

# Informatics 143

## Information Visualization

### Lecture 2

*Duplication of course material for any commercial purpose without  
the explicit written permission of the professor is prohibited.*

*These course materials are based on books from Claus O. Wilke, Kieran Healy, Edward R. Tufte,  
Alberto Cairo, Colin Ware, Tamara Munzner, and others.  
Powerpoint theme by Prof. André van der Hoek.*

# Our definition of information visualization

---



*The activity of transforming abstract data into visual representations to improve human cognition.*

**In this course :**

- Abstract data will be numerical and non-numerical
- Visual representations will be *mostly* static

# Why visualization

---

- Tell a story about the data
- Discover a story from the data
  - “The purpose of computing is insight, not numbers” Richard Hamming

# Why visualization

---



- Amplifies cognition
  - Expand working memory
  - Reduce searching time
  - Pattern detection
  - Inference

# Challenges

---



- What are the principles for designing effective visualizations?
- What are the tools to produce visualization?
- How to deal with large data?
- But to begin with:

*Is there a common language?*

# Mapping data into aesthetics

---



- Visualization:
  - Input: data values
  - Output: visual elements

# Mapping data into aesthetics

---

- Visualization:
  - Input: data values
  - Output: visual elements



# Mapping data into aesthetics

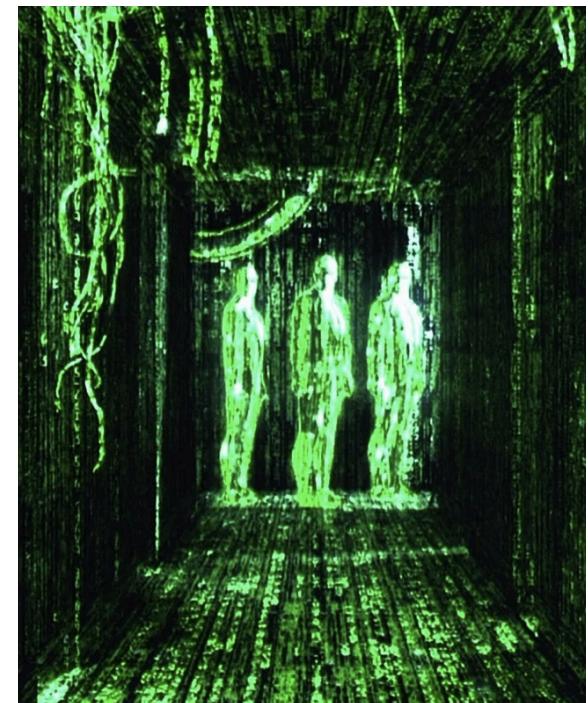
- Visualization:
  - Input: data values
  - Output: visual elements



+



=



# Mapping data into aesthetics

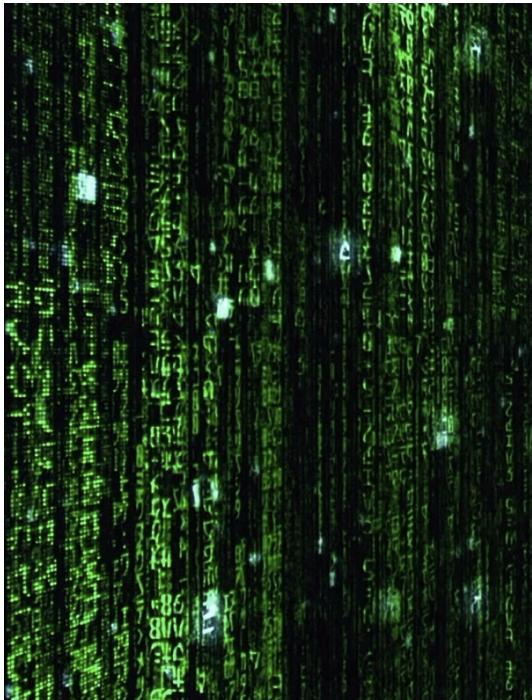
---

- Visualization:
  - Input: data values
  - Output: visual elements



# Mapping data into aesthetics

- Visualization:
  - Input: data values
  - Output: visual elements



Magic process  
3

2	5	5	8	4	7	0	4	0
5	6	8	1	6	8	0	0	4
1	0	4	6	7	9	9	4	5
0	6	4	4	6	4	0	4	0
6	2	7	9	7	5	2	7	2
2	3	1	4	4	4	7	7	5
4	0	3	3	9	0	5	9	7
0	SYSTEM FAILURE							
3	2	3	9	8	0	3	6	0
5	1	9	8	7	8	2	4	4
8	7	0	0	5	2	4	7	9
7	5	9	7	8	5	6	5	3
4	4	2	0	7	6	4	8	8
6	0	4	0	3	3	3	3	3
8	9	3	3	7	1	3	4	9

# Mapping data into aesthetics

---



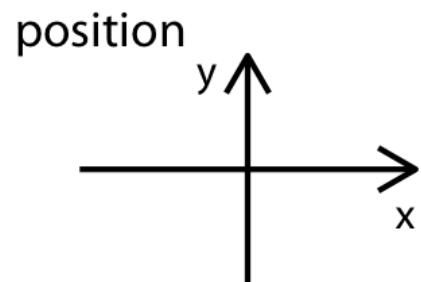
- Visualization:
  - Input: data values
  - Output: visual elements
- Visualization can be seen as a *map* between *quantifiable features to aesthetics*

# Aesthetics

---



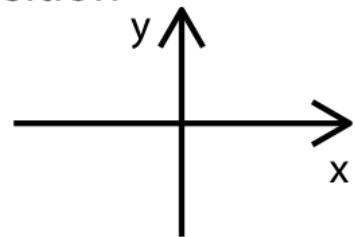
- Describe aspects of graphical elements



# Aesthetics

- Describe aspects of graphical elements

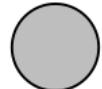
position



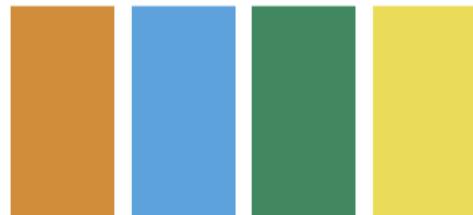
shape



size



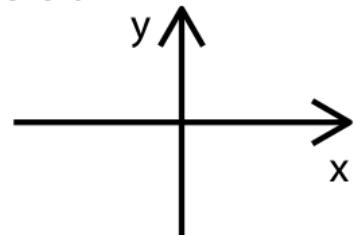
color



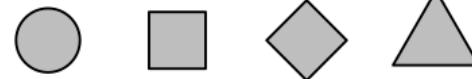
# Aesthetics

- Describe aspects of graphical elements

position



shape



size



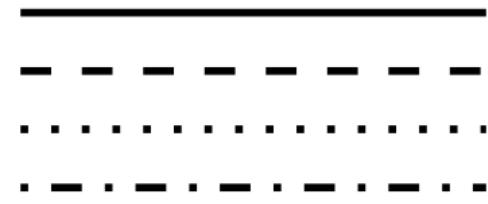
color



line width



line type

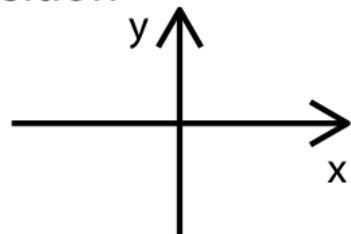


# Aesthetics

---

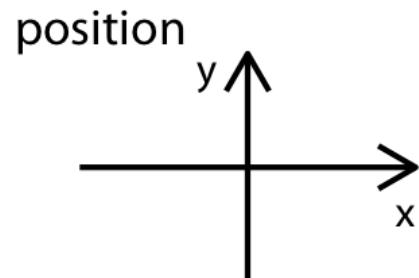
- Describe aspects of graphical elements

Always required  
position

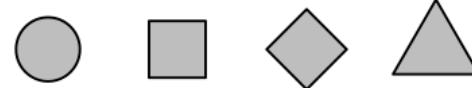


# Aesthetics

- Describe aspects of graphical elements



Better for discrete  
shape



line type

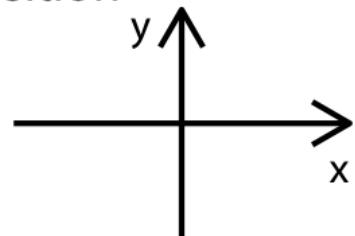


Better for discrete

# Aesthetics

- Describe aspects of graphical elements

position



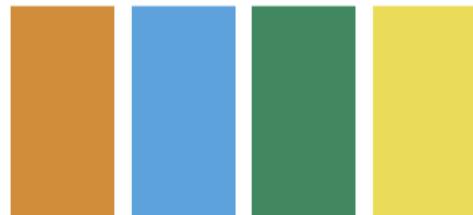
shape



size



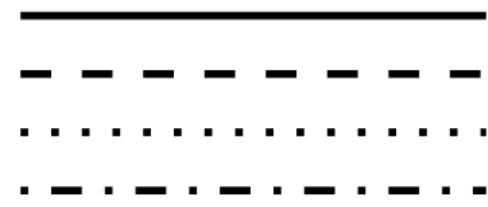
color



line width



line type





# Data types

---

- **Quantitative**
  - Numerical data
  - Continuous or discrete
  - Usually can be mapped into reals
  - Always sortable
  - Not necessarily finite/bounded

# Data types

---



- **Quantitative**
  - Numerical data
  - Continuous or discrete
  - Usually can be mapped into reals
  - Always sortable
  - Not necessarily finite/bounded
  
- **Qualitative**
  - Categorical data
  - Always discrete
  - Usually can be mapped into integers
  - Can be sortable or not
  - Usually a finite

# Data types

- **Quantitative**

- Numerical data
- Continuous or discrete
- Usually can be mapped into reals
- Always sortable
- Not necessarily finite/bounded

quantitative/numerical  
continuous      1.3, 5.7, 83,  
                                 $1.5 \times 10^{-2}$

quantitative/numerical  
discrete      1, 2, 3, 4

- **Qualitative**

- Categorical, data
- Always discrete
- *Usually* can be mapped into integers
- Can be sortable or not
- Usually a finite

qualitative/categorical  
unordered      dog, cat, fish

qualitative/categorical  
ordered      good, fair, poor

# Data types

Month	Day	Location	Station ID	Temperature
Jan	1	Chicago	USW00014819	25.6
Jan	1	San Diego	USW00093107	55.2
Jan	1	Houston	USW00012918	53.9
Jan	1	Death Valley	USC00042319	51.0
Jan	2	Chicago	USW00014819	25.5
Jan	2	San Diego	USW00093107	55.3
Jan	2	Houston	USW00012918	53.8
Jan	2	Death Valley	USC00042319	51.2
Jan	3	Chicago	USW00014819	25.3
Jan	3	San Diego	USW00093107	55.3
Jan	3	Death Valley	USC00042319	51.3
Jan	3	Houston	USW00012918	53.8



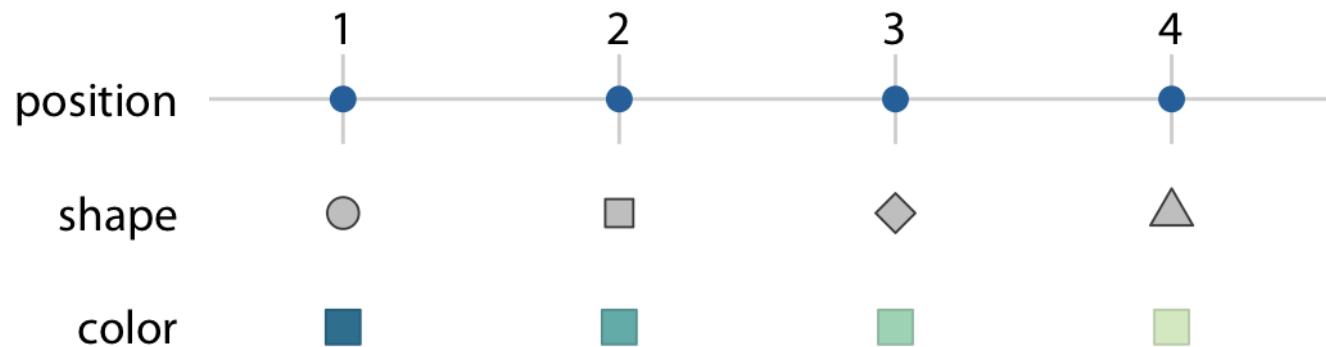
# Scales

---

- Map data values onto aesthetics
  - For each data value → some aesthetics property

# Scales

- Map data values onto aesthetics
  - For each data value → some aesthetics property
  - **Important: uniqueness.**
  - Example:
    - Pairs of real numbers → linear coordinates
    - Real numbers → point sizes, different colors
    - Classes of animals → shapes of figures



# Scales

Month	Day	Location	Station ID	Temperature
Jan	1	Chicago	USW00014819	25.6
Jan	1	San Diego	USW00093107	55.2
Jan	1	Houston	USW00012918	53.9
Jan	1	Death Valley	USC00042319	51.0
Jan	2	Chicago	USW00014819	25.5
Jan	2	San Diego	USW00093107	55.3
Jan	2	Houston	USW00012918	53.8
Jan	2	Death Valley	USC00042319	51.2
Jan	3	Chicago	USW00014819	25.3
Jan	3	San Diego	USW00093107	55.3
Jan	3	Death Valley	USC00042319	51.3
Jan	3	Houston	USW00012918	53.8

# Scales

Month	Day	Location	Station ID	Temperature
Jan	1	Chicago	USW00014819	25.6
Jan	1	San Diego	USW00093107	55.2
Jan	1	Houston	USW00012918	53.9
Jan	1	Death Valley	USC00042319	51.0
Jan	2	Chicago	USW00014819	25.5
Jan	2	San Diego	USW00093107	55.3
Jan	2	Houston	USW00012918	53.8
Jan	2	Death Valley	USC00042319	51.2
Jan	3	Chicago	USW00014819	25.3
Jan	3	San Diego	USW00093107	55.3
Jan	3	Death Valley	USC00042319	51.3
Jan	3	Houston	USW00012918	53.8

+

**Selected mapping**

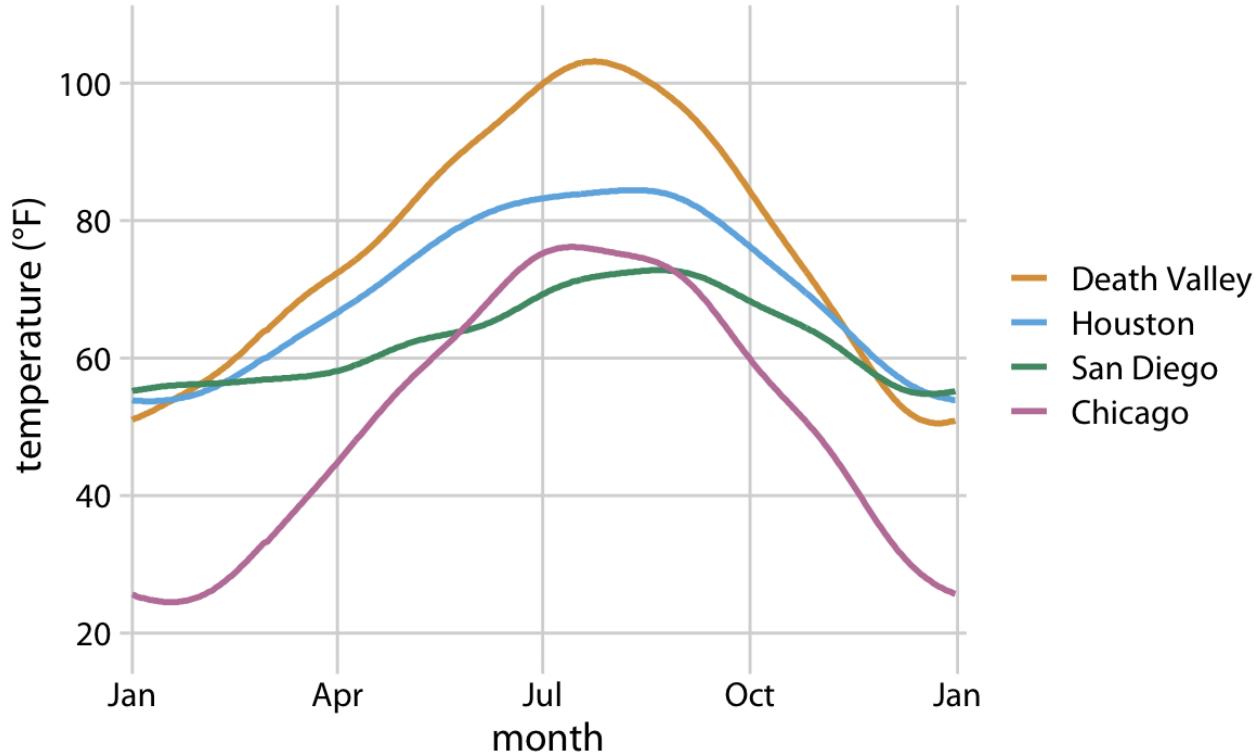
**Temperature : y axis**

**Day of the year: x axis**

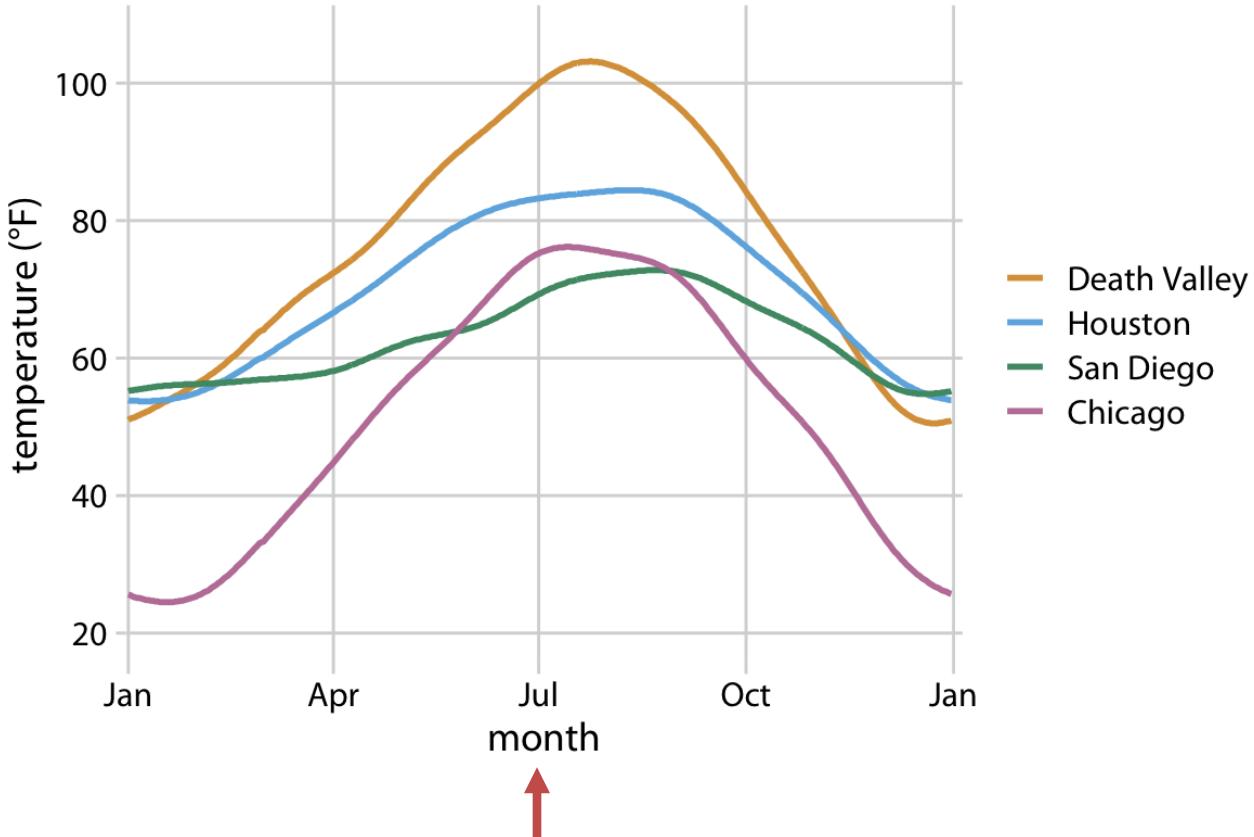
**Location : color**

# Scales

---



# Scales



Note: This required a transformation between  
Month+Day into a continuous variable

