

# CMPT 733 – Big Data Programming II

## Visualization Principles for Data Science

Instructor                  Steven Bergner

Course website            <https://sfu-db.github.io/bigdata-cmpt733/>

Source: Ch. 6.4 - 6.6 of "Principles of Data Science" by Lau, Gonzales, Nolan  
Slides adapted from Nolan, Dudoit, Perez, & Lau (CC BY-NC-ND 4.0)

# Sources

## Books

- Tamara Munzner “Visualization Analysis and Design”, 2014
- Lau, Gonzalez, Nolan “Principles and Techniques of Data Science”

## Slides

- Torsten Möller’s Visualization course, Spring 2018
- UC Berkley Data 100 (Lau, Nolan, Dudoit, Perez)

# Defining Visualization (Vis)

**Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.**

[“Visualization Analysis and Design” by T. Munzner, 2014]

## Why have a human in the loop?

- Not needed when automatic solution is trusted
- Good for ill-specified analysis problems
  - Common setting: “What questions can we ask?”

# Why have a human in the loop?

Computer-based visualization systems provide visual representations of datasets designed to help **people** carry out **tasks** more effectively.

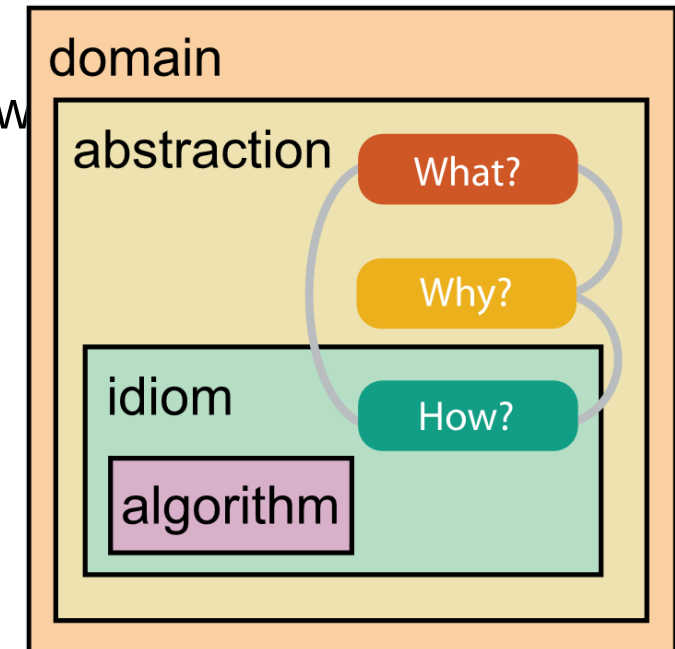
Munzner, T. (2014)

- Long-term use**
- Exploratory analysis of scientific data
  - Presentation of known results

- Short-term use**
- For **developers** of automatic solutions:
    - Understand requirements for model development
    - Refine/debug and determine parameters
  - For **end users** of automatic solutions: verify, build trust

# Analysis framework: four levels

- **Domain** situation: Who are the target users?
- **Abstraction**: Translate from specifics of domain to vocabulary of vis
- **What** is shown? *Data abstraction*
  - Don't just draw what you're given: transform to new
- **Why** is the user looking at it? *Task abstraction*
- **How** is it shown? **Idiom (Vis technique)**
  - Visual encoding idiom: How to draw
  - Interaction idiom: How to manipulate
- **Algorithm**: efficient computation



# Resource limitations

- **Computational** limits
  - Processing time and system memory
- **Human** limits
  - Human attention and memory
  - Understanding abstractions
- **Display** limits
  - Pixels are precious
  - Information density tradeoff: Info encoding vs unused whitespace

# **Understand Data, Task, and Encoding**

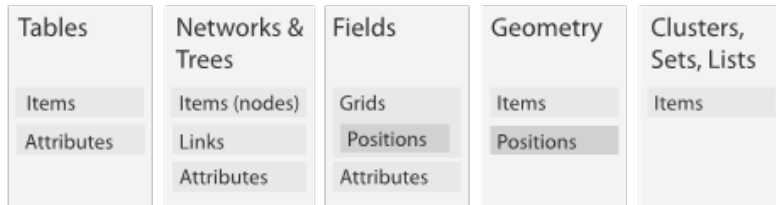
## What?

### Datasets

#### → Data Types

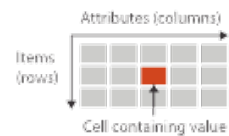
→ Items → Attributes → Links → Positions → Grids

#### → Data and Dataset Types

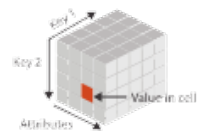


#### → Dataset Types

→ Tables



→ Multidimensional Table



→ Geometry (Spatial)



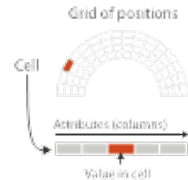
→ Networks



→ Trees



→ Fields (Continuous)



#### → Dataset Availability

→ Static



→ Dynamic



### Attributes

#### → Attribute Types

→ Categorical

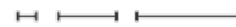


→ Ordered

→ Ordinal



→ Quantitative



#### → Ordering Direction

→ Sequential



→ Diverging



→ Cyclic



# Data Types

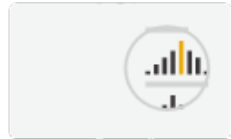
- Items and attributes as rows and columns of tables
- Position and time are special attributes
- Spatial data on grids makes computation easier

## Why?

### Actions

#### → Analyze

→ Consume



→ Present



→ Enjoy



→ Produce

→ Annotate



→ Record



→ Derive



#### → Search

	Target known	Target unknown
Location known	Lookup	Browse
Location unknown	Locate	Explore

