13: Reduce Items and Attributes
14: Embed: Focus + Context

Enrico Puppo
Department of Computer Science, Bioengineering, Robotics and Systems Engineering
University of Genova

90529 Data Visualization
10 December 2020

https://2020.aulaweb.unige.it/course/view.php?id=4293

Credits:
material in these slides is partially taken from
- T. Munzner, University of British Columbia
- A. Lex, University of Utah

Idiom design choices: Part 2

Manipulate

- Change

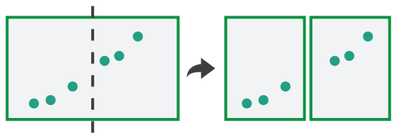
- Select

- Navigate

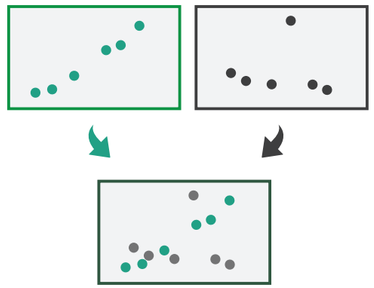
Facet

- Juxtapose

- Partition



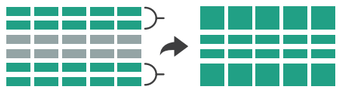
- Superimpose



Reduce

- Filter

- Aggregate



- Embed

Reduce items and attributes

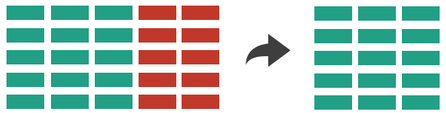
- reduce/increase: inverses
- filter
  - pro: straightforward and intuitive
    - to understand and compute
  - con: out of sight, out of mind
- aggregation
  - pro: inform about whole set
  - con: difficult to avoid losing signal
- not mutually exclusive
  - combine filter, aggregate
  - combine reduce, change, facet

Reducing Items and Attributes

Filter

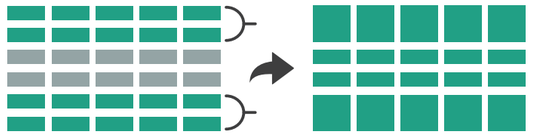
→ Items

→ Attributes

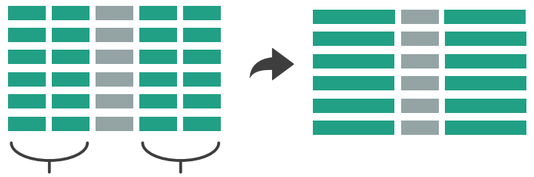


Aggregate

→ Items



→ Attributes



Reduce

Filter

Aggregate

Embed

Filter

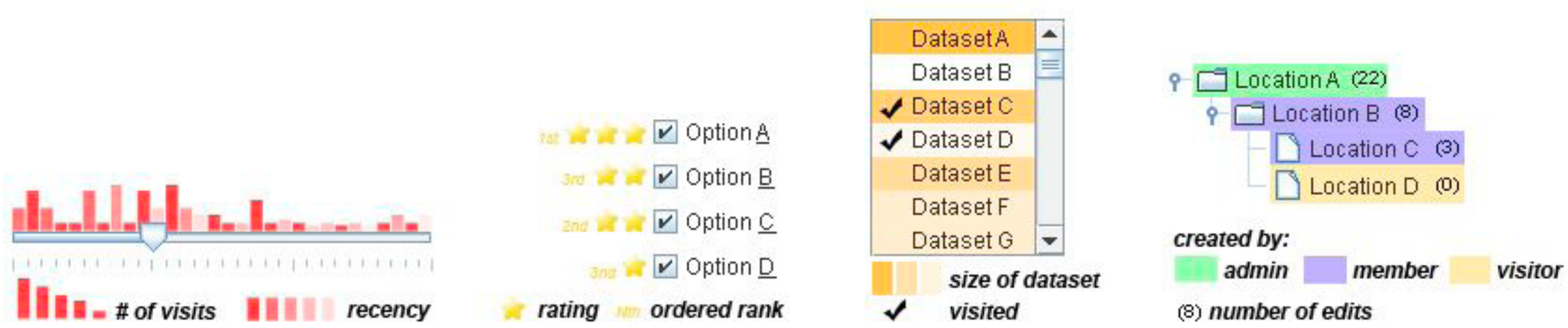
- General idea:
  - reduce visual cluttering
  - simplify view by leaving out some of the (less interesting) items or attributes
- Filter items:
  - focus on items of interest
  - attributes are used as keys for filtering:
    - select categories and/or ranges of values
- Filter attributes:
  - focus on aspects of interest
  - attributes are directly selected/deselected

Dynamic queries vs Filters

- Both coupling between encoding and interaction:
  - user can immediately see the results of an action
- Queries:
  - start with none, add in elements
  - user must know what to look for
- Filters:
  - start with all, remove elements
  - user must know where to focus
- Approach depends also on dataset size:
  - filters usually preferred, because preliminary exploration is supported
  - "start with all" not possible for large datasets: dynamic queries may be better

Idiom: **scented widgets**

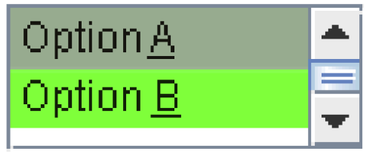
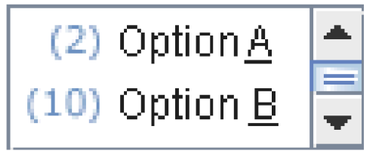
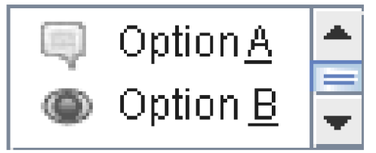
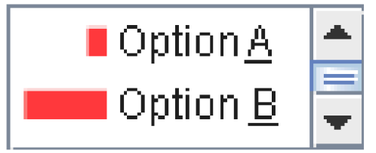
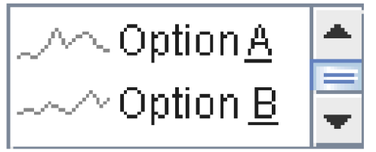
- augment widgets for filtering to show ***information scent***
  - cues to show whether value in drilling down further vs looking elsewhere
- concise, in part of screen normally considered control panel



[Scented Widgets: Improving Navigation Cues with Embedded Visualizations. Willett, Heer, and Agrawala. IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis 2007) 13:6 (2007), 1129–1136.]

Idiom: **scented widgets**

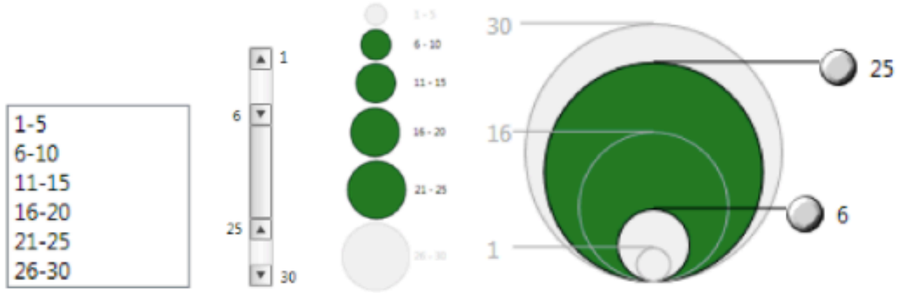
- Marks and/or channels are used to decorate the interaction widgets (buttons, sliders, …)

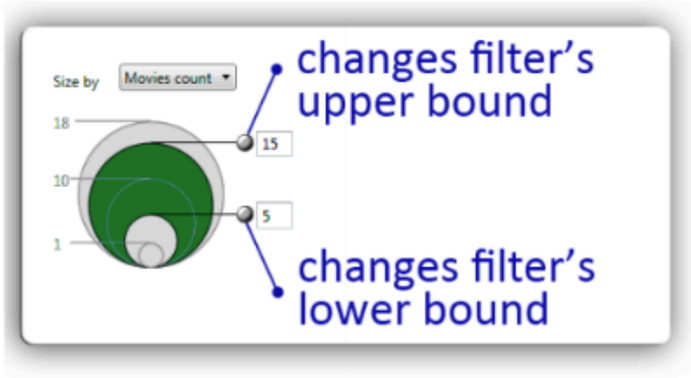
Name	Description	Example
Hue	Varies the hue of the widget (or of a visualization embedded in it)	
Saturation	Varies the saturation of the widget (or of a visualization embedded in it)	
Opacity	Varies the saturation of the widget (or of a visualization embedded in it)	
Text	Inserts one or more small text figures into the widget	
Icon	Inserts one or more small icons into the widget.	
Bar Chart	Inserts one or more small bar chart visualizations into the widget	
Line Chart	Inserts one or more small line chart visualizations into the widget	

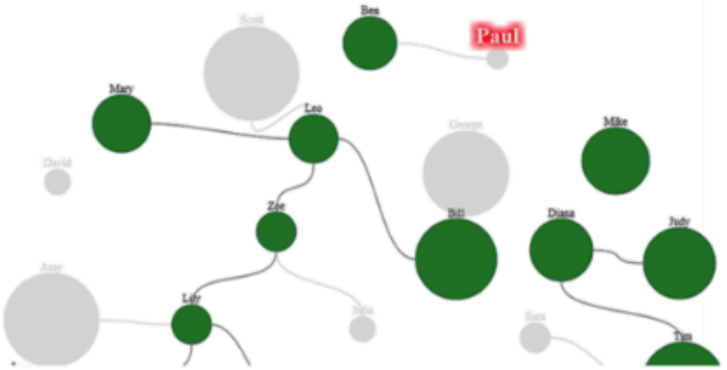
[Scented Widgets: Improving Navigation Cues with Embedded Visualizations. Willett, Heer, and Agrawala. IEEE Trans. Visualization and Computer Graphics (Proc. InfoVis 2007) 13:6 (2007), 1129–1136.]

