Interactive Visual Data Analysis

04 – Visual mapping

Objectives

- How to build a visual representation of data: Learn about the basic building blocks
- How can data be encoded visually: Learn the fundamentals of mapping data to visual representations

Overview

- Visual encoding
  - Marks and visual variables
  - Visual mapping
- Presentation
  - Presenting marks and views on the display

Motivation

- **Visual representations** are **most crucial** for visual data analysis
- Many different options for mapping data to visual representations
- Visual representations should fulfill the quality criteria expressiveness, effectiveness, and efficiency
- **The representation effect:**

> "The representation effect: Human **performance varies** enormously (10–100:1) **with** different **representations**.
>
> — Hanrahan, 2009

Motivation

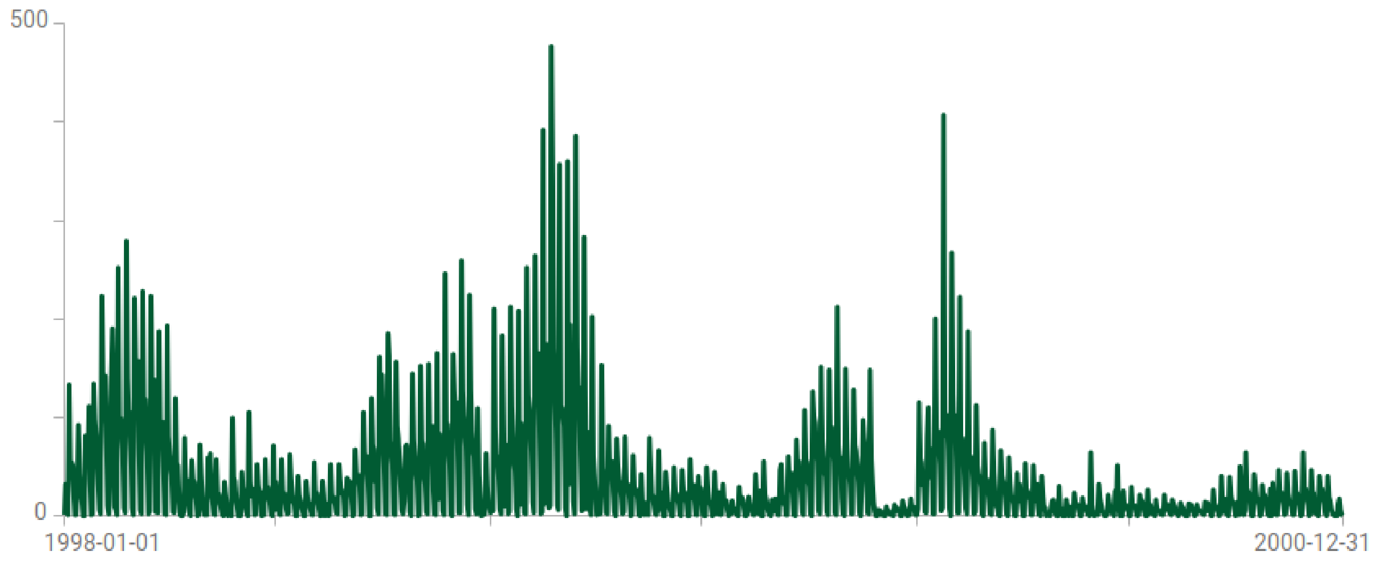
Illustration of the representation effect

- Data: Temporal data with number of people reported sick per day
- Representation: Visual mapping using different visualization techniques and different parameterizations
  - Line plot
  - Spiral display
    - Cycle length 32 days
    - Cycle length 28 days

Motivation

Illustration of the representation effect

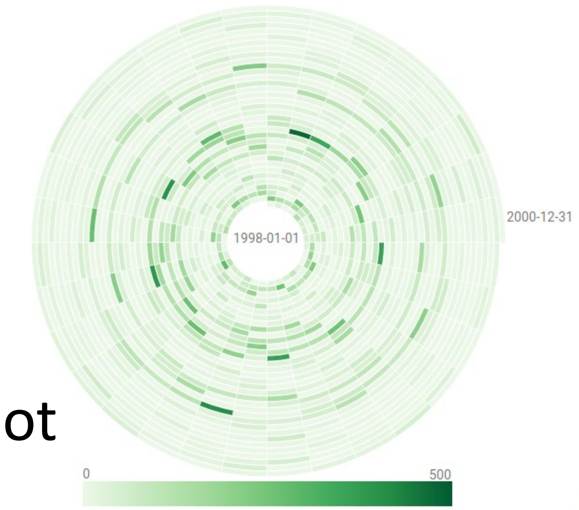
- Line plot
  - We can locate peaks
  - We see phases of moderate and low numbers of cases
  - We do not see a clear trend
  - Think about: Are there any cyclic patterns?



Motivation

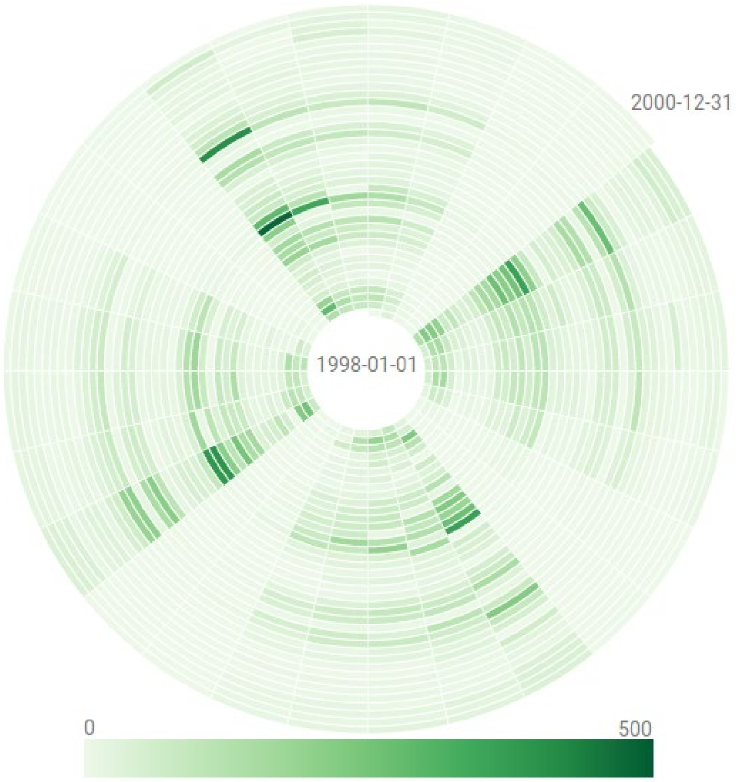
Illustration of the representation effect

- Spiral display
  - Data color-coded along spiral
  - With cycle length 32, we cannot see any pattern
  - With cycle length 28, we see a clear cyclic pattern
  - Think about: What do these patterns tell us?
  - Think about: What else can we see?



Cycle length 32 days

Cycle length 28 days



Motivation

Illustration of the representation effect

- We have seen how different visual representations can or cannot support different analysis questions
  - Peaks can be easily identified and located in time using the line plot, yet cyclic patterns are not visible
  - A color-coded spiral display (if cycle length is set appropriately) can show us recurring and seasonal patterns in the data, identifying peaks is possible as well, but locating them in time is more difficult

- Bottom line: **We need to design for the data and the task at hand!**

Visual encoding and presentation

For designing high-quality visual representations we need to know

- **How to encode data visually?**
  - Visual encoding
- **How to present data meaningfully?**
  - Presentation

Visual encoding

- Fundamental ideas for data graphics established by French cartographer Jacques Bertin (1967, 1983)
- Visual representations are composed of
  - **Marks** (graphical primitives)
  - **Visual variables** (visual appearance of primitives)