#### → Query

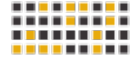
→ Identify



→ Compare



→ Summarize



### Targets

#### → All Data

→ Trends



→ Outliers



→ Features



#### → Attributes

→ One

→ Distribution



→ Extremes



→ Many

→ Dependency



→ Correlation



→ Similarity



#### → Network Data

→ Topology



→ Paths



#### → Spatial Data

→ Shape



# Tasks

- Actions

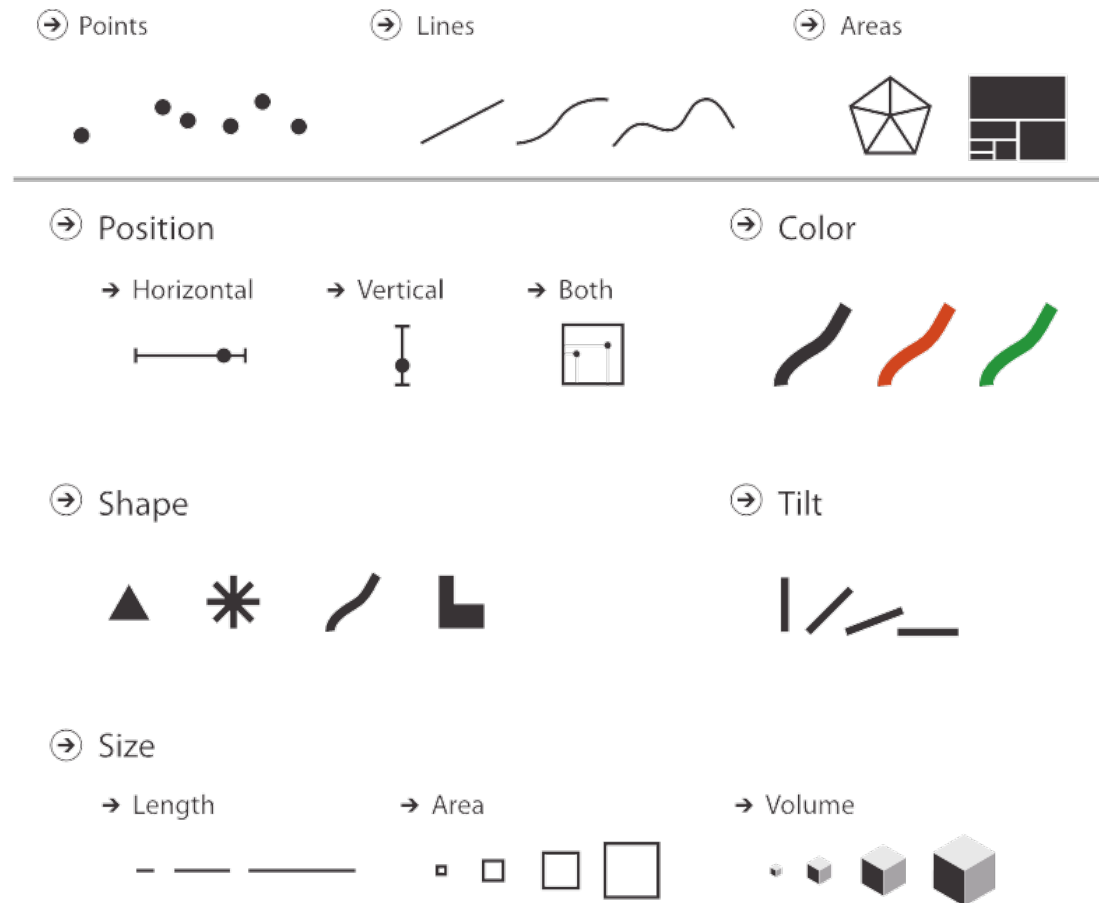
- Analyze
- Search
- Query

- Targets

- Item & Attributes
- Topology & Shape

# Visual Encoding – How?

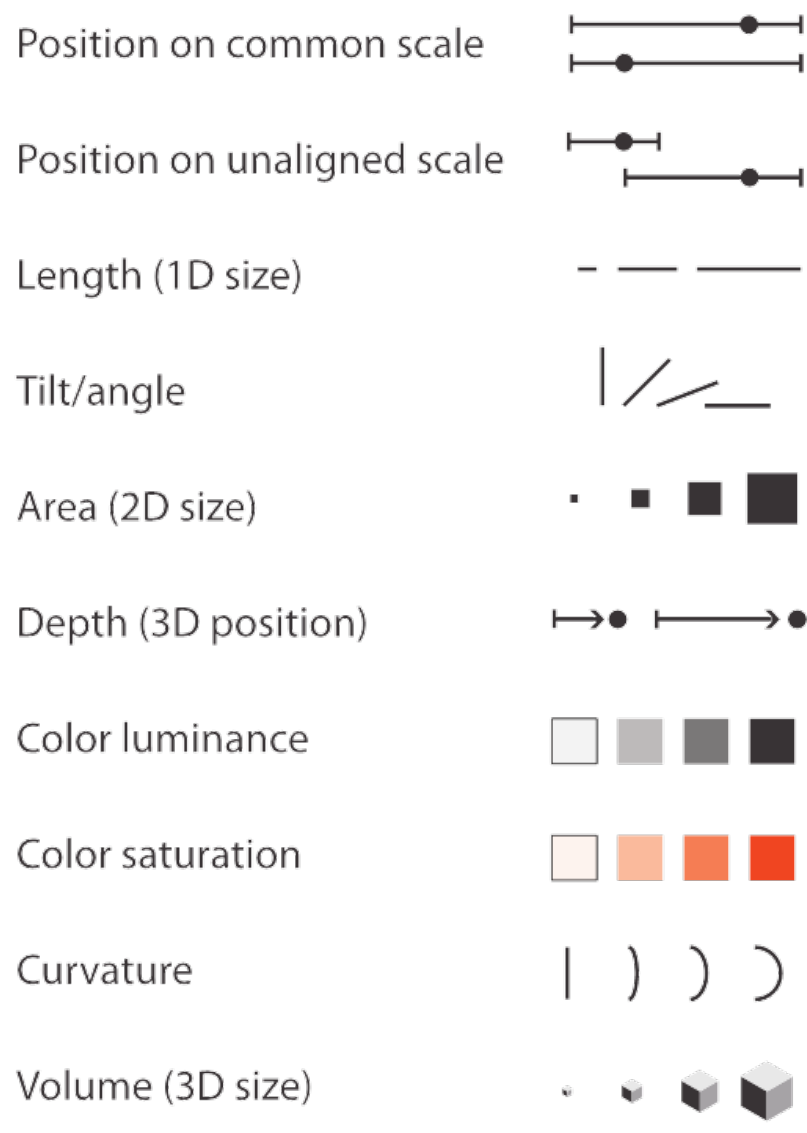
- Marks
  - Geometric primitives
- Channels
  - Appearance of marks
  - Redundant coding of data with multiple channels is possible