# Scales

Month	Day	Location	Station ID	Temperature
Jan	1	Chicago	USW00014819	25.6
Jan	1	San Diego	USW00093107	55.2
Jan	1	Houston	USW00012918	53.9
Jan	1	Death Valley	USC00042319	51.0
Jan	2	Chicago	USW00014819	25.5
Jan	2	San Diego	USW00093107	55.3
Jan	2	Houston	USW00012918	53.8
Jan	2	Death Valley	USC00042319	51.2
Jan	3	Chicago	USW00014819	25.3
Jan	3	San Diego	USW00093107	55.3
Jan	3	Death Valley	USC00042319	51.3
Jan	3	Houston	USW00012918	53.8

+

## Selected mapping

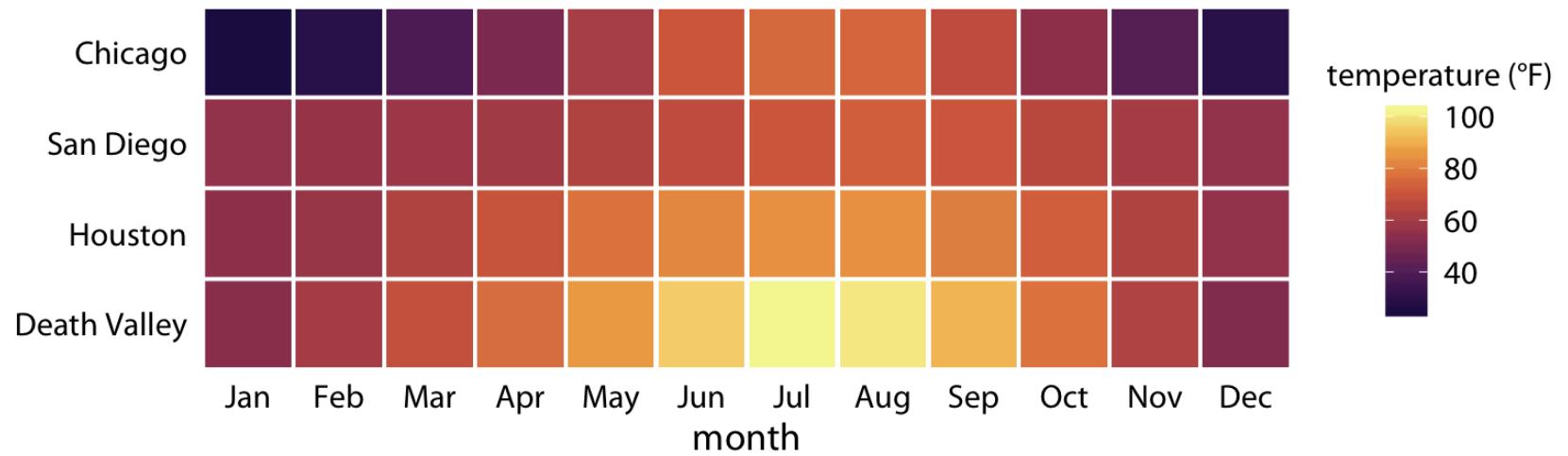
**Temperature** : color (before, y axis)