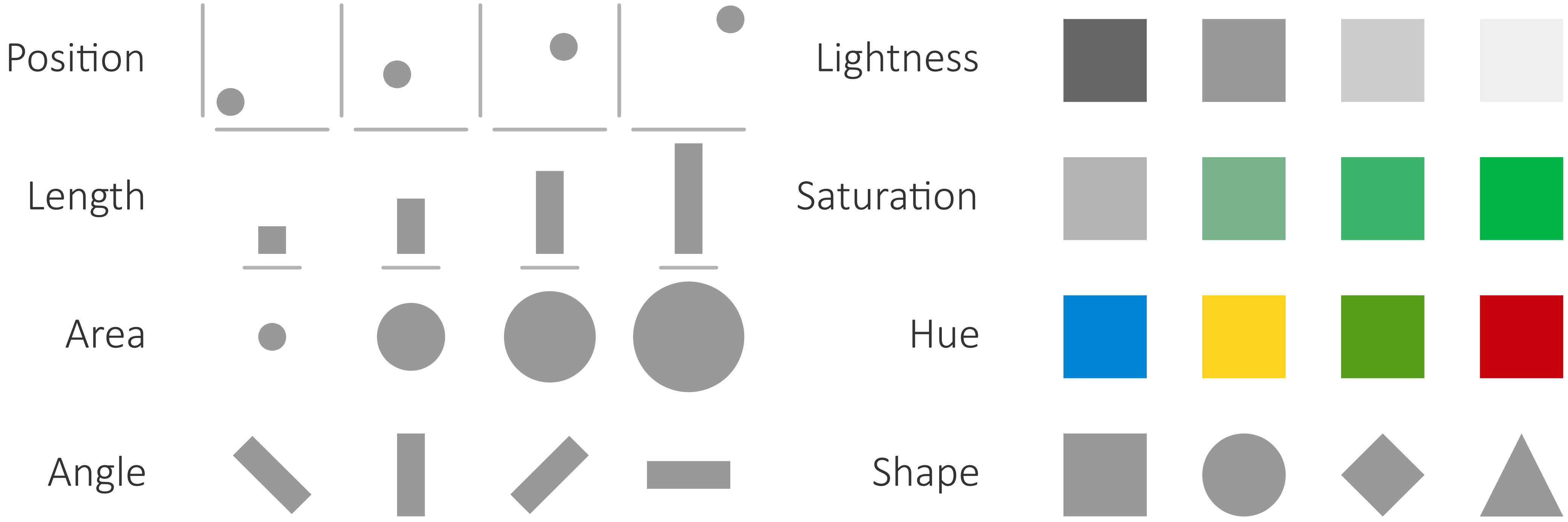
Visualizing data **means creating graphical primitives** and specifying their **visual appearance** according to the underlying data.

Visual encoding

- **Marks**
  - Basic building blocks of visual representations
  - Distinguish different types of marks based on their dimensionality (0D points, 1D lines, 2D areas, 3D bodies)
- **Visual variables**
  - Carry the actual information
  - Can be varied within a reasonable range
  - Different variations can be perceived as different

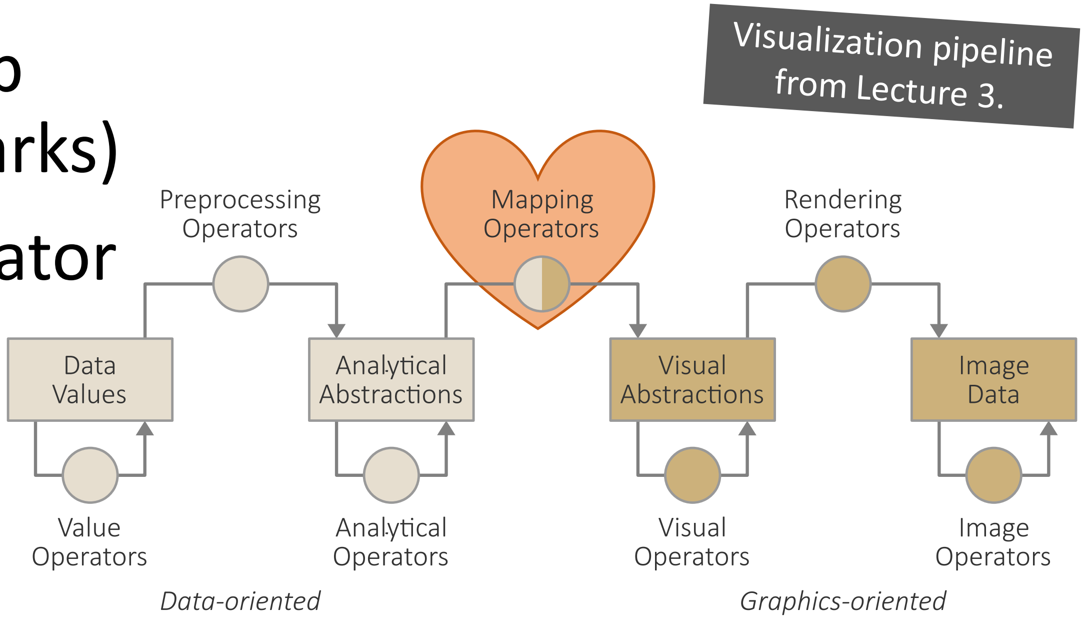
Visual encoding

Visual variables



Visual encoding

- At the heart of visual encoding, we map data variables to visual variables (of marks)
- **“Visual mapping”** transformation operator in the visualization pipeline: From analytical abstractions to visual abstractions



- Two questions need to be answered:
  - **What to map?**
  - **How to map?**

Visual encoding

What to map?

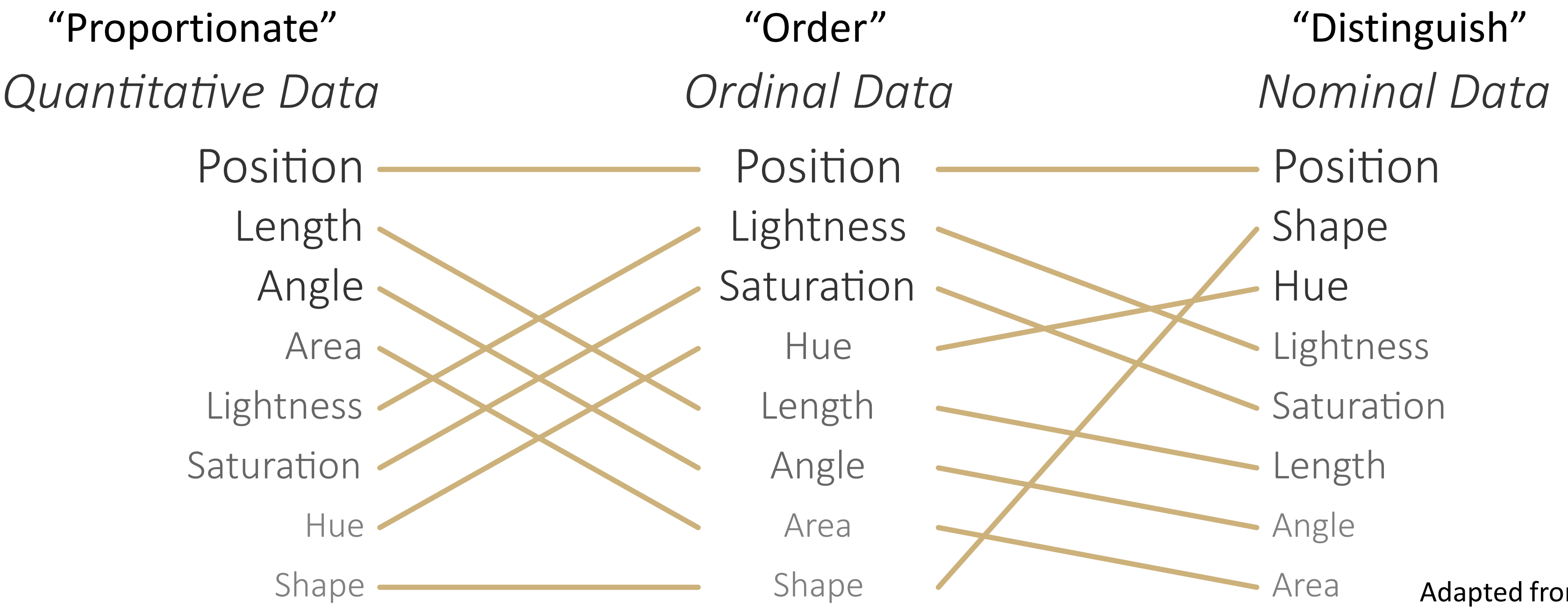
- Student: “Which data variable(s) should be mapped to which visual variable(s)?”
- Teacher: “You should map the most relevant data variables to the most effective visual variables!”

- Student: “But which data variables are relevant and which visual variables are the most effective?”
- Teacher: “Relevant data variables are prescribed by the analysis tasks and the most effective visual variables are data-dependent and have been found via perceptual studies!”

Visual encoding

What to map?

- Different visual variables are differently effective for nominal, ordinal, and quantitative data ([Cleveland and McGill, 1984]) and ([Heer and Bostock, 2010])



Adapted from Mackinlay and Winslow (2014)

Visual encoding

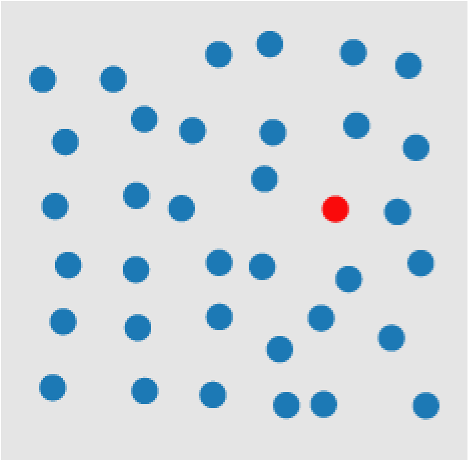
How to map?