Idiom: **interactive legends**

- Controls combining the visual representation of static legends with interaction mechanisms of widgets
- Define and control visual display together







Riche 2010

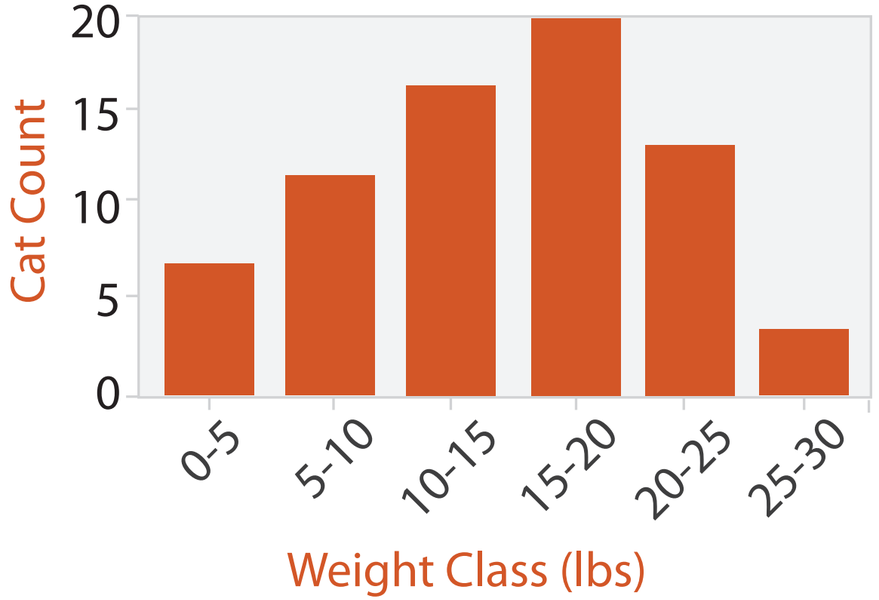
Aggregate

- General idea:
  - represent a group of elements as a new derived element
  - possible for both items and attributes
- Basic derived attributes:
  - average, minimum, maximum, count, sum
- Aggregate items can provide selection keys for filtering
- Interactive vs automatic aggregation/deaggregation

- Pro: helps summarizing and finding trends; suitable to hierarchical approach (on-the-fly aggregation at different LODs)
- Con: loss of detail, possibly misleading (see Ancombe's quartet)

Idiom: **histogram**

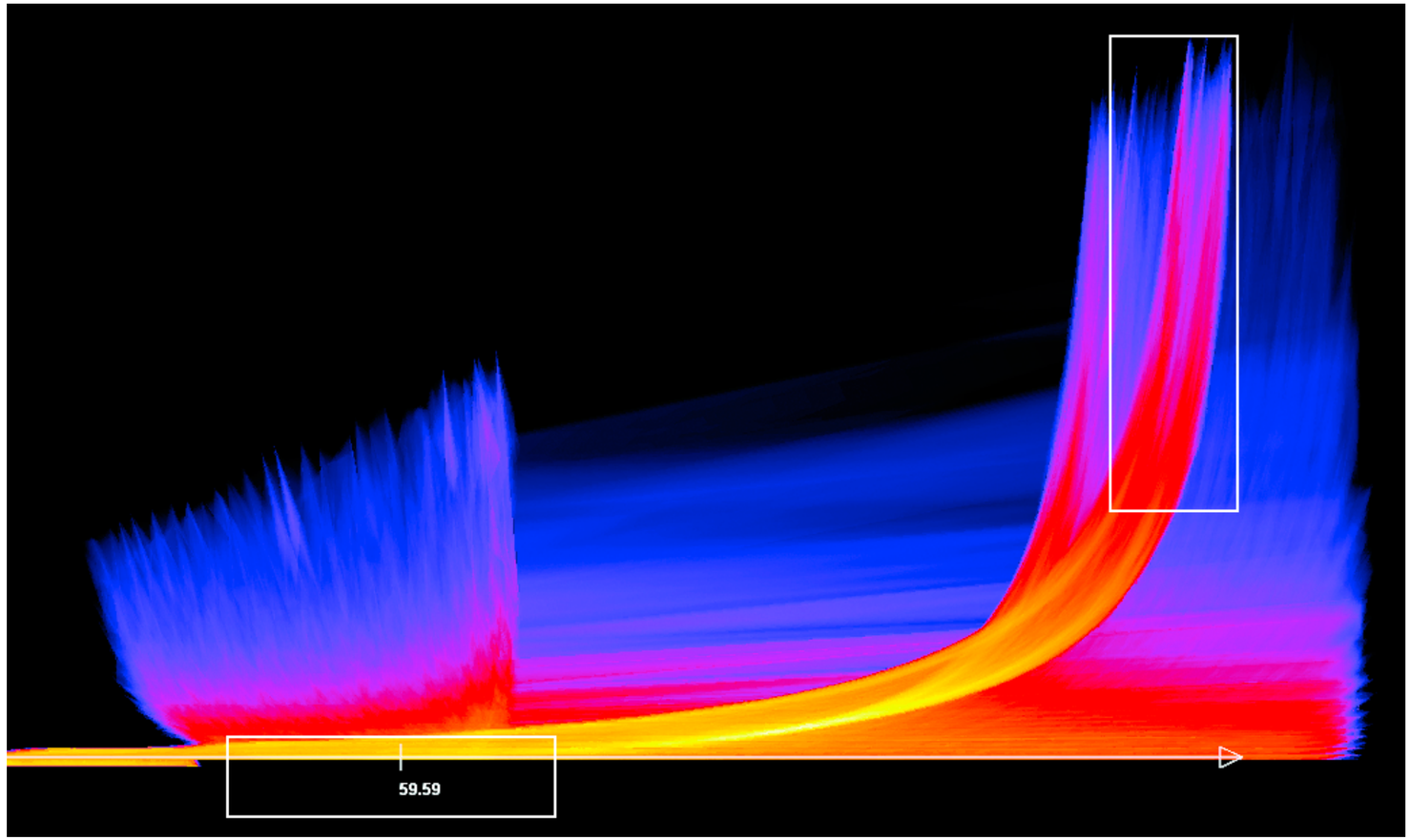
- static item aggregation
- task: find distribution
- data: table
- derived data
  - new table: keys are bins, values are counts
- bin size crucial
  - pattern can change dramatically depending on discretization
  - opportunity for interaction: control bin size on the fly



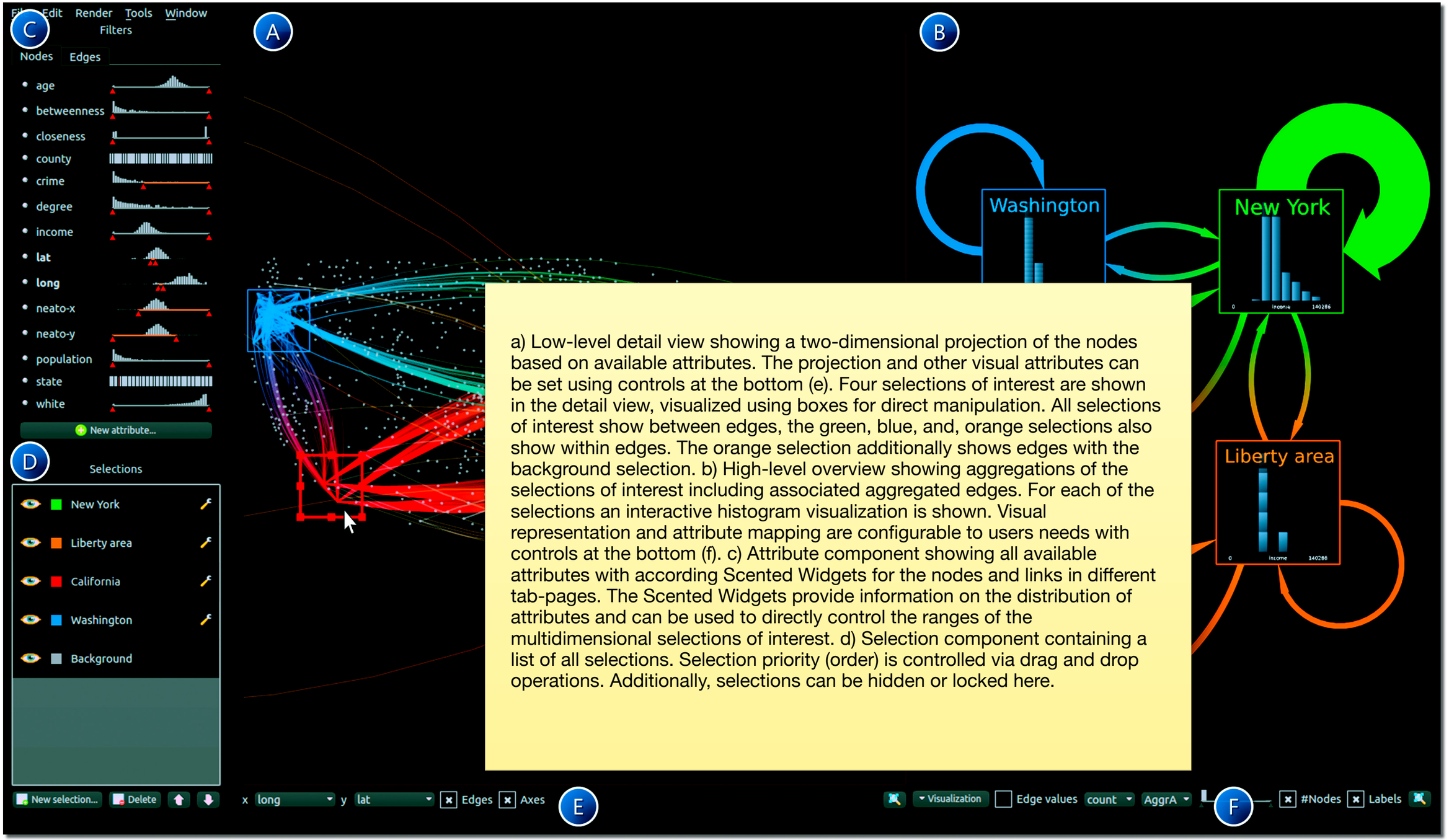
- not a barchart!!!
  - each mark represents count of items in a given range