**Day of the year**: x axis

**Location** : y axis (before, color)

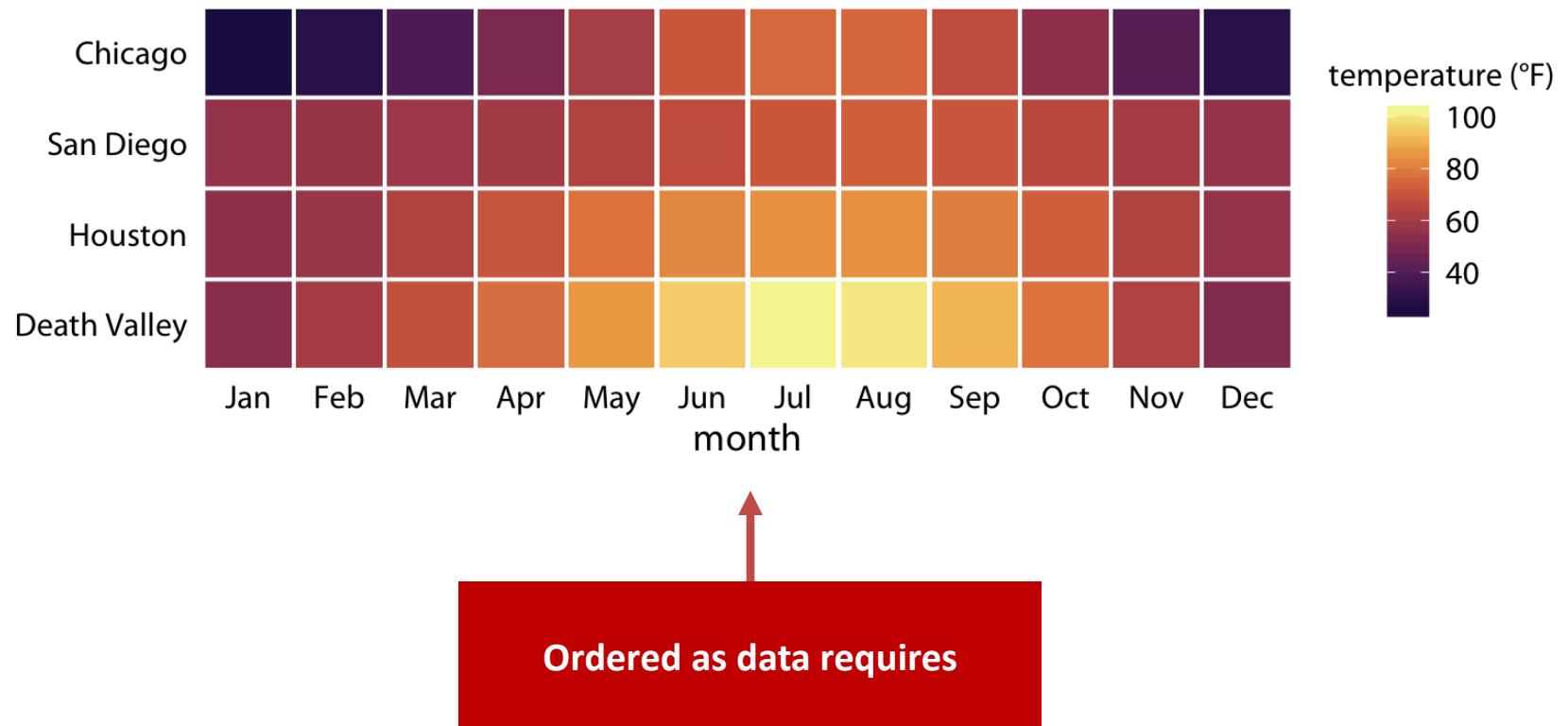
# Scales

---



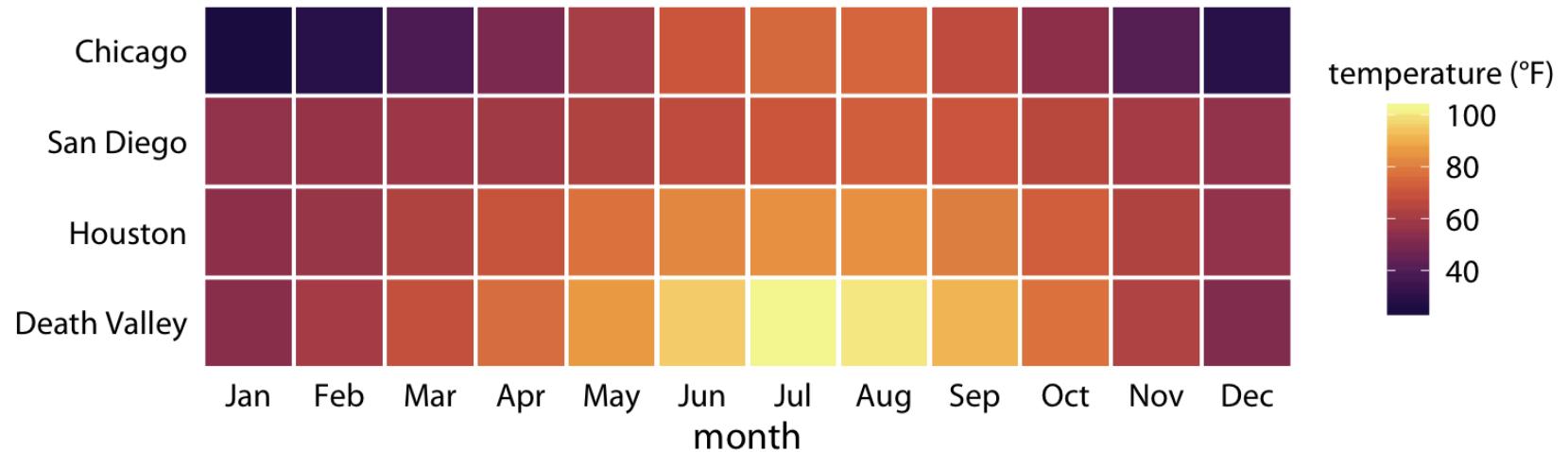
# Scales

---



# Scales

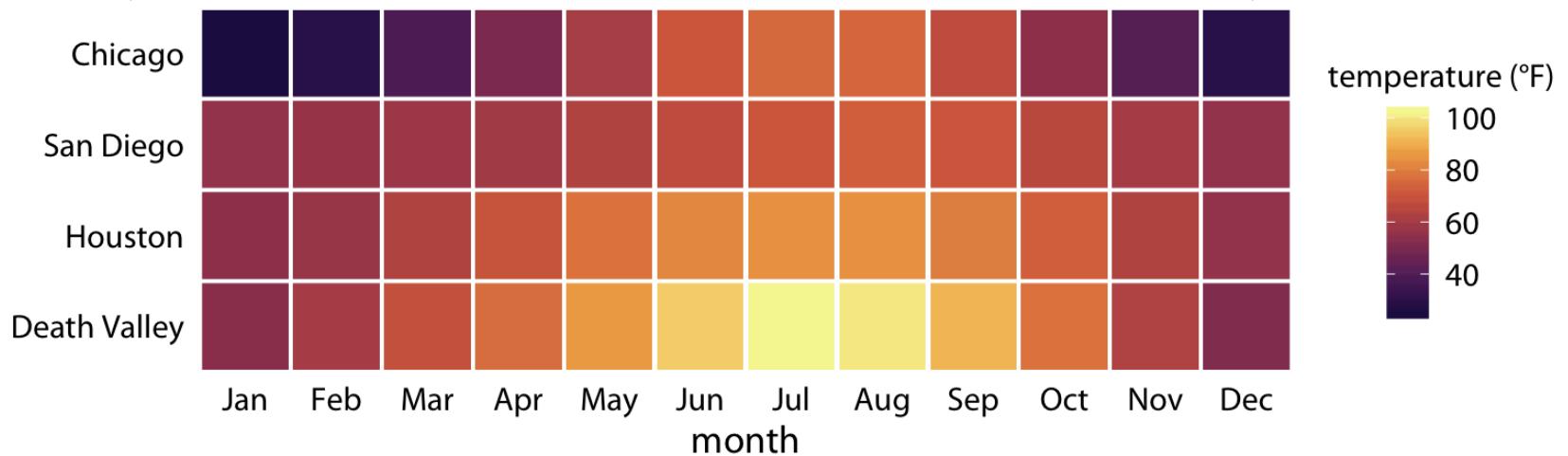
Arbitrary ordering (in this case, mean temperature)



Ordered as data requires

# Scales

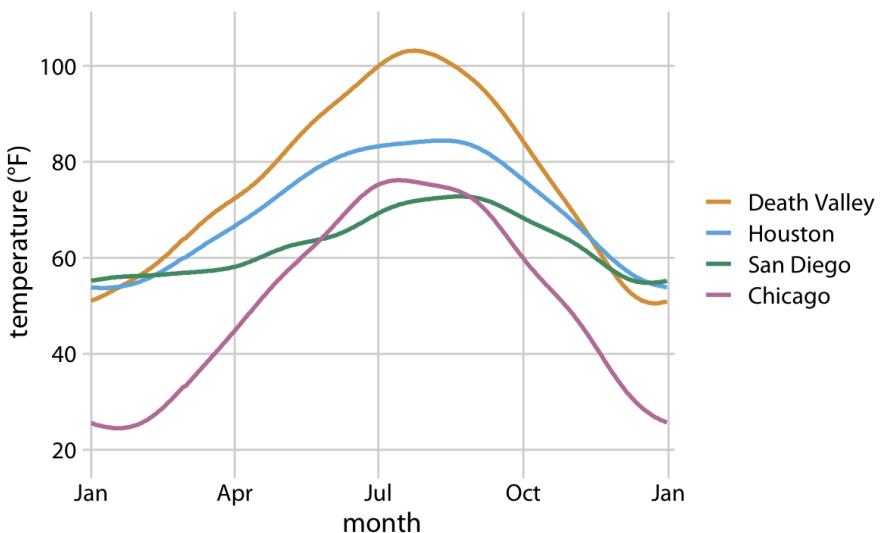
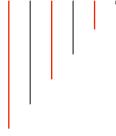
Arbitrary ordering (in this case, mean temperature)



The color of each cell encodes the mean of several data points of temperature

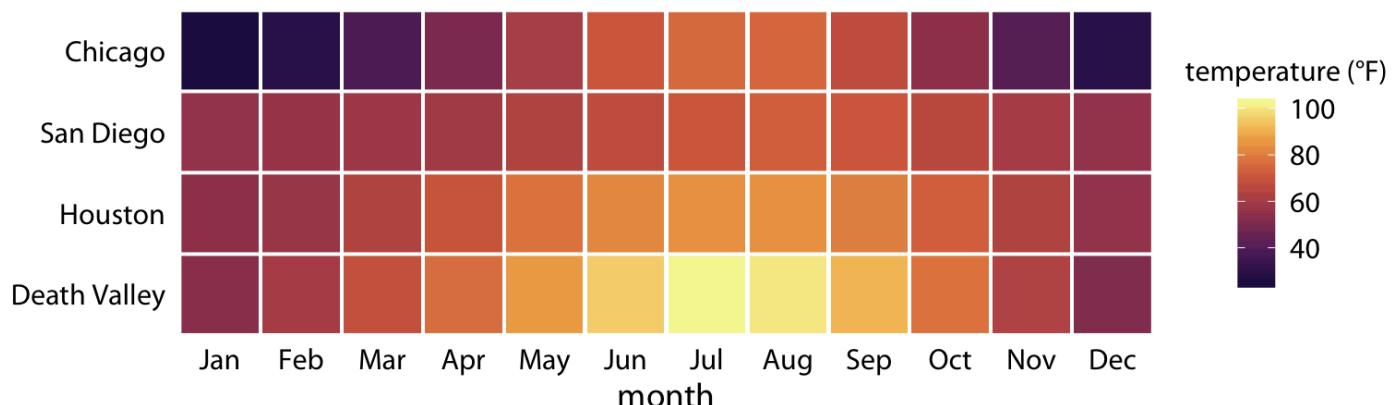
Ordered as data requires

# Scales



?

or





The choice of the map depends on:

The **story** you want to **tell** with the visualization

The **patterns** that you want to **discover** in the data

# Scales

---



- **Important: Keep scales to the minimum required**
  - It is easier for human cognition: it simplifies understanding!

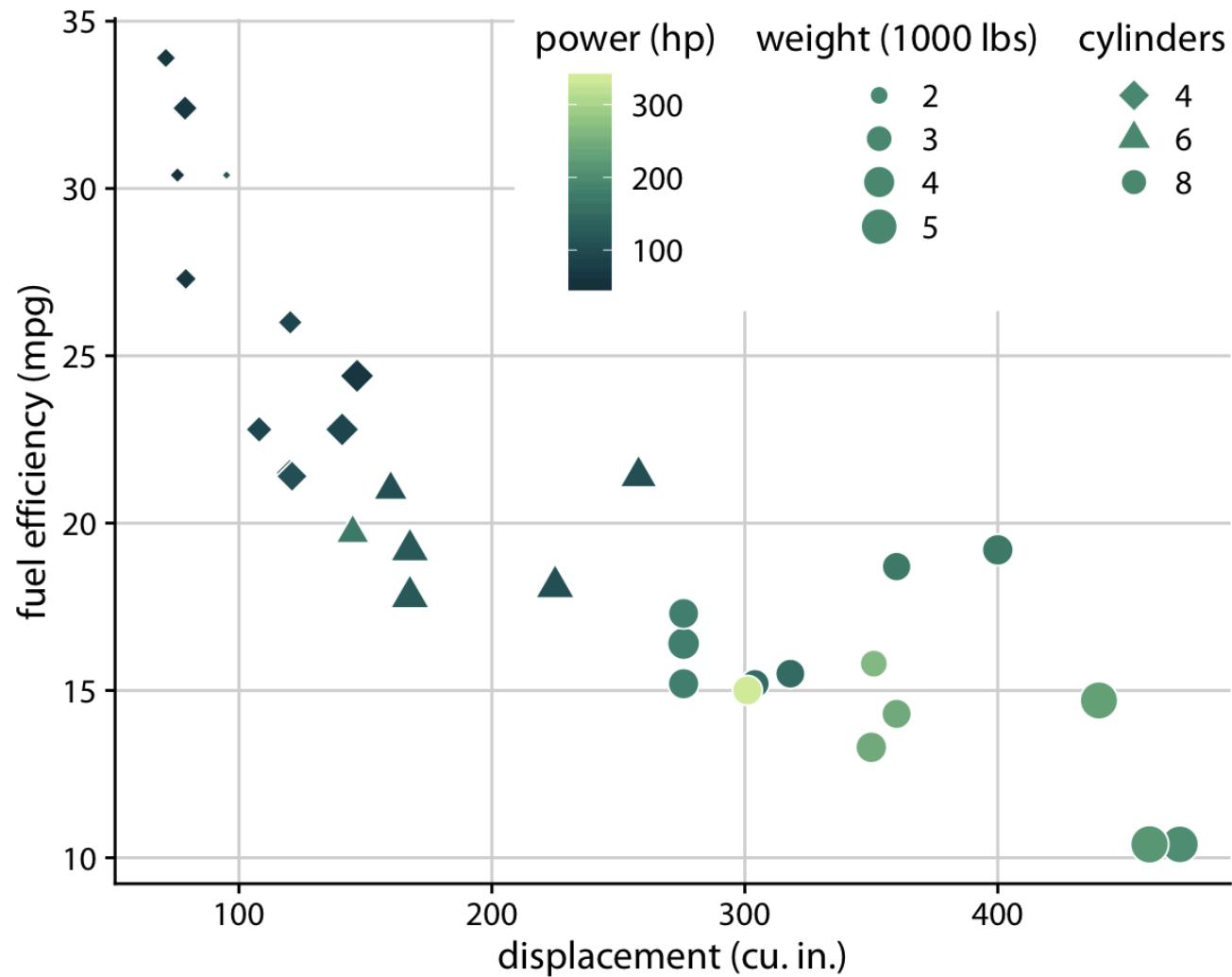
# Scales

---



- **Important: Keep scales to the minimum required**
  - It is easier for human cognition: it simplifies understanding!
- *Typically:*
  - Two positions
  - One color
- But many other possibilities! E.g.
  - Two positions
  - One color
  - One size
  - One shape

# Scales



# Coordinate systems and axes

---



- First step in the construction of a visual representation

# Coordinate systems and axes

---



- First step in the construction of a visual representation
- **Where the different data values are located?**
- **How they are related to each other?**

# Coordinate systems and axes

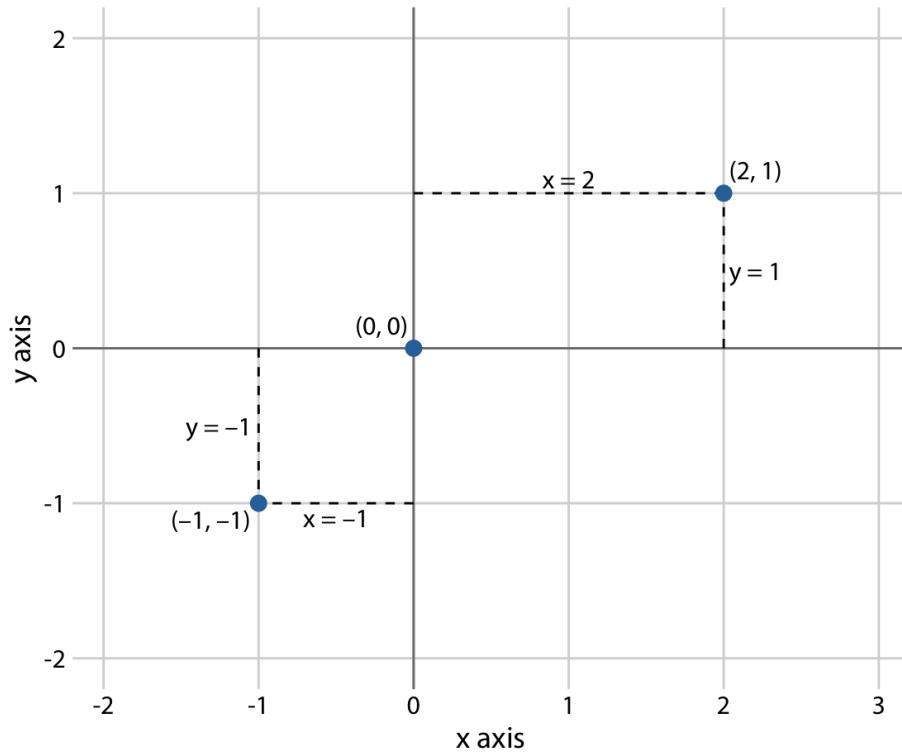
---

- First step in the construction of a visual representation
- **Where the different data values are located?**
- **How they are related to each other?**
  - Horizontally? Vertically? At an angle? Along a circle? On the surface of a sphere? Like a spiral?
  - Linearly? Logarithmically? Discretized? Continuous?

The choice of the position scales and how they are arranged geometrically is what sets a coordinate system in the context of visualization.

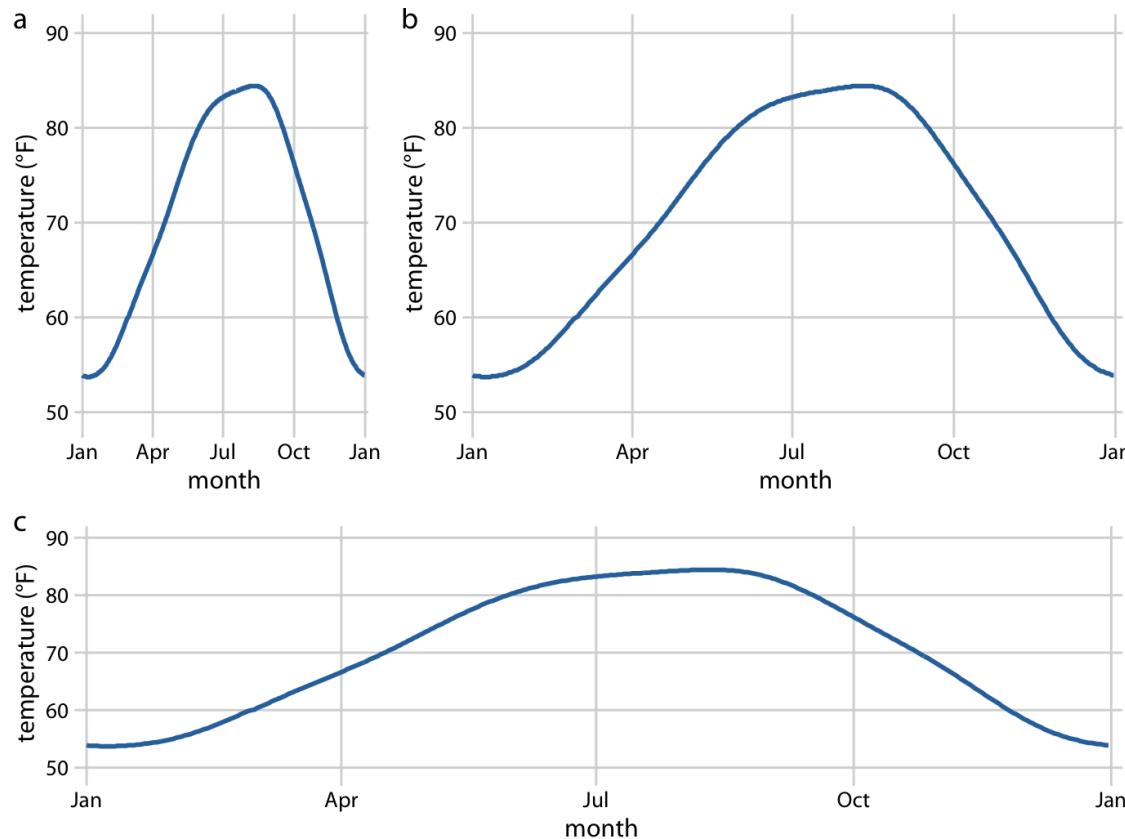
# Coordinate systems : Cartesian

- Most used system in visualization
  - *Intuitive for most people*
  - In 2D, a pair  $(p_1, p_2)$  of Reals maps to a position  $(x, y)$  in a space where x and y axis are orthogonal
  - Even spacing
  - Continuous axes



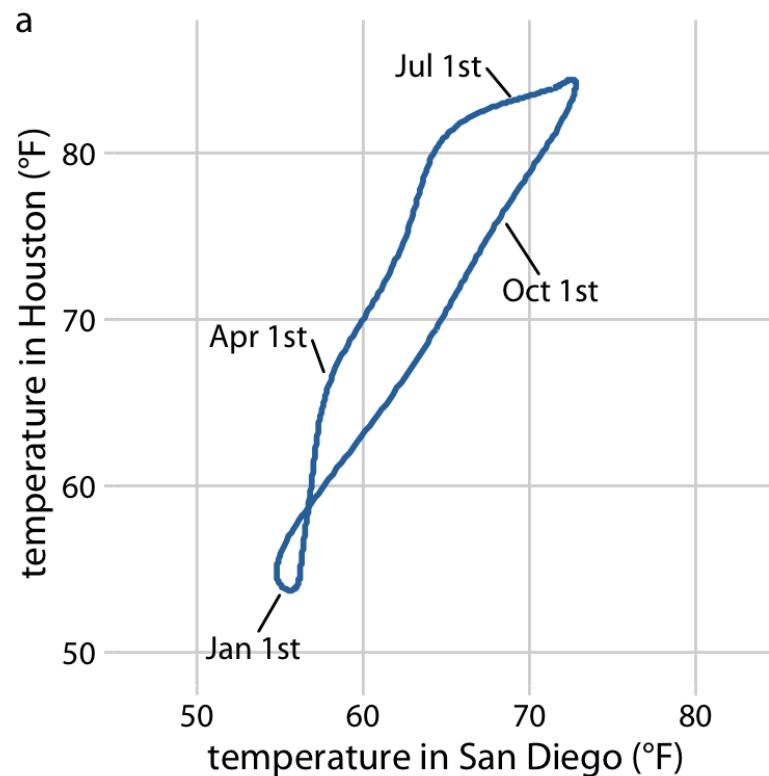
# Coordinate systems : Cartesian

- Note: axes can represent different units!
  - *In this case, aspect ratio is arbitrary ( but be carefull with choices! )*



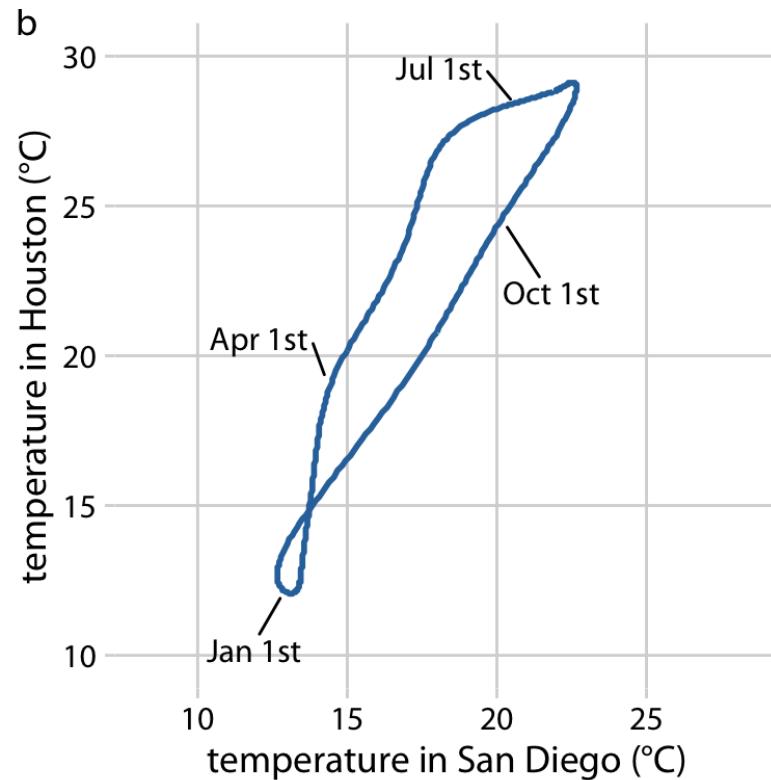
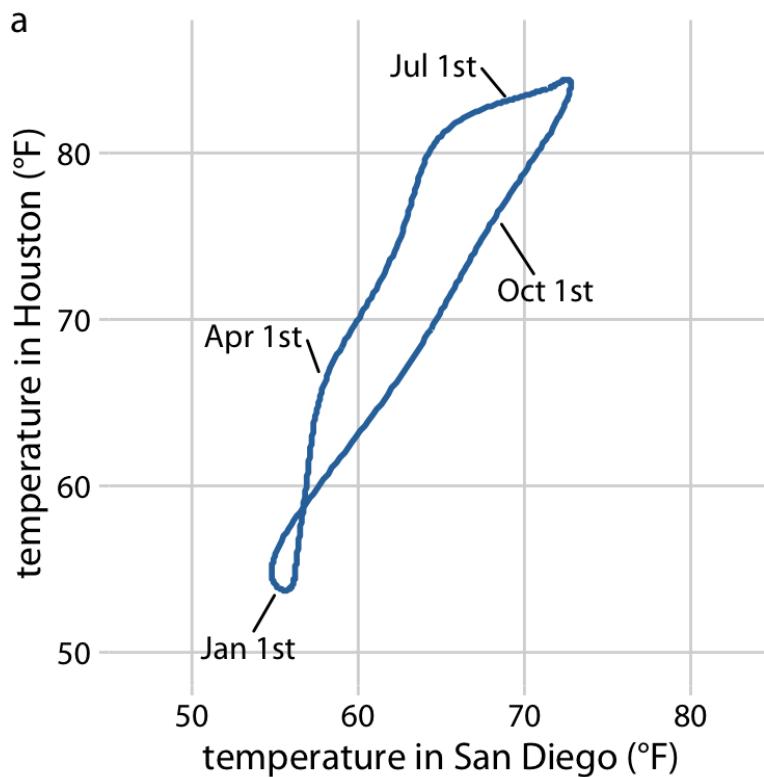
# Coordinate systems : Cartesian

- Note: If units are the same, you should keep a unitary aspect ratio!