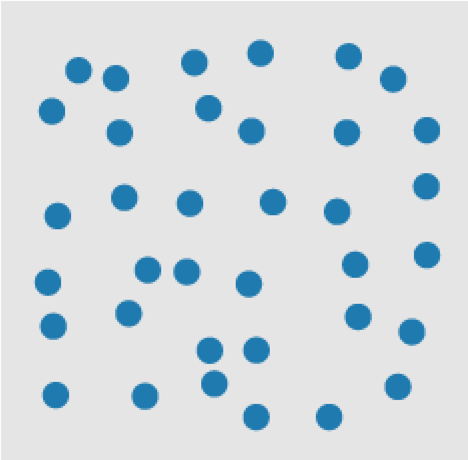
- Student: “Which values or intervals of a data variable should be mapped to which portions of a visual variable?”
- Teacher: “Map the data such that the important information stands out! Ideally, the important information can be perceived pre-attentively.”

- Student: “Pre-att.. What?”
- Teacher: “**Pre-attentive**. Things that can be **perceived** in the blink of an eye **without** requiring our full **attention** are perceived pre-attentively.”

Visual encoding

Excurse: Pre-attentive processing





https://www.csc2.ncsu.edu/faculty/healey/PP/

Visual encoding

How to map? – By the example of color-coding

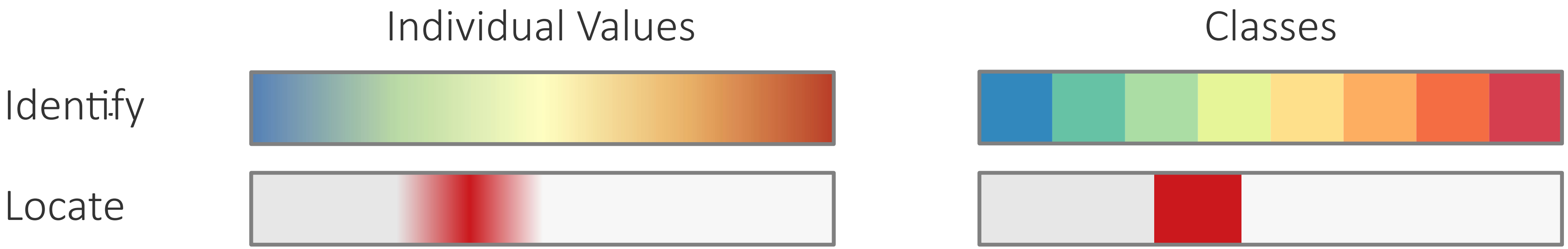
- Ideally, each data value is mapped to a unique identifiable color, though this ideal is limited by output device and human perception
- General recommendations
  - Use **perceptual color spaces** (CIE lab, HSV, etc.), not technical color spaces (RGB, CMYK)
  - Use **perceptually uniform color maps** (perceived color differences are proportional to differences in data), not just some color maps
  - Use **well-established and tested color maps**, not just some fancy colors

https://colorbrewer2.org

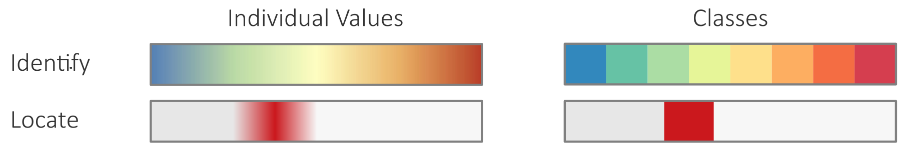
Visual encoding

How to map? – By the example of color-coding

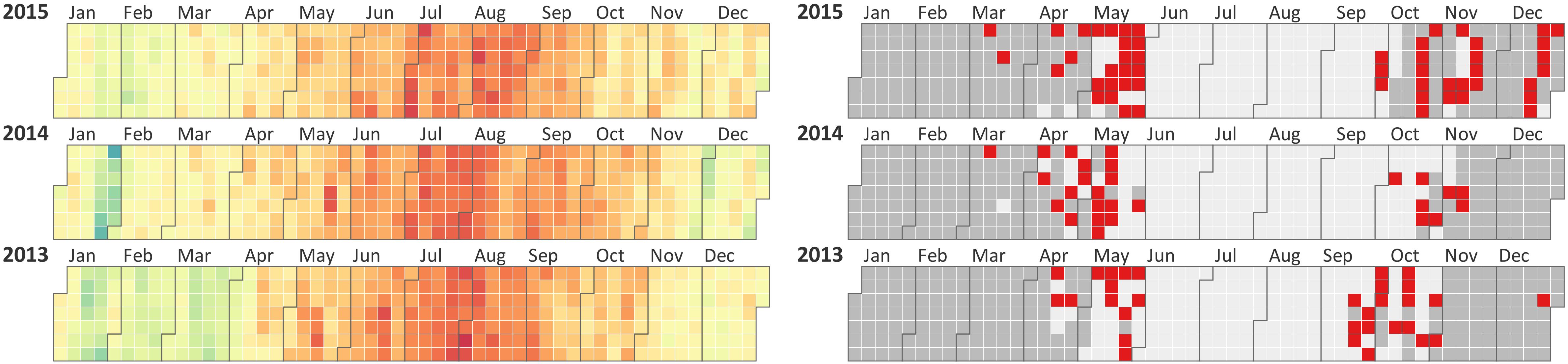
- Data: Individual values vs. classes → Continuous vs. segmented maps
- Task: Identify vs. locate → Uniform vs. highlighting maps



Visual encoding



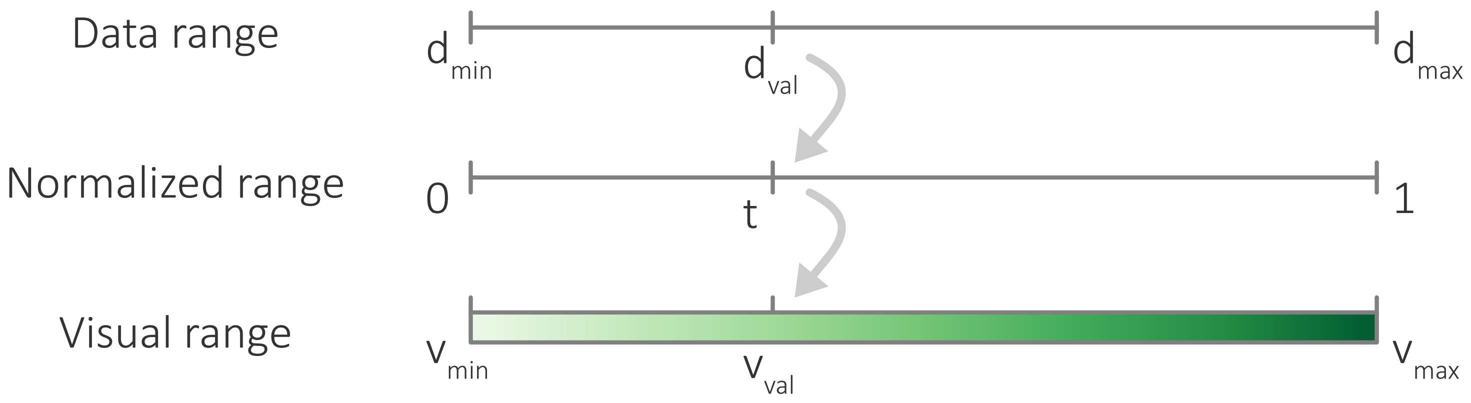
How to map? – By the example of color-coding



Visual encoding

How to map? – By the example of color-coding

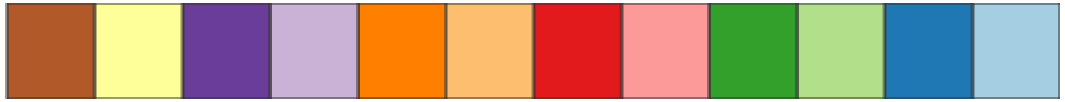
- Basic mapping of a range of numeric values to colors
  - Normalize data value $t = \frac{d_{val} - d_{min}}{d_{max} - d_{min}}$
  - Linearly map normalized value to color $v_{val} = (1 - t) \cdot v_{min} + t \cdot v_{max}$



Visual encoding

How to map? – By the example of color-coding

- Basic mapping of nominal values to colors
  - Assign each nominal value to distinct color from qualitative color map
    - Colors must be clearly differentiable
    - Colors must **not** suggest any order
    - Usually possible for up to 12 colors



  - Map nominal values to numbers (before mapping these to colors)
    - Give nominal values a quantitative meaning based on how data are distributed
    - More details in Rosario et al. (2004)