Continuous scatterplot

- static item aggregation
- data: table
- derived data: table
  - key attribs x,y for pixels
  - quant attrib: overplot density
- dense space-filling 2D matrix
- color: sequential categorical hue + ordered luminance colormap
- 1D: continuous histogram



[Continuous Scatterplots. Bachthaler and Weiskopf. IEEE TVCG (Proc. Vis 08) 14:6 (2008), 1428–1435. 2008.]



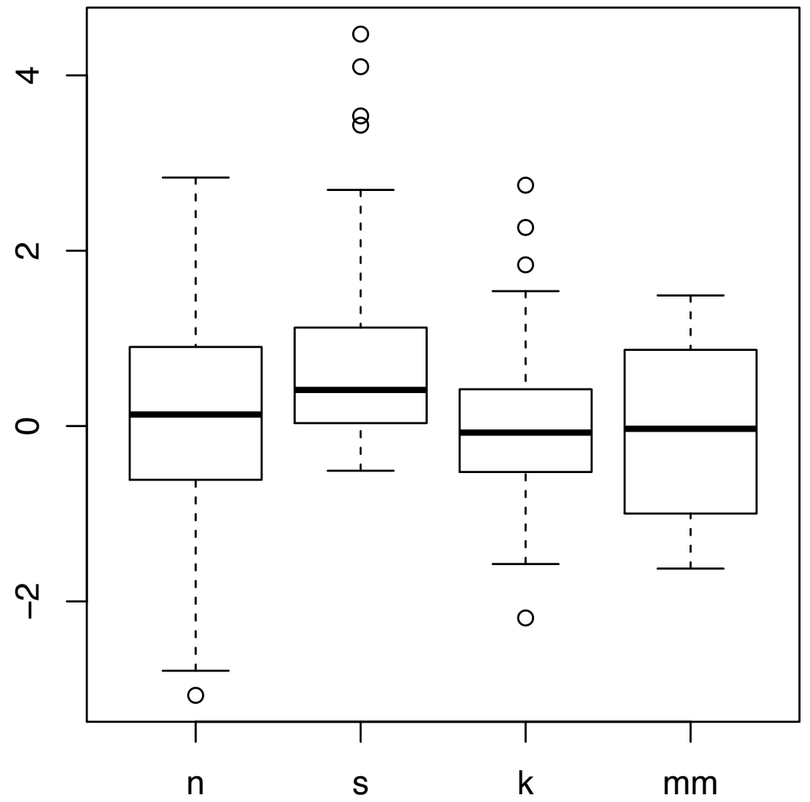
[Multivariate Network Exploration and Presentation: From Detail to Overview via Selections and Aggregations. van den Elzen and van Wijk, TVCG 20(12) 2014.]

Screenshot showing ICLIC and its four views: the attribute view (top left), the image view (top right), the data view (bottom left), and the category view (bottom right). (a) Attributes can be sorted by name, frequency, and relevance. (b) Attributes can be searched by a regular expression. (c) Switch between RSVP and pivot mode. (d) The size of images can be adjusted, increasing or decreasing the number of images shown. (e) Convenience buttons for no/all/invert selection. (f) Search field to select images of which at least one value matches the query. (g) Legend that shows the number of images on each stack. (h) Attributes are presented as scented widgets, striped bars indicate undefined values. (i) Attributes can be dragged onto the axes of the image view. (j) Bars indicate the impact of the filter steps. The filter horizon, between the two steps, separates the pre- and post filters. (k) Selected images are added to, or removed from categories via these buttons. (l) Radio buttons for filtering images (not) in the category. (m) Status bar with detailed statistics about hovered elements.

[ICLIC: Interactive categorization of large image collections. van der Corput and van Wijk. Proc. PacificVis 2016.]

Idiom: **boxplot**

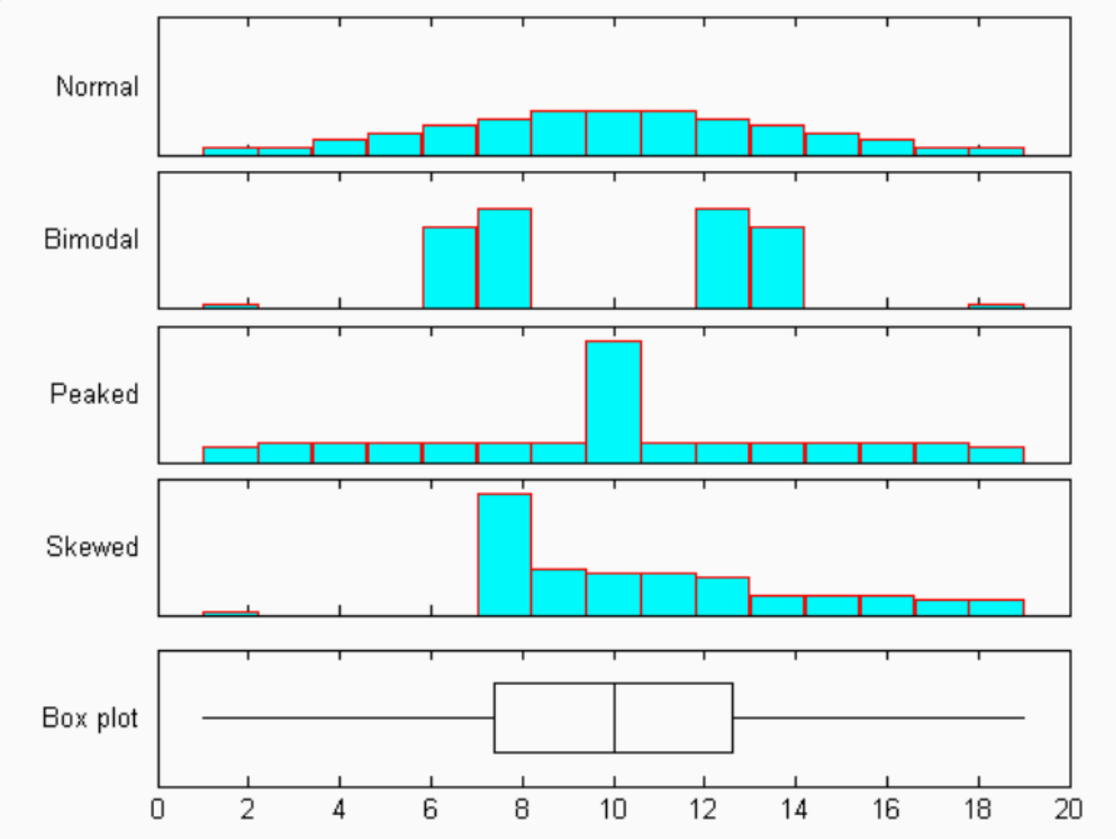
- static item aggregation
- task: find distribution
- data: table
- derived data
  - 5 quant attributes
    - median: central line
    - lower and upper quartile: boxes
    - lower upper fences: whiskers
      - values beyond which items are outliers
  - outliers beyond fence cutoffs explicitly shown