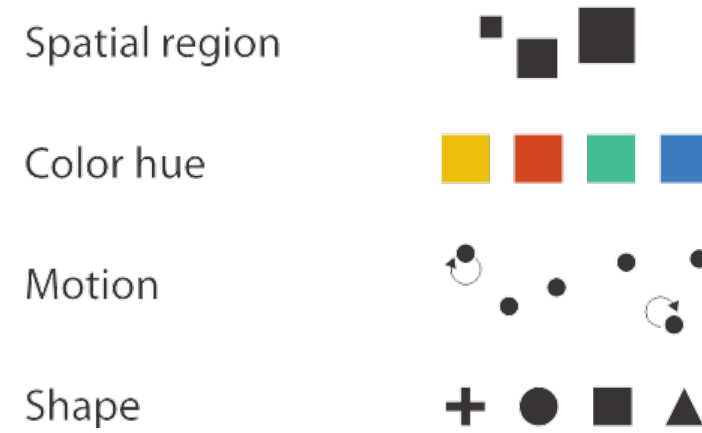
[T. Munzner, 2014]

# **Design Principles for Task Effective Visualization**

## ➔ Magnitude Channels: **Ordered** Attributes



## ➔ Identity Channels: **Categorical** Attributes



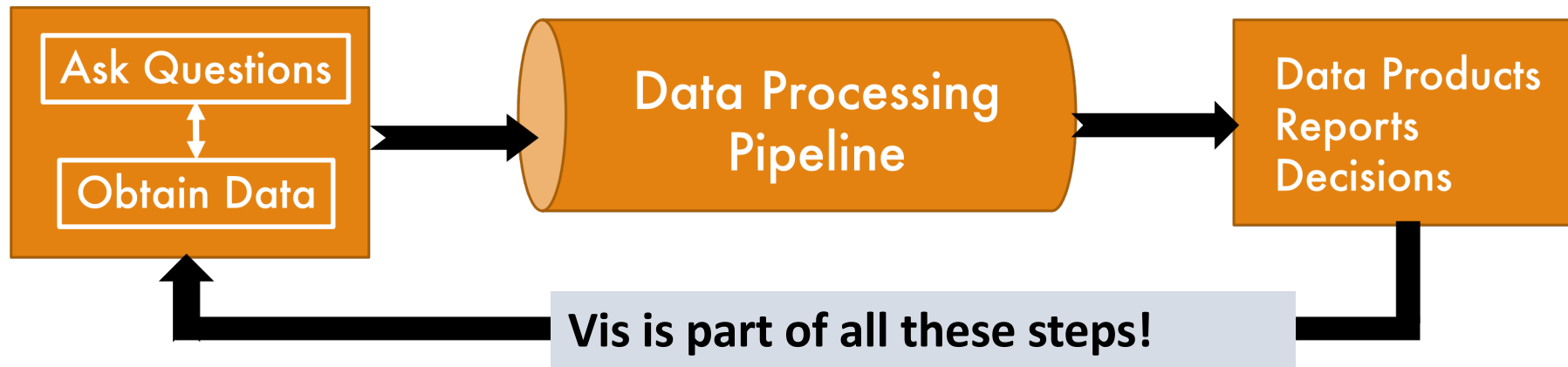
Expressiveness principle

- **Match channel characteristics and data type**

Effectiveness principle

- **Encode important attributes with higher ranked channels**

# Recap: Data Science Lifecycle



## Related Processes

### Big Data Journey

- Business transformations as a company becomes more data-centric

### Data Visualization *Process*

- Acquire, Parse, Filter, Mine, Represent, Refine, Interact [Ben Fry '07, Visualizing Data]

### Data Visualization *Pipeline*

- Analyse (Wrangling), Filter, Map to visual properties, Render geometry

# Six Principles Today

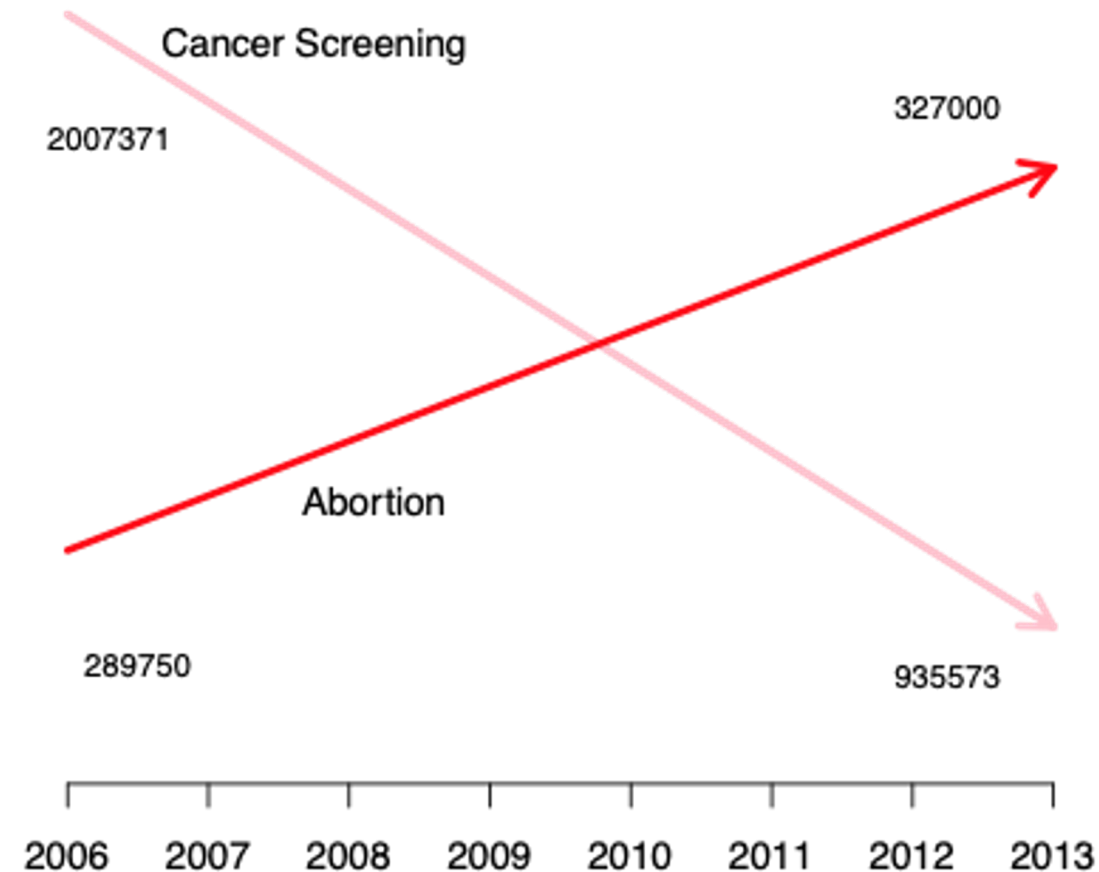
1. Scale
2. Conditioning
3. Perception
4. Transformations
5. Context
6. Smoothing

Explored via three case studies.