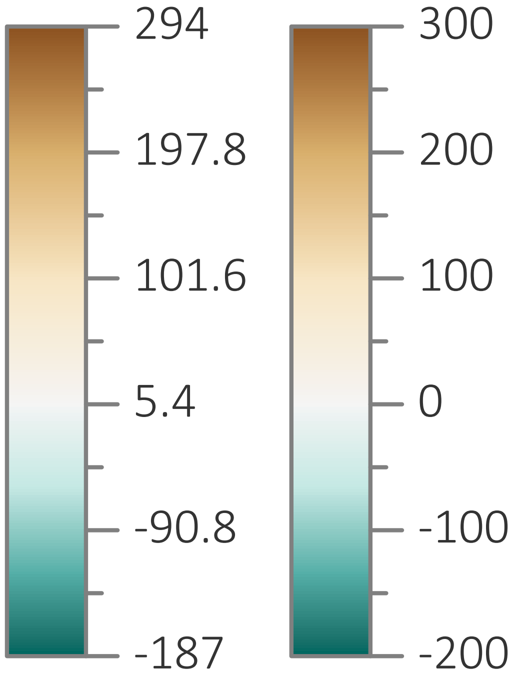
Visual encoding

How to map? – By the example of color-coding

- Some practical concerns…
  - Color legend labels can be difficult to interpret
  - Linear mapping might not go well with the data's distribution
  - Color mapping for visual comparison
- … and their solutions
  - Value range expansion
  - Logarithmic mapping
  - Box-whisker mapping
  - Merging of color maps

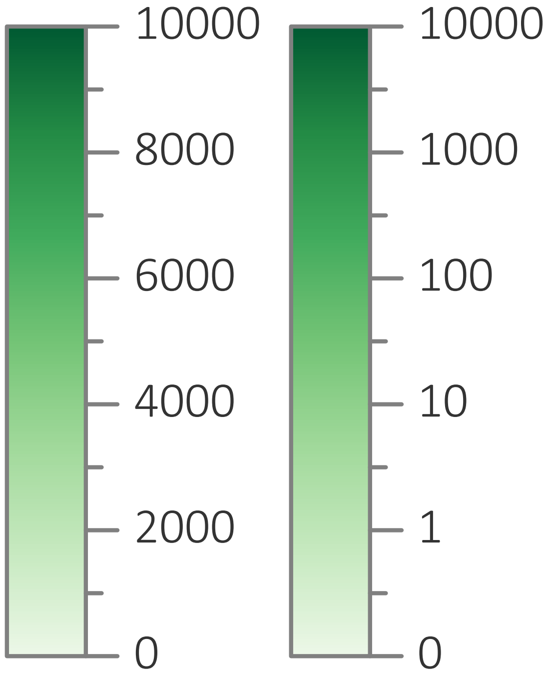
Visual encoding

How to map? – By the example of color-coding



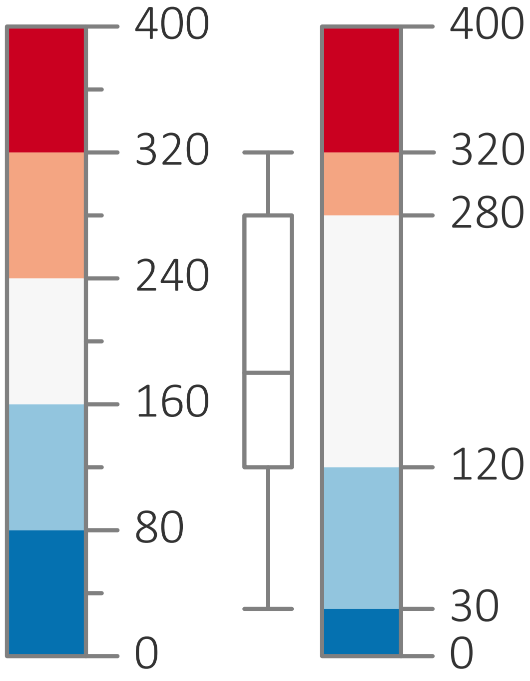
Value range expansion

Make labels multiples of 2 or powers of 10



Logarithmic mapping

$$t = \frac{\log(d_{val} - d_{min})}{\log(d_{max} - d_{min})}$$



Box-Whisker mapping

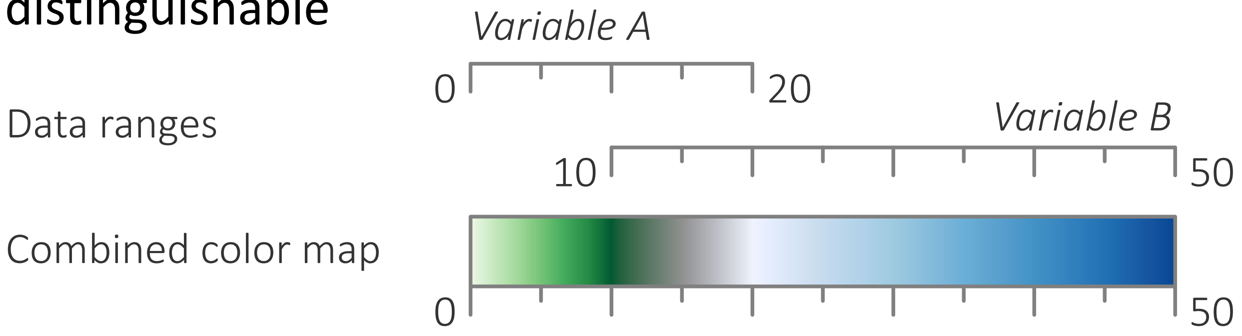
Map based on data distribution's interquartile range

Visual encoding

How to map? – By the example of color-coding

Color-coding for visual comparison tasks

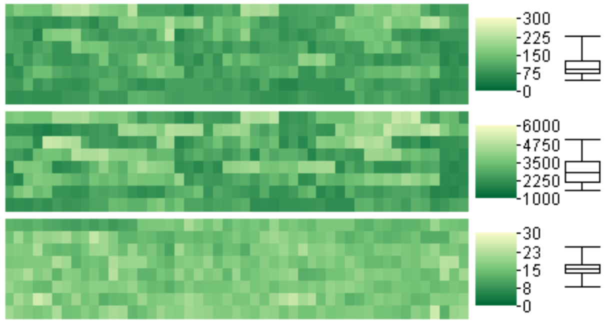
- Conflicting needs
  - Need to distinguish values of individual variables
    → Calls for separate color maps, but comparison not possible
  - Need to be able to compare values between variables
    → Calls for global color map, but local data variations might get washed out
  - Better solution: **Merging of color maps** → Comparison is possible, individual values remain distinguishable



Visual encoding

How to map? – By the example of color-coding

Color-coding for visual comparison tasks

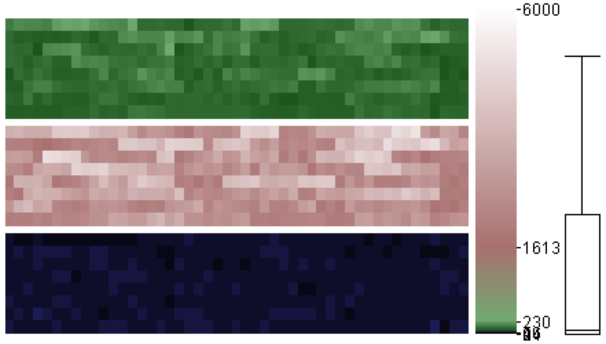


Individual color maps per variable

Values differentiable but not comparable

Global color map for all variables

Values comparable but not differentiable



Dedicated comparison color map

Values comparable and also differentiable (to a certain degree)

Visual encoding

Using multiple visual variables

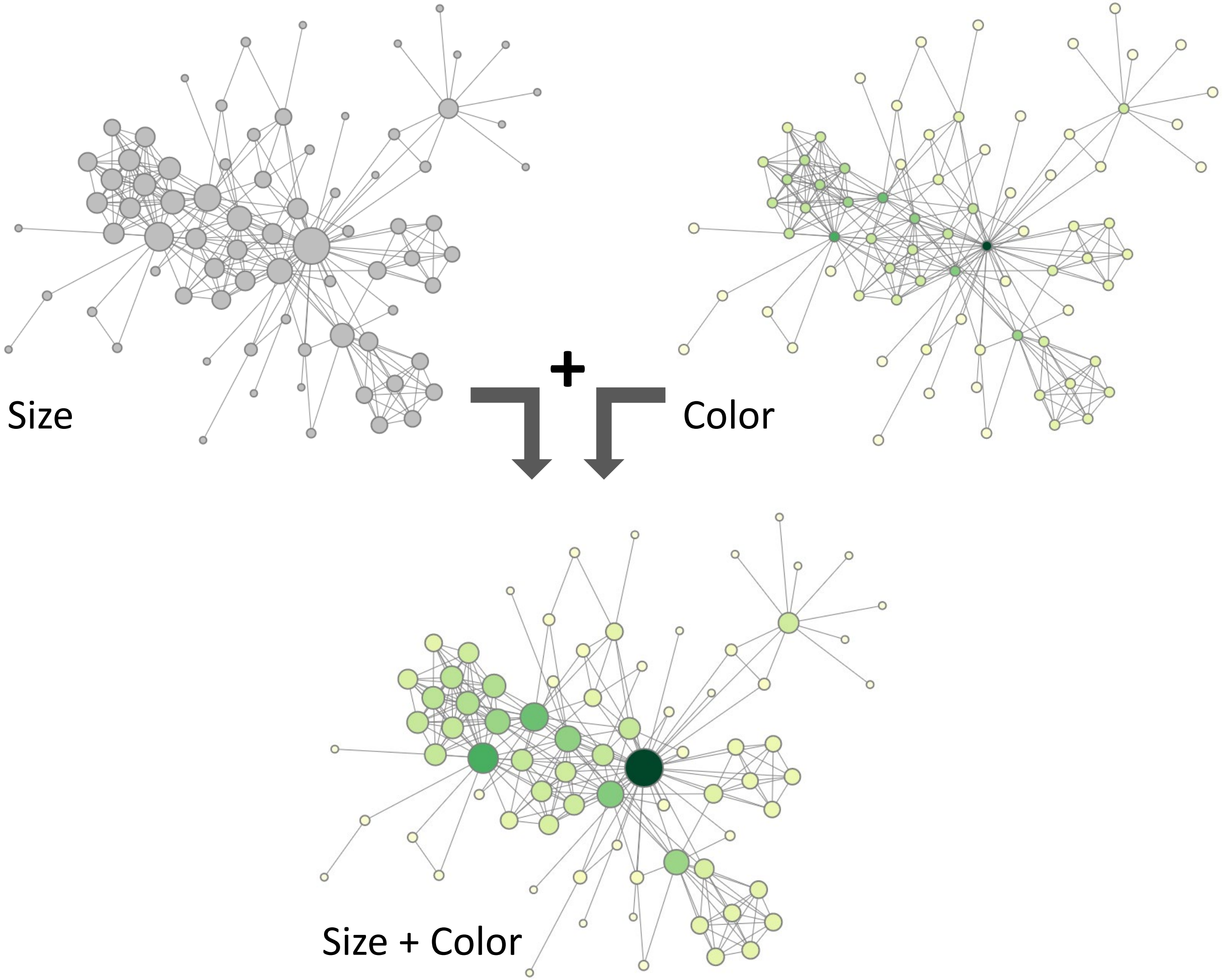
- By combining multiple visual variables, more expressive, effective, and efficient visual encoding can be obtained
- Two basic options
  - 1-to-*n* mapping: 1 data value mapped to *n* visual variables
    - Can make visual representation more effective and efficient (make data easier to see)
  - *n*-to-*n* mapping: *n* data values mapped to *n* visual variables
    - Can make visual representation more expressive (show more data)

Important: Don't overdo!

Visual encoding

1-to-*n* mapping

- Example: Node-link diagram from first lecture
- Map node degree to size *and* color



https://vcg.informatik.uni-rostock.de/~ct/software/iGraph.js/iGraph.html

Visual encoding

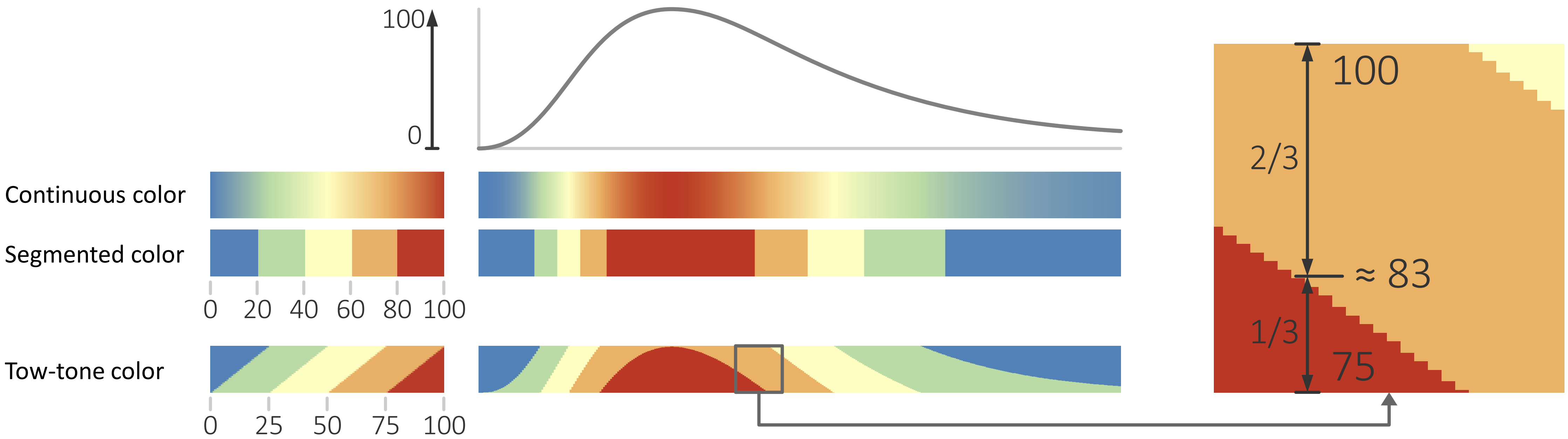
1-to-*n* mapping

- Example: **Two-tone coloring** combines color plus length (Saito et al., 2005)
  - Properties of mapping data to color and length
    - Color: Data can be estimated quickly, but are more difficult to read precisely
    - Length: Data can be read precisely, but visualization requires substantial display space
  - Basic idea: Combine advantages of color and length
    - Two colors guide the rough estimation of data values
    - The proportion of the two colors (length) allows reading the precise data value
  - Result:
    - Values can be quickly estimated and precisely read
    - Only little display space is required

Visual encoding

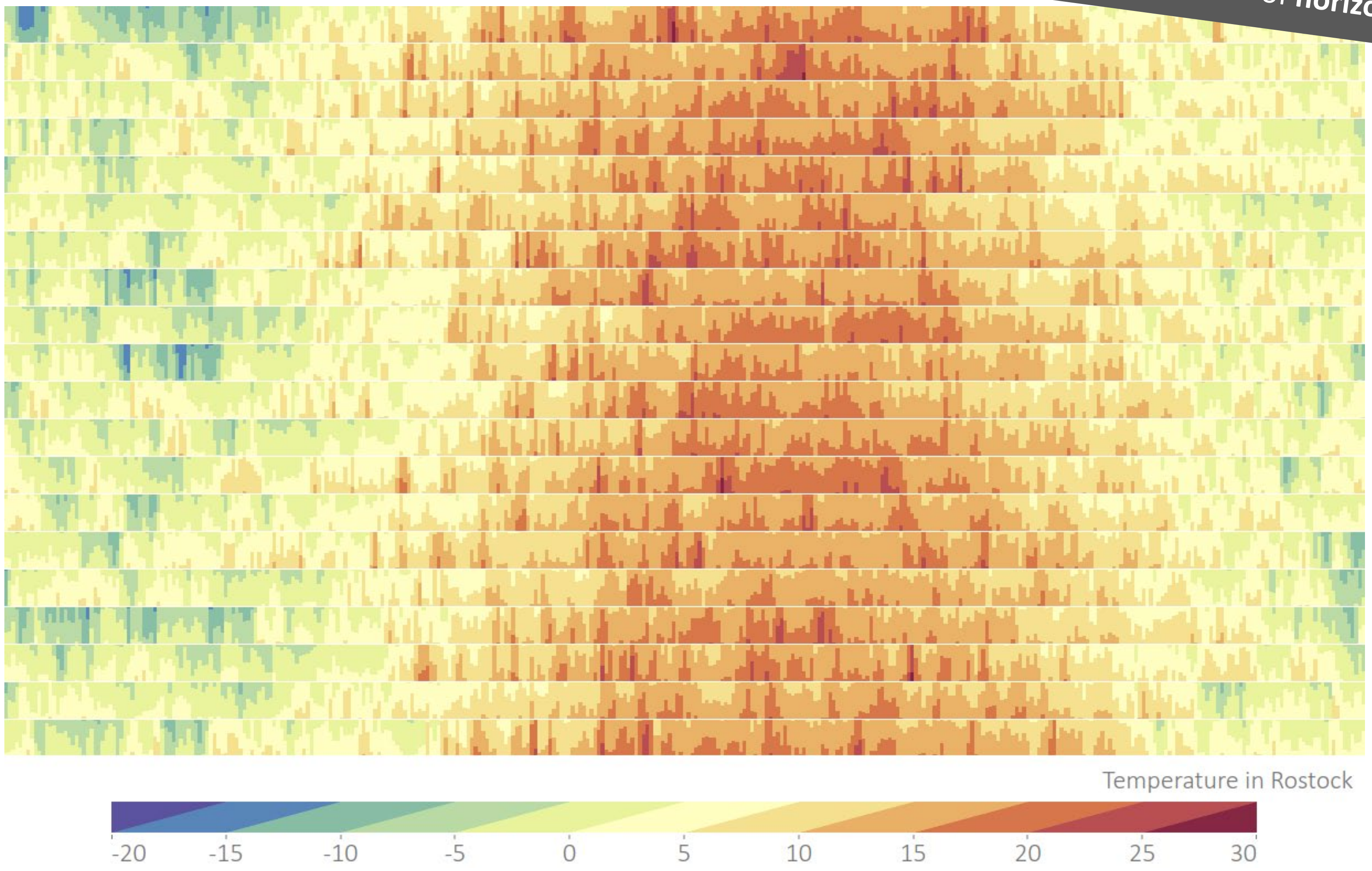
1-to-*n* mapping

- Example: **Two-tone coloring** combines color plus length (Saito et al., 2005)



