

Information Visualization

Reduce: Aggregation & Filtering

Project Peer Reviews

Tamara Munzner

Department of Computer Science
University of British Columbia

Week 12, 16 Nov 2022

<https://www.cs.ubc.ca/~tmm/courses/547-22>

Today

- first: project peer reviews
 - join your matched teams
 - you've already read other team's written update
 - let me know by private Piazza post if your counterpart(s) weren't prepared
 - record discussion/thoughts in gdoc (freeform)
 - first A critiques B; then B critiques A
- break
- Q&A / mini-lecture this time
 - Ch 13, Reduce

Peer reviews

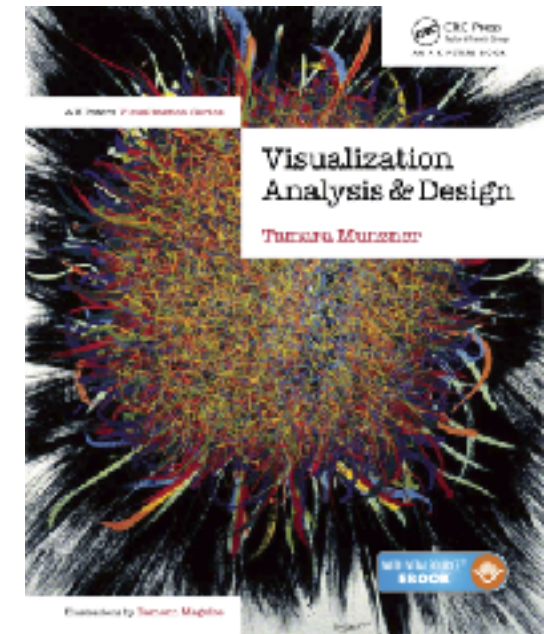
- rough structure (adapt as you like, aim for ~45-60 min)
 - talk through initial thoughts when read updates
 - ask clarifying questions
 - get demo to see look/feel & any interaction
 - discuss tradeoffs, design choices, suggestions
 - when conversation winds down, critiquers record braindump (if not done as you go)
 - write DONE at top of your gdoc section & switch!
- tips on giving feedback
 - state what you think is good about the work, and why you think so
 - state what you think needs improvement, including why/rationale
 - offer specific suggestions on how to improve it, as followup
 - keep your feedback focused on the work, not the person who did it

Upcoming

- next week (W12)
 - async: last week of readings / discussion (light, 2 readings)
 - Ch 14: Embed - Focus+Context
 - paper: Visualizing Dataflow Graphs of Deep Learning Models in TensorFlow.
Kanit Wongsuphasawat, Daniel Smilkov, James Wexler, Jimbo Wilson, Dandelion Mané, Doug Fritz, Dilip Krishnan, Fernanda B. Viégas, and Martin Wattenberg.
IEEE TVCG (Proc. VAST 2017) 24(1):1-12, 2018.
[type: design study]
 - in class: post-update meetings with Tamara
 - oral feedback on project progress, after I've read them

Q&A / Backup Slides

Visualization Analysis & Design



Reduce: Aggregation & Filtering (Ch 13)

Tamara Munzner

Department of Computer Science
University of British Columbia

[@tamaramunzner](#)

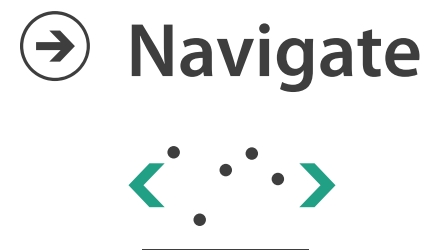
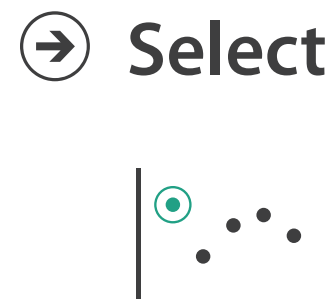
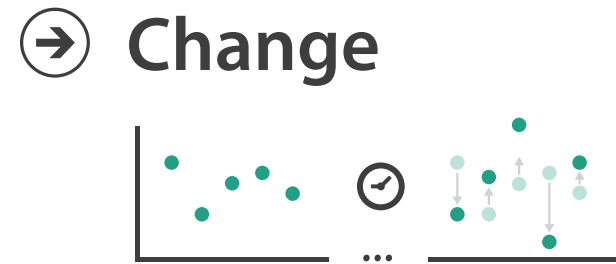
How to handle complexity: 3 previous strategies

→ *Derive*

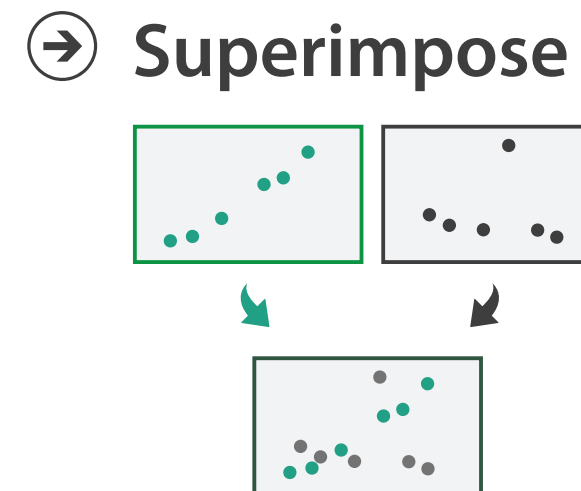
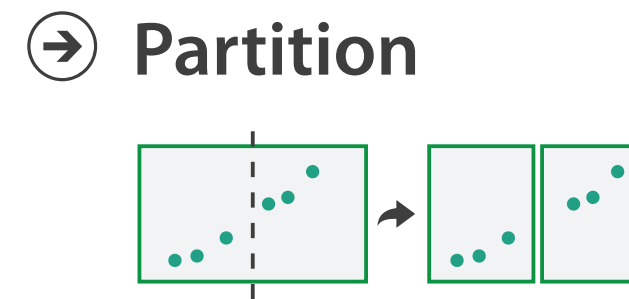
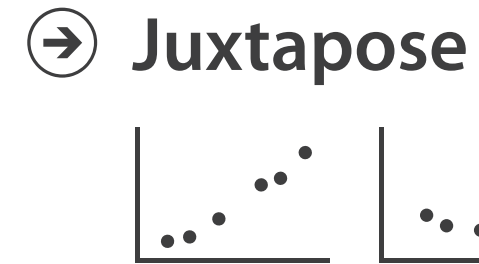


- derive new data to show within view
- change view over time
- facet across multiple views

Manipulate



Facet



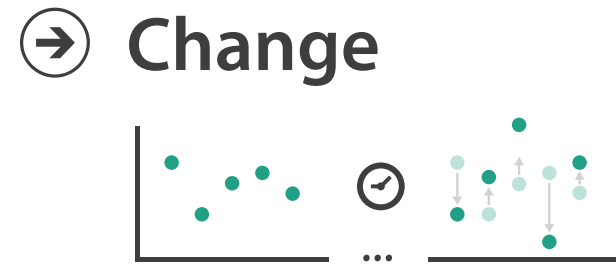
How to handle complexity: 3 previous strategies + 1 more

→ *Derive*

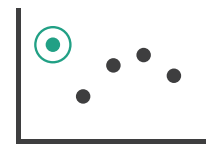


- derive new data to show within view
- change view over time
- facet across multiple views
- reduce items/attributes within single view

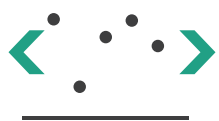
Manipulate



→ Select

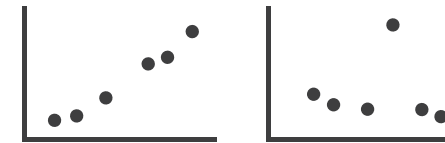


→ Navigate

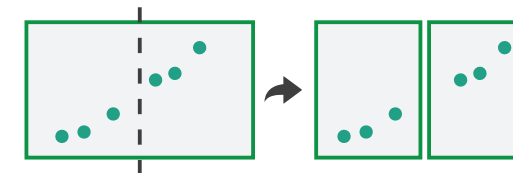


Facet

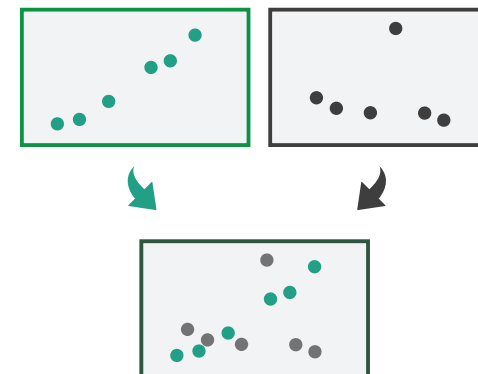
→ Juxtapose



→ Partition



→ Superimpose

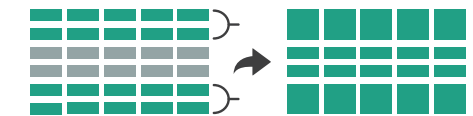


Reduce

→ Filter



→ Aggregate



→ Embed



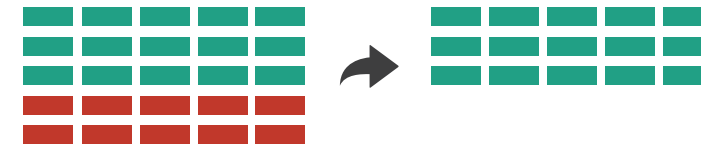
Reduce items and attributes

- reduce/increase: inverses
- filter
 - pro: straightforward and intuitive
 - to understand and compute
 - con: out of sight, out of mind

Reducing Items and Attributes

➔ Filter

➔ Items



➔ Attributes



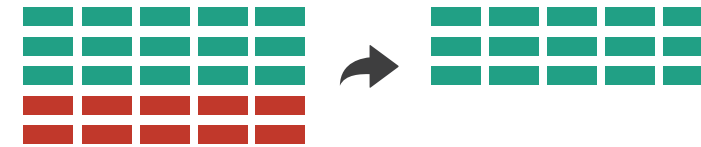
Reduce items and attributes

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

Reducing Items and Attributes

➔ Filter

➔ Items



➔ Attributes

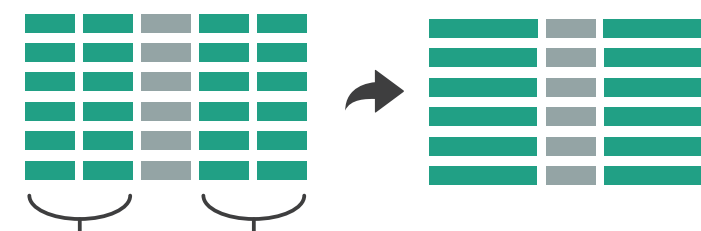


➔ Aggregate

➔ Items



➔ Attributes



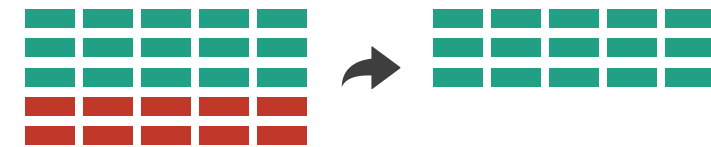
Filter

- eliminate some elements
 - either items or attributes
- according to what?
 - any possible function that partitions dataset into two sets
 - attribute values bigger/smaller than x
 - noise/signal
- filters vs queries
 - query: start with nothing, add in elements
 - filters: start with everything, remove elements
 - best approach depends on dataset size

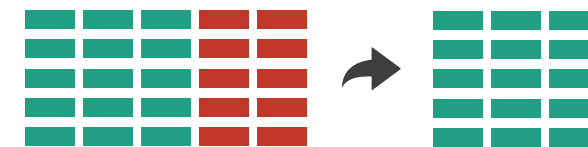
Reducing Items and Attributes

➔ Filter

→ Items

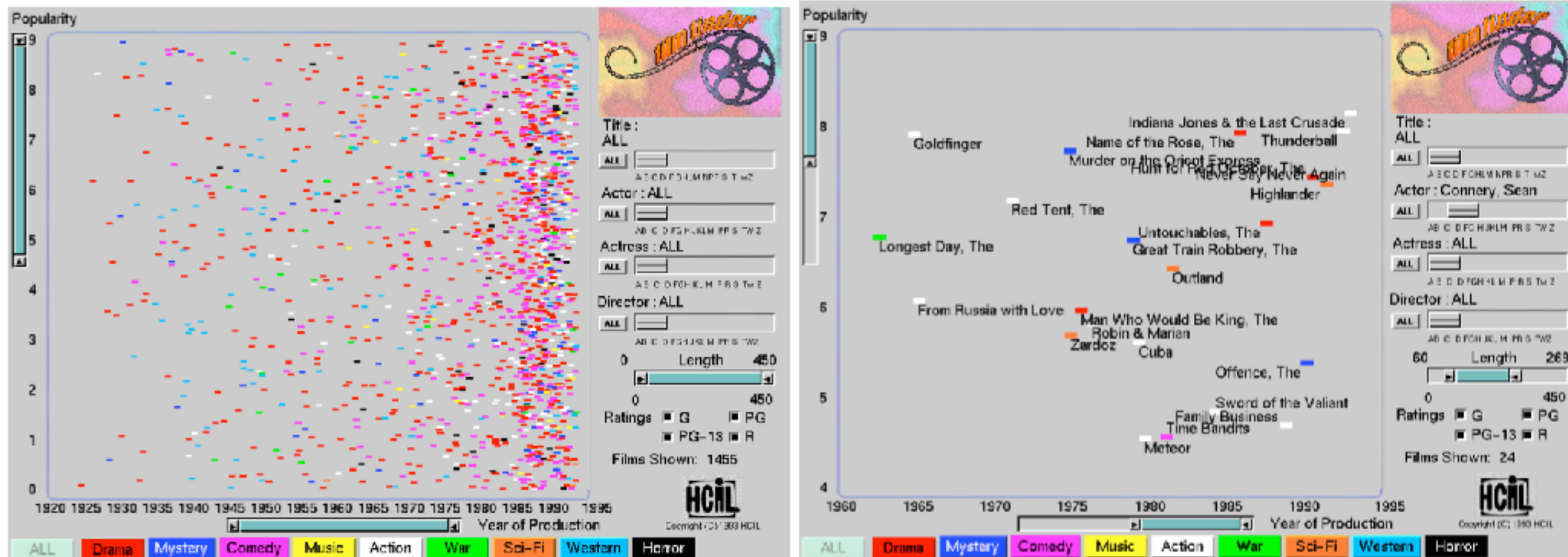


→ Attributes



Idiom: **FilmFinder**

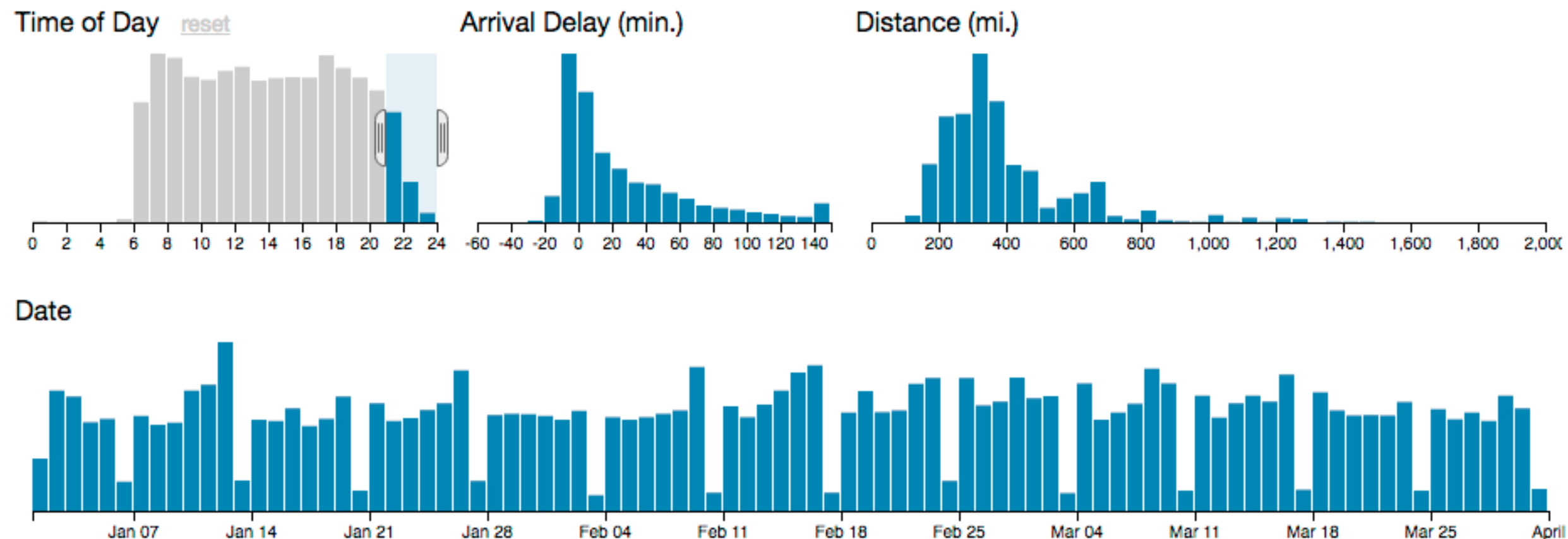
- dynamic queries/filters for items
 - tightly coupled interaction and visual encoding idioms, so user can immediately see results of action