# Coordinate systems : Cartesian

- Cartesian systems are invariant under linear transformations



# Nonlinear scales in the axes

---



- **Linear scales are usually easier for the brain to interpret**
  - *But probably because people are used to them*
  - So stick to it if you can

# Nonlinear scales in the axes

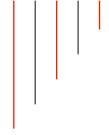
---



- **Linear scales are usually easier for the brain to interpret**
  - *But probably because people are used to them*
  - So stick to it if you can
- **But in many cases nonlinear scales are preferred:**
  - I.e. lots of small and very big numbers in the same dataset

# Nonlinear scales in the axes

---

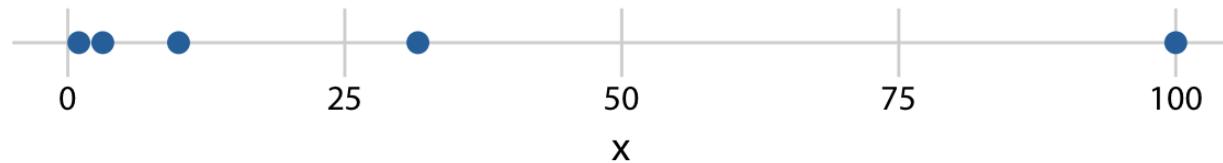


- **Linear scales are usually easier for the brain to interpret**
  - *But probably because people are used to them*
  - So stick to it if you can
- **But in many cases nonlinear scales are preferred:**
  - I.e. lots of small and very big numbers in the same dataset
- Some other scales:
  - **Logarithmic scale** (“log scale”)
  - Square-root scale (“sqrt scale”)
  - More atypical: Hyperbolic arcsin scale
    - *Behaves like log scale for big numbers, and like linear for small numbers*

# Example for $\{1, 3.16, 10, 31.6, 100\}$

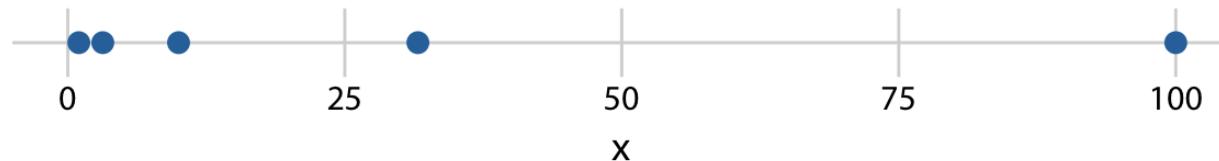
---

original data, linear scale

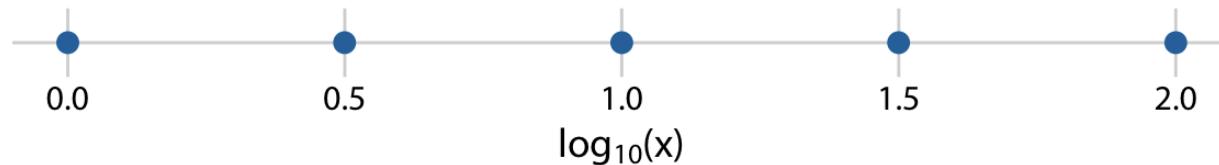


# Example for {1, 3.16, 10, 31.6, 100}

original data, linear scale

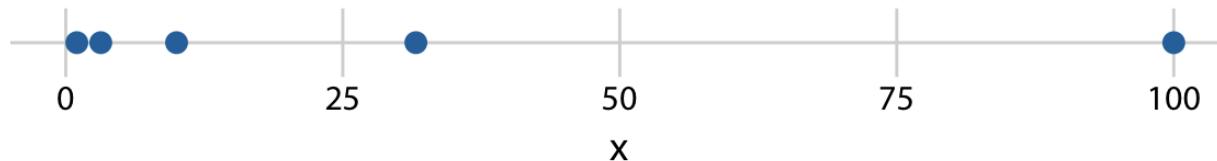


log-transformed data, linear scale

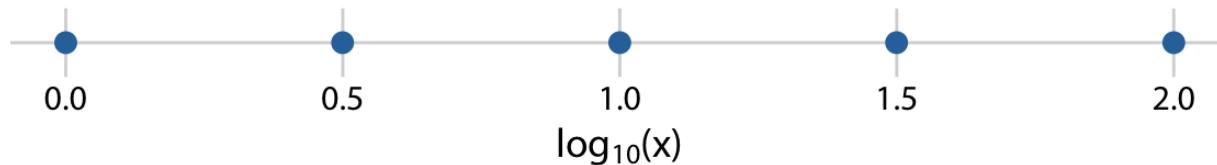


# Example for {1, 3.16, 10, 31.6, 100}

original data, linear scale



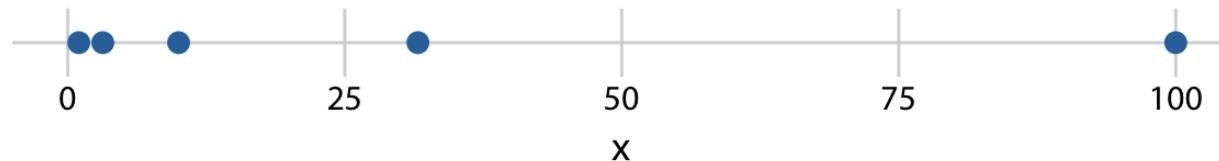
log-transformed data, linear scale



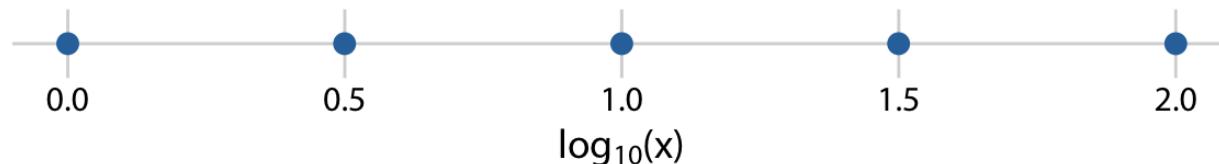
If you do a log-transformation on the data, make sure to specify the base of the log!