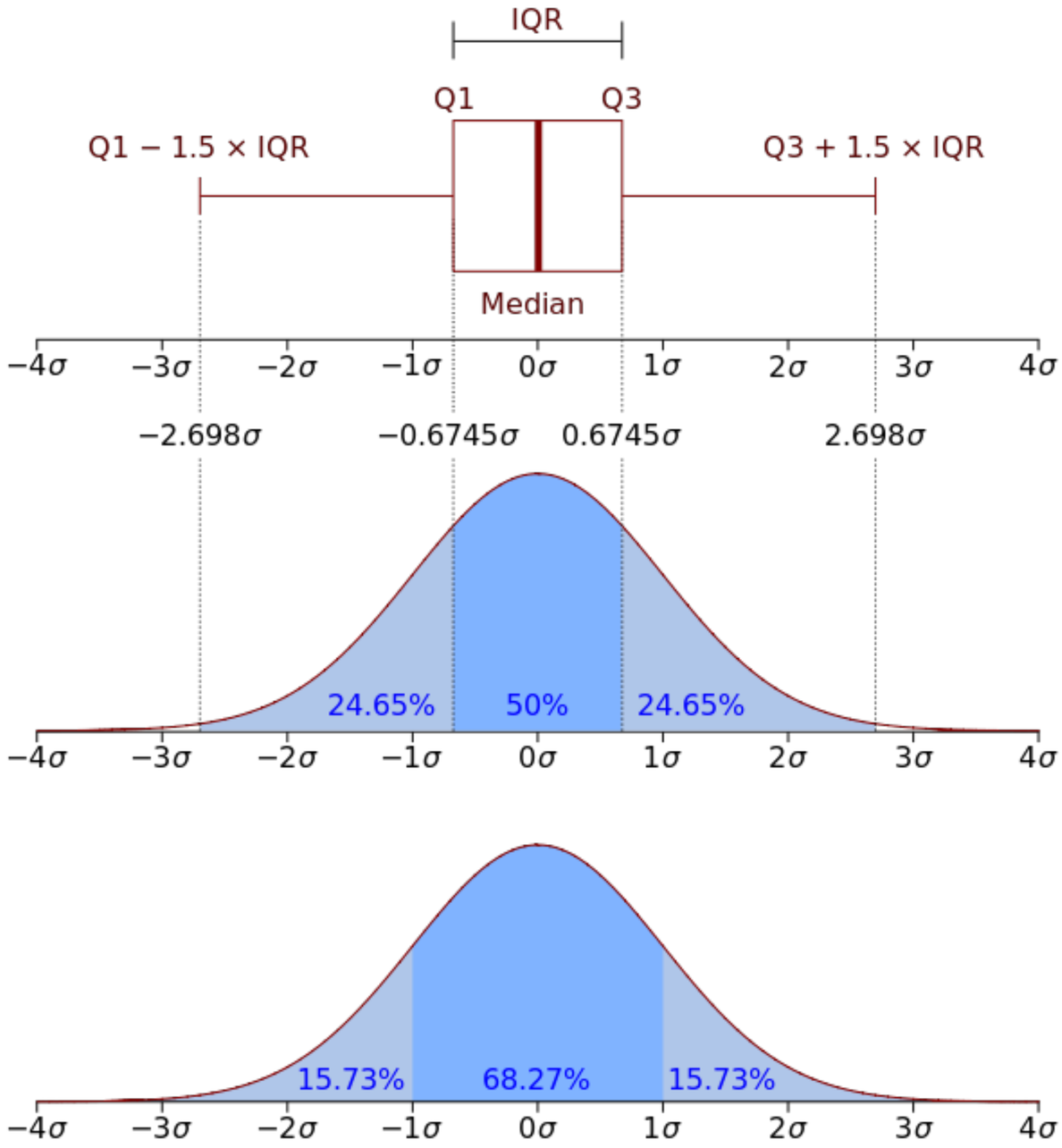


[40 years of boxplots. Wickham and Stryjewski. 2012. had.co.nz]

Idiom: **boxplot**

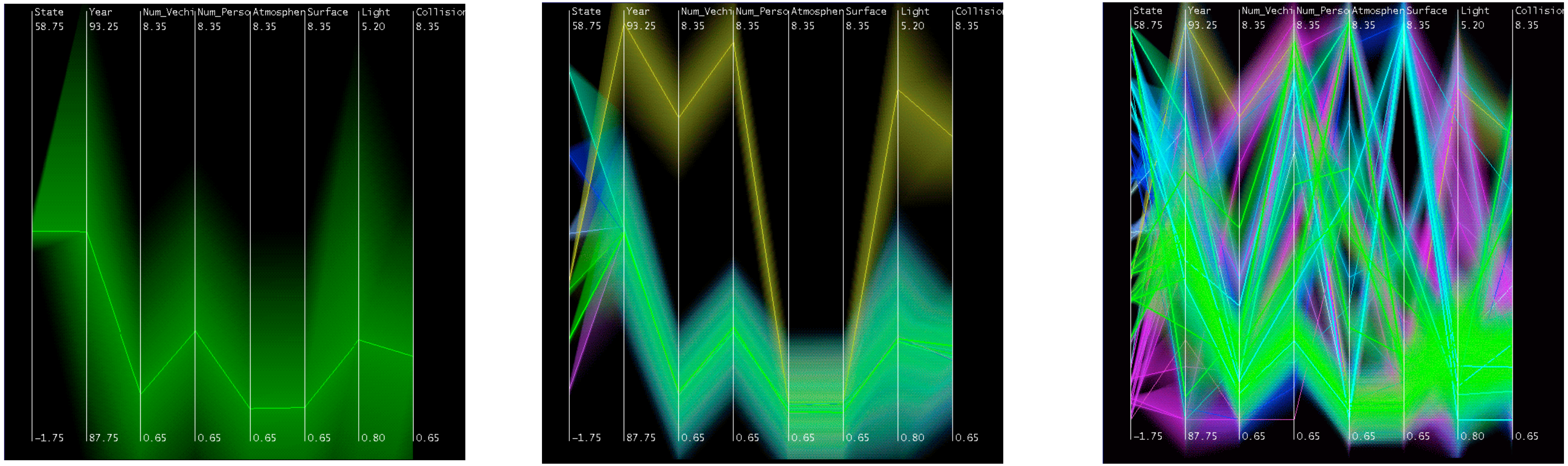
- *Warning*:
  - rather good for normal distribution
  - not so great for non-normal distribution
  - especially bad for bi-modal distribution





Idiom: **Hierarchical parallel coordinates**

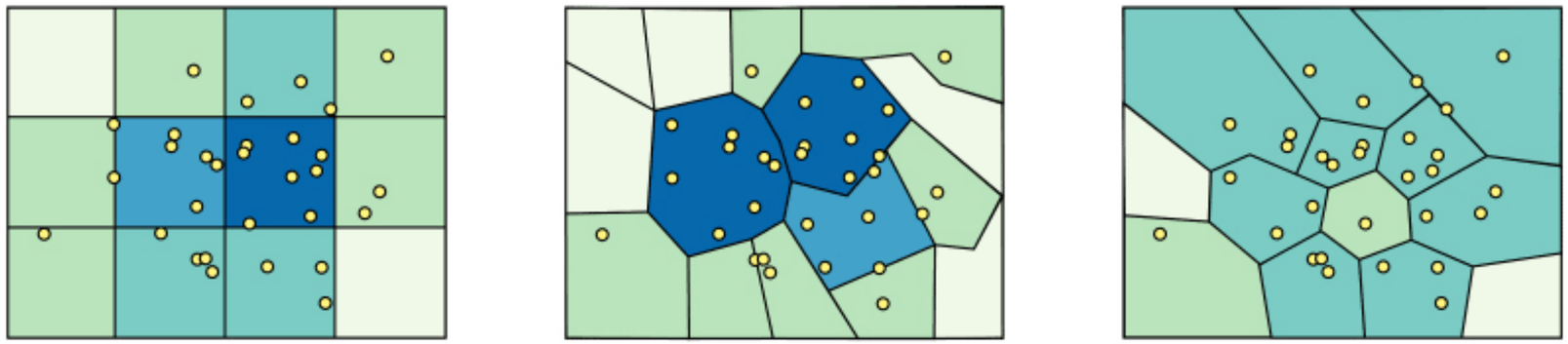
- dynamic item aggregation
- derived data: ***hierarchical clustering***
- encoding:
  - cluster band with variable transparency, line at mean, width by min/max values
  - color by proximity in hierarchy



[Hierarchical Parallel Coordinates for Exploration of Large Datasets. Fua, Ward, and Rundensteiner. Proc. IEEE Visualization Conference (Vis '99), pp. 43– 50, 1999.]