# Case 1: Planned Parenthood 2015 Hearing

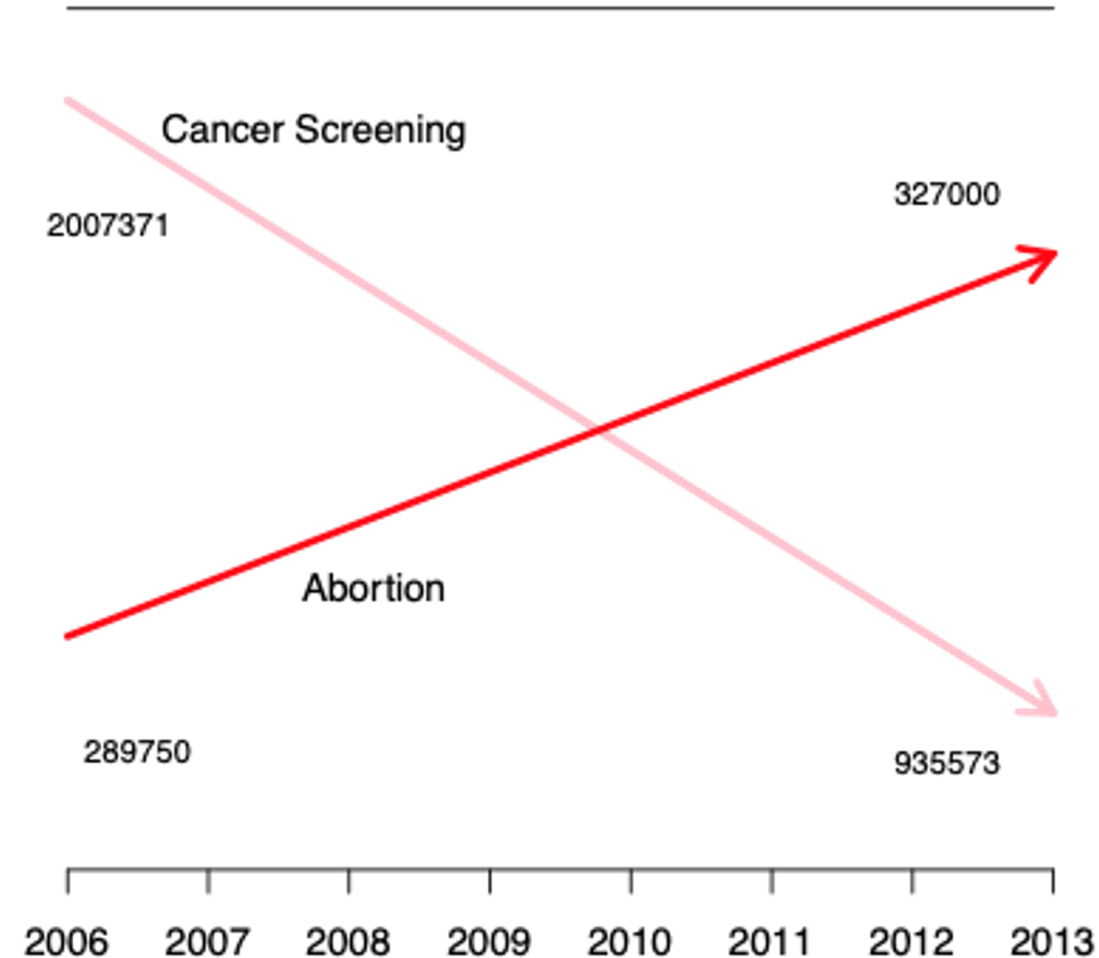
---

- Investigation of federal funding of Planned Parenthood in light of fetal tissue controversy
- Congressman Chaffetz (R-UT) showed plot which originally appeared in a report by Americans United for Life (<http://www.aul.org/>)



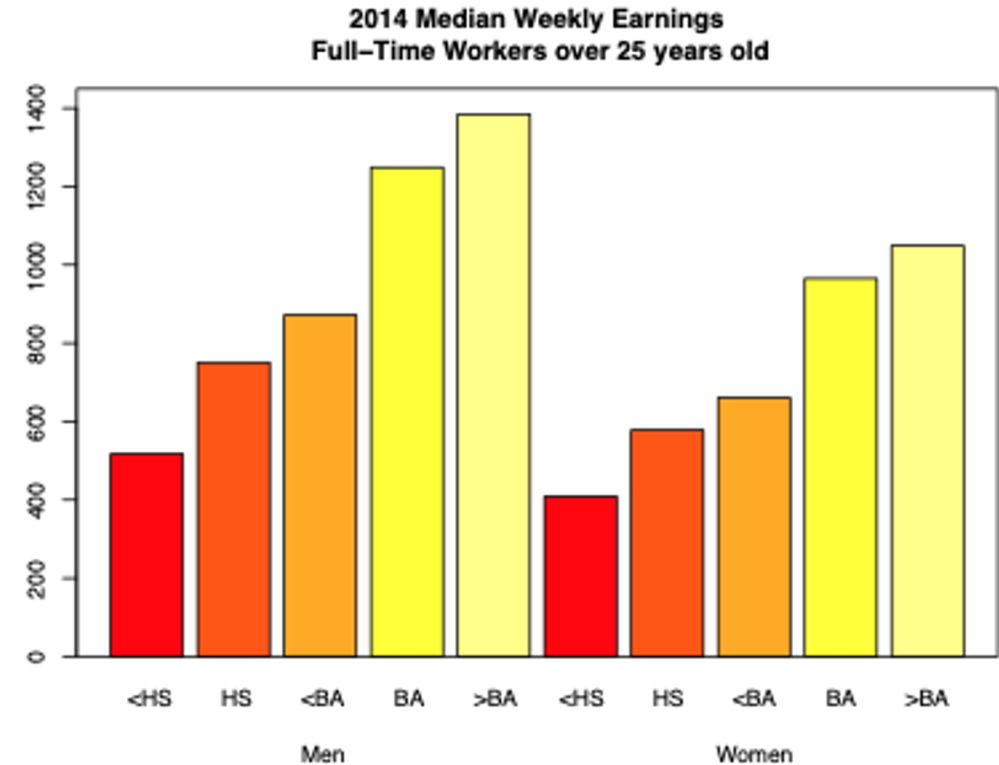
# Case 1: Planned Parenthood 2015 Hearing

- Procedures: cancer screenings and abortions
- How many data points are plotted?
- What is suspicious?
- What message is this plot trying to convey?