Idiom: **cross filtering**

System: **Crossfilter**

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



<http://square.github.io/crossfilter/>

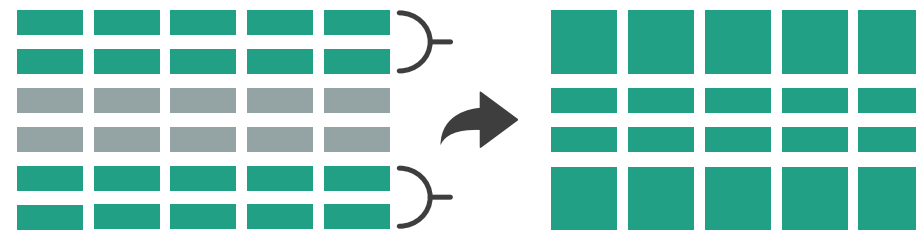
<https://observablehq.com/@uwdata/interaction>

Aggregate

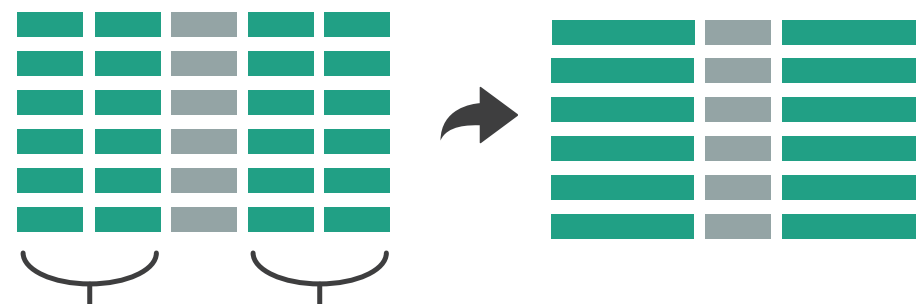
- a group of elements is represented by a smaller number of derived elements

➔ Aggregate

➔ Items

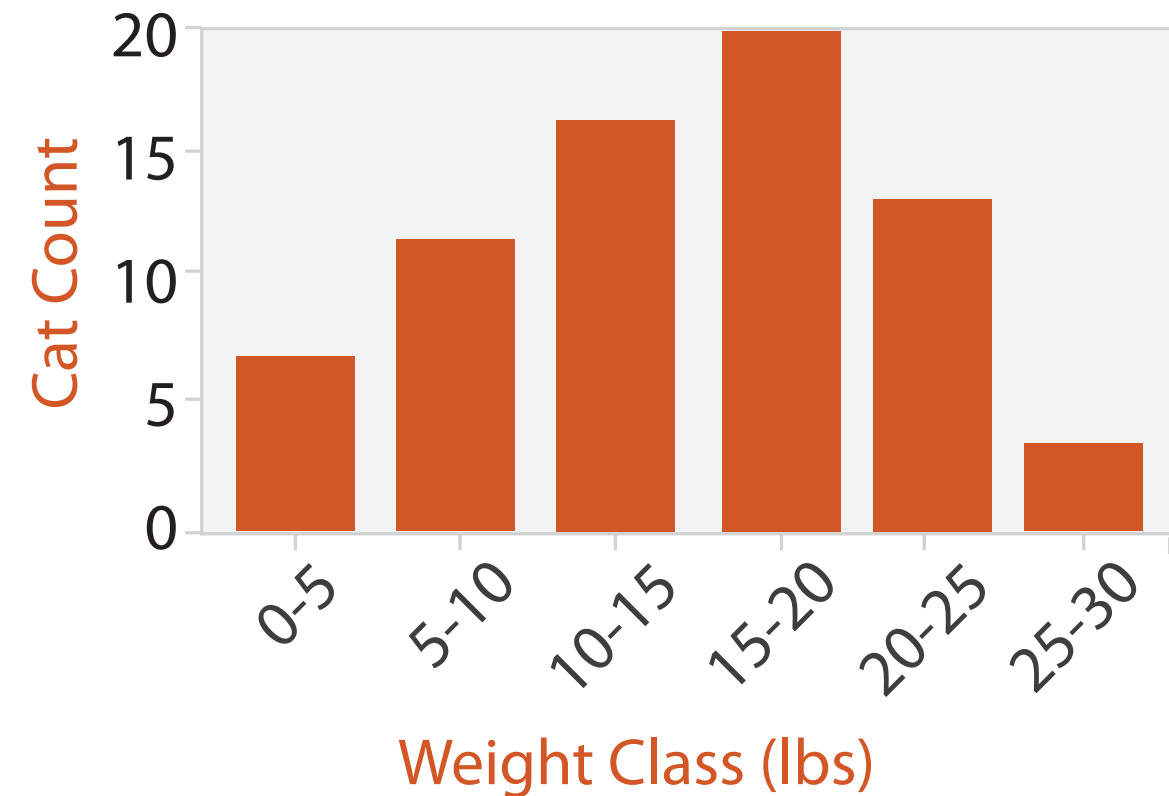


➔ Attributes



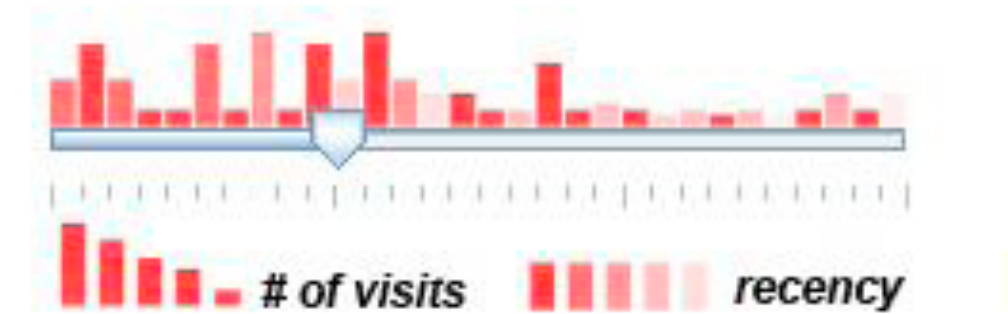
Idiom: **histogram**

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



Idiom: **scented widgets**

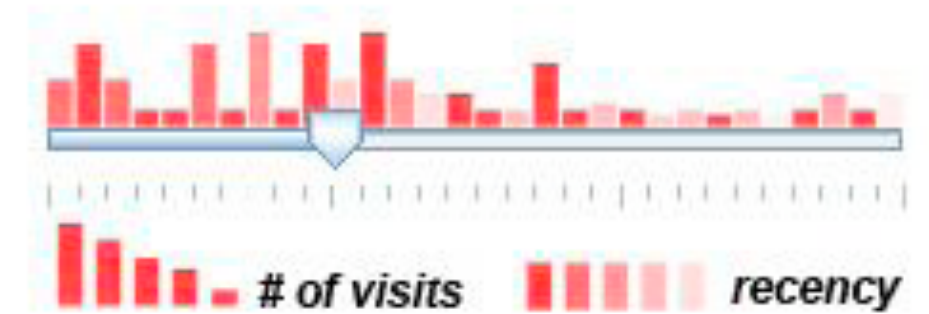
- augmented widgets show *information scent*
 - better cues for *information foraging*: show whether value in drilling down further vs looking elsewhere
- concise use of space: histogram on slider



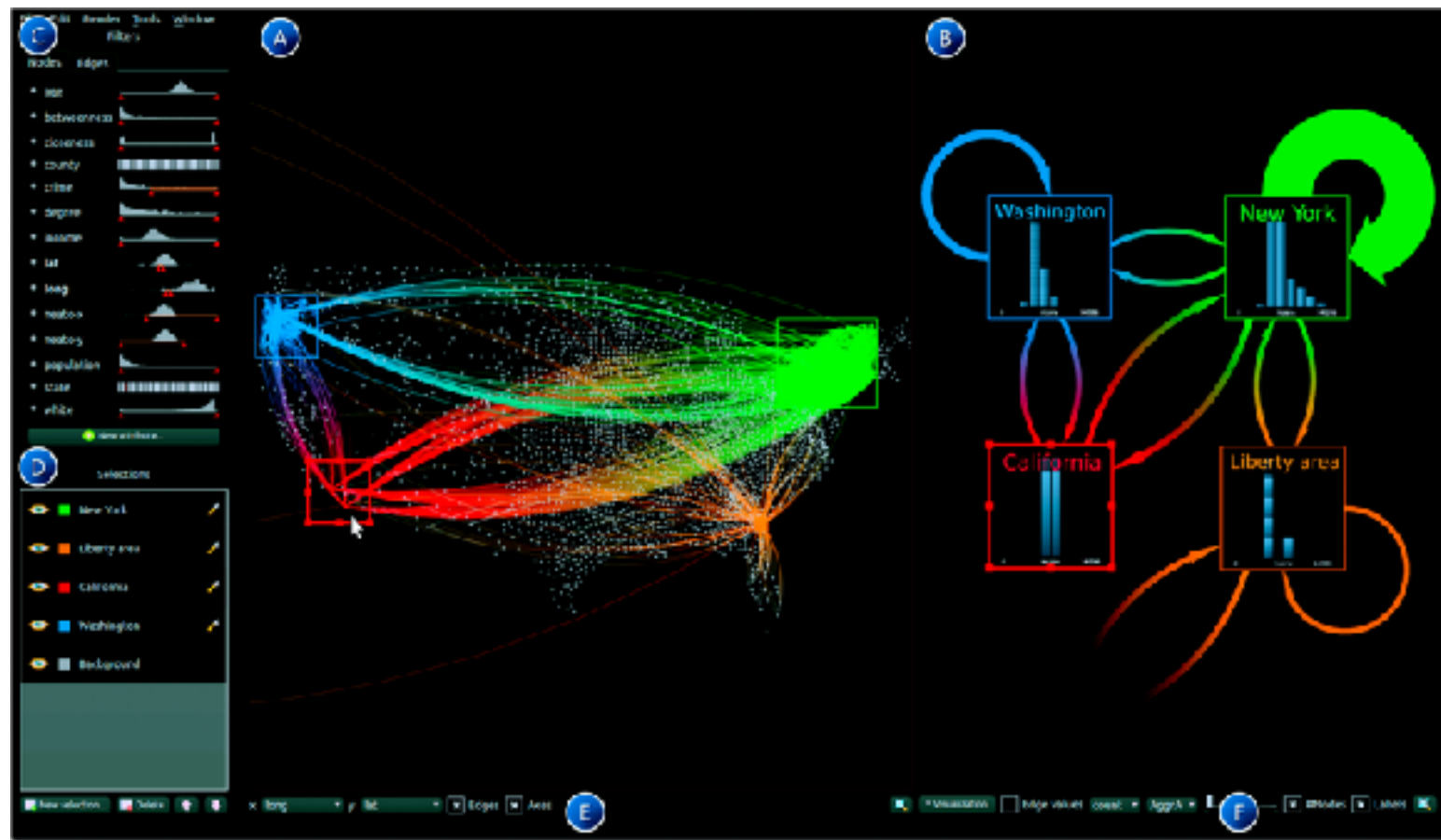
[Scented Widgets: Improving Navigation Cues with Embedded Visualizations. Willett, Heer, and Agrawala. IEEE TVCG (Proc. InfoVis 2007) 13:6 (2007), 1129–1136.]

Idiom: **scented widgets**

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



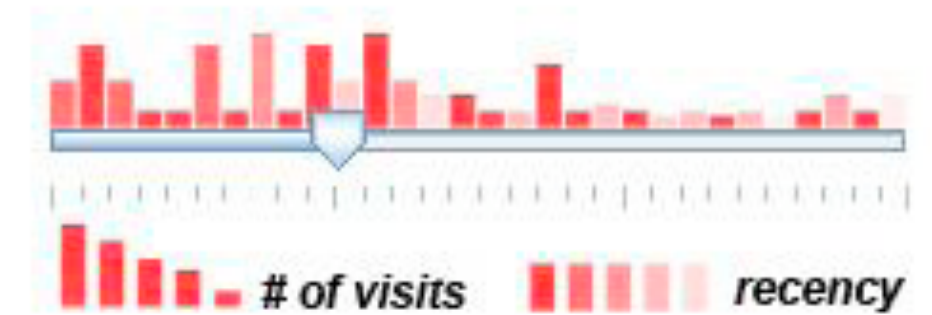
[Scented Widgets: Improving Navigation Cues with Embedded Visualizations. Willett, Heer, and Agrawala. IEEE TVCG (Proc. InfoVis 2007) 13:6 (2007), 1129–1136.]



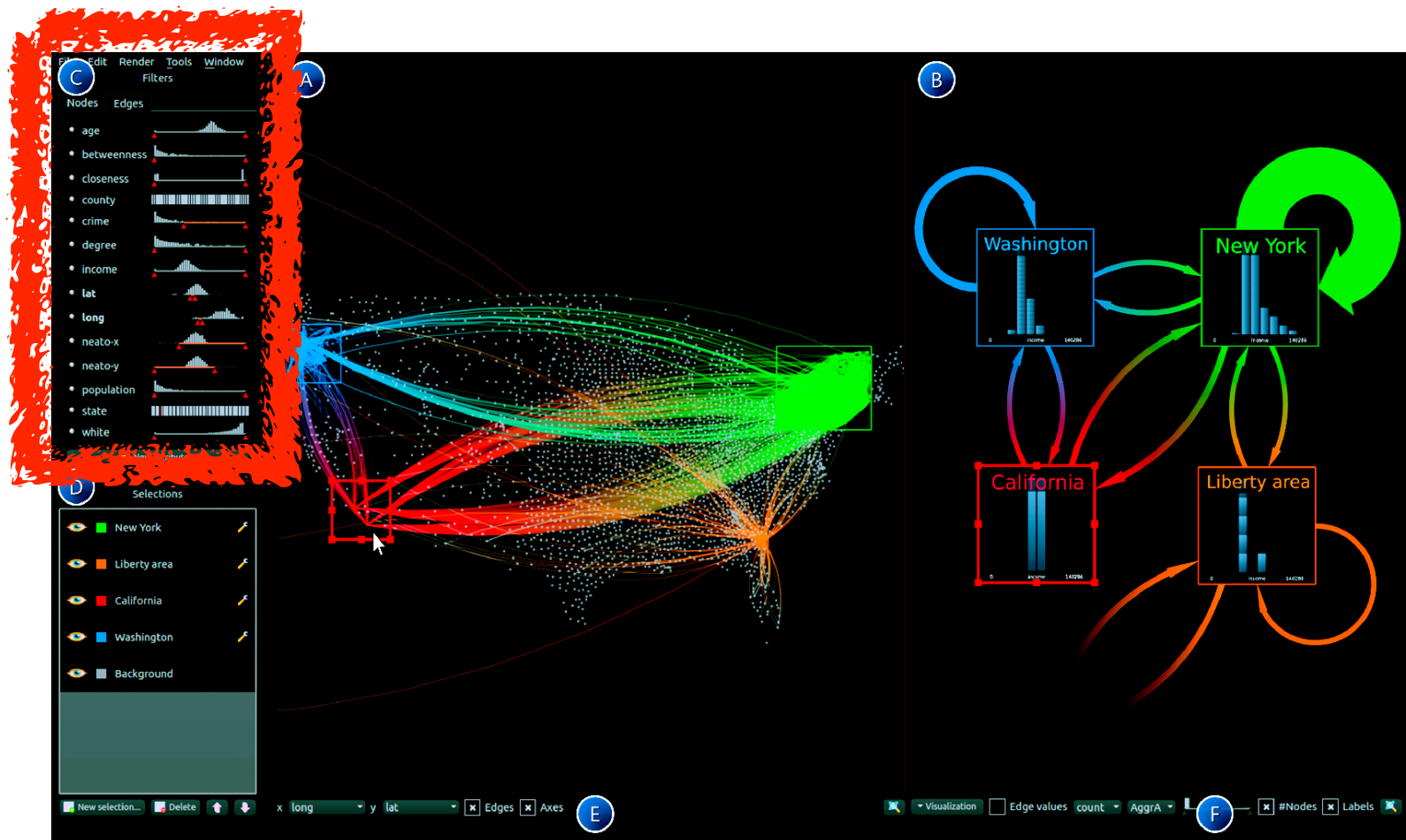
[Multivariate Network Exploration and Presentation: From Detail to Overview via Selections and Aggregations. van den Elzen, van Wijk, IEEE TVCG 20(12): 2014 (Proc. InfoVis 2014).]