Spatial aggregation

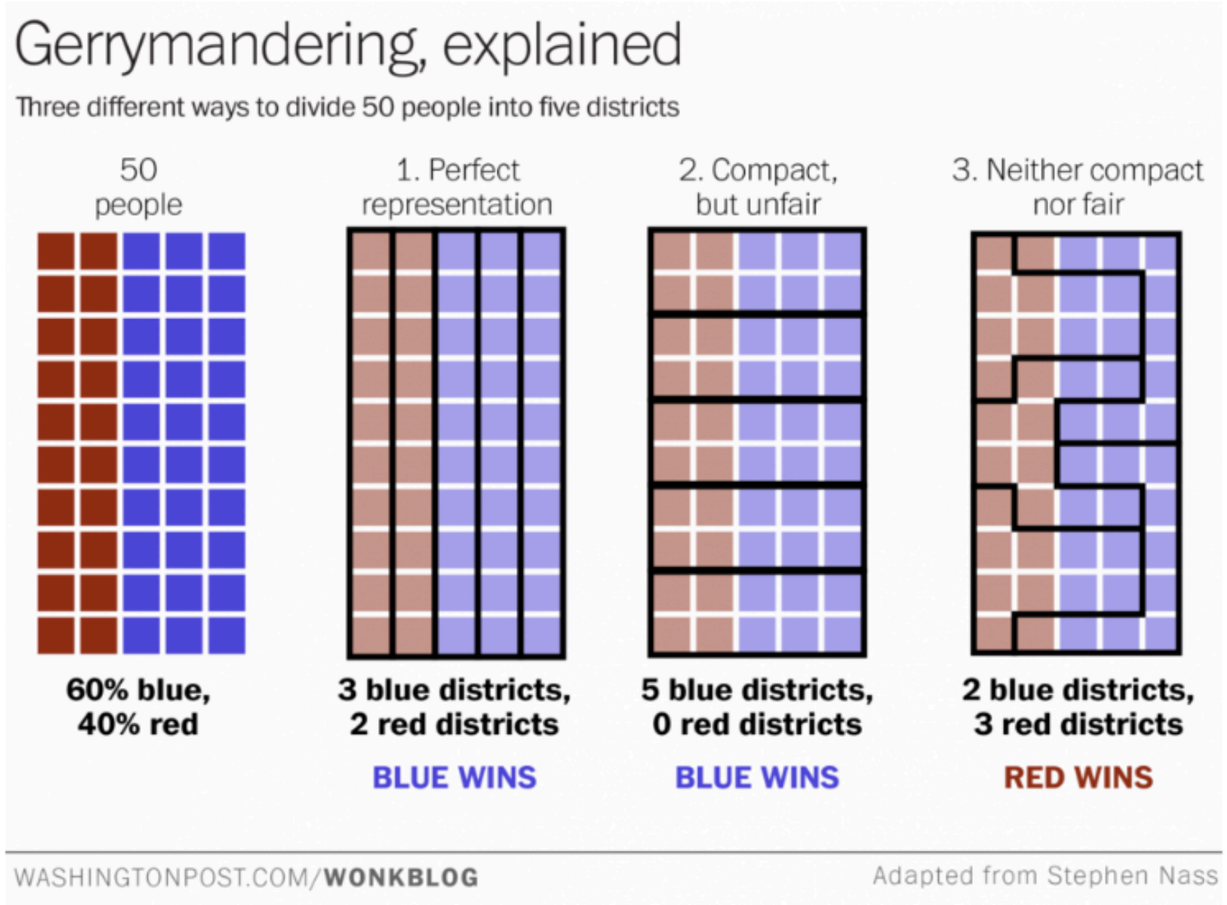
- MAUP: Modifiable Areal Unit Problem
  - imposition of artificial units of spatial reporting on continuous geographical phenomena
  - generation of artificial spatial patterns
  - any set of rules creates bias

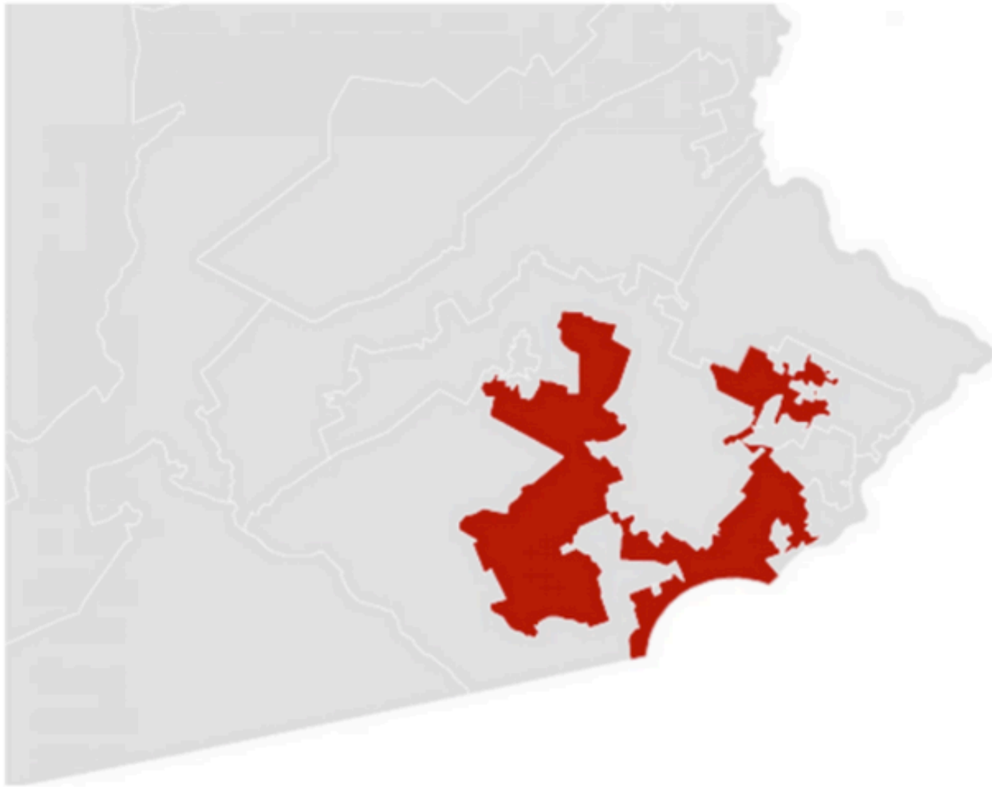


http://gispopsci.org/maup/

Spatial aggregation

- Gerrymandering: manipulation of electoral districts for political gain





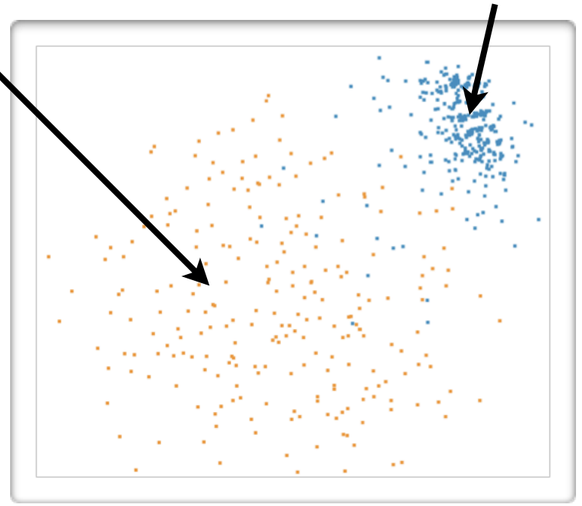
A real district in Pennsylvania
Democrats won 51% of the vote
but only 5 out of 18 house seats

Dimensionality reduction

- attribute aggregation
  - derive low-dimensional target space from high-dimensional measured space
  - use when you cannot directly measure what you care about
    - true dimensionality of dataset conjectured to be smaller than dimensionality of measurements
    - latent factors, hidden variables

Tumor
Measurement Data

→ DR →



data: 9D measured space

derived data: 2D target space

Dimensionality reduction for documents

Task 1

In
HD data

Out
2D data

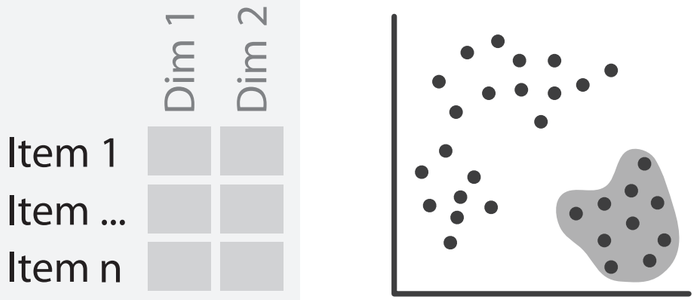
What?

- In High-dimensional data
- Out 2D data

Why?

- Produce
- Derive

Task 2



In
2D data

Out
Scatterplot
Clusters & points

What?

- In 2D data
- Out Scatterplot
- Out Clusters & points

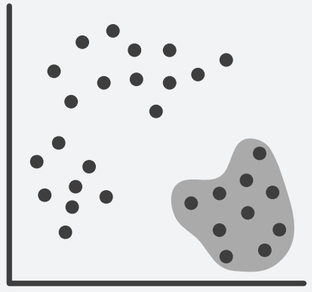
Why?

- Discover
- Explore
- Identify

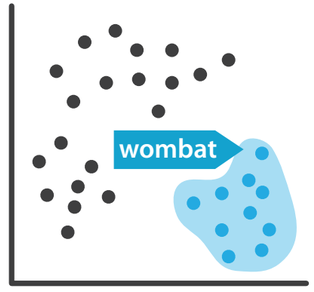
How?

- Encode
- Navigate
- Select

Task 3



In
Scatterplot
Clusters & points



Out
Labels for
clusters

What?

- In Scatterplot
- In Clusters & points
- Out Labels for clusters

Why?

- Produce
- Annotate

Dimensionality vs attribute reduction

- vocabulary use in field not consistent
  - dimension/attribute
- attribute reduction: *reduce set* with filtering
  - includes orthographic projection
- dimensionality reduction: *create smaller set* of new dims/attribs
  - typically implies dimensional aggregation, not just filtering
  - vocabulary: projection/mapping

Estimating true dimensionality

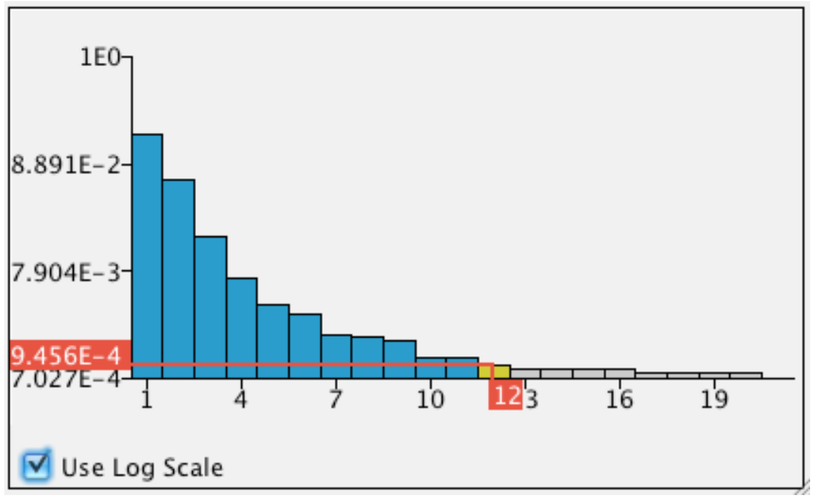
- how do you know when you would benefit from DR?
  - consider error for low-dim projection vs high-dim projection
- no single correct answer; many metrics proposed
  - cumulative variance that is not accounted for
- Multidimensional Scaling:
  - Map points to lower dimensional space by minimising strain/stress
  - strain: match variations in distance (vs actual distance values)
  - stress: difference between inter-point distances in high and low dims

$$stress(D, \Delta) = \sqrt{\frac{\sum_{ij}(d_{ij} - \delta_{ij})^2}{\sum_{ij} \delta_{ij}^2}}$$

- D: matrix of lowD distances
- Δ: matrix of hiD distances δ_{ij}

Estimating true dimensionality

- scree plots as simple way: error against # attribs

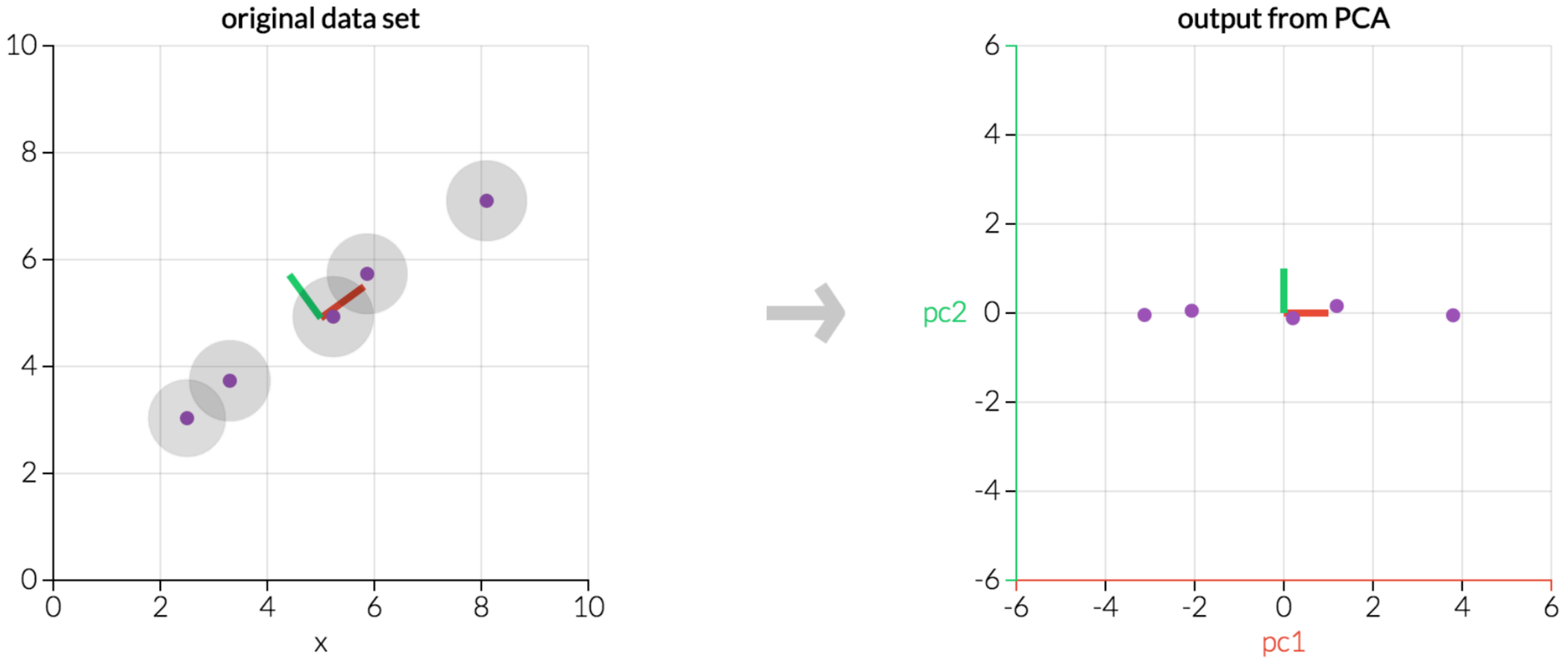


  - original dataset: 294 dims
  - estimate: almost all variance preserved with < 20 dims

[Fig 2. DimStiller: Workflows for dimensional analysis and reduction. Ingram et al. Proc. VAST 2010, p 3-10]

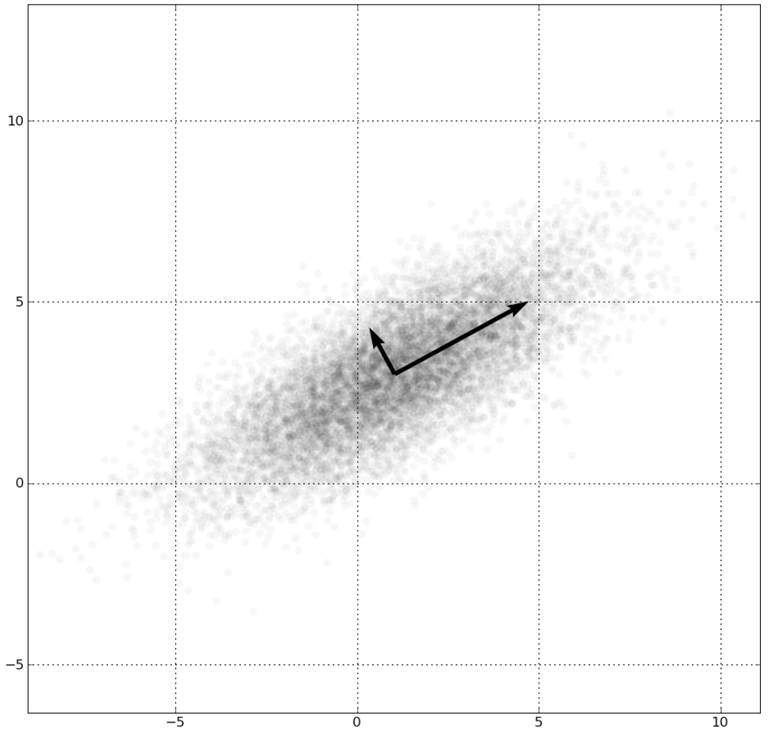
Linear dimensionality reduction

- principal components analysis (PCA)
  - describe location of each point as linear combination of weights for each axis
  - finding axes: first with most variance, second with next most, ...



PCA

- # Input dimensions *n* can be arbitrarily many
- # Output dimensions *d* decided by user
  - each of the *d* axes in output is a linear combination of the *n* axes in input
  - far more complex than selecting *d* out of *n* dimensions



Nonlinear dimensionality reduction

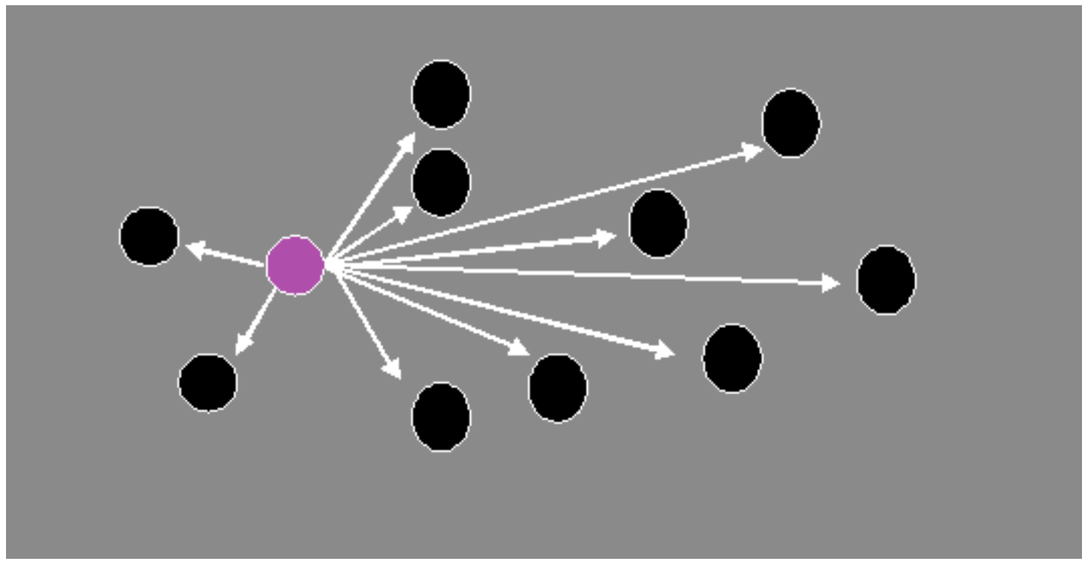
- many techniques proposed
  - MDS, charting, isomap, LLE, T-SNE
  - many literatures: visualization, machine learning, optimization, psychology, ...
- pro: can handle curved rather than linear structure
- cons: lose all ties to original dims/attribs
  - new dimensions cannot be easily related to originals