Visual encoding

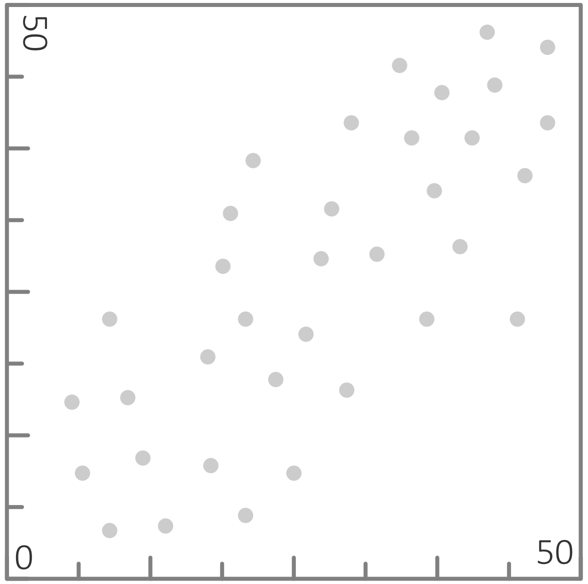
This kind of visual representation also goes by the name of **horizon graph**.



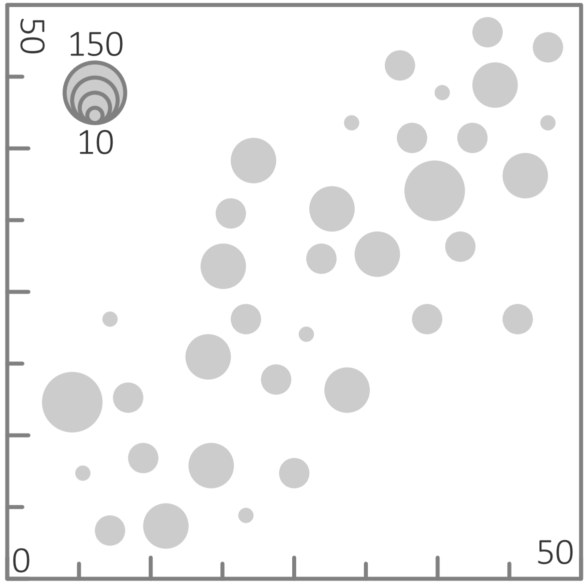
Visual encoding

***n*-to-*n* mapping**

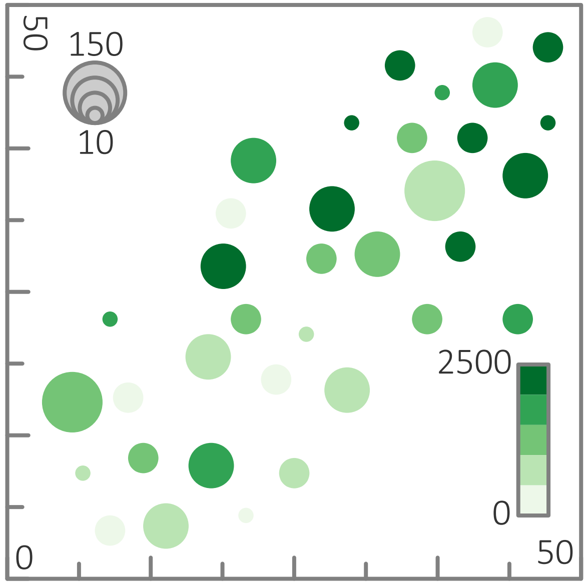
- Example: Scatter plot encoding 5 variables



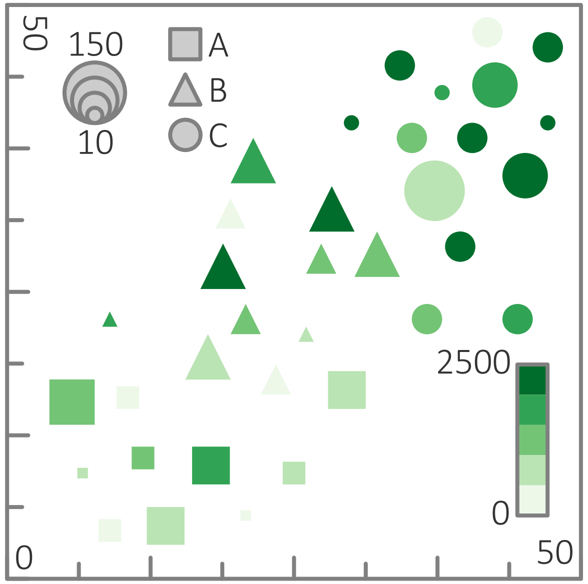
2 variables encoded in position



1 more var. encoded via size



1 more var. encoded by color



1 more var. encoded in shape

Visual encoding

Using multiple visual variables

- Important: Don't overdo!

> "In anything at all, **perfection is** finally attained not when there is no longer anything to add, but **when there is no longer anything to take away** [..]
>
> — de Saint-Exupéry, 1939

Visual encoding

Summary

- Basic ingredients: Marks and visual variables
- Two design questions:
  - **What to map?**
    → "Map most relevant data variables to most effective visual variables!"
  - **How to map?**
    → Make important information pre-attentively perceivable
    → Basic methods can be enhanced in various ways
    - Value range expansion
    - Visual mapping based on data distribution
    - Use of multiple visual variables

Visual encoding and presentation

For designing high-quality visual representations we need to know

- How to encode data visually?
  - Visual encoding
- **How to present data meaningfully?**
  - Presentation

Presentation

- Visual encoding is concerned with generating marks and assigning their visual appearance and compile them to views

- **Presentation** is concerned with presenting marks and views to users
- Three key questions are relevant:
  1. Should we present the marks in 2D or 3D?
  2. Should we present all marks or focus on subsets of interest?
  3. How can we arrange multiple views so that users can use them in concert?

Presentation

1. Presenting in 2D or 3D?

- Think about: What are advantages and disadvantages of 2D and 3D visual representations?

Presentation

1. Presenting in 2D or 3D?

- 2D is arguably more abstract and easier to explore
- Human perception naturally more adept to 3 dimensions
- 3rd dimension can serve as carrier of additional information
- 3D can be more difficult to interpret due to projection and perspective distortion
- In 3D, occlusion can be more of a problem than in 2D

There is no definite answer w.r.t. to 2D or 3D

Decision must be made carefully on case-by-case basis

Presentation

2. Presenting all data or data of interest?

- Two essential communicative goals
  1. **Convey an overall picture**
  2. **Provide details where necessary**
- Both goals become more and more difficult to achieve for larger data
  - Conveying overview usually requires neglecting details
  - Presenting details usually makes it necessary to sacrifice completeness
- Two fundamental strategies
  - **Overview+detail**
  - **Focus+context**