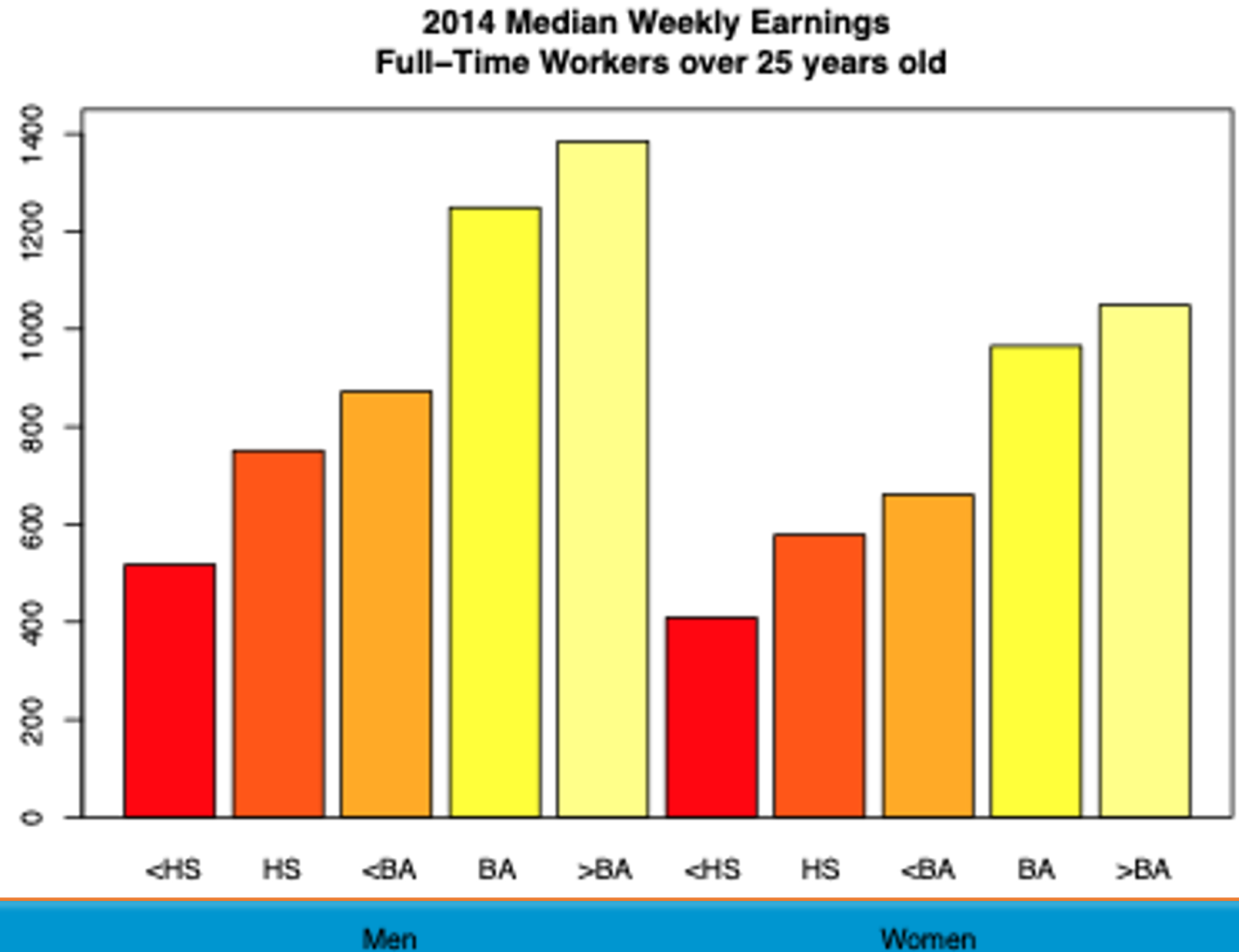
# Case 2: Median Weekly Earnings

- Bureau of Labor Statistics surveys economics of labor
- [www.bls.gov](http://www.bls.gov) - Web interface to a report generating app
- Plot of median weekly earnings for males and females by education level



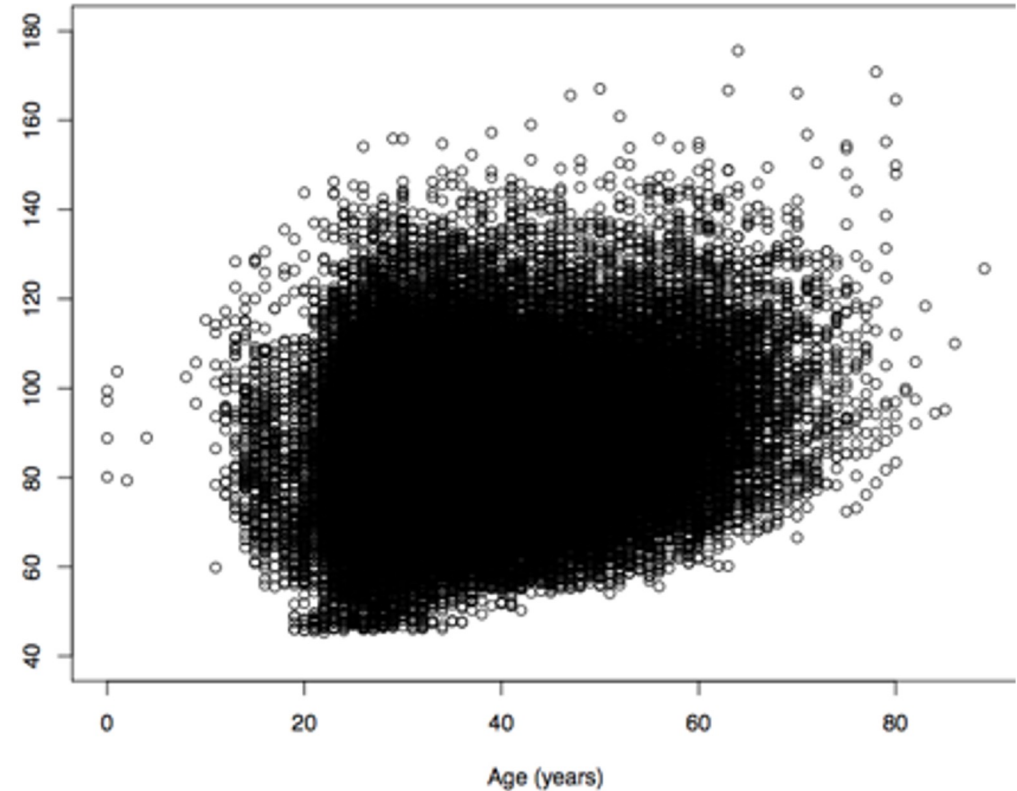
# Case 2: Median Weekly Earnings

- What comparisons are easily made with this plot?
- What comparisons are most interesting and important?