Idiom: scented widgets

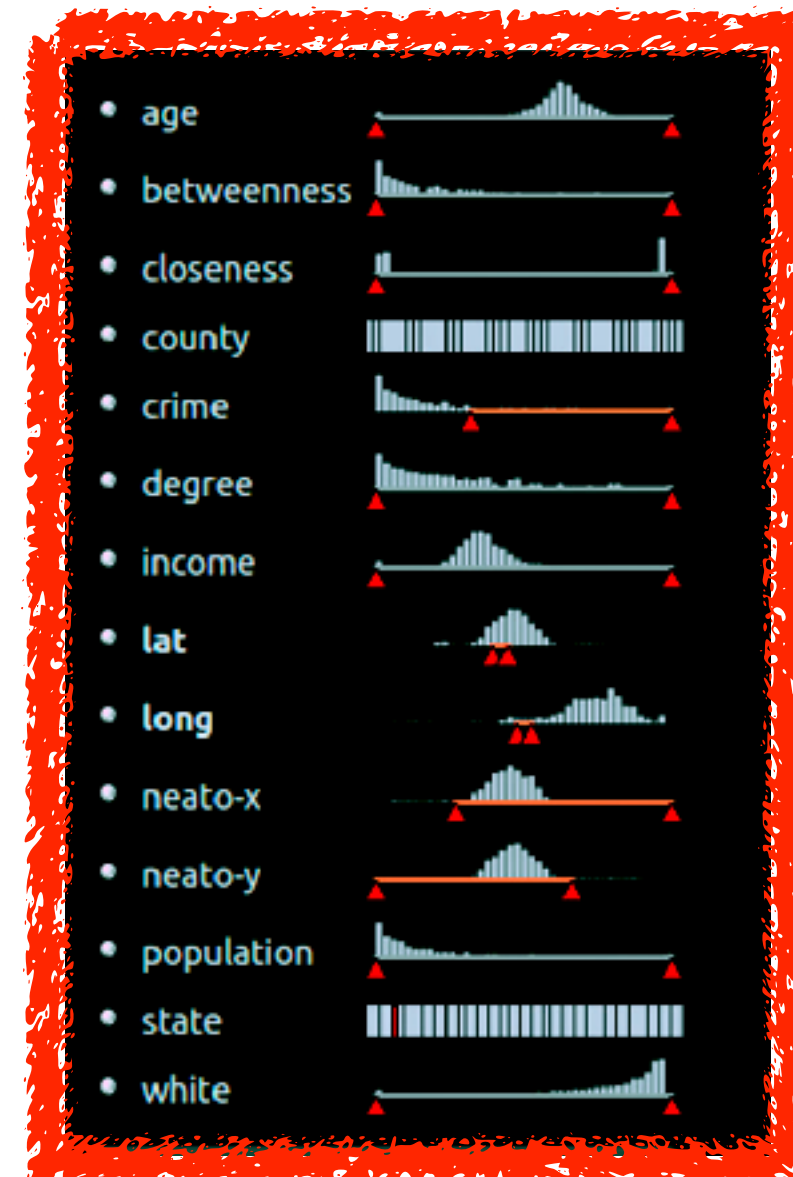
- augmented widgets show *information scent*
 - better cues for *information foraging*: show whether value in drilling down further vs looking elsewhere
- concise use of space: histogram on slider



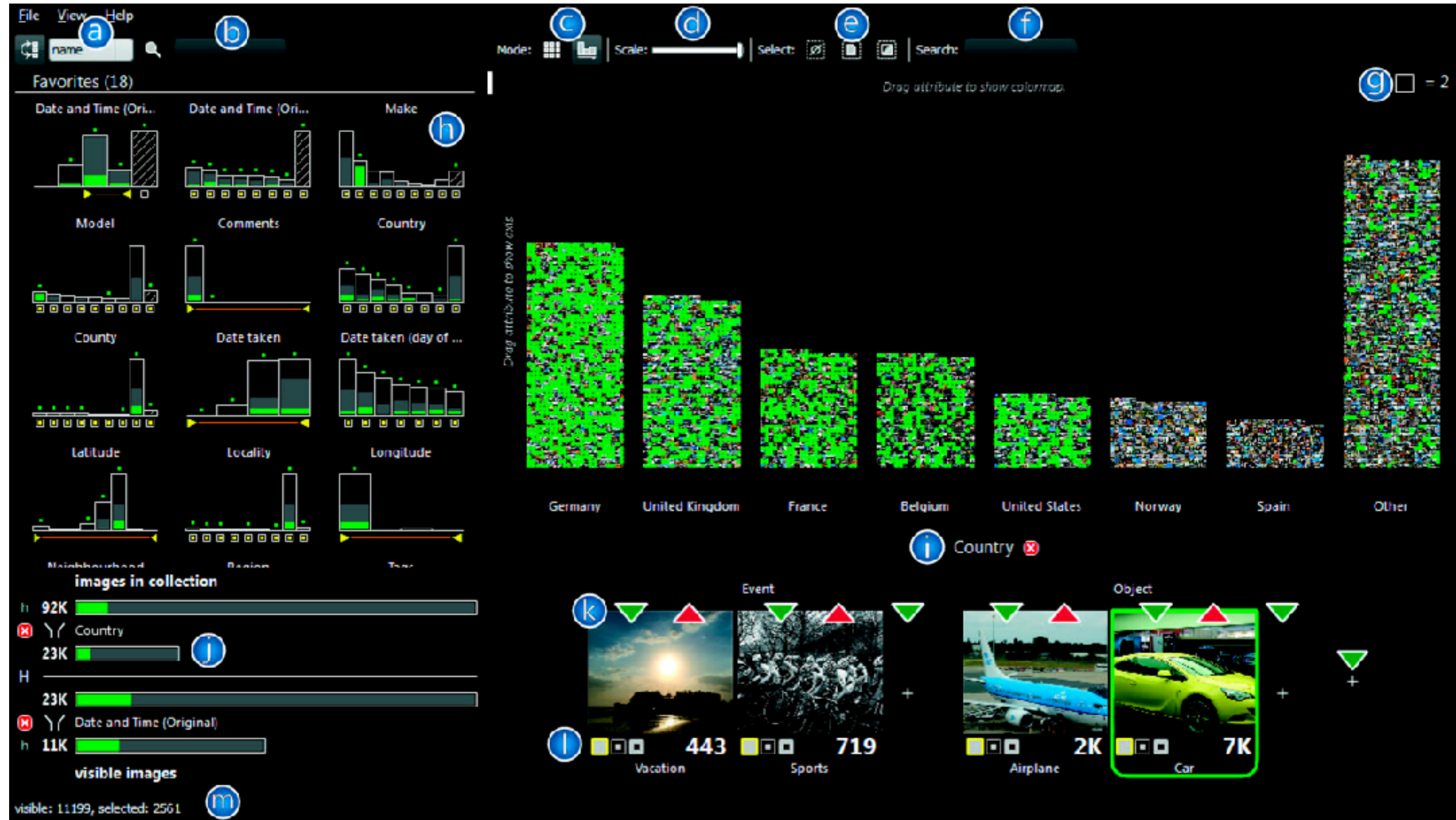
[Scented Widgets: Improving Navigation Cues with Embedded Visualizations. Willett, Heer, and Agrawala. IEEE TVCG (Proc. InfoVis 2007) 13:6 (2007), 1129–1136.]



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



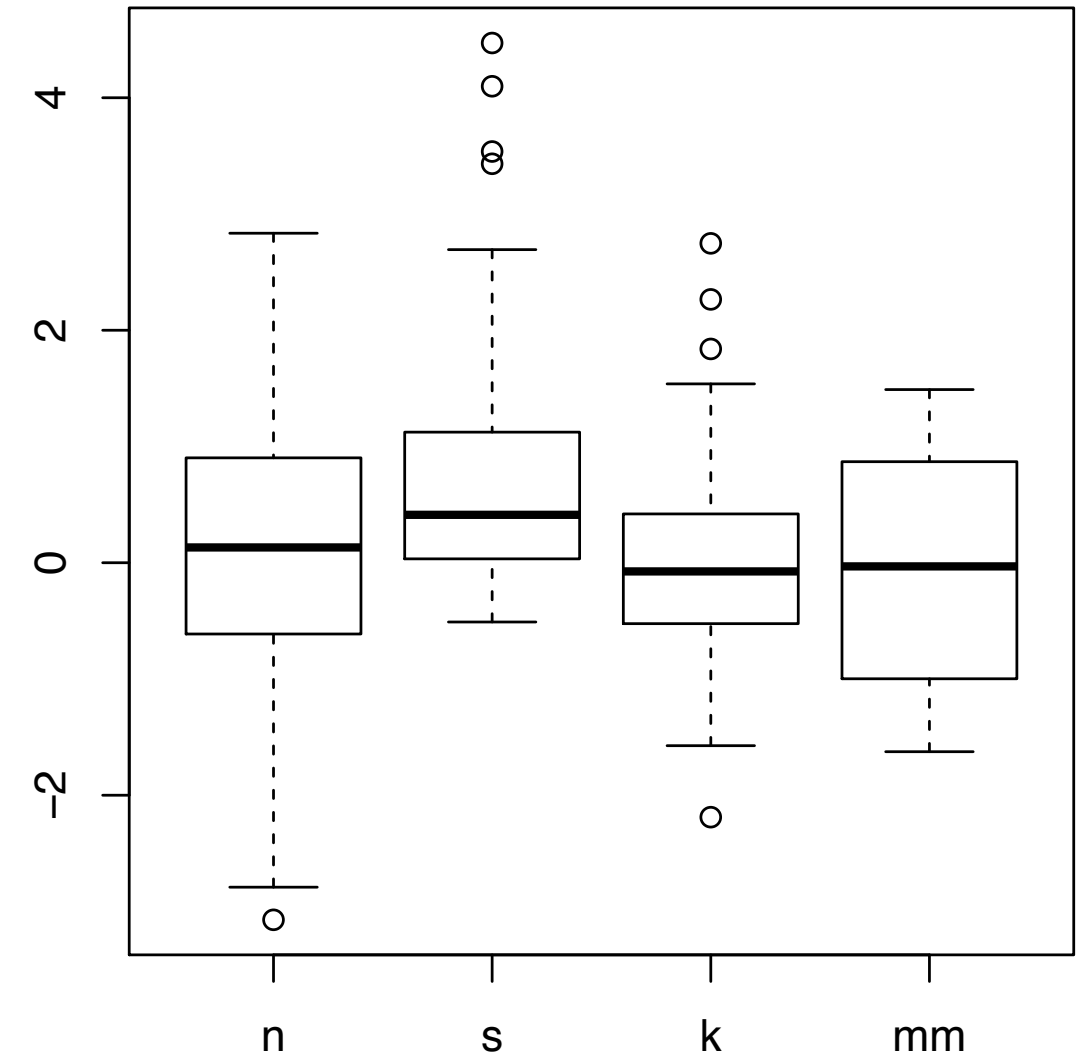
Scented histogram bisliders: detailed



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

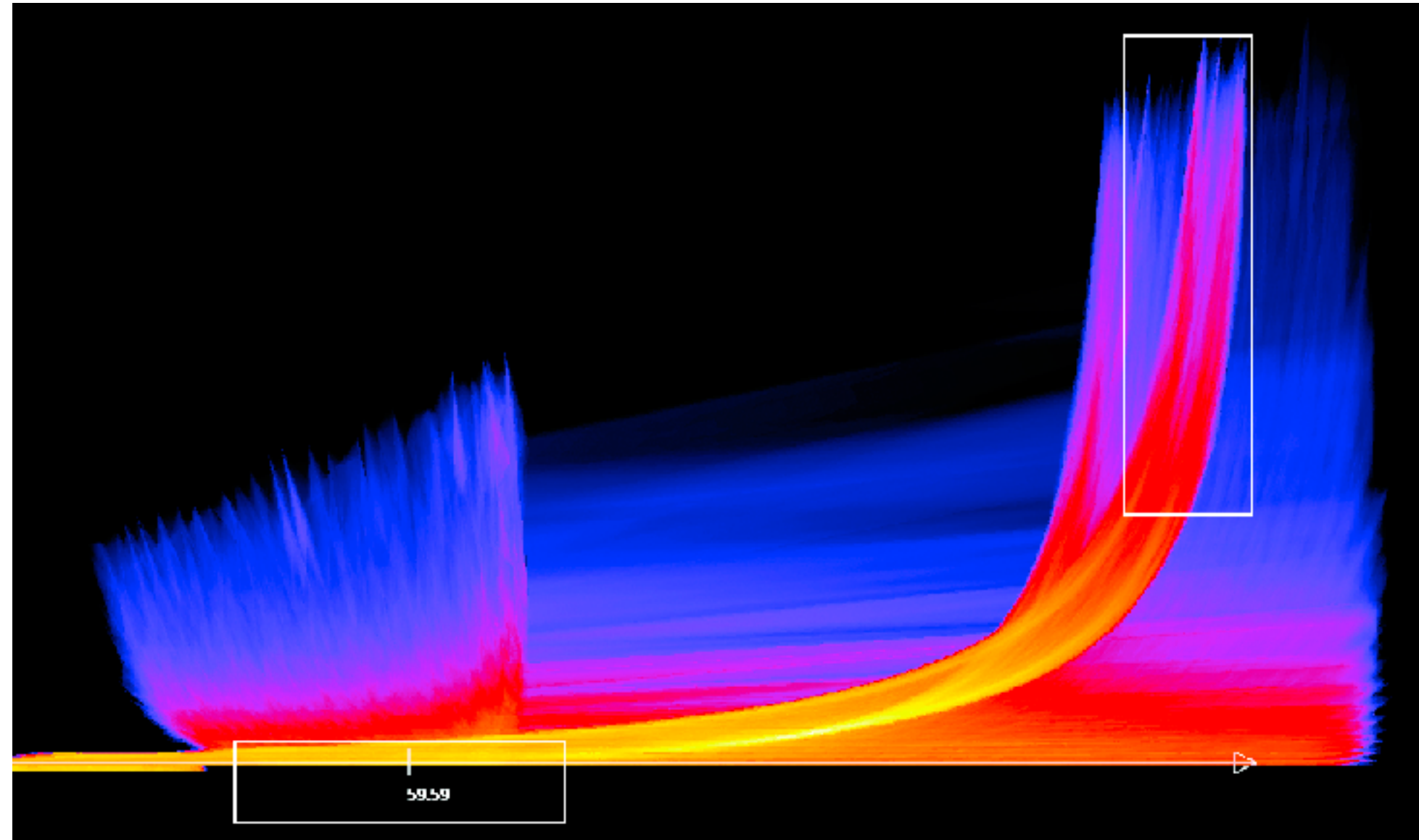
Idiom: **boxplot**

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



Idiom: 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
- scalability
 - no limits on overplotting:
millions of items



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

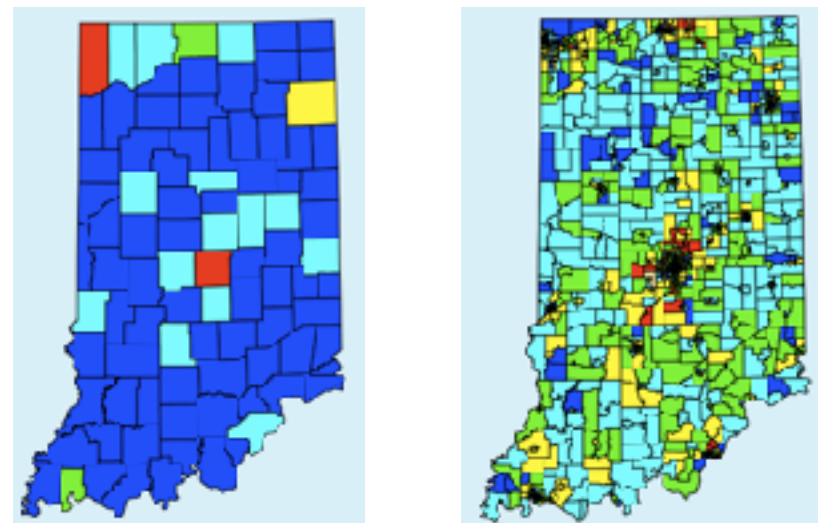
Spatial aggregation

- MAUP: Modifiable Areal Unit Problem
 - changing boundaries of cartographic regions can yield dramatically different results
 - zone effects



[http://www.e-education.psu.edu/geog486/l4_p7.html, Fig 4.cg.6]