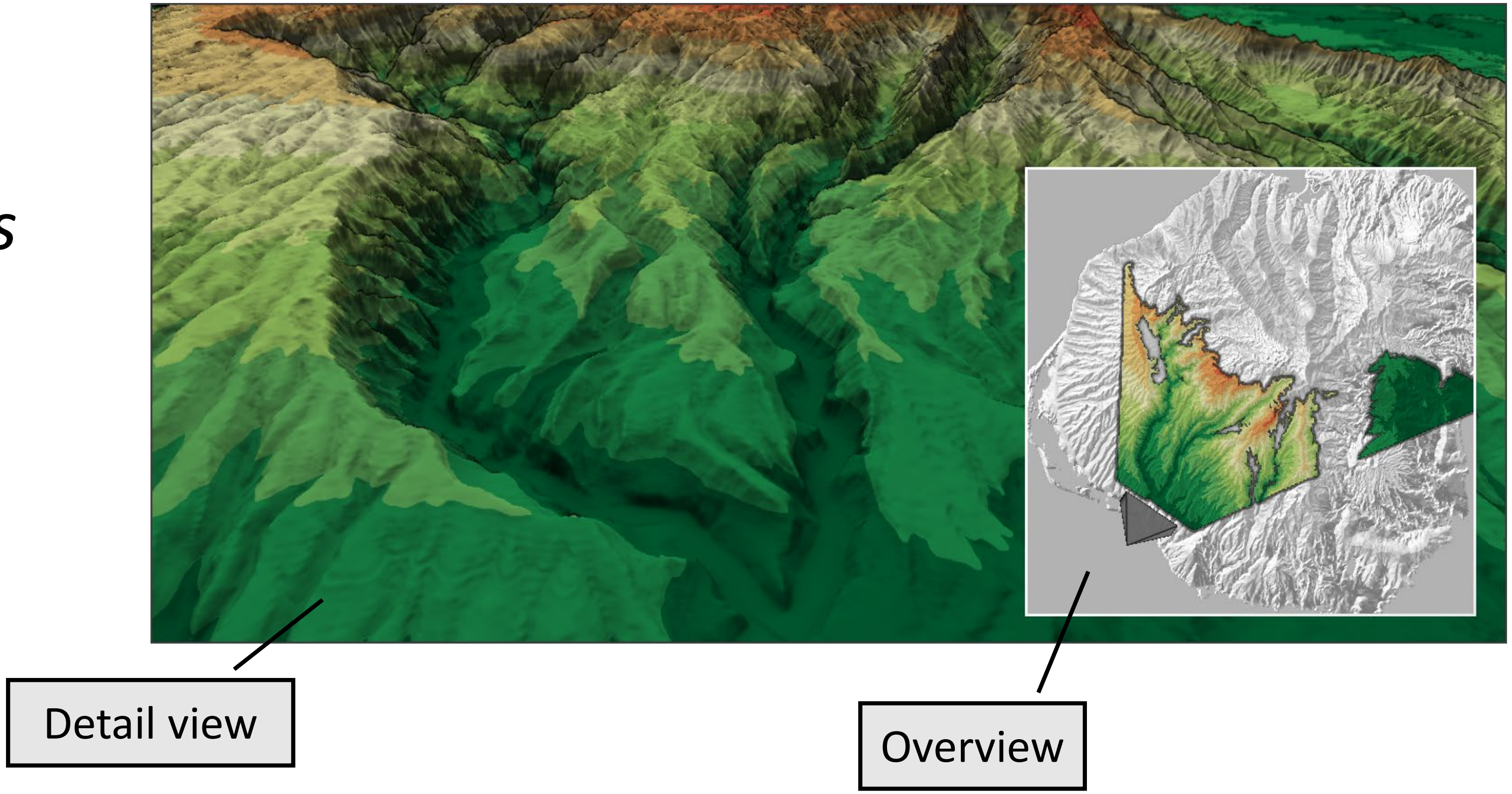
Presentation

2. Presenting all data or data of interest?

Overview + detail:

Show two *separate views*

- **Overview**
  - All data
  - Reduced detail
- **Detail view**
  - Selected data
  - Full detail



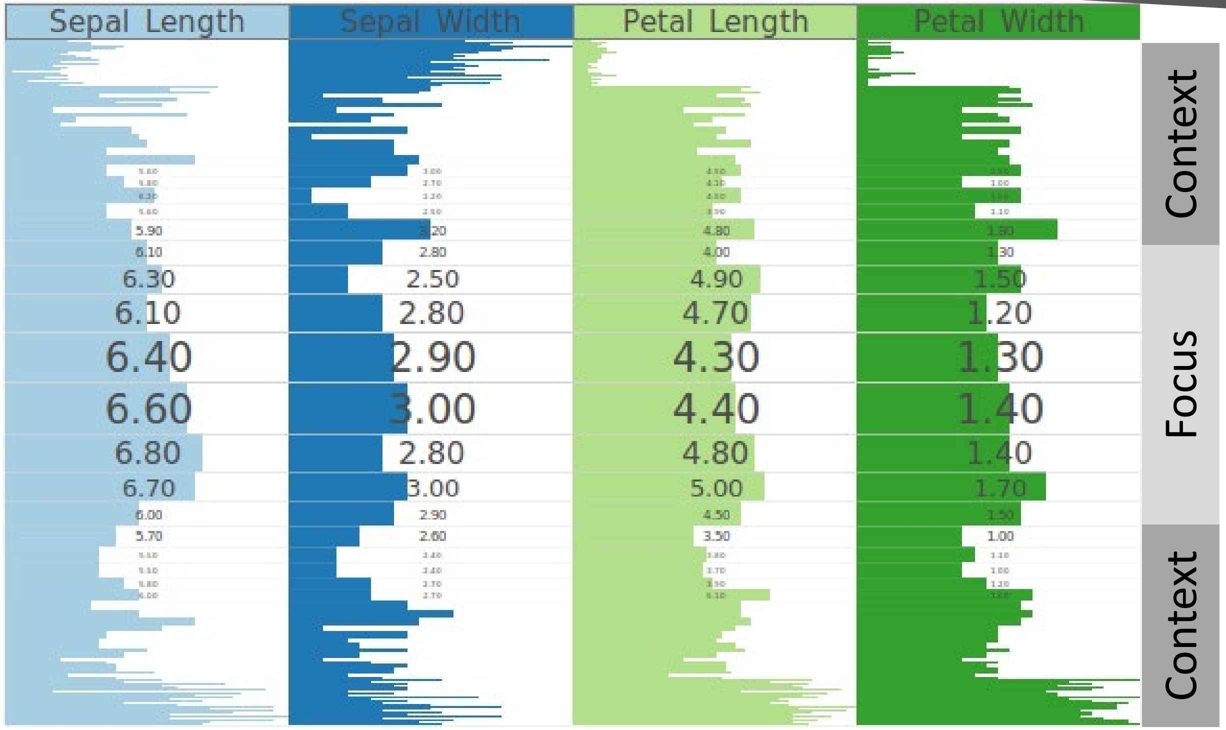
Presentation

2. Presenting all data or data of interest?

The **Table Lens** is a focus+context technique, here implemented via a **fisheye distortion** mapping.

Focus + context:
Smoothly embed in a single view

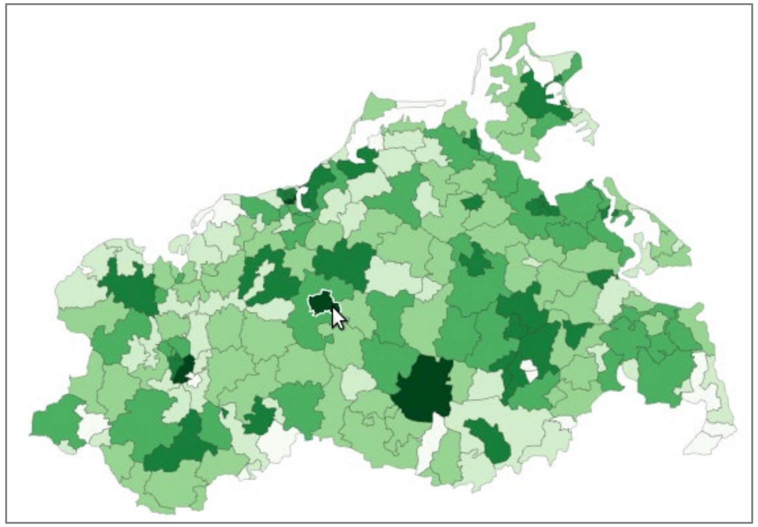
- **Focus**
  - Selected data
  - Full detail
- **Context**
  - All data
  - Reduced detail