# Case 3: Cherry Blossom Runners

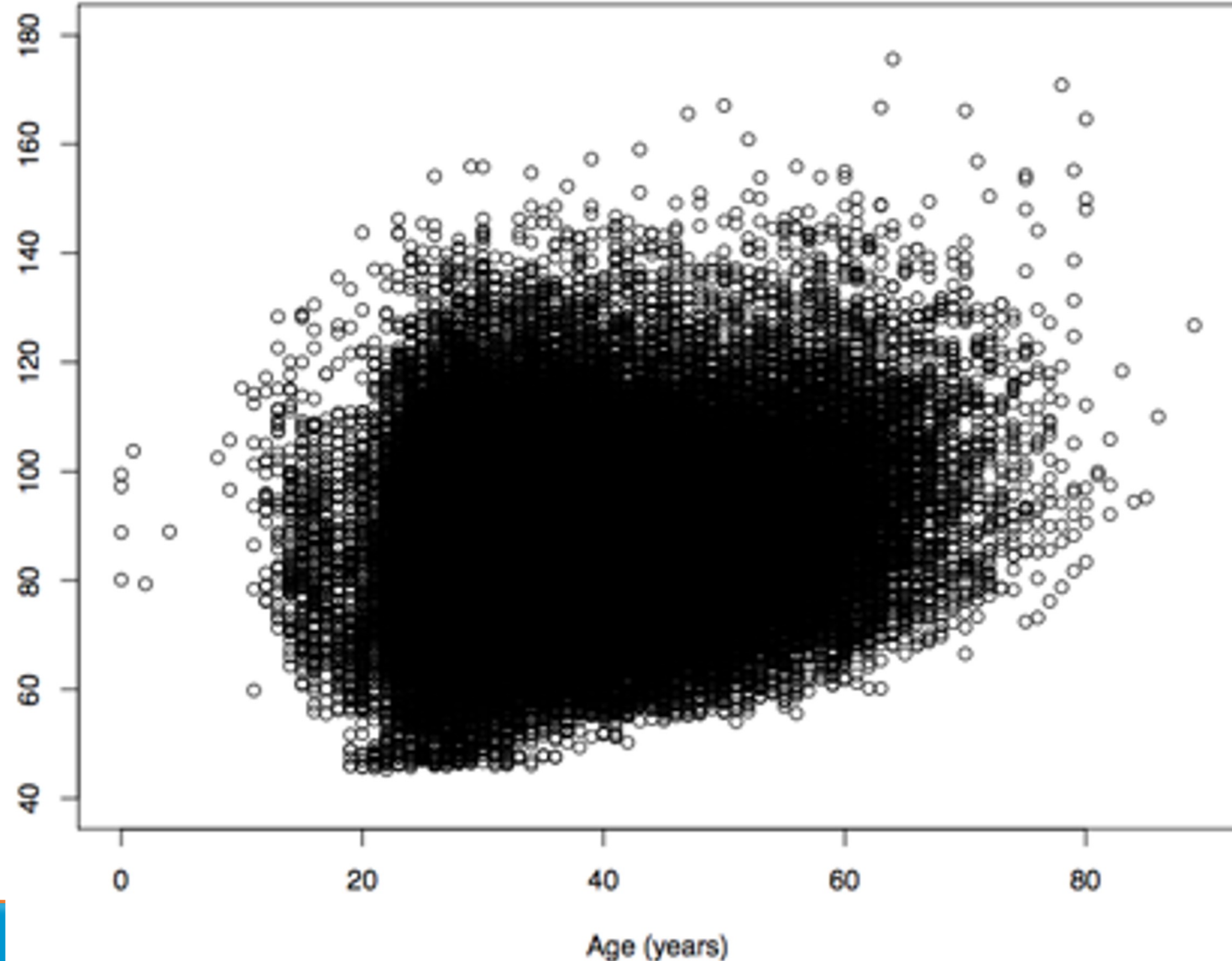
- 10 mi run in DC every April
- Results available from 1999-2019
- In 2019 over 17,000 runners
- Scatter plot of run time (min) against age (yrs)