- scale effects

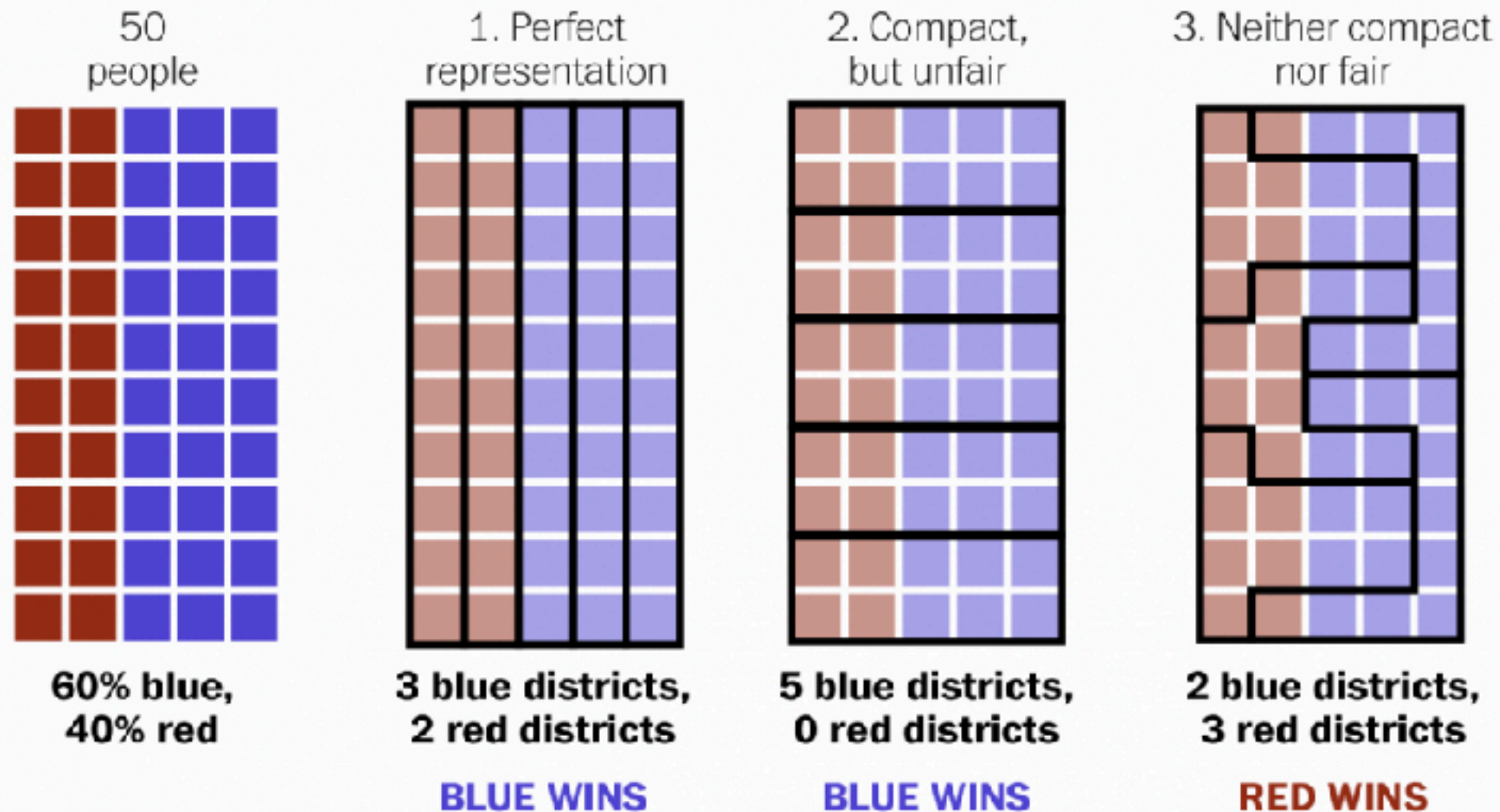


<https://blog.cartographica.com/blog/2011/5/19/the-modifiable-areal-unit-problem-in-gis.html>

Gerrymandering: MAUP for political gain

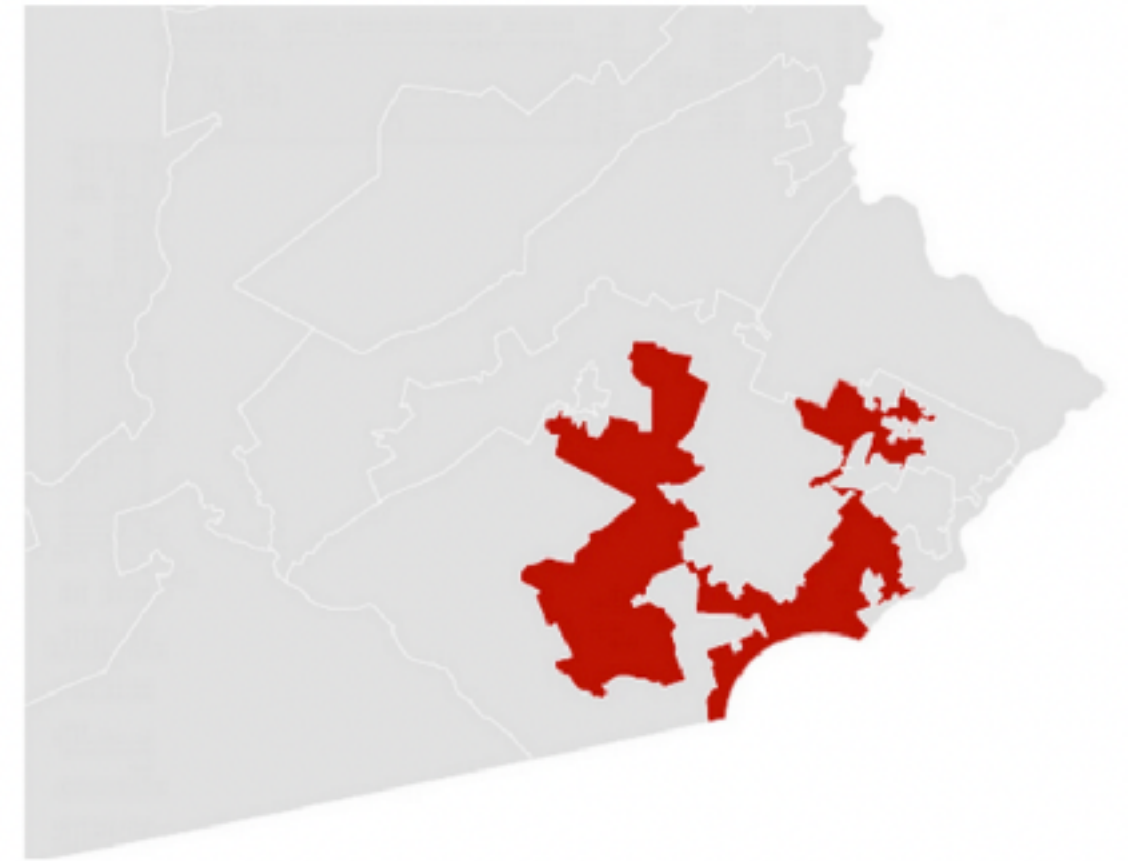
Gerrymandering, explained

Three different ways to divide 50 people into five districts



WASHINGTONPOST.COM/WONKBLOG

Adapted from Stephen Nass



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

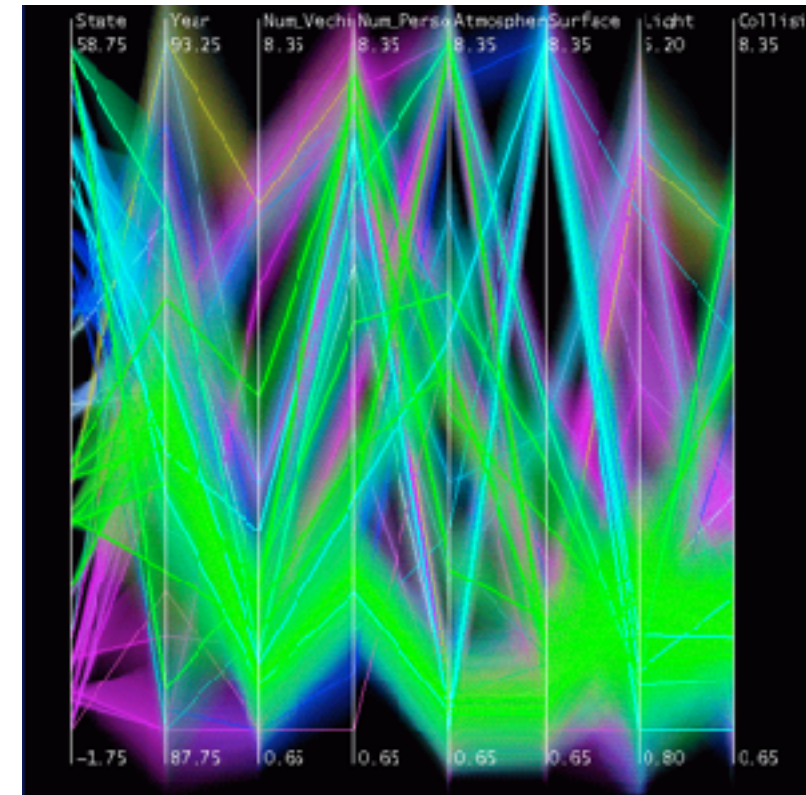
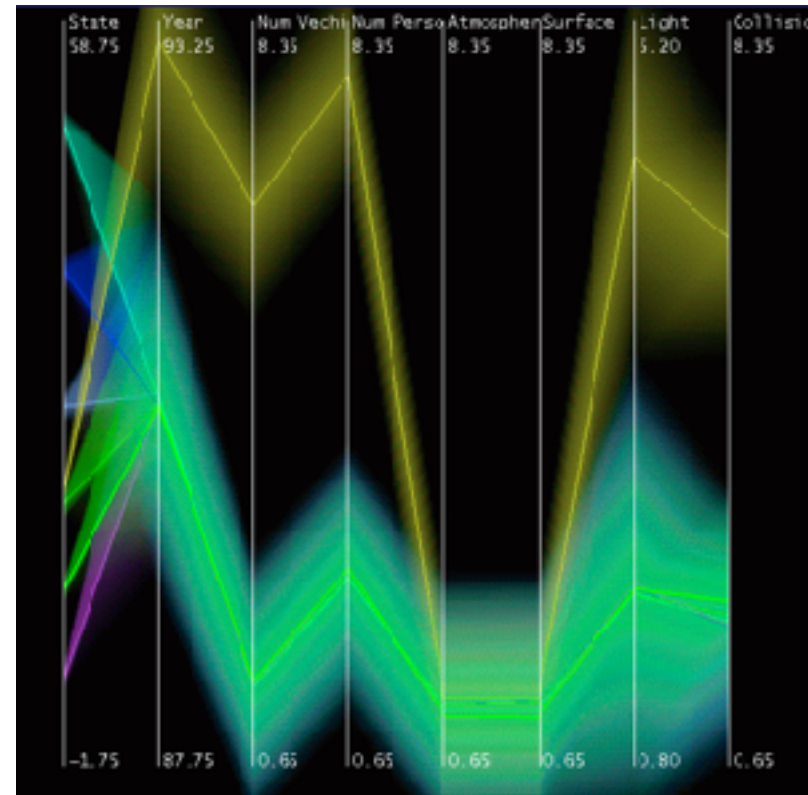
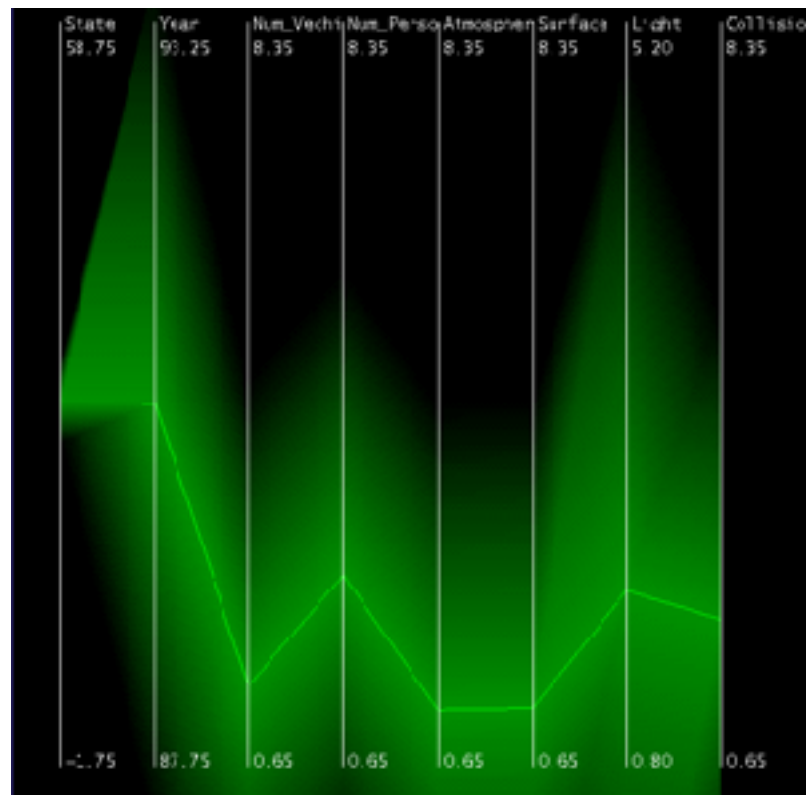
<https://www.washingtonpost.com/news/wonk/wp/2015/03/01/this-is-the-best-explanation-of-gerrymandering-you-will-ever-see/>

Dynamic aggregation: Clustering

- clustering: classification of items into similar bins
 - based on similarity measure
 - hierarchical algorithms produce "similarity tree": cluster hierarchy
 - agglomerative clustering: start w/ each node as own cluster, then iteratively merge
- cluster hierarchy: derived data used w/ many dynamic aggregation idioms
 - cluster more homogeneous than whole dataset
 - statistical measures & distribution more meaningful

Idiom: **Hierarchical parallel coordinates**

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

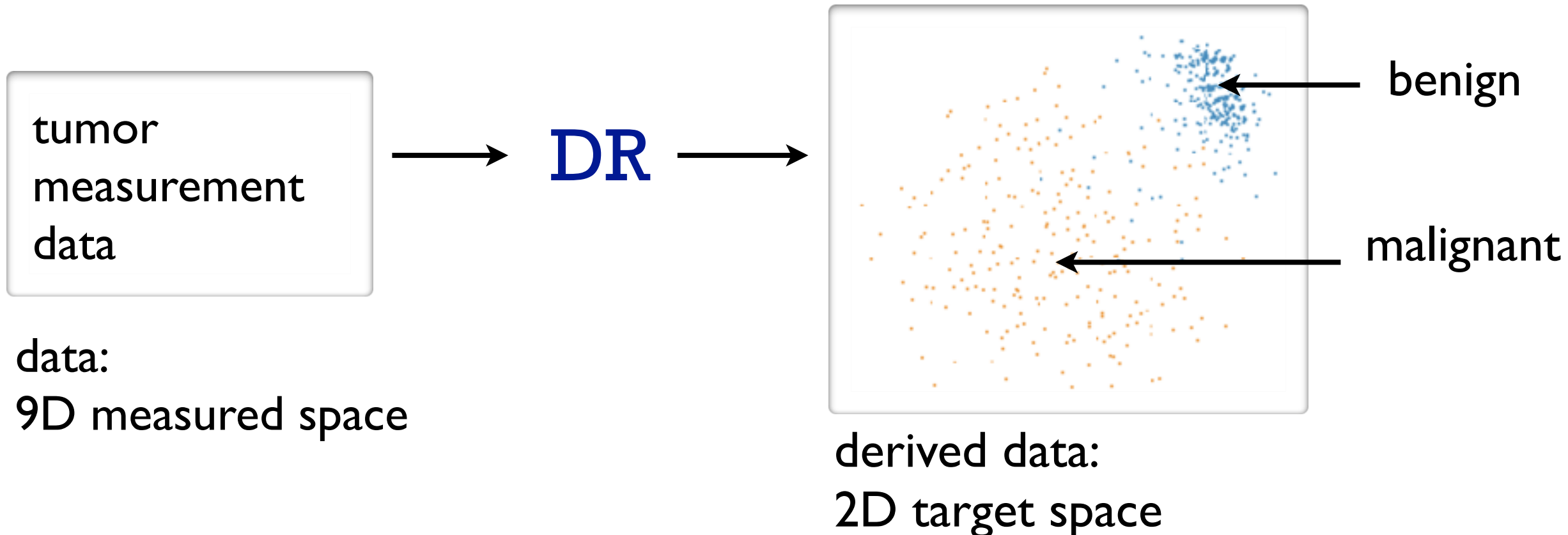


[Hierarchical Parallel Coordinates for Exploration of Large Datasets. Fua, Ward, and Rundensteiner.
Proc. IEEE Visualization Conference (Vis '99), pp. 43– 50, 1999.]

Dimensionality Reduction

Attribute aggregation: Dimensionality reduction

- attribute aggregation
 - derive low-dimensional target space from high-dimensional measured space
 - capture most of variance with minimal error
 - use when you can't directly measure what you care about
 - true dimensionality of dataset conjectured to be smaller than dimensionality of measurements
 - latent factors, hidden variables



Dimensionality vs attribute reduction

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

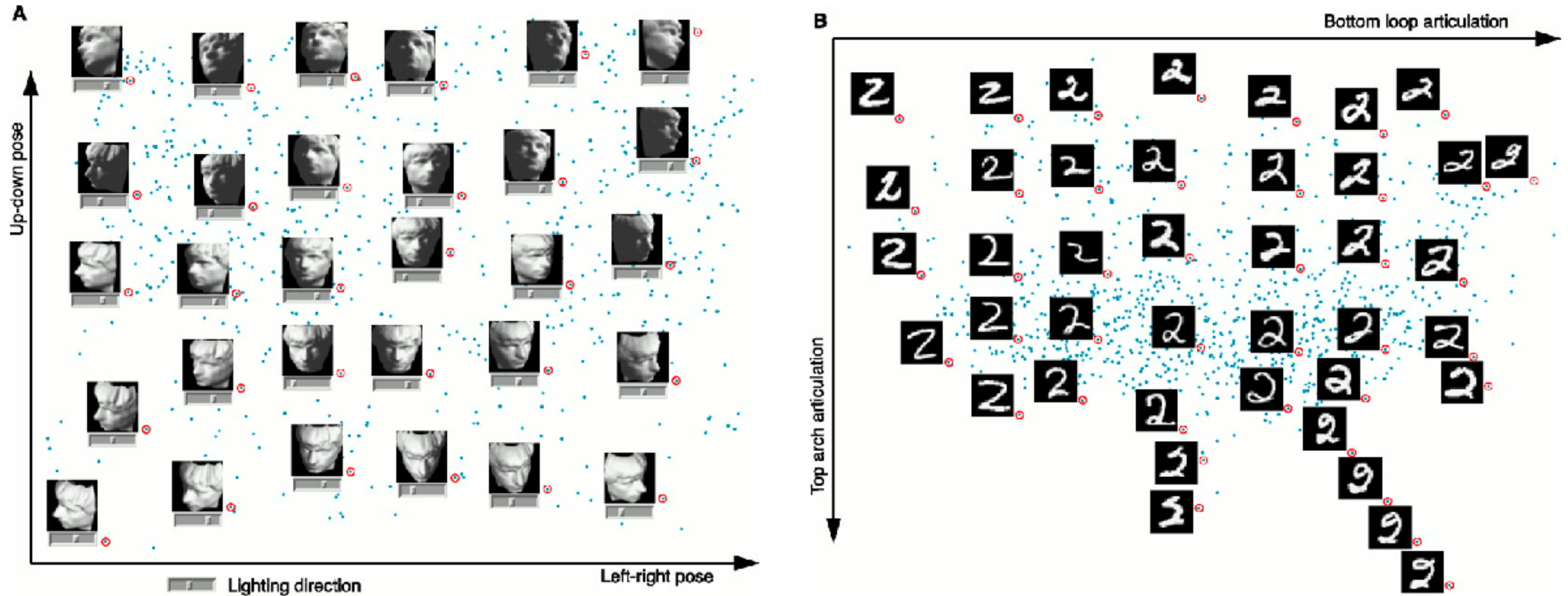
Dimensionality reduction & visualization

- why do people do DR?
 - improve performance of downstream algorithm
 - avoid curse of dimensionality
 - data analysis
 - if look at the output: visual data analysis
- abstract tasks when visualizing DR data
 - dimension-oriented tasks
 - naming synthesized dims, mapping synthesized dims to original dims
 - cluster-oriented tasks
 - verifying clusters, naming clusters, matching clusters and classes

[Visualizing Dimensionally-Reduced Data: Interviews with Analysts and a Characterization of Task Sequences. Brehmer, Sedlmair, Ingram, and Munzner. Proc. BELIV 2014.]

Dimension-oriented tasks

- naming synthesized dims: inspect data represented by lowD points

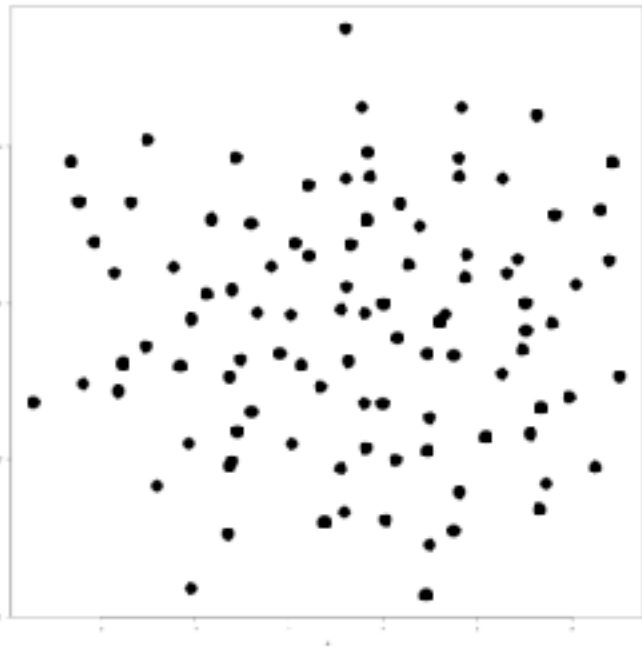


[A global geometric framework for nonlinear dimensionality reduction. Tenenbaum, de Silva, and Langford. Science, 290(5500):2319–2323, 2000.]

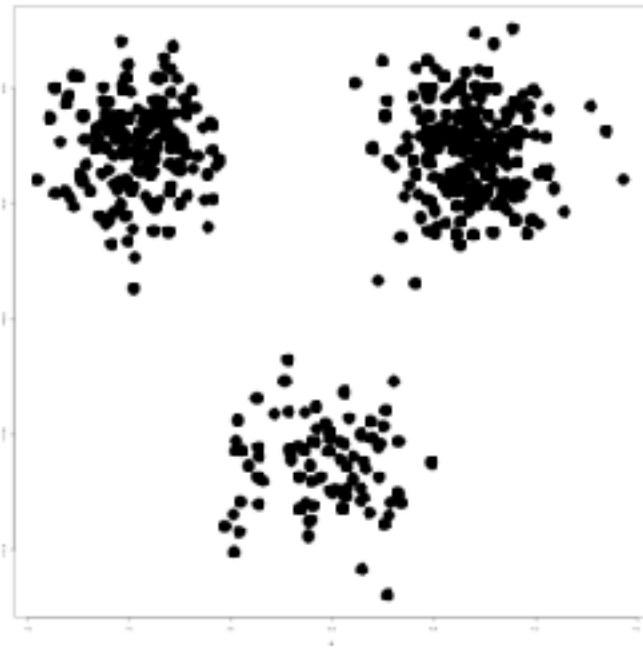
Cluster-oriented tasks

- verifying, naming, matching to classes

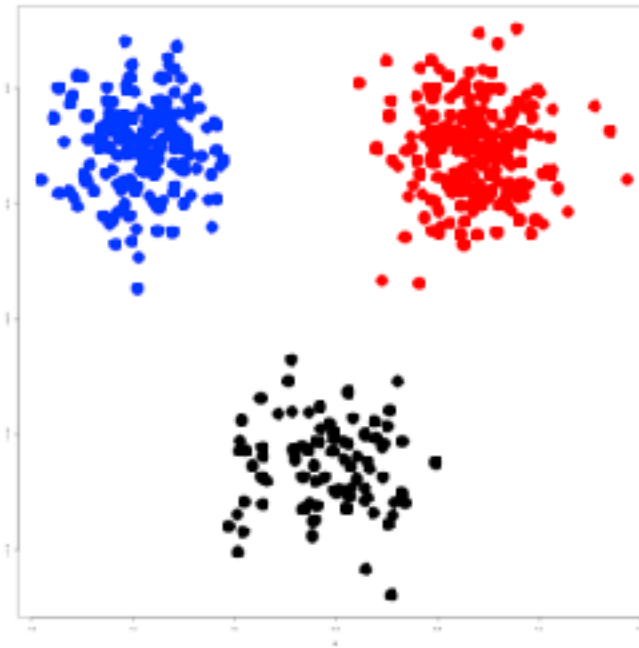
no discernable clusters



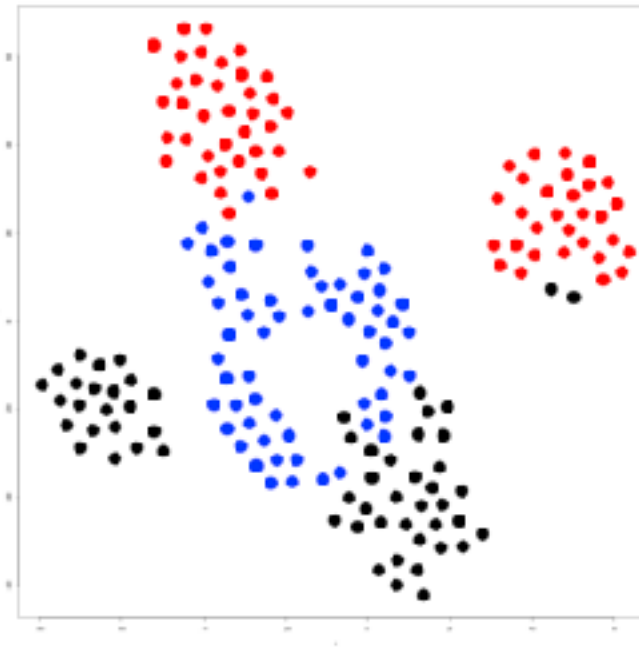
clearly discernable clusters



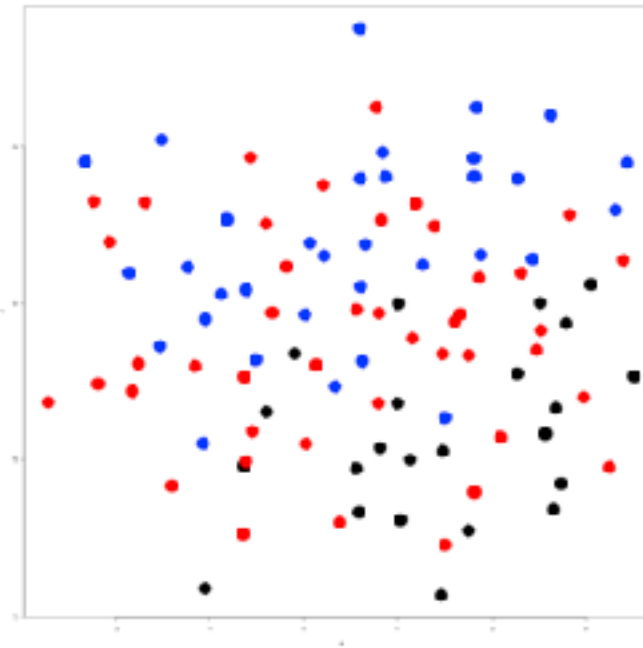
clear match cluster/class



partial match cluster/class



no match cluster/class



[Visualizing Dimensionally-Reduced Data: Interviews with Analysts and a Characterization of Task Sequences. Brehmer, Sedlmair, Ingram, and Munzner. Proc. BELIV 2014.]

Idiom: Dimensionality reduction for documents

Task 1



In HD data → **Out** 2D data

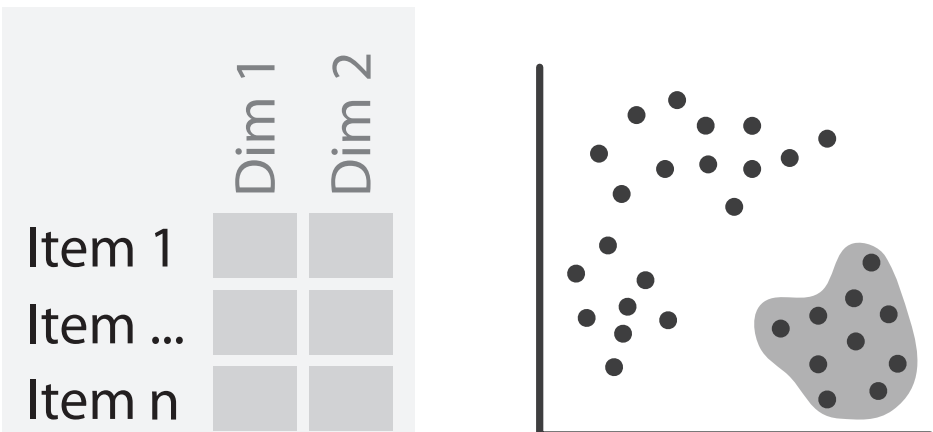
What?

- **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

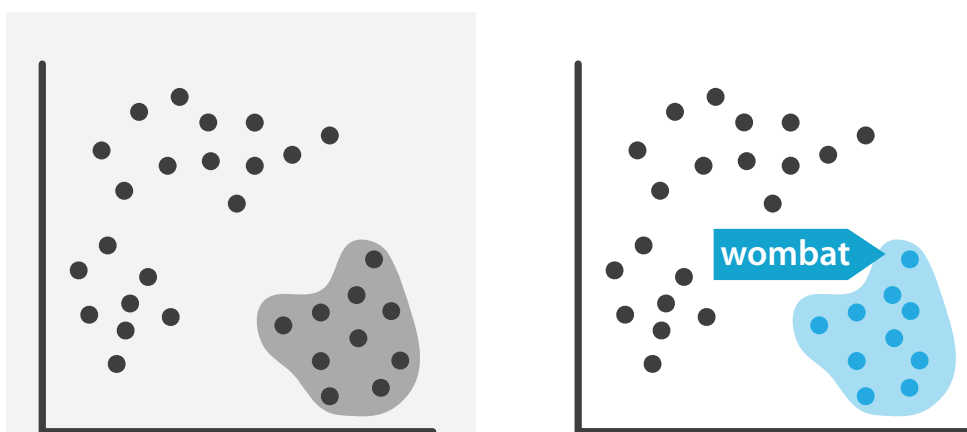
Why?

- 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

Latest algorithms: t-SNE, UMAP

- t-SNE <https://distill.pub/2016/misread-tsne/>
- UMAP <https://pair-code.github.io/understanding-umap/>



How to Use t-SNE Effectively

Although extremely useful for visualizing high-dimensional data, t-SNE plots can sometimes be mysterious or misleading. By exploring how it behaves in simple cases, we can learn to use it more effectively.

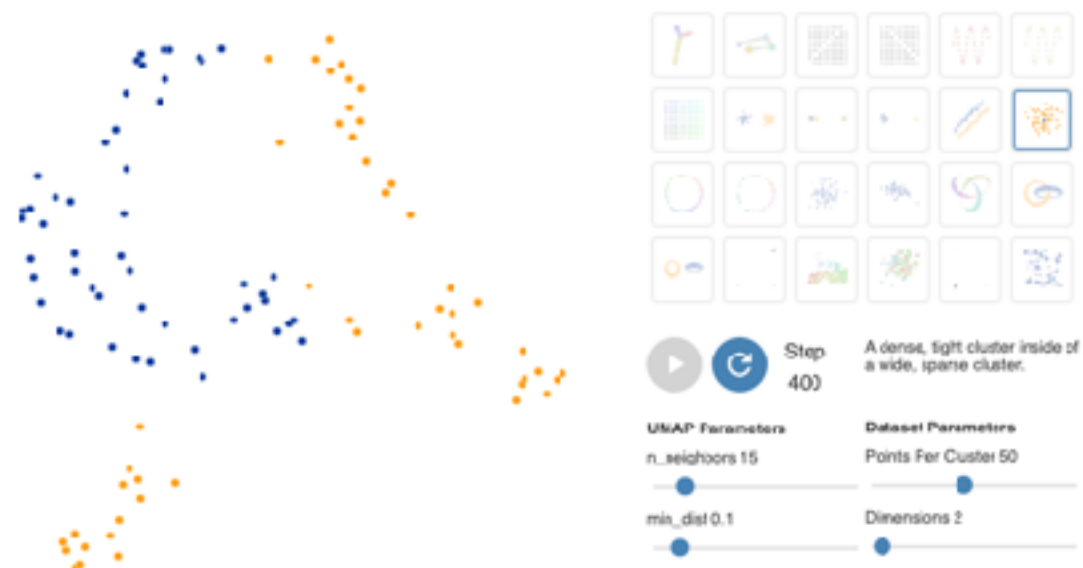


Understanding UMAP

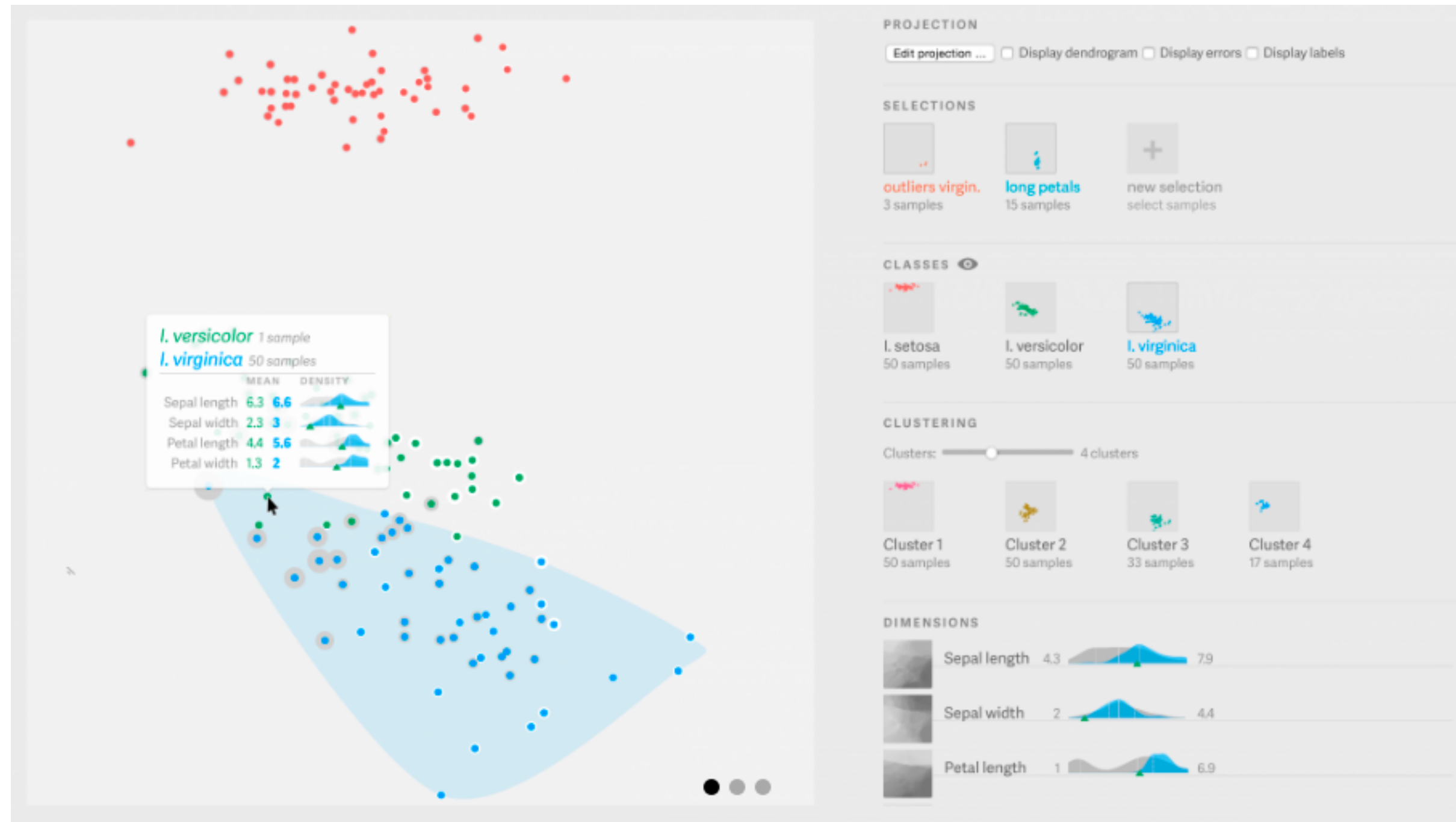
Andy Coenen, Adam Pearce | Google PAIR

Dimensionality reduction is a powerful tool for machine learning practitioners to visualize and understand large, high dimensional datasets. One of the most widely used techniques for visualization is t-SNE, but its performance suffers with large datasets and using it correctly can be challenging.

UMAP is a new technique by McInnes et al. that offers a number of advantages over t-SNE, most notably increased speed and better preservation of the data's global structure. In this article, we'll take a look at the theory behind UMAP in order to better understand how the algorithm works, how to use it effectively, and how its performance compares with t-SNE.



Interacting with dimensionally reduced data

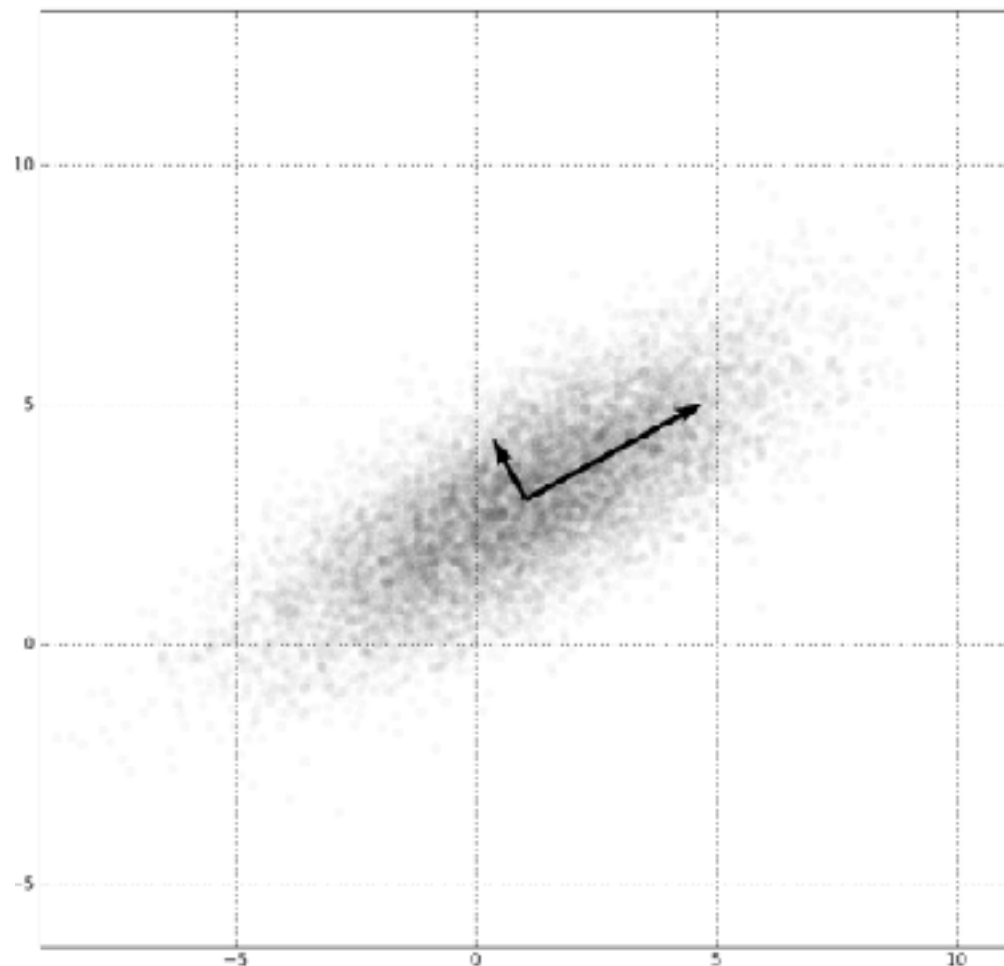


[\[https://uclab.fh-potsdam.de/projects/probing-projections/\]](https://uclab.fh-potsdam.de/projects/probing-projections/)

[Probing Projections: Interaction Techniques for Interpreting Arrangements and Errors of Dimensionality Reductions. Stahnke, Dörk, Müller, and Thom. IEEE TVCG (Proc. InfoVis 2015) 22(1):629-38 2016.]

Linear dimensionality reduction

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



[<http://en.wikipedia.org/wiki/File:GaussianScatterPCA.png>]

Nonlinear dimensionality reduction

- pro: can handle curved rather than linear structure
- cons: lose all ties to original dims/attribs
 - new dimensions often cannot be easily related to originals
 - mapping synthesized dims to original dims task is difficult
- many techniques proposed
 - many literatures: visualization, machine learning, optimization, psychology, ...
 - techniques: t-SNE, MDS (multidimensional scaling), charting, isomap, LLE, ...
 - t-SNE: excellent for clusters
 - but some trickiness remains: <http://distill.pub/2016/misread-tsne/>
 - MDS: confusingly, entire family of techniques, both linear and nonlinear
 - minimize stress or strain metrics
 - early formulations equivalent to PCA

VDA with DR example: nonlinear vs linear

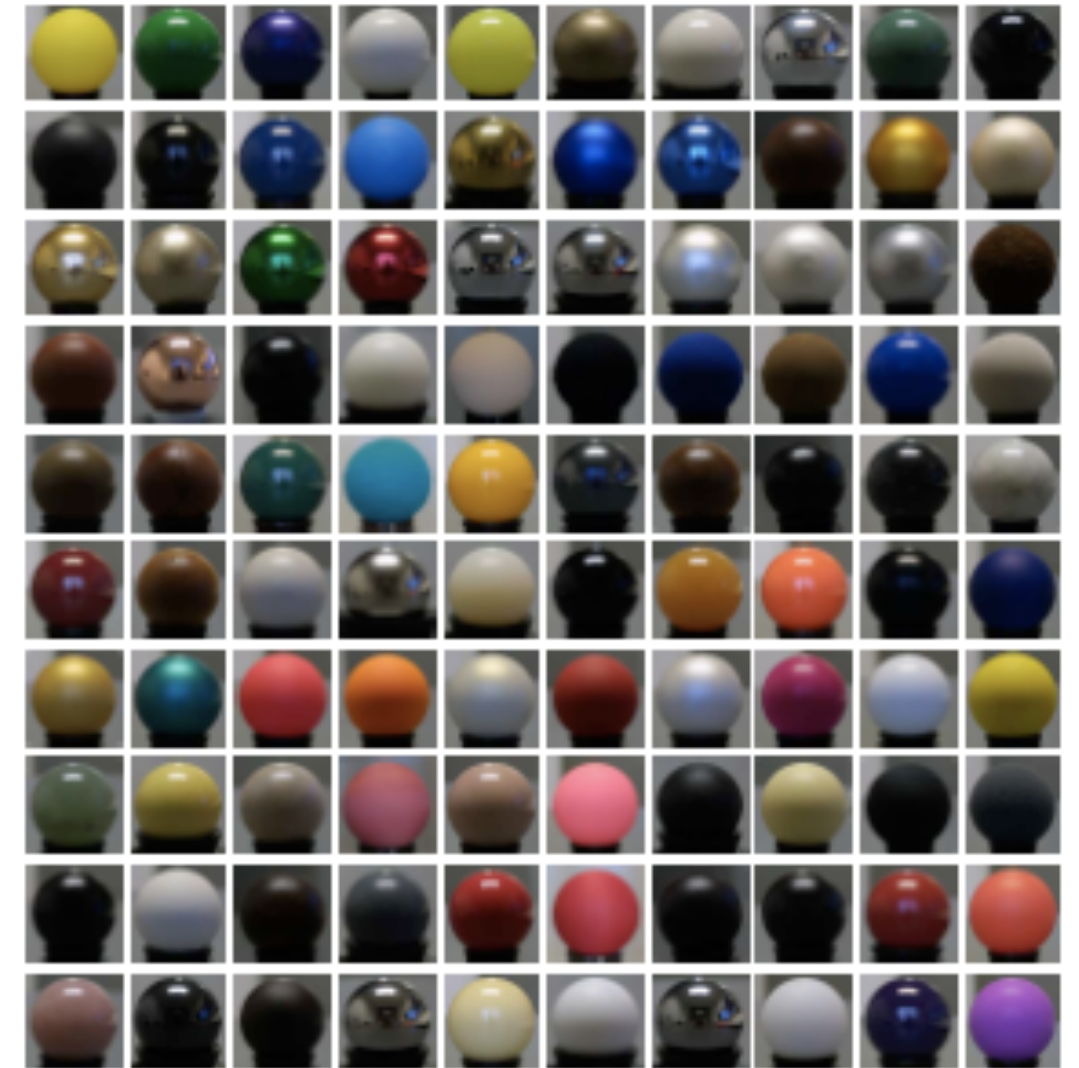
- DR for computer graphics reflectance model
 - goal: simulate how light bounces off materials to make realistic pictures
 - computer graphics: BRDF (reflectance)
 - idea: measure what light does with real materials



[Fig 2. Matusik, Pfister, Brand, and McMillan. A Data-Driven Reflectance Model. SIGGRAPH 2003]

Capturing & using material reflectance

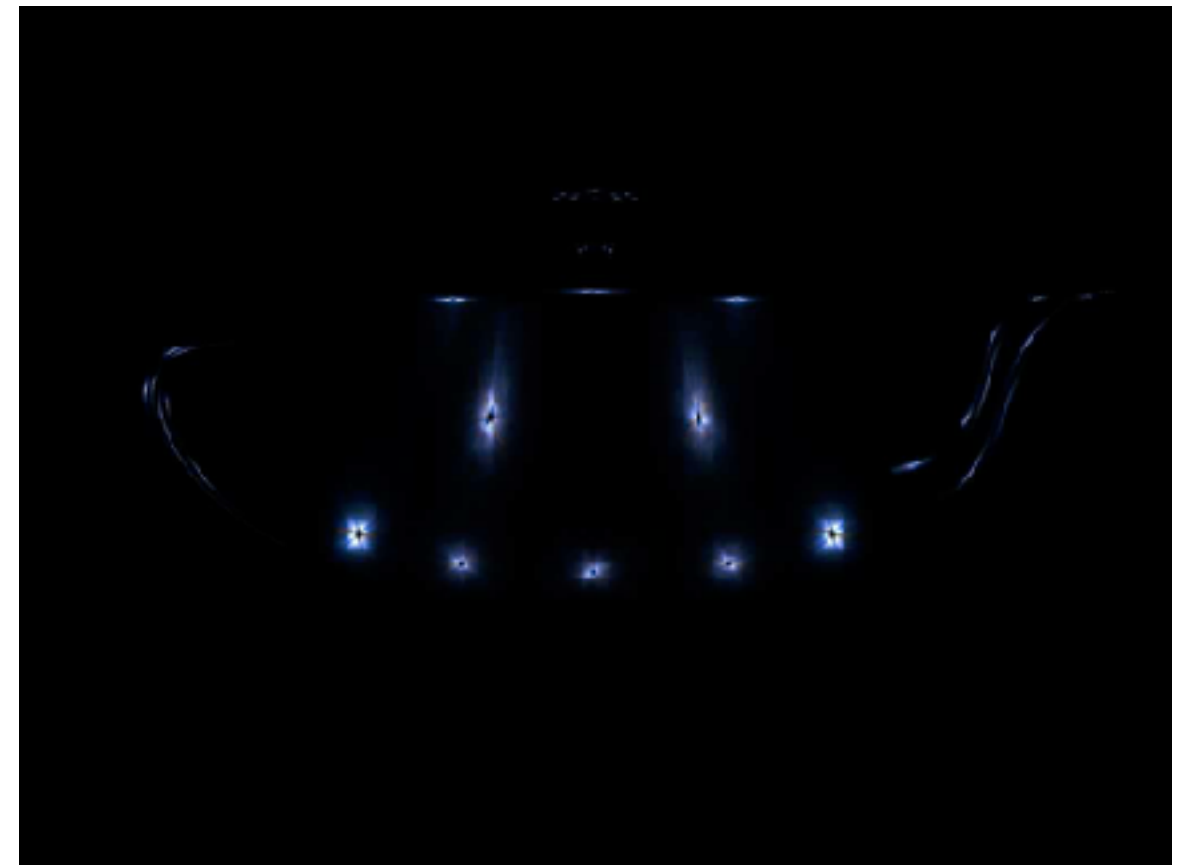
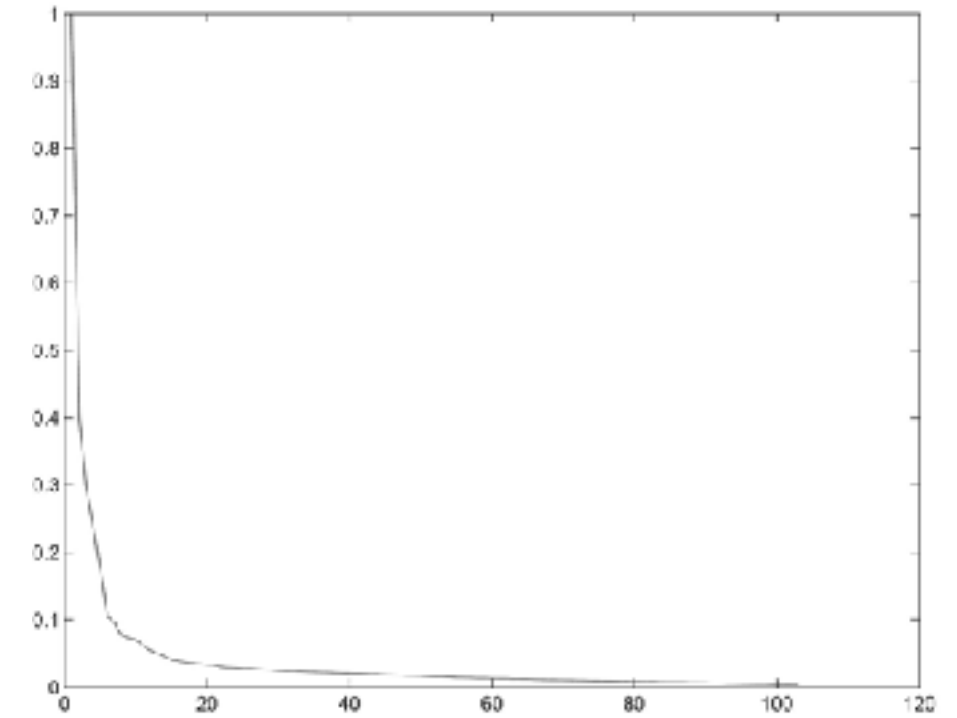
- reflectance measurement: interaction of light with real materials (spheres)
- result: 104 high-res images of material
 - each image 4M pixels
- goal: image synthesis
 - simulate completely new materials
- need for more concise model
 - 104 materials * 4M pixels = 400M dims
 - want concise model with meaningful knobs
 - how shiny/greasy/metallic
 - DR to the rescue!