MDS: Multidimensional Scaling

- confusingly: entire family of methods, linear and nonlinear!
- classical scaling: minimize strain
  - early formulation equivalent to PCA (linear)
  - Nystrom/spectral methods approximate eigenvectors: O(N)
    - Landmark MDS [de Silva 2004], PivotMDS [Brandes & Pich 2006]
  - limitations: quality for very high dimensional sparse data
- distance scaling: minimize stress
  - nonlinear optimization: $O(N^2)$
    - SMACOF [de Leeuw 1977]
  - force-directed placement: $O(N^2)$
    - Stochastic Force [Chalmers 1996]
    - limitations: quality problems from local minima
  - Glimmer: O(N)

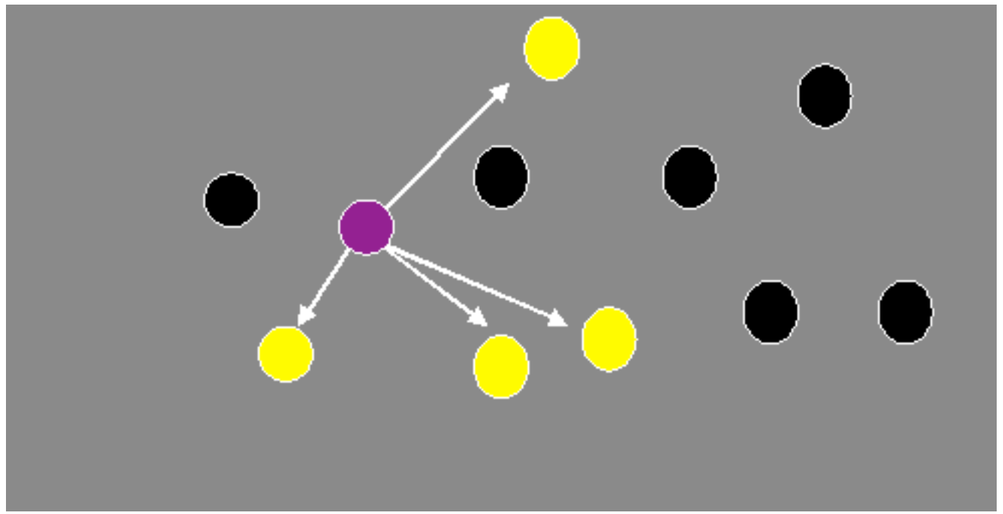
Spring-based MDS: naive

- repeat for all points
  - compute spring force to all other points
  - difference between high dim, low dim distance
  - move to better location using computed forces
- compute distances between all points
  - $O(N^2)$ iteration, $O(N^3)$ algorithm



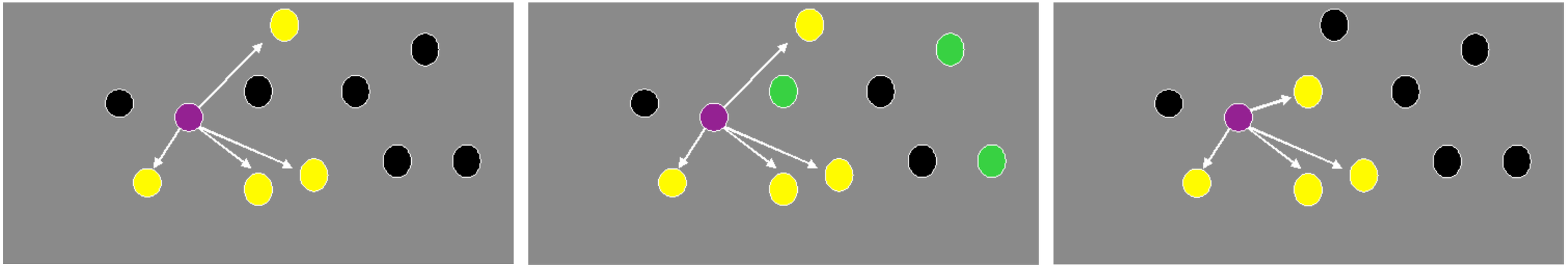
Faster spring model: Stochastic

- compare distances only with a few points
  - maintain small local neighborhood set



Faster spring model: Stochastic

- compare distances only with a few points
  - maintain small local neighborhood set
  - each time pick some randoms, swap in if closer
- small constant: 6 locals, 3 randoms (typically)
  - O(N) iteration, $O(N^2)$ algorithm



Embed: Focus+Context

- General idea:
  - *Focus*: selected set of interesting elements (items / attributes)
  - *Context*: all the remaining elements
  - Embed detailed information about the *focus* as well as overview information about the *context* within the *same view*

- Sophisticated combination of filtering and aggregation
- Many possible idioms
- Embedded details may be treated as separate views or glyphs

- Focus usually user selected:
  - fundamental synthesis between encoding and interaction

Embed: Focus+Context

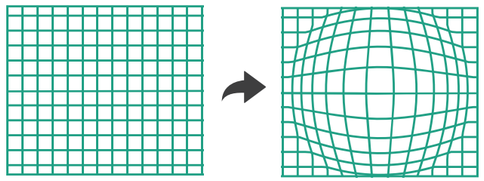
- combine information within single view
- elide
  - selectively filter and aggregate
- superimpose layer
  - local lens
- distortion design choices
  - region shape: radial, rectilinear, complex
  - how many regions: one, many
  - region extent: local, global
  - interaction metaphor

Embed

Elide Data

Superimpose Layer

Distort Geometry



Degree of interest (DOI)

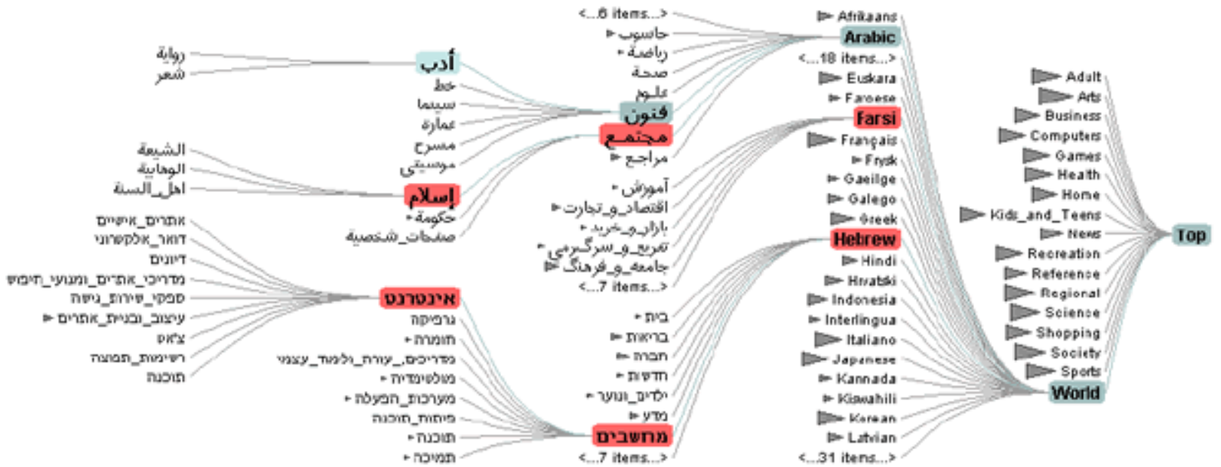
- based on observation that humans often represent their own neighborhood in detail, yet only major landmarks far away
- goal is balance between local detail and global context

$$DOI(x) = API(x) - D(x,y)$$

  - API: a priori interest
  - D: a distance function to the current focus y
  - can have multiple foci