# Example for {1, 3.16, 10, 31.6, 100}

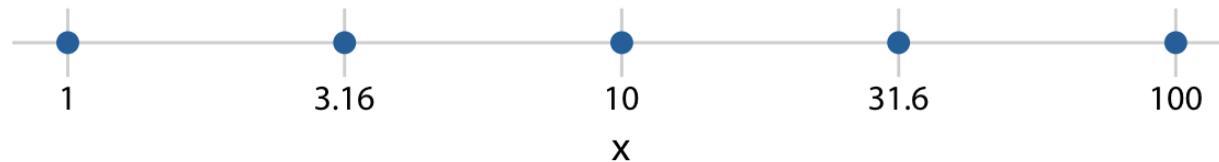
original data, linear scale



log-transformed data, linear scale



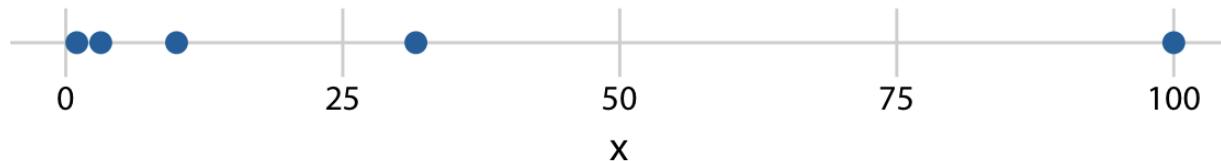
original data, logarithmic scale



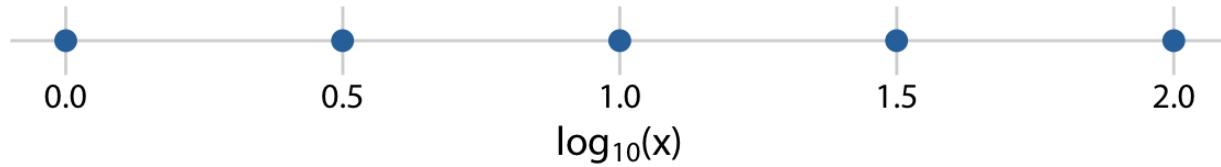
# Example for {1, 3.16, 10, 31.6, 100}



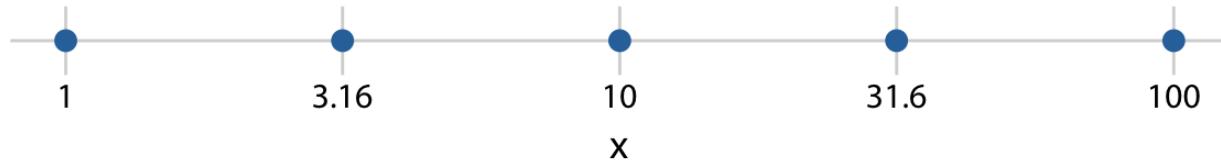
original data, linear scale



log-transformed data, linear scale



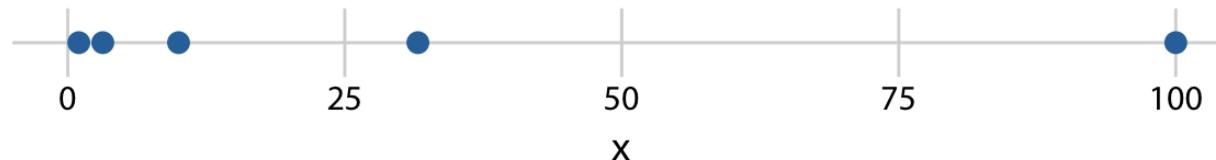
original data, logarithmic scale



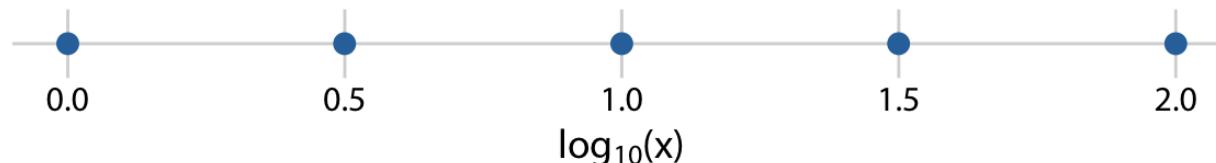
Requires less  
effort to  
analyse

# What not to do!

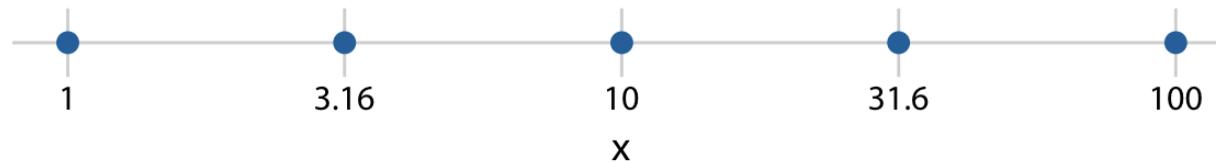
original data, linear scale



log-transformed data, linear scale

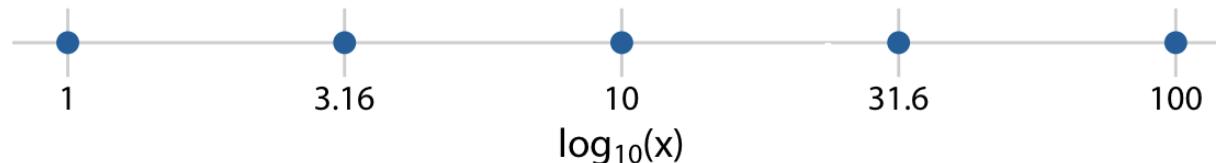


original data, logarithmic scale



logarithmic scale with incorrect axis title

wrong



Never!

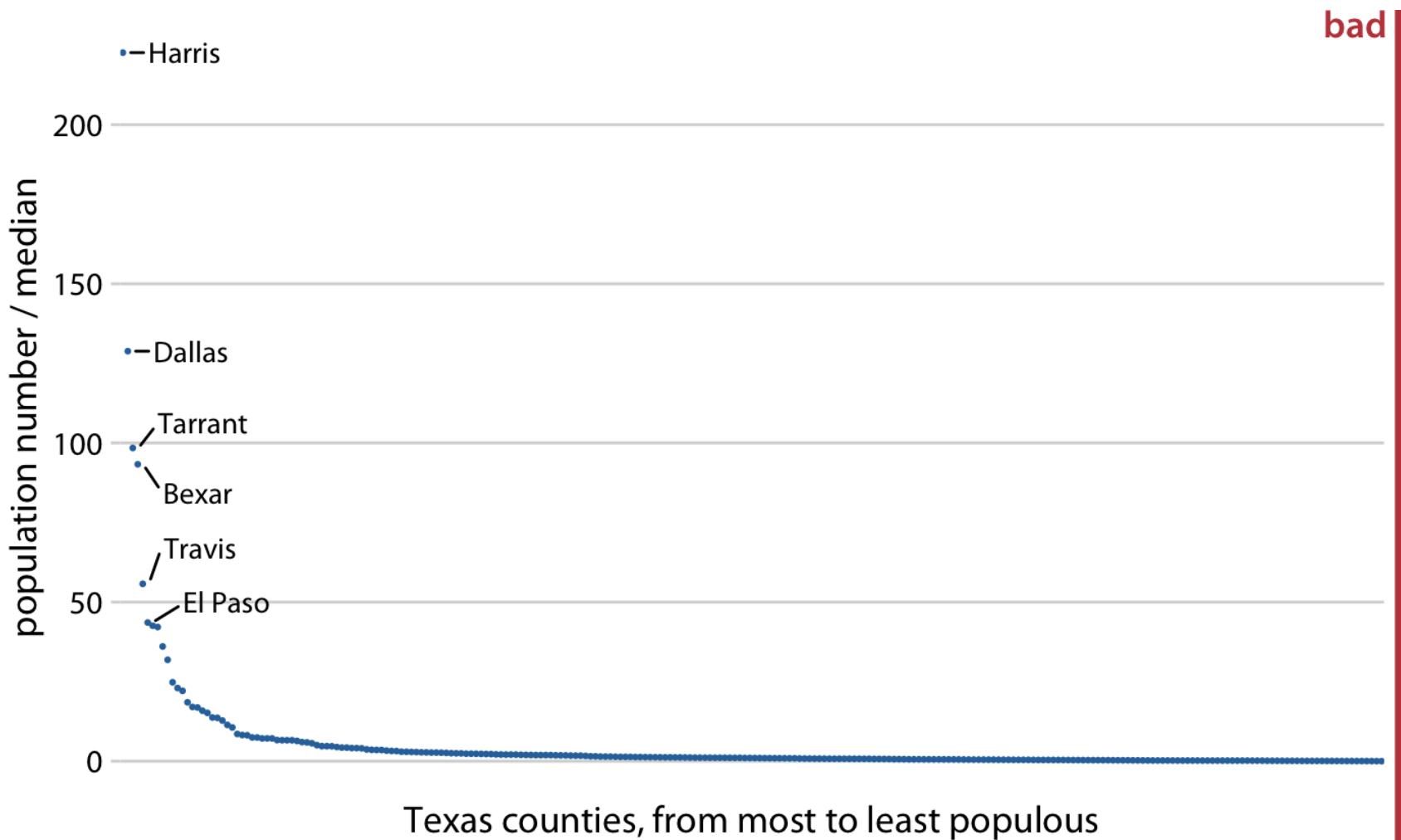
# When to use log scales ?

---

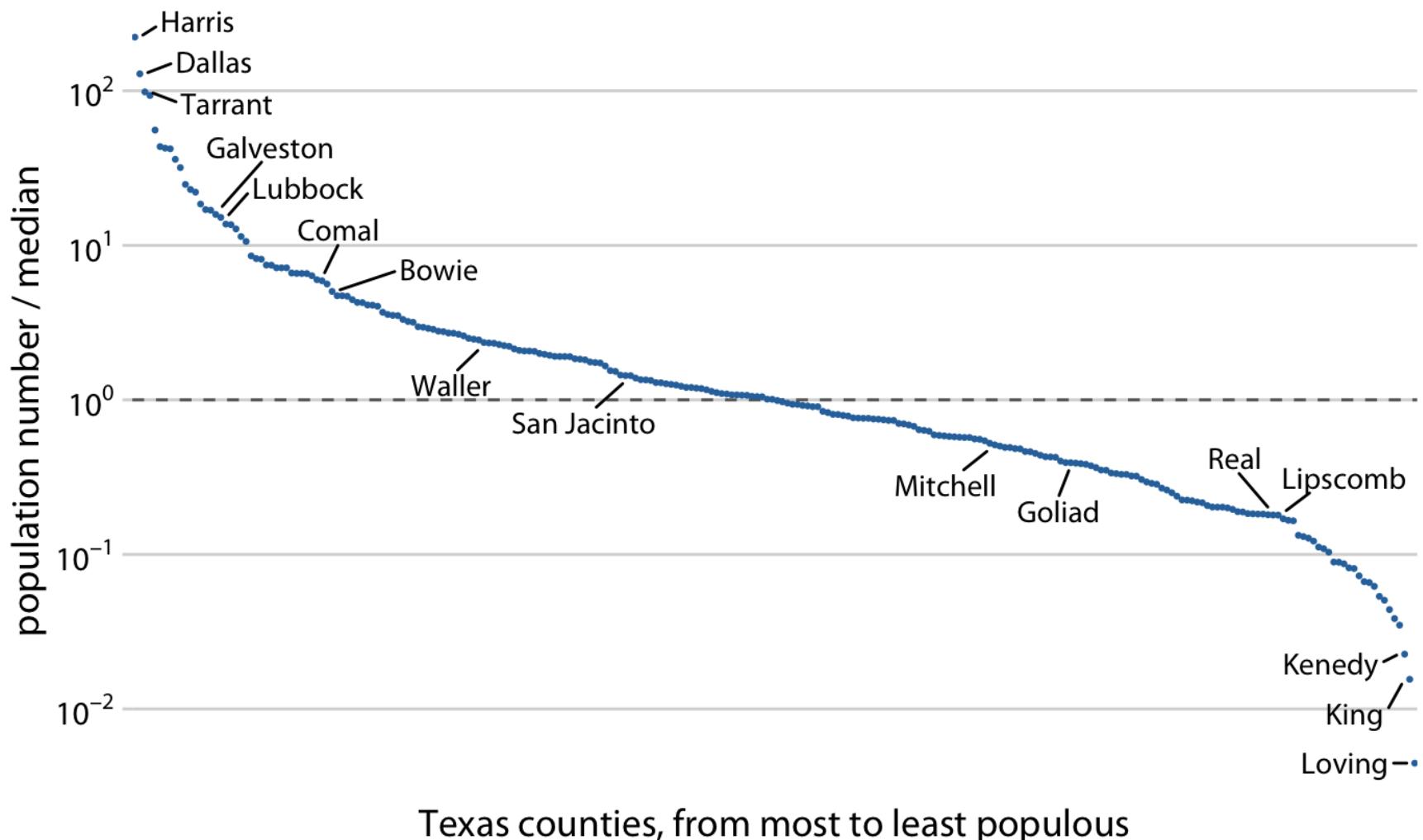


- Very large skewness
  - i.e. : when your data contains a few points that are much much larger than the rest.
- When your data is showing multiplicative factors or ratios

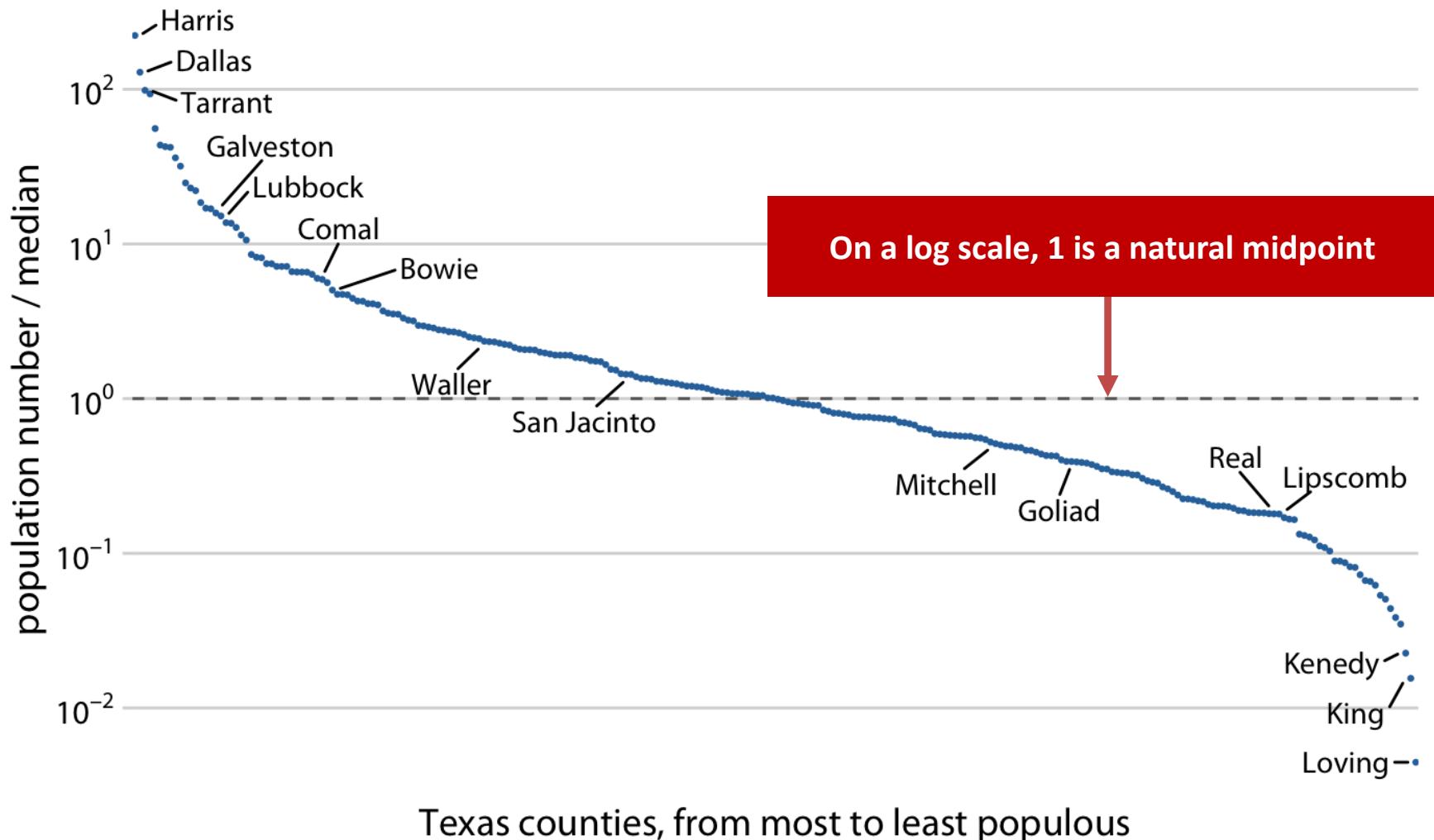
# When to use log scales ?



# When to use log scales ?



# When to use log scales ?



# When to use log scales ?

---

**Important drawback of a logscale :**

**zeros cannot exist!**

$$\log(0) = -\infty$$

# When to use log scales ?

---

**Important drawback of a logscale :**  
**zeros cannot exist.**

**Possible solution : *Hyperbolic arcsine***

$$\text{arcsinh}x = \ln \left( x + \sqrt{(x^2 + 1)} \right)$$

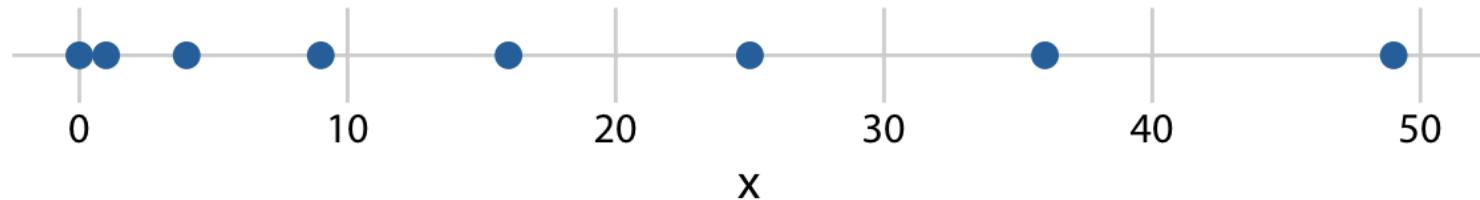
# A more common solution: square-root scale

original data, linear scale

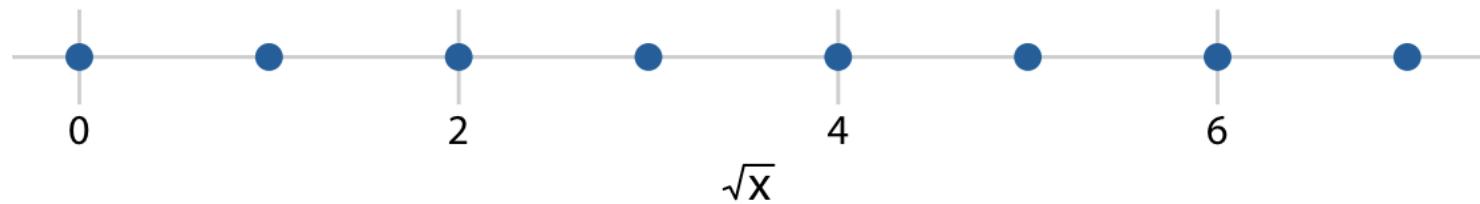


# A more common solution: square-root scale

original data, linear scale

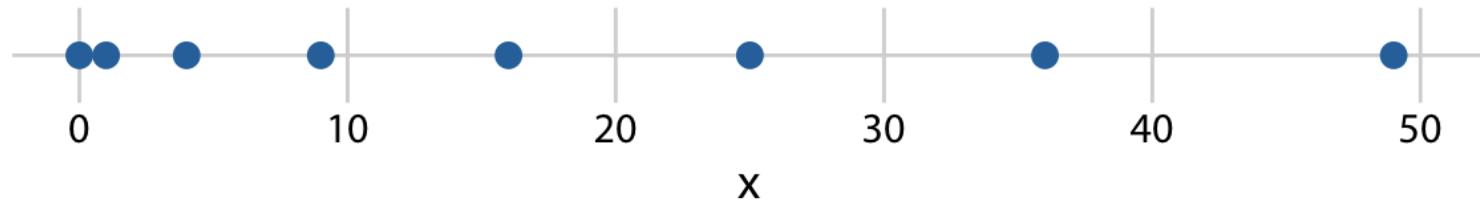


square-root-transformed data, linear scale

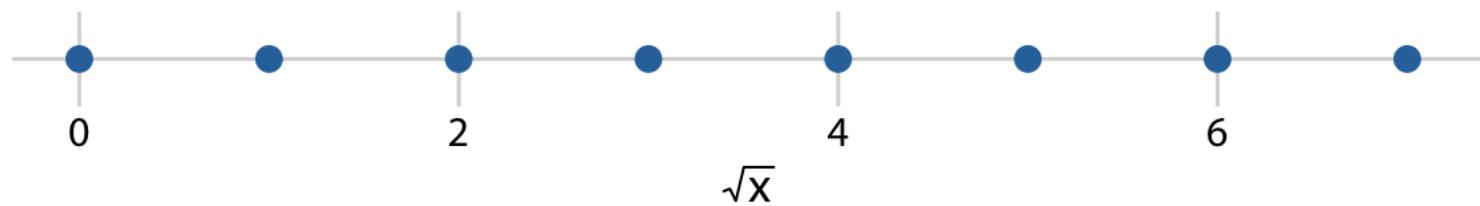


# A more common solution: square-root scale

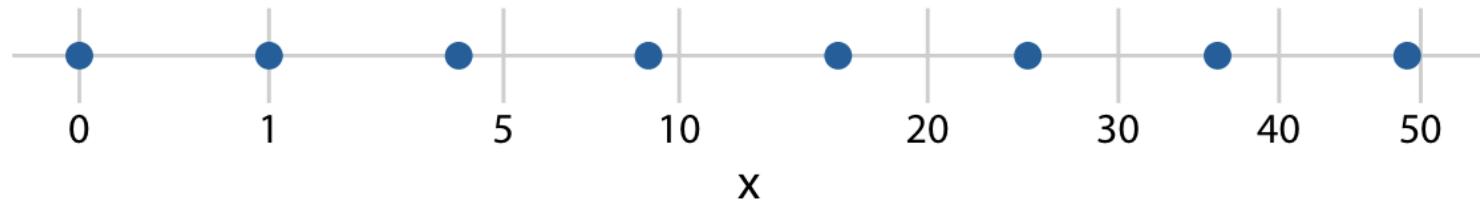
original data, linear scale



square-root-transformed data, linear scale



original data, square-root scale



# A more common solution: square-root scale

- Natural scale for data in squared units