[Figs 5/6. Matusik et al. A Data-Driven Reflectance Model. SIGGRAPH 2003]

Linear DR

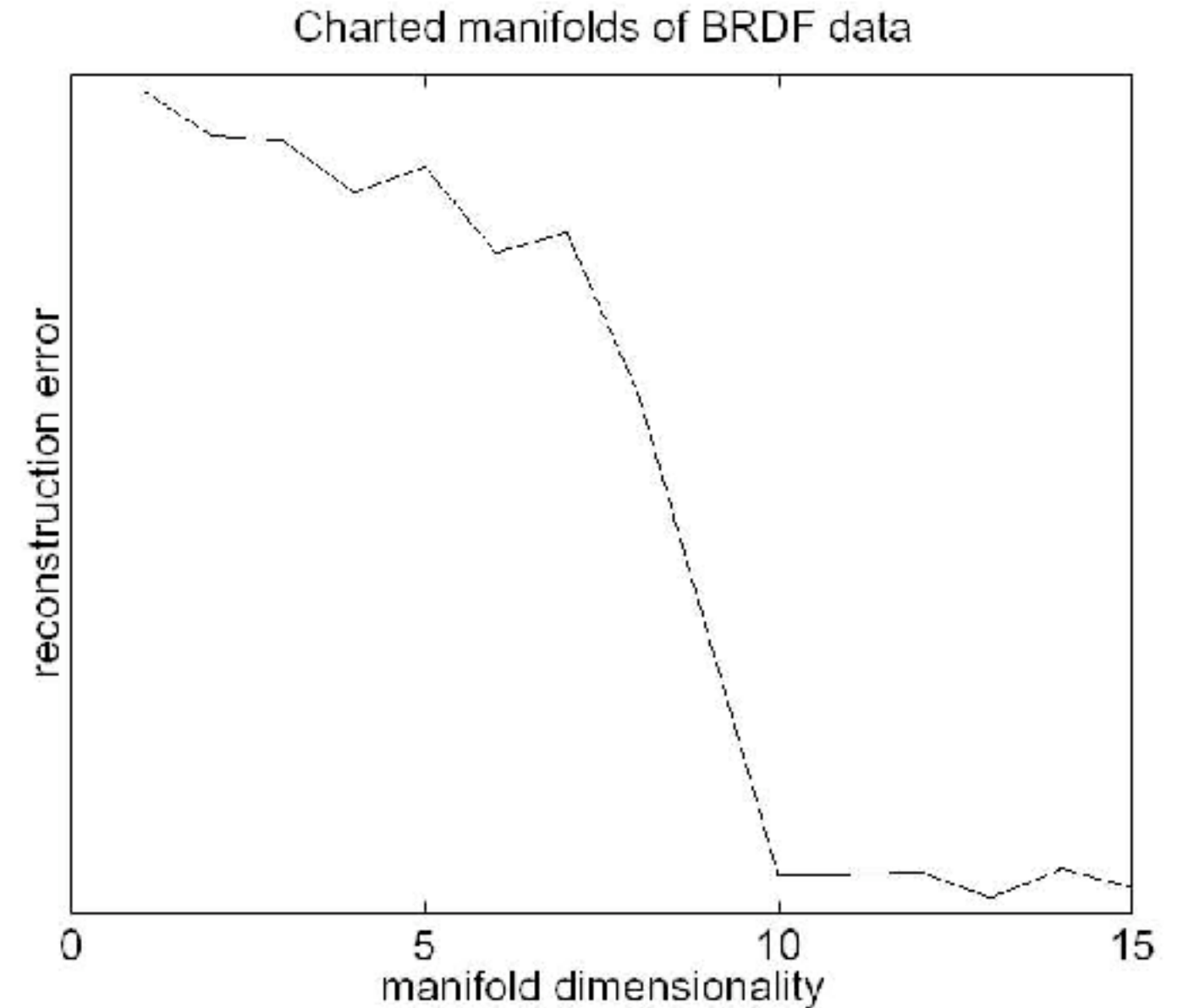
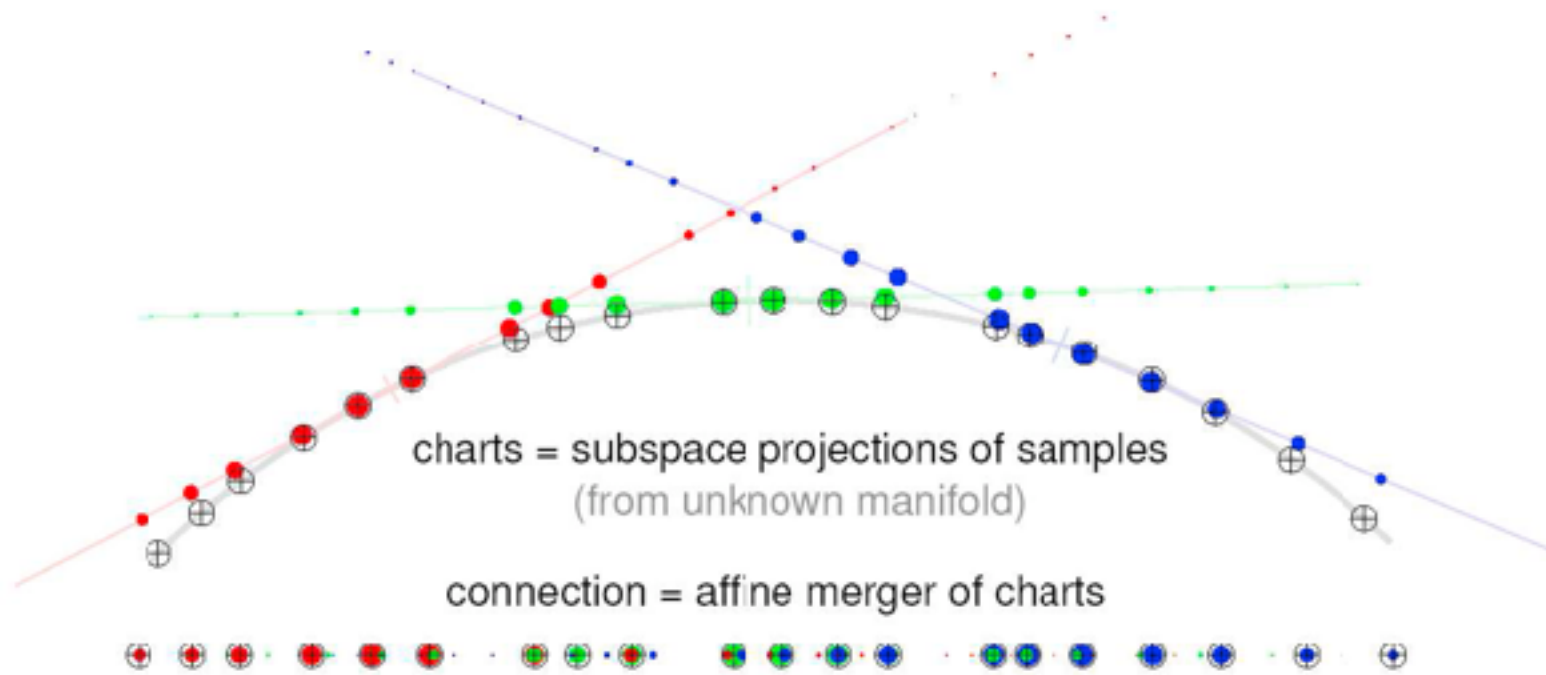
- first try: PCA (linear)
- result: error falls off sharply after ~45 dimensions
 - scree plots: error vs number of dimensions in lowD projection
- problem: physically impossible intermediate points when simulating new materials
 - specular highlights cannot have holes!



[Figs 6/7. Matusik et al. A Data-Driven Reflectance Model. SIGGRAPH 2003]

Nonlinear DR

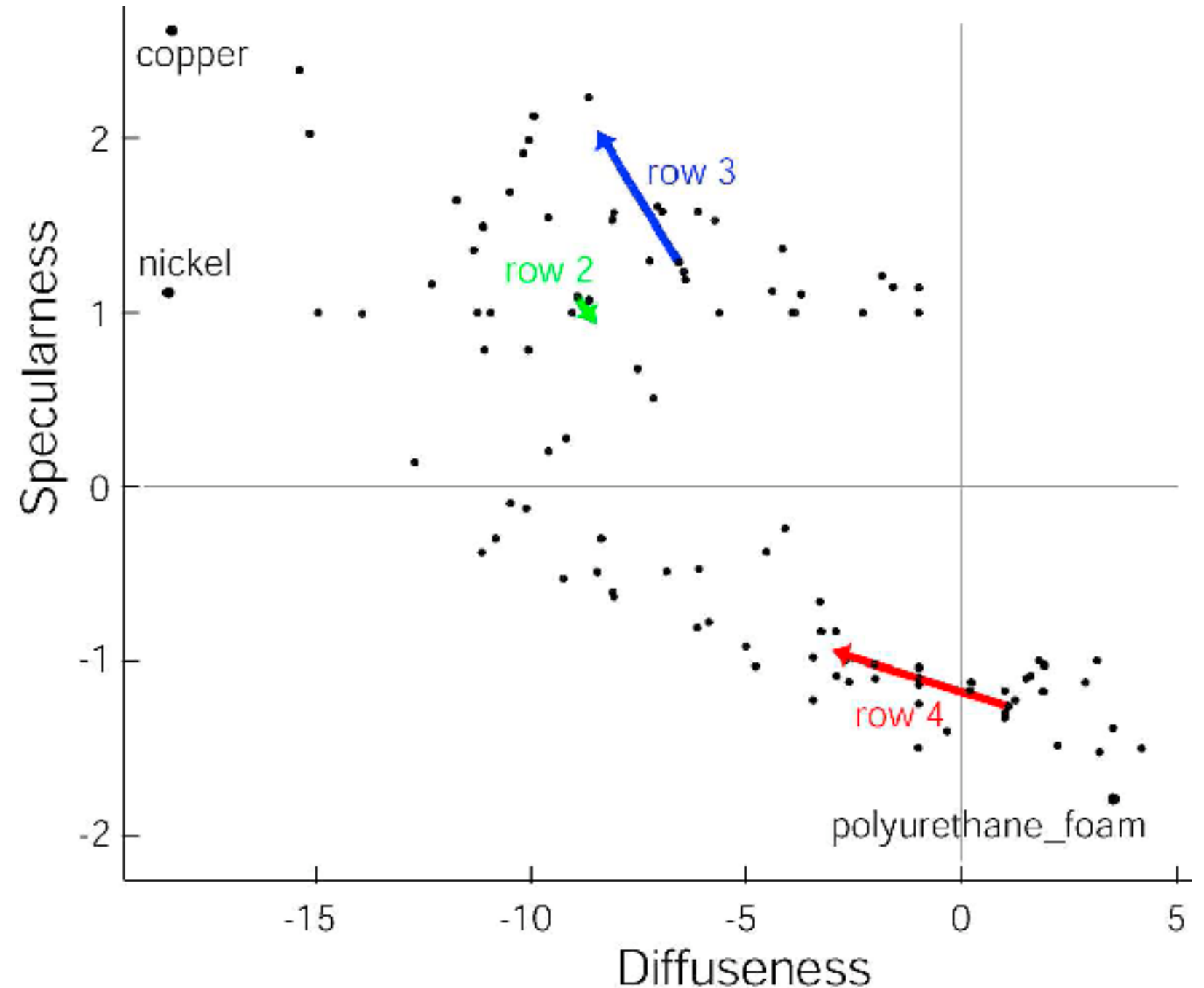
- second try: charting (nonlinear DR technique)
 - scree plot suggests 10-15 dims
 - note: dim estimate depends on technique used!



[Fig 10/11. Matusik et al. A Data-Driven Reflectance Model. SIGGRAPH 2003]

Finding semantics for synthetic dimensions

- look for meaning in scatterplots
 - synthetic dims created by algorithm but named by human analysts
 - points represent real-world images (spheres)
 - people inspect images corresponding to points to decide if axis could have meaningful name
- cross-check meaning
 - arrows show simulated images (teapots) made from model
 - check if those match dimension semantics