Idiom: **DOI Tree**

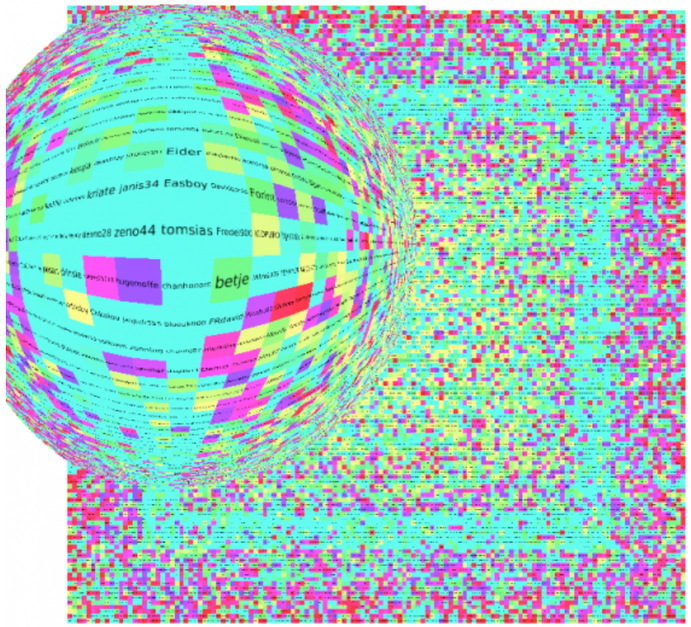
- Interactive tree with animated transitions that fit within bounded region
- Layout depends on the user's estimated DOI
- Use:
  - logical filtering based on DOI
  - geometric distortion of node size
  - semantic zooming
- elide
  - some items dynamically filtered out
  - some items dynamically aggregated together
  - some items shown in detail

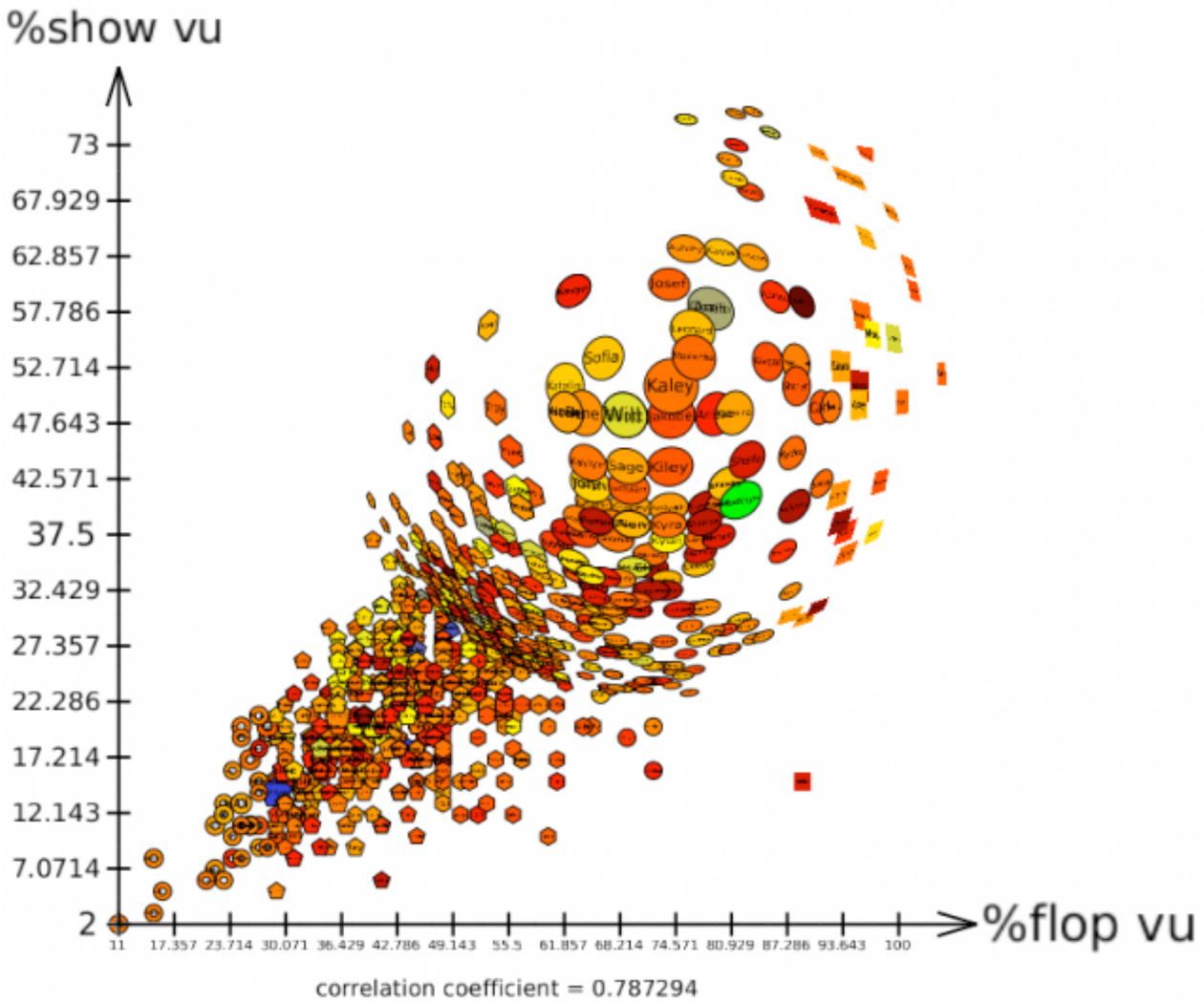


[DOITrees Revisited: Scalable, Space-Constrained Visualization of Hierarchical Data. Heer and Card. Proc. Advanced Visual Interfaces (AVI), pp. 421–424, 2004.]

Idiom: **Fisheye Lens**

- distort geometry
  - shape: radial
  - focus: single extent
  - extent: local
  - metaphor: draggable lens





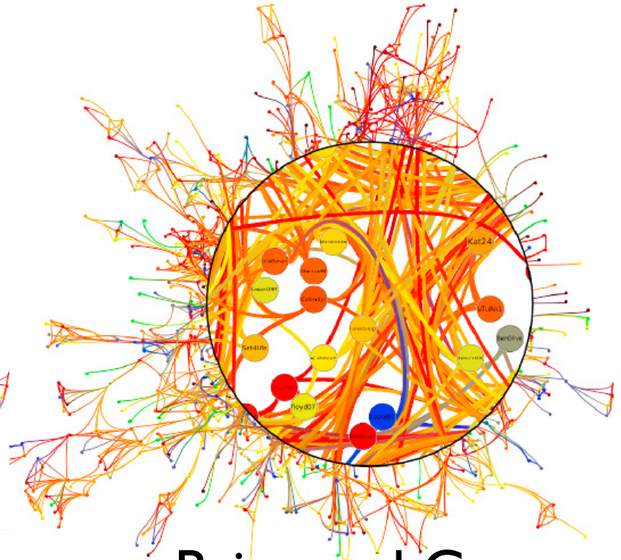
http://tulip.labri.fr/TulipDrupal/?q=node/351
http://tulip.labri.fr/TulipDrupal/?q=node/371

Distortion costs and benefits

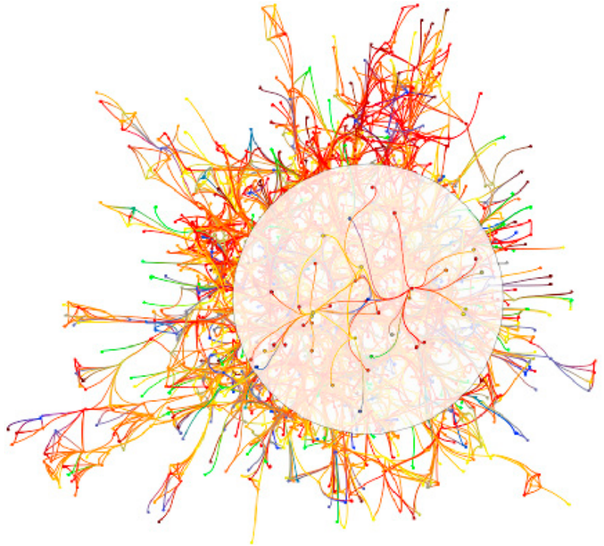
- benefits
  - combine focus and context information in single view
- costs
  - length comparisons impaired
    - network/tree topology comparisons unaffected: connection, containment
  - effects of distortion unclear if original structure unfamiliar
  - object constancy/tracking maybe impaired

fisheye lens

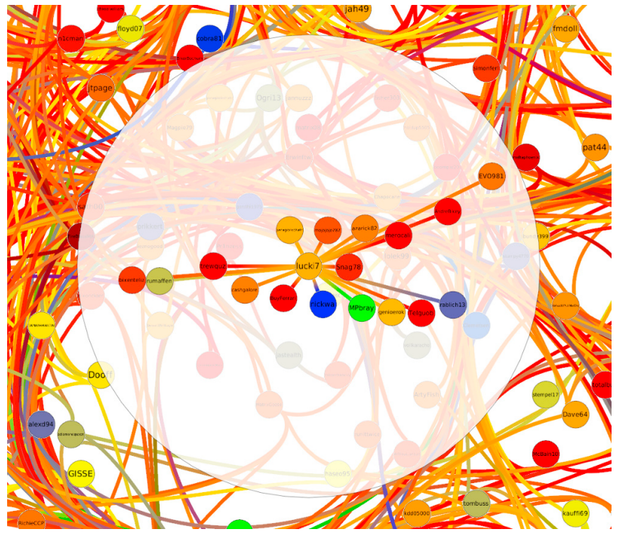
magnifying lens



neighborhood layering



Bring and Go



[Living Flows: Enhanced Exploration of Edge-Bundled Graphs Based on GPU-Intensive Edge Rendering. Lambert, Auber, and Melançon. Proc. Intl. Conf. Information Visualisation (IV), pp. 523–530, 2010.]