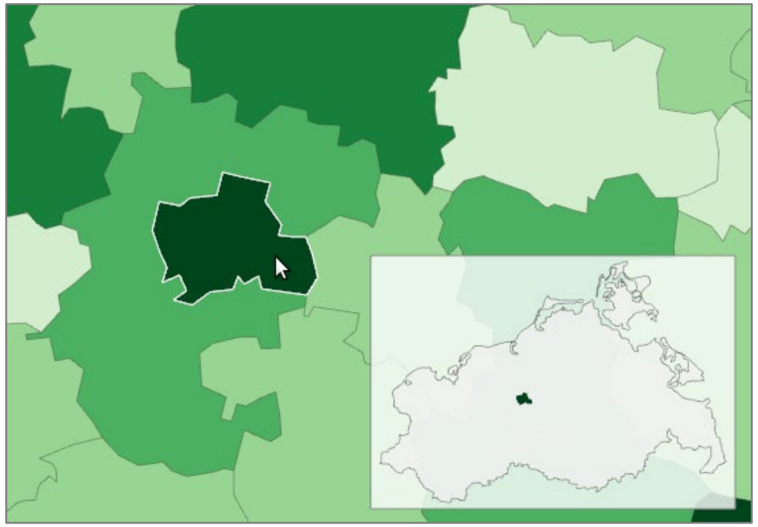
Presentation

2. Presenting all data or data of interest?

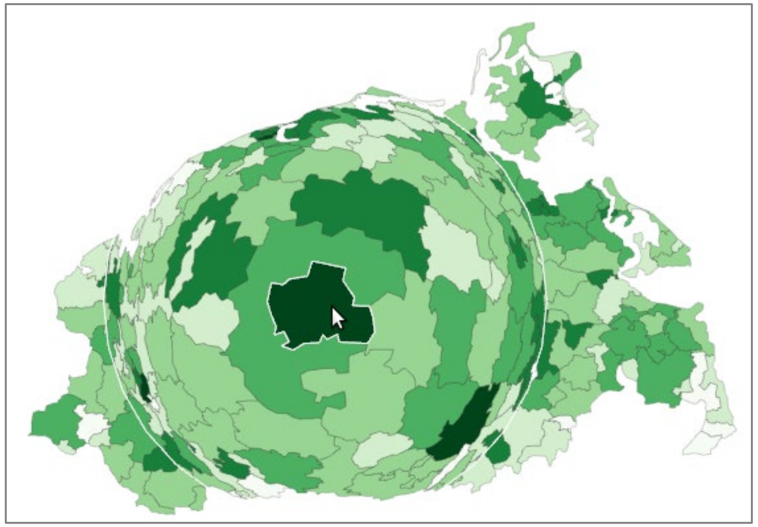
- Comparison of overview + detail vs. focus + context



Regular map



Overview + detail (large detail, small overview)



Focus + context (fisheye distortion)

Presentation

2. Presenting all data or data of interest?

Fisheye distortion ([Sarkar and Brown, 1994](#))

- Push points that are close to the focus away from it
- Parametric mapping from current distance (to focus) *d* to new distance *d'*
- Magnification factor *m* defines how far points are pushed away from focus
- Only affect points with $d < d_{max}$

$$d' = d_{max} \cdot \frac{(m+1) \cdot \frac{d}{d_{max}}}{m \cdot \frac{d}{d_{max}} + 1}$$

Presentation

3. Presenting multiple views in space or in time?

- To understand complex data, we often need multiple views, each concentrating on different aspects of the data
- Such multiple views can be presented
  - **In space:** Side by side
  - **In time:** One after the other

Presentation

3. Presenting multiple views in space or in time?

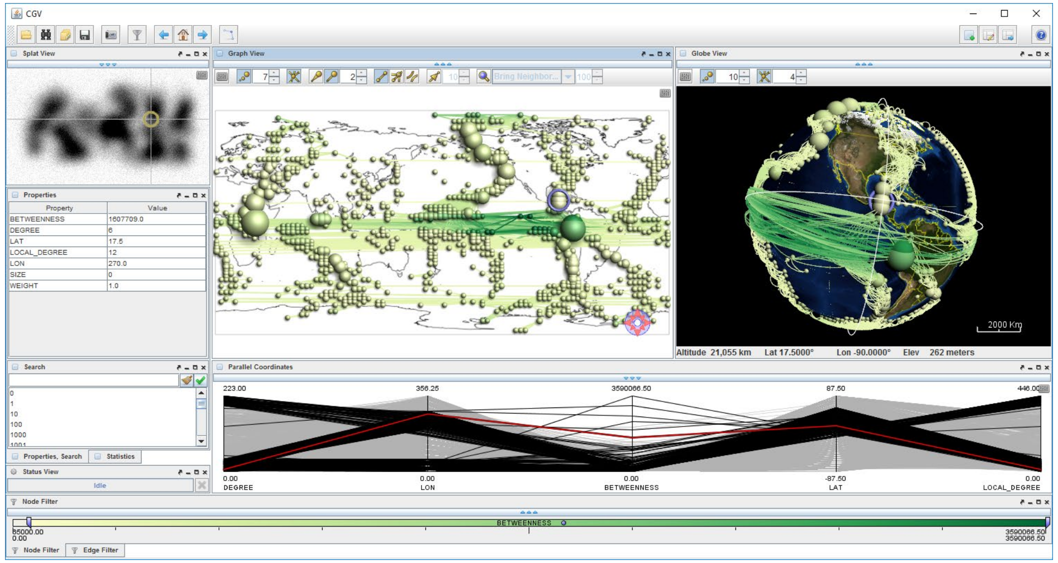
Arranging views in space

- **Fixed:** Fixed arrangement designed by expert for efficiency
  - No effort required on user side
  - No flexibility (cannot adjust view size or adapt to device)
- **Free:** Users arrange views freely
  - View arrangement can be costly to set up
  - Full flexibility
- **Constrained:** Users refine pre-defined arrangement within reasonable bounds
  - Compromise between costs and flexibility
  - Useful constraints: Overlap-free partition of view space, maintain aspect ratio of views

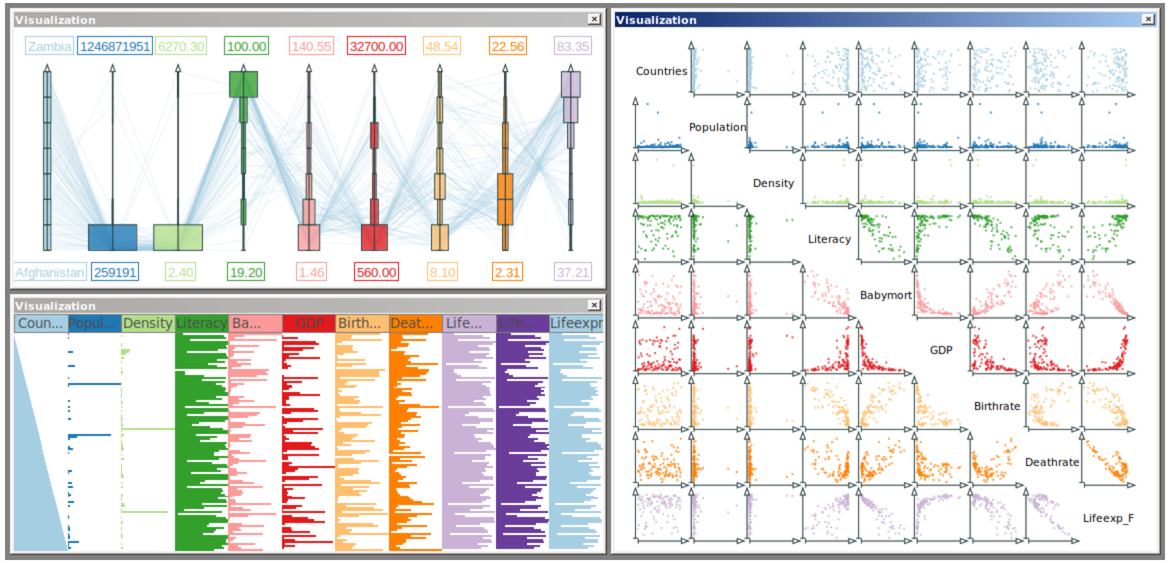
Presentation

3. Presenting multiple views in space or in time?

Arranging views in space



Graph visualization with multiple coordinated views



Multi-view visualization of multivariate data

Presentation

3. Presenting multiple views in space or in time?

Arranging views in space

- Think about: How should a visual representation respond to size changes?

- Answers can be found in recent research on **“Responsive Visualization”** (Hoffswell et al., 2020)

Presentation

3. Presenting multiple views in space or in time?

Presenting multiple views in time

- Sequencing views in time
  - **Slide show:** slow progression of views
  - **Animation:** many views per time unit (see assignments for learning more!)
- Pros & cons
  - Spatial arrangement allows for detailed comparison
  - Temporal sequence is suited to represent development over time
  - Animations can be difficult to follow (need careful design and interactive control)

Summary

For designing high-quality visual representations we need to know

- **How to encode data visually?**
- **How to present data meaningfully?**

We now know **how data can be encoded and presented visually**. Next, we **look at concrete visualization techniques** for different data classes.

Assignments

1. Learn more about **pre-attentive processing** at https://www.csc2.ncsu.edu/faculty/healey/PP/!
2. Go to https://colorbrewer2.org and learn about **sequential**, **diverging**, and **qualitative** color scales!
3. Learn more about fisheye distortion at https://observablehq.com/@benmaier/a-visually-more-appealing-fisheye-function!
4. Read "Animation for Visualization: Opportunities and Drawbacks" by Danyel Fisher!

Questions

1. What does the representation effect tell us?
2. What are marks and visual variables? Give examples!
3. How would you select visual variables for the visual encoding?
4. Characterize color maps for identification and location tasks!
5. What is pre-attentive processing?
6. What types of color maps are suited for what types of data?
7. How can ColorBrewer help us find suitable color maps?
8. What are the advantages of using multiple visual variables?
9. What can be benefits and problems when presenting data in 3D?
10. What is the difference between overview+detail and focus+context?
11. How can views be arranged in space and be sequenced in time?