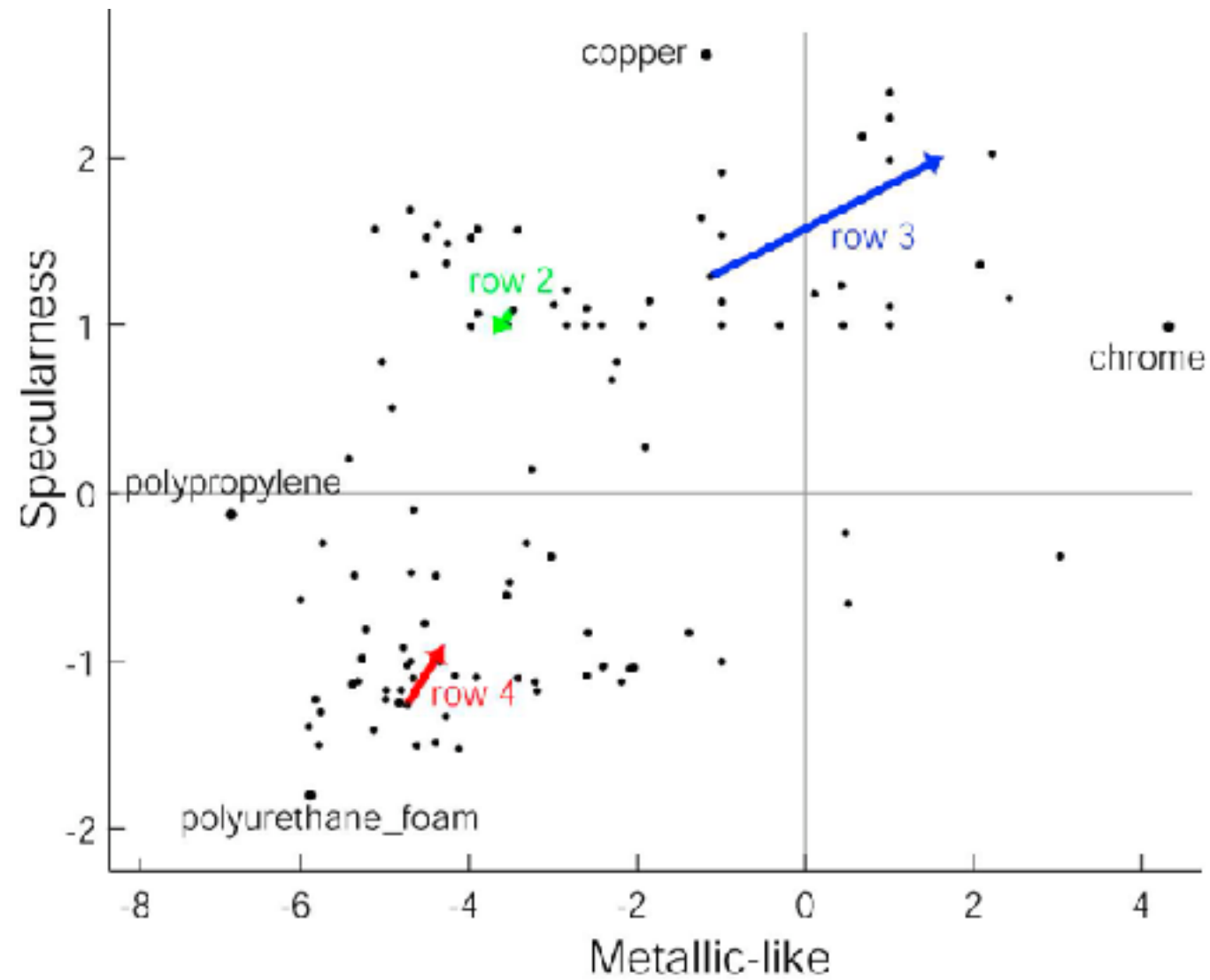


row 4

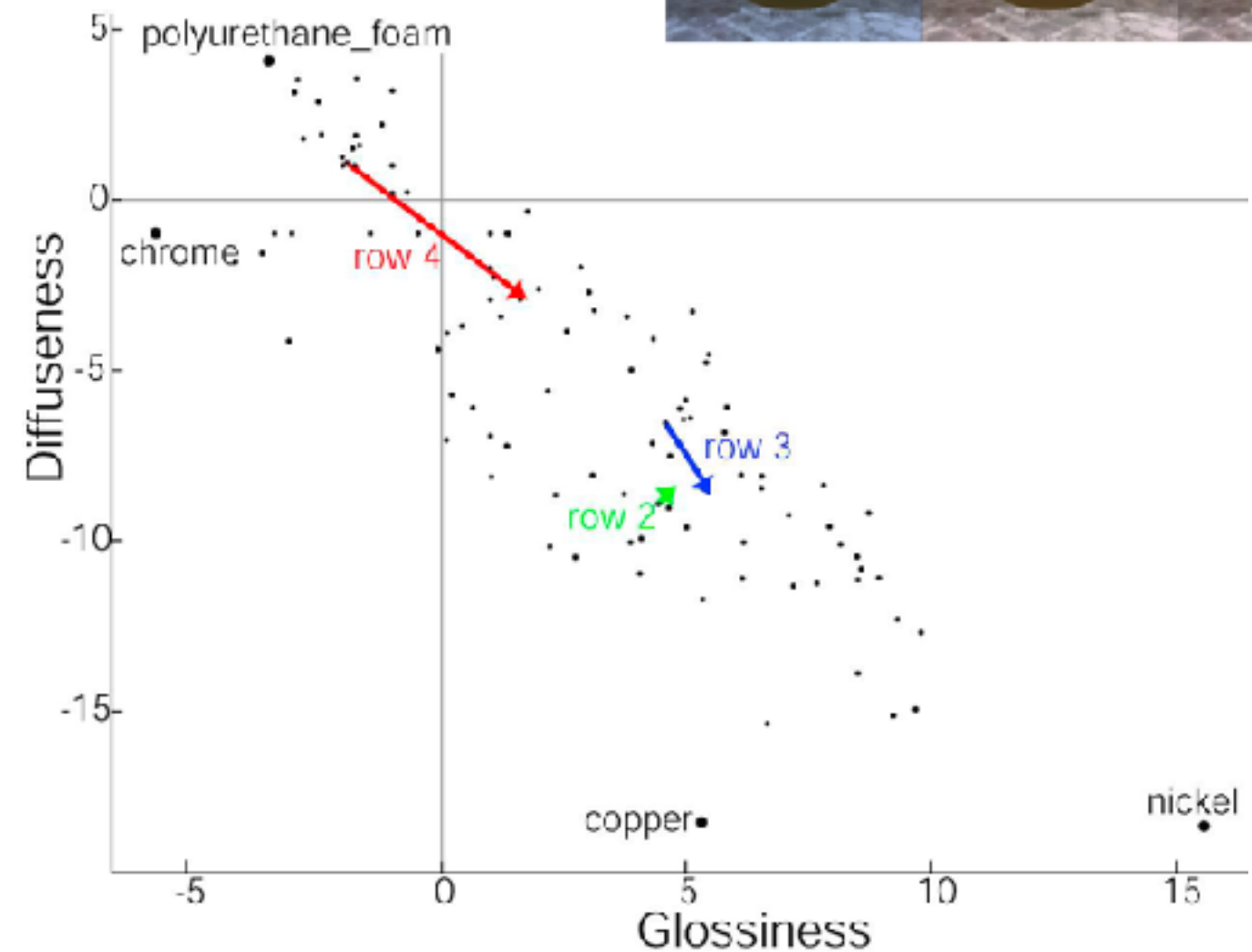


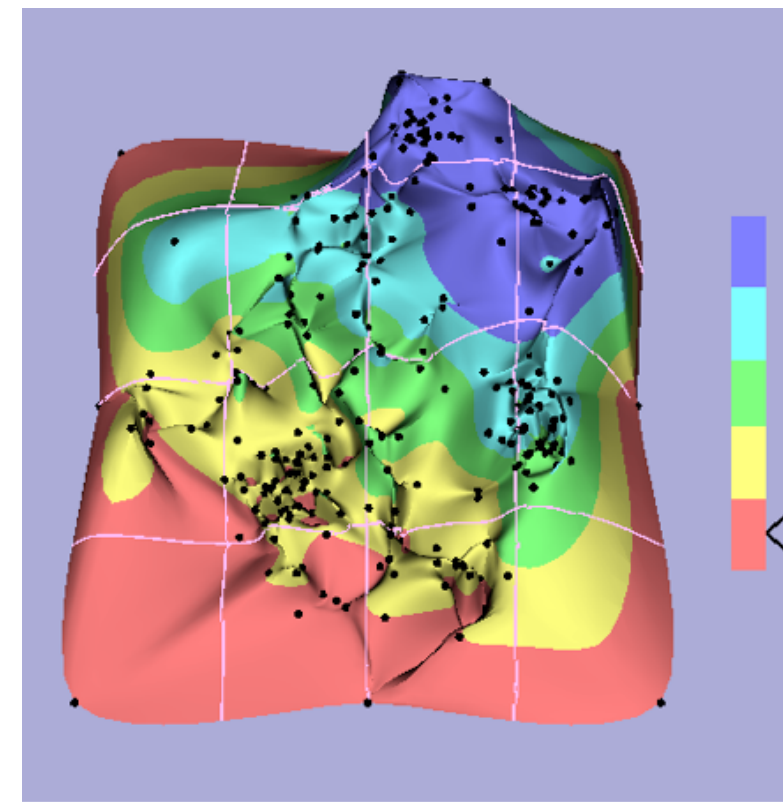
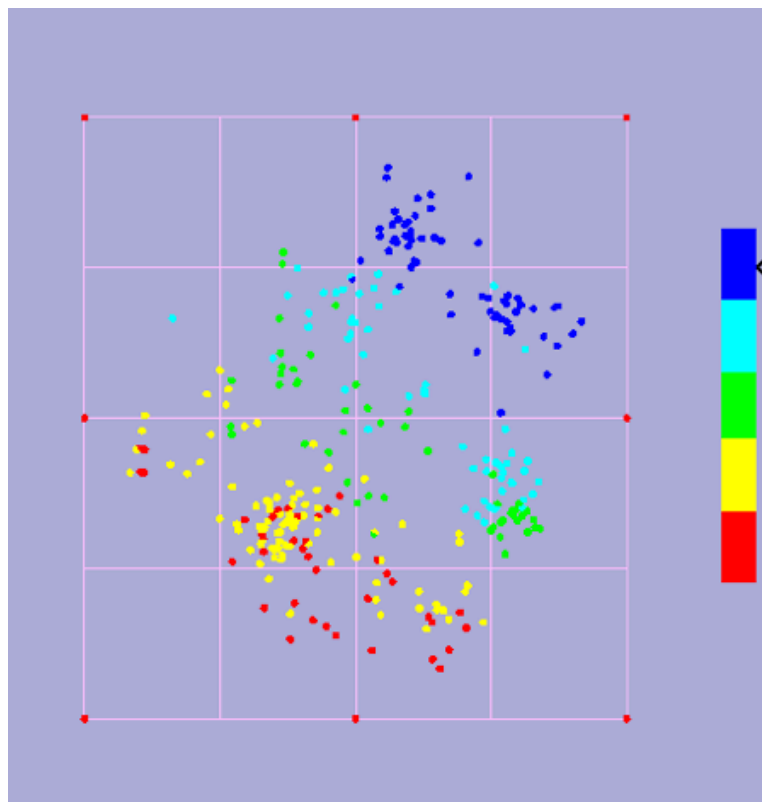
Understanding synthetic dimensions

Specular-Metallic



Diffuseness-Glossiness





Spatialization Design

Comparing Points and Landscapes

joint work with:

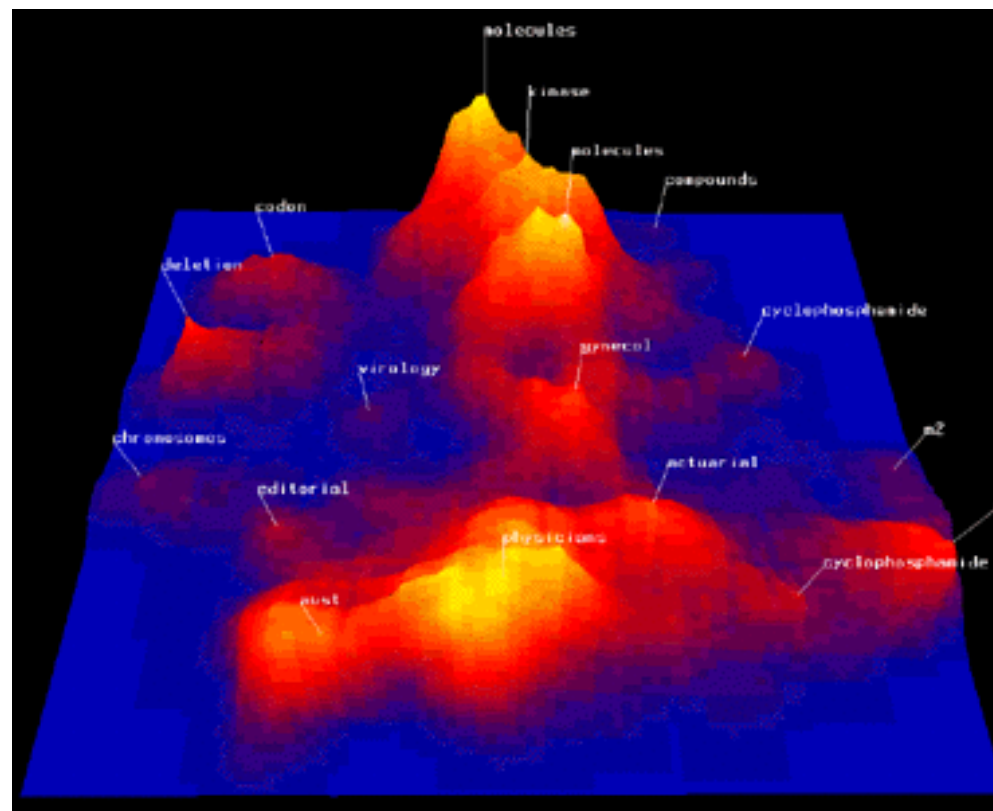
Melanie Tory, David W. Sprague, Fuqu Wu, Wing Yan So

<http://webhome.cs.uvic.ca/~mtory/publications/infovis2007.pdf>

Spatialization Design: Comparing Points and Landscapes.
Tory, Sprague, Wu, So, and Munzner.
IEEE TVCG 13(6):1262–1269, 2007 (Proc. InfoVis 07).

Information Landscapes

- 2D or 3D landscape from set of DR points
 - height based on density
- oddly popular choice in DR
 - despite known occlusion/distortion problems with 3D
 - assertions: pattern recognition, spatial reasoning, familiar



Themescape:
[<http://www.k-n-o-r-z.de/publ/example/retrieval1.htm>]

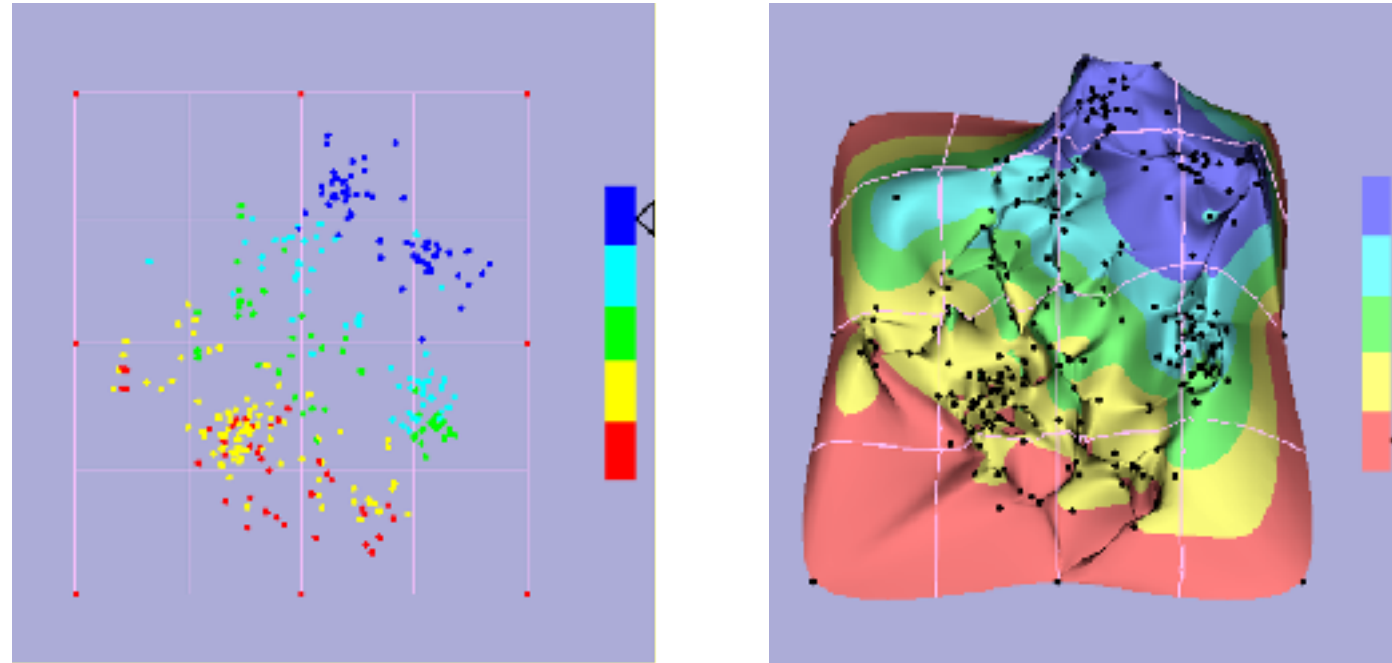


[Guide to MicroPatent Aureka 9 ThemeScape]

Understanding User Task

- abstract: search involving spatial areas and estimation

Estimate which grid cell has the most points of the target color



- domain-specific examples

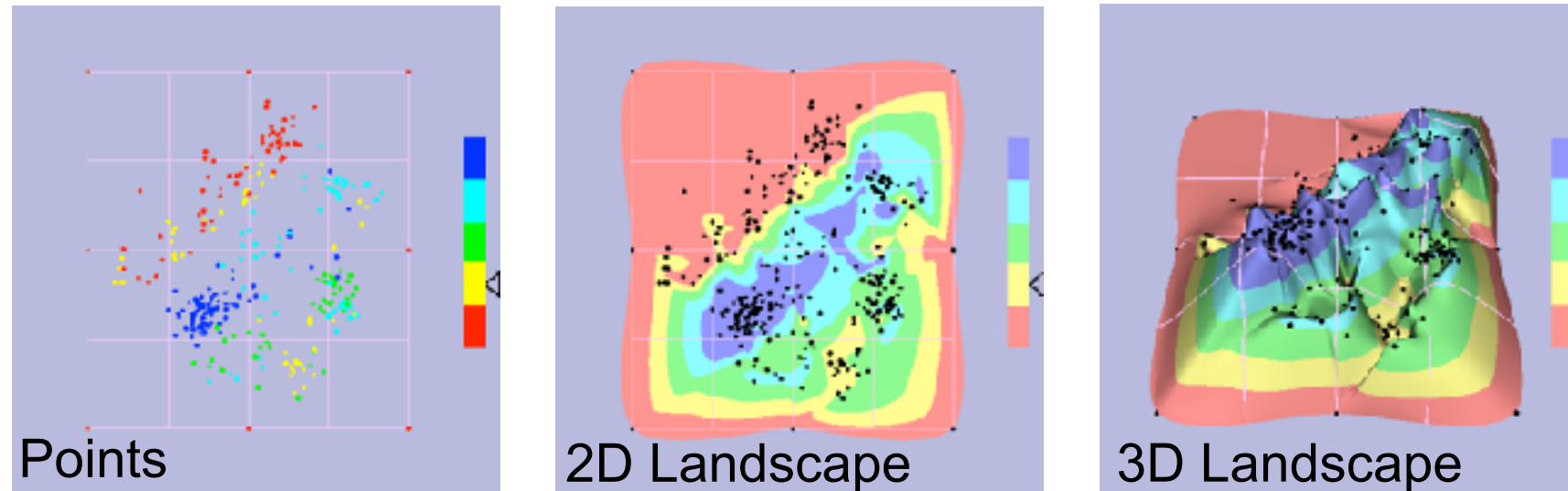
“Where in the display are people with high incomes?”

“Does this area also have high education levels?”

“Does this area correspond to a particular work sector?”

- non-trivial complexity yet fast response time
- frequent subtask in pilot test of real data analysis

Lab Study: Test Human Response Time and Error



- hypotheses
 - points are better than landscapes
 - result: yes!
 - much better: 2-4 × faster, 5-14 × more accurate
 - 2D landscapes (color only) better than 3D landscapes (color + height redundantly encoded)
 - result: yes
 - significantly faster, no significant difference in accuracy

How?

Encode

➔ Arrange

➔ Express



➔ Separate



➔ Order



➔ Align



➔ Use



➔ Map

from **categorical** and **ordered** attributes

➔ Color

➔ Hue



➔ Saturation



➔ Luminance



➔ Size, Angle, Curvature, ...



➔ Shape



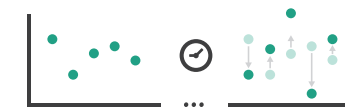
➔ Motion

Direction, Rate, Frequency, ...

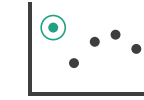


Manipulate

➔ Change



➔ Select



➔ Navigate

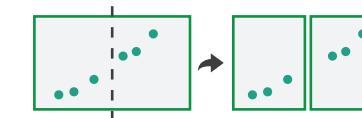


Facet

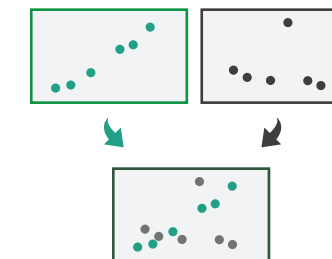
➔ Juxtapose



➔ Partition



➔ Superimpose



Reduce

➔ Filter



➔ Aggregate



➔ Embed



What?

Why?

How?