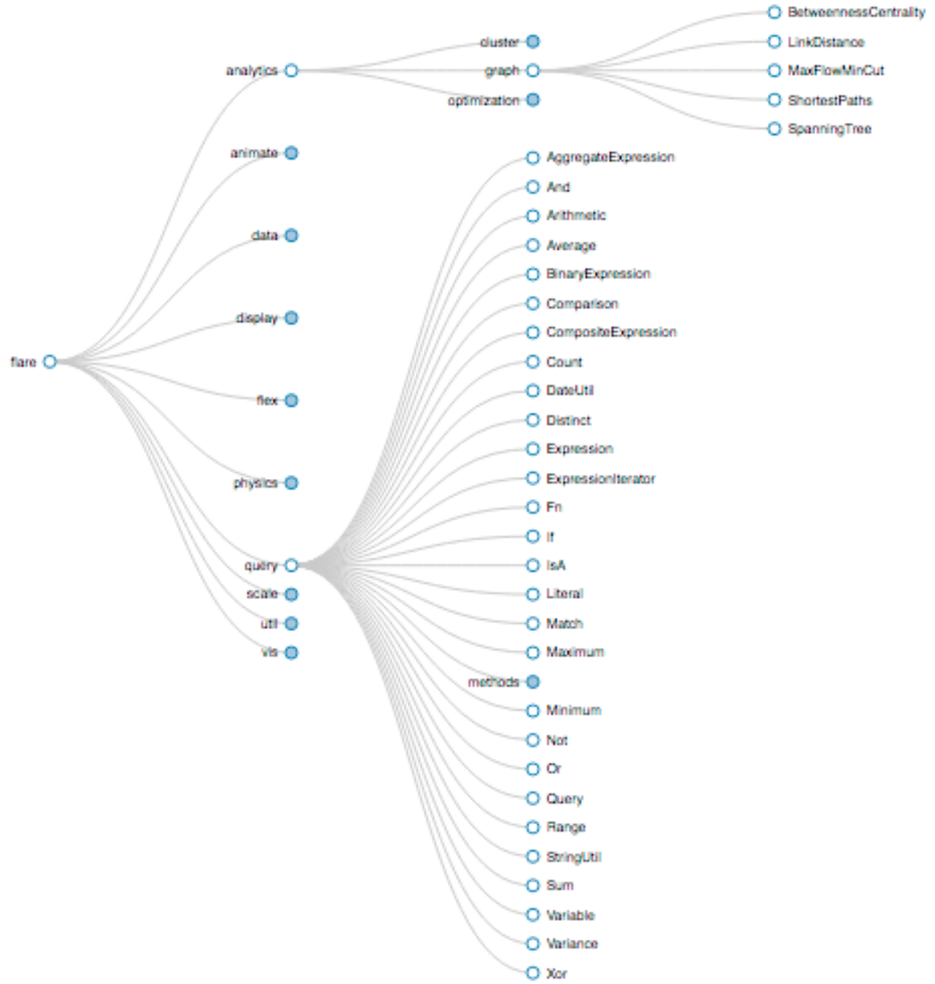




[Stacked to Grouped Bars] https://observablehq.com/@d3/stacked-to-grouped-bars

Idiom: **Animated transition - tree detail**

- animated transition
  - network drilldown/rollup

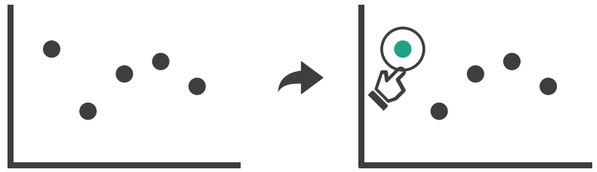


[Collapsible Tree] https://observablehq.com/@d3/collapsible-tree

Manipulate

- Change over Time

- Select



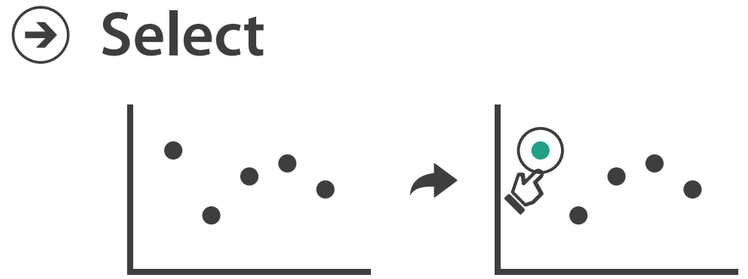
Interaction technology

- what do you design for?
  - mouse & keyboard on desktop?
    - large screens, hover, multiple clicks
  - touch interaction on mobile?
    - small screens, no hover, just tap
  - gestures from video / sensors?
    - ergonomic reality vs movie bombast
  - eye tracking?

Data visualization and the news - Gregor Aisch (37 min)
vimeo.com/182590214

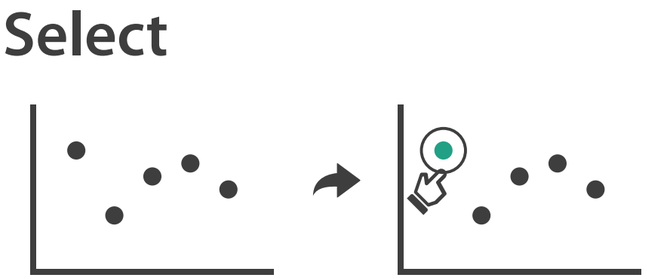
I Hate Tom Cruise - Alex Kauffmann (5 min)
www.youtube.com/watch?v=QXLfT9sFcbc

Selection



- selection: basic operation for most interaction
- design choices
  - how many selection types?
    - interaction modalities
      - click/tap (heavyweight) vs hover (lightweight but not available on most touchscreens)
      - multiple click types (shift-click, option-click, …)
      - proximity beyond click/hover (touching vs nearby vs distant)
    - application semantics
      - adding to selection set vs replacing selection
      - can selection be null?
        - ex: toggle so nothing selected if click on background
      - primary vs secondary (ex: source/target nodes in network)
      - group membership (add/delete items, name group, …)

Highlighting

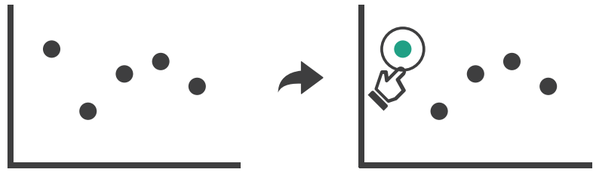


- highlight: change visual encoding for selection targets
  - visual feedback closely tied to but separable from selection (interaction)
- design choices: typical visual channels
  - change item color
    - but hides existing color coding
  - add outline mark
  - change size (ex: increase outline mark linewidth)
  - change shape (ex: from solid to dashed line for link mark)
- unusual channels: motion
  - motion: usually avoid for single view
    - with multiple views, could justify to draw attention to other views

Manipulate

Change over Time

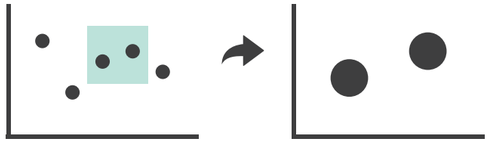
Select



Navigate

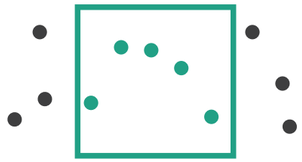
Item Reduction

- *Zoom*
  Geometric or *Semantic*



- *Pan/Translate*

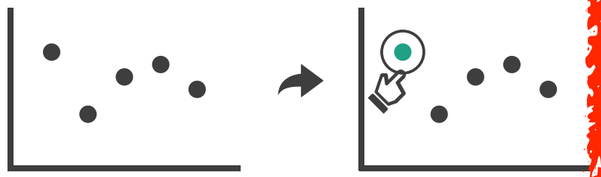
- *Constrained*



Manipulate

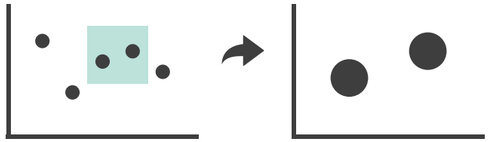
Change over Time

Select



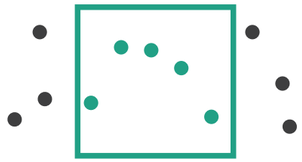
Navigate

- *Zoom*
  Geometric



- *Pan/Translate*

- *Constrained*



Navigate: Changing viewpoint/visibility

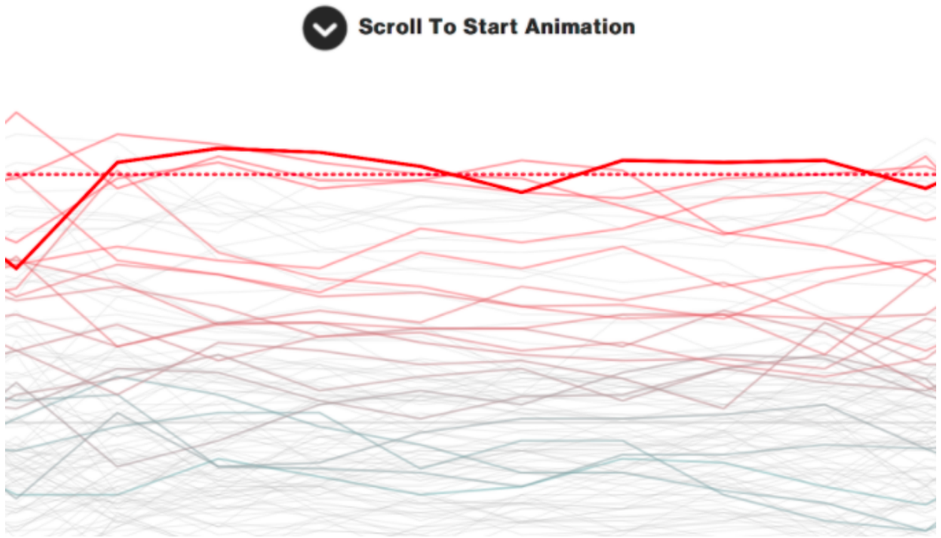
- change viewpoint
  - changes which items are visible within view
- camera metaphor
  - pan/translate/scroll
    - move up/down/sideways

Navigate

→ *Pan/Translate*

Idiom: **Scrollytelling**

- how: navigate page by scrolling (panning down)
- pros:
  - familiar & intuitive, from standard web browsing
  - linear (only up & down) vs possible overload of click-based interface choices
- cons:
  - full-screen mode may lack affordances
  - scrolljacking, no direct access
  - unexpected behaviour
  - continuous control for discrete steps



[How to Scroll, Bostock](https://bost.ocks.org/mike/scroll/)

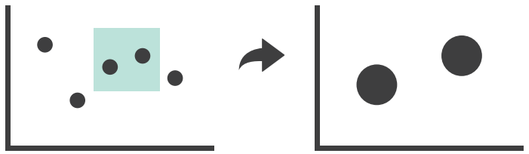
https://eagereyes.org/blog/2016/the-scrollytelling-scourge

Navigate: Changing viewpoint/visibility

- change viewpoint
  - changes which items are visible within view
- camera metaphor
  - pan/translate/scroll
    - move up/down/sideways
  - rotate/spin
    - typically in 3D
  - zoom in/out
    - enlarge/shrink world == move camera closer/further
    - geometric zoom: standard, like moving physical object

Navigate

→ *Zoom*
Geometric



→ *Pan/Translate*

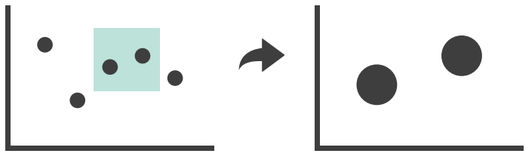
Navigate: Unconstrained vs constrained

- unconstrained navigation
  - easy to implement for designer
  - hard to control for user
    - easy to overshoot/undershoot
- constrained navigation
  - typically uses animated transitions
  - trajectory automatically computed based on selection
    - just click; selection ends up framed nicely in final viewport

Navigate

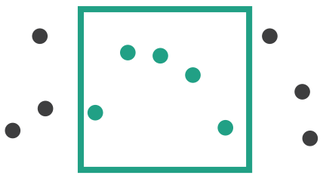
→ Item Reduction

→ *Zoom*
Geometric or *Semantic*



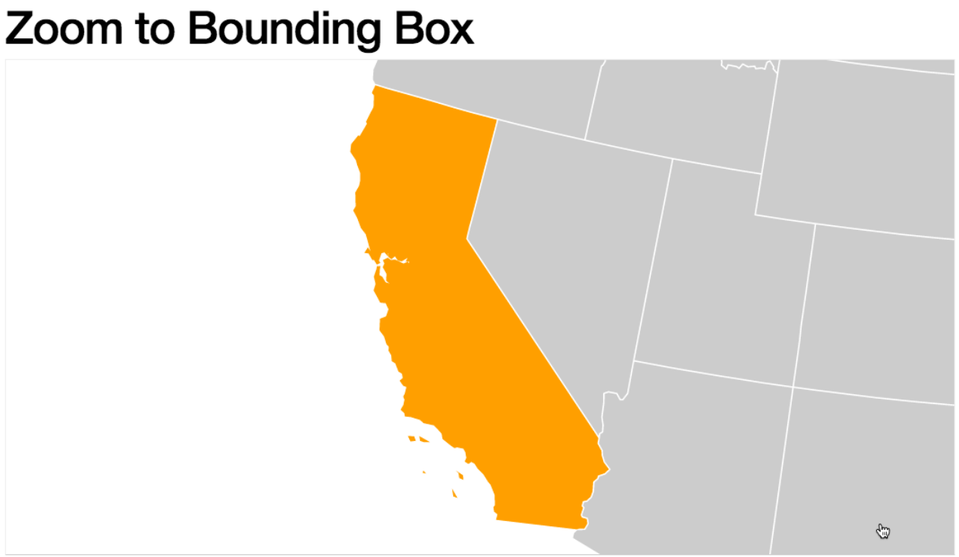
→ *Pan/Translate*

→ *Constrained*



Idiom: **Animated transition + constrained navigation**

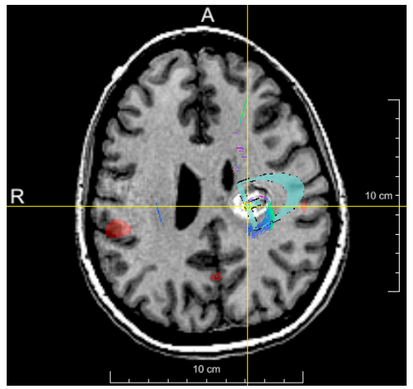
- example: geographic map
  - simple zoom, only viewport changes, shapes preserved

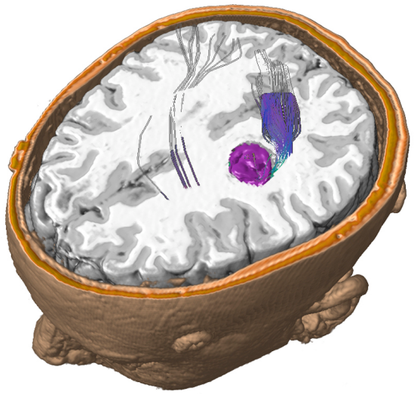


[Zoom to Bounding Box] https://observablehq.com/@d3/zoom-to-bounding-box

Navigate: Reducing attributes

- continuation of camera metaphor
  - slice
    - show only items matching specific value for given attribute: slicing plane
    - axis aligned, or arbitrary alignment
  - cut
    - show only items on far slide of plane from camera
  - project
    - change mathematics of image creation
      - orthographic
      - perspective
      - many others: Mercator, cabinet, ...

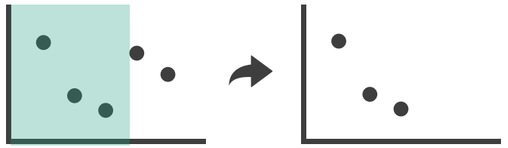




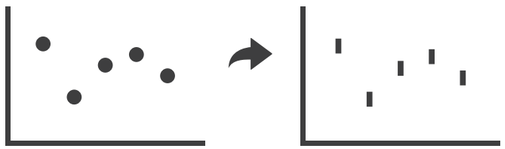
➔ Attribute Reduction

➔ *Slice*

➔ *Cut*



➔ *Project*



[Interactive Visualization of Multimodal Volume Data for Neurosurgical Tumor Treatment. Rieder, Ritter, Raspe, and Peitgen. Computer Graphics Forum (Proc. EuroVis 2008) 27:3 (2008), 1055–1062.]

Interaction benefits

- interaction pros
  - major advantage of computer-based vs paper-based visualization
  - flexible, powerful, intuitive
    - exploratory data analysis: change as you go during analysis process
    - fluid task switching: different visual encodings support different tasks
  - animated transitions provide excellent support
    - empirical evidence that animated transitions help people stay oriented

Interaction limitations

- interaction has a time cost
  - sometimes minor, sometimes significant
  - degenerates to human-powered search in worst case
- remembering previous state imposes cognitive load
- controls may take screen real estate
  - or invisible functionality may be difficult to discover (lack of affordances)
- users may not interact as planned by designer
  - NYTimes logs show ~90% don't interact beyond scrollytelling - Aisch, 2016