<http://www.cherryblossom.org/>

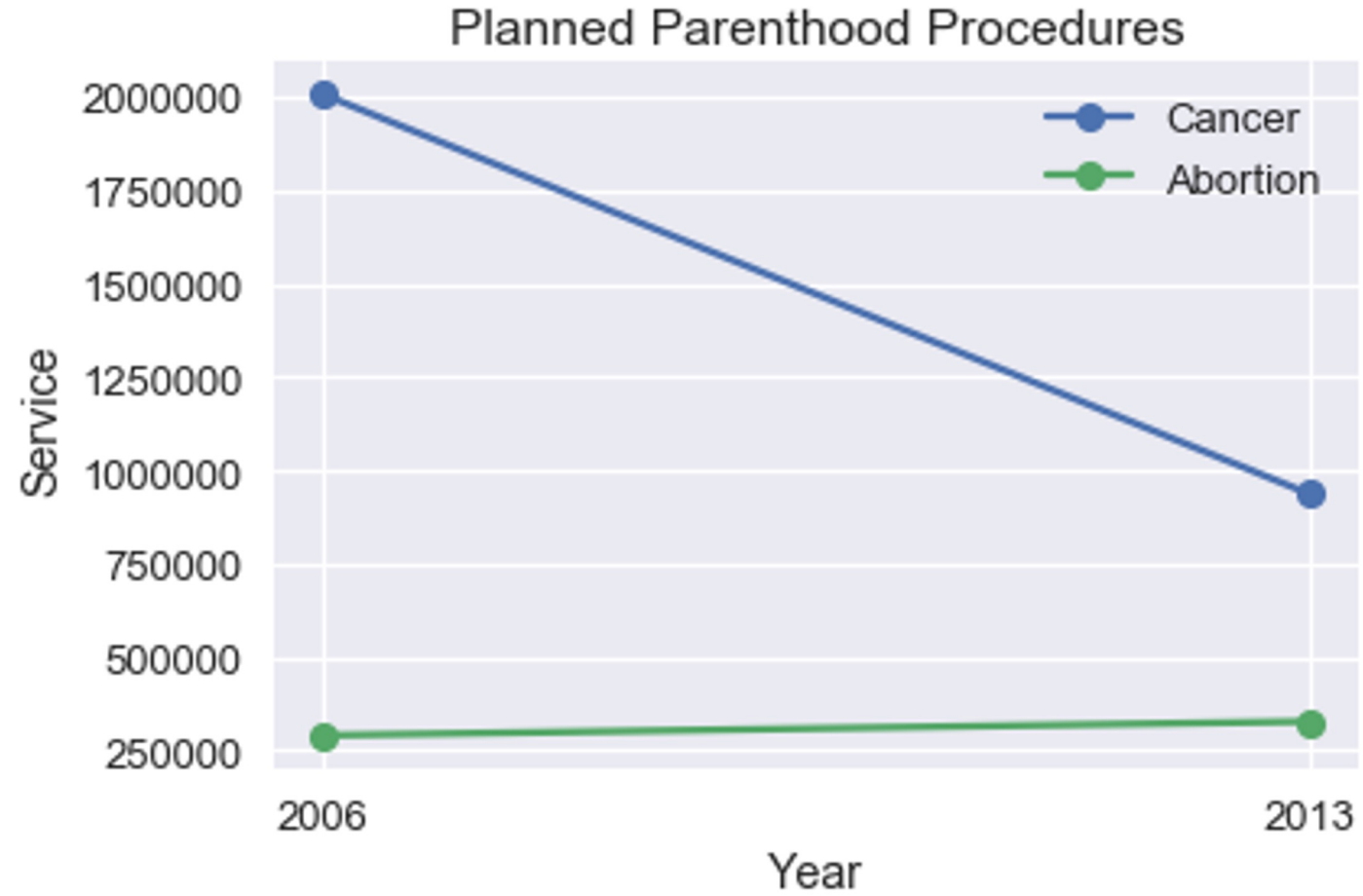
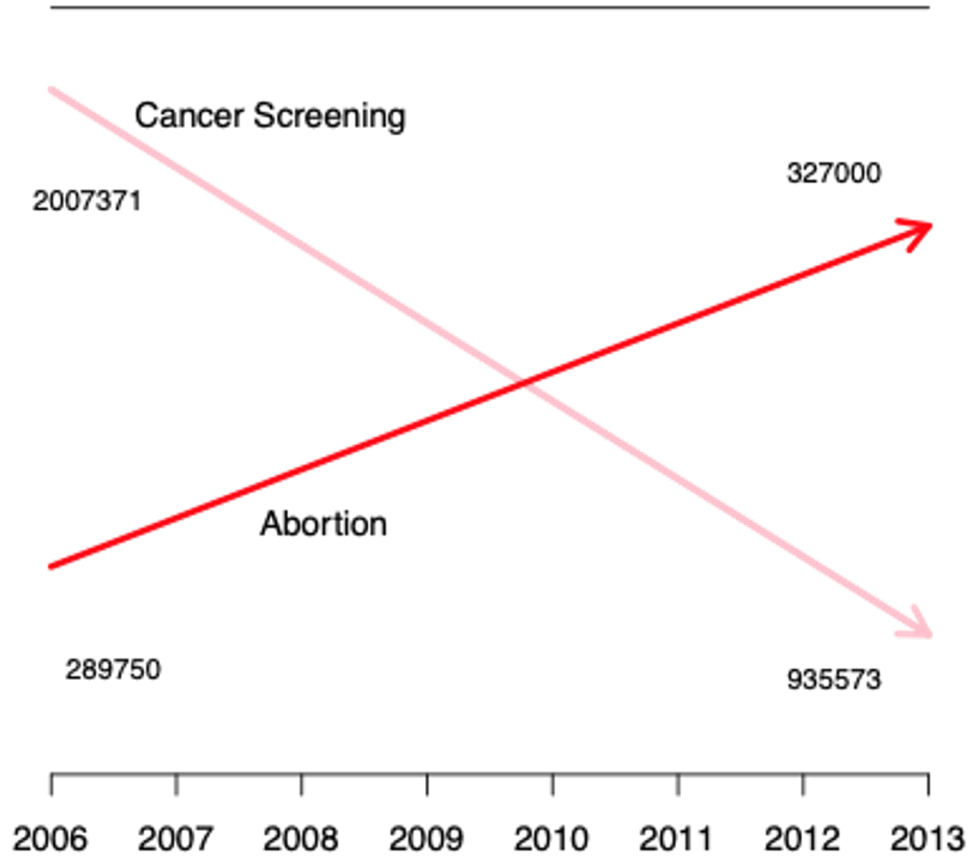
# Case 3: Cherry Blossom Runners

- 70,000+ points in the plot!
- What's the relationship between run time and age?



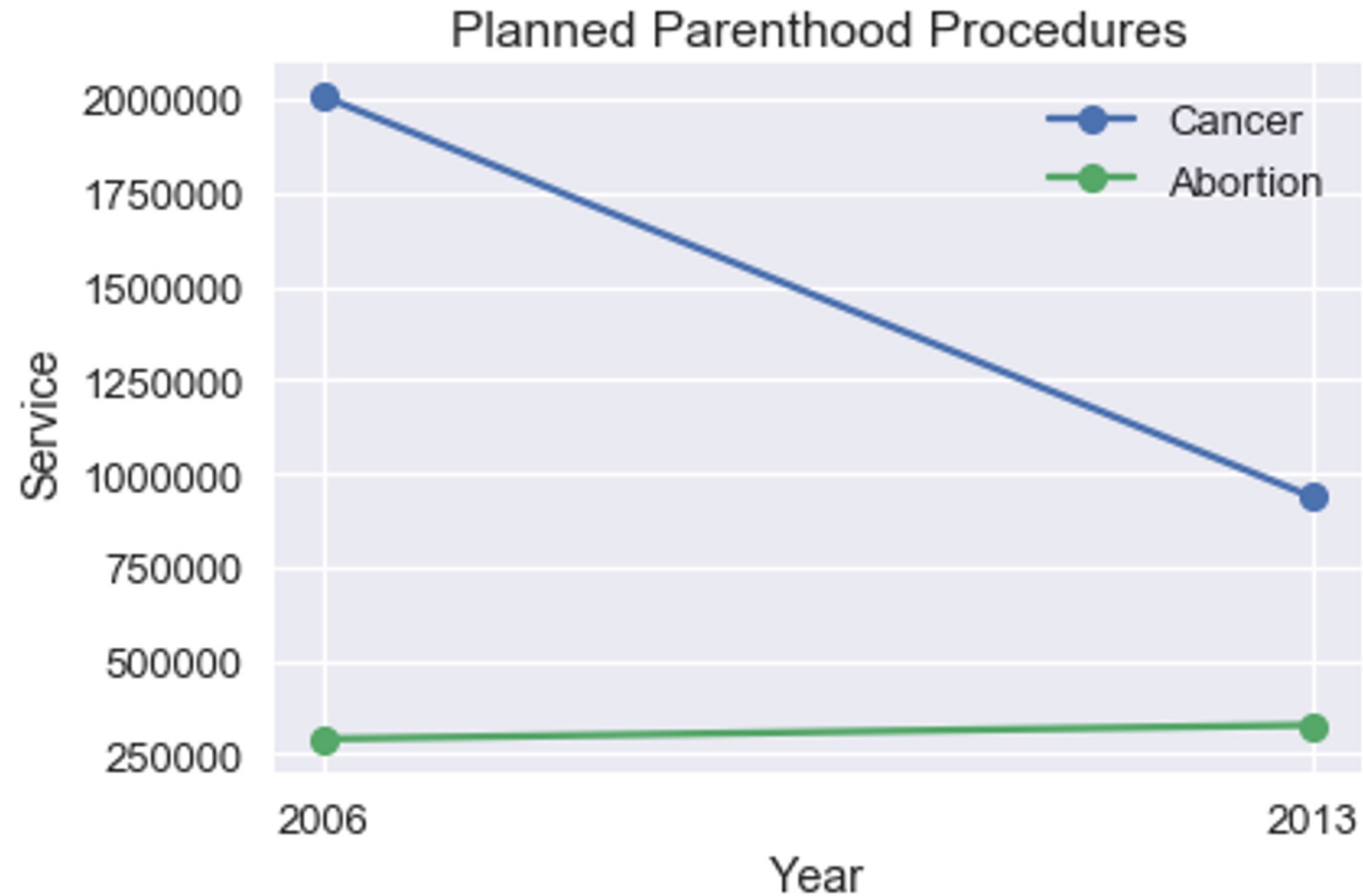
# Principles of Scale

# Scale



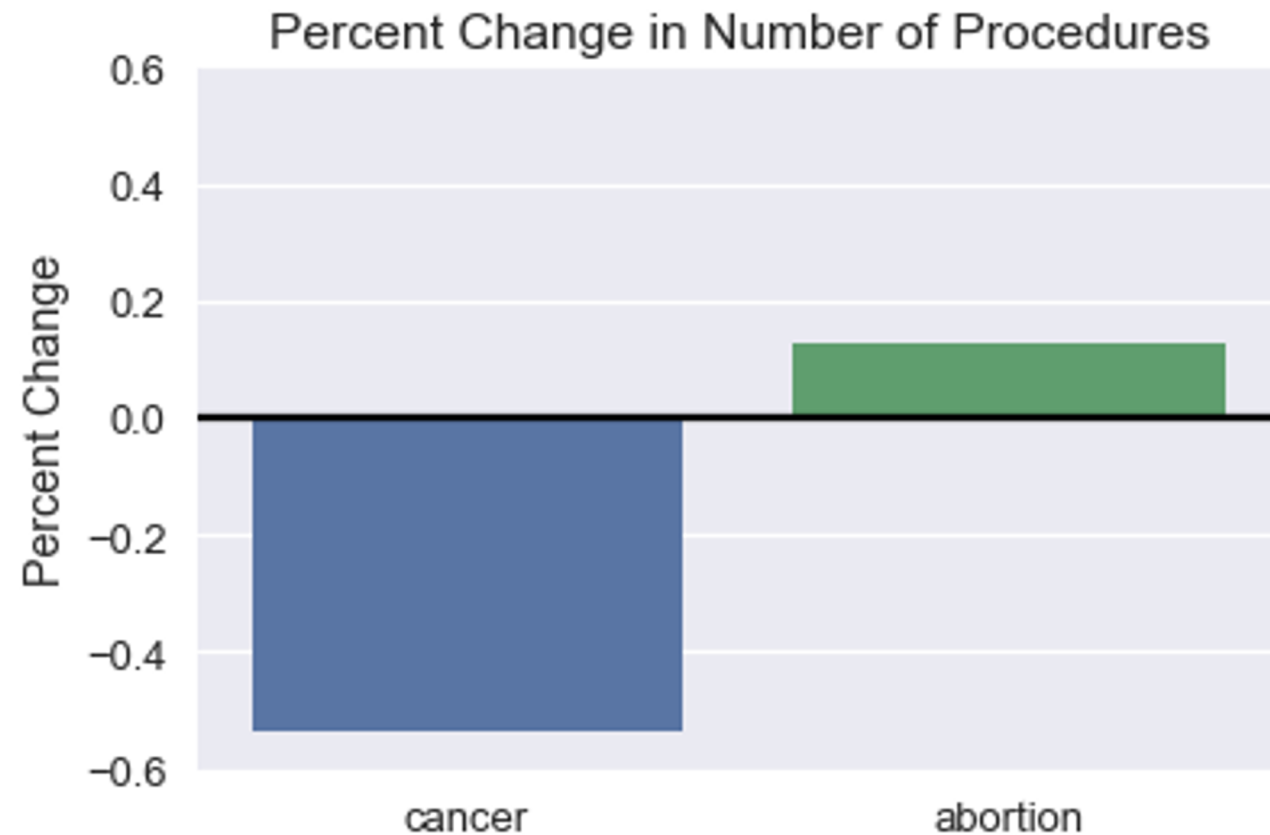
# Keep consistent axis scales

- Don't change scale mid-axis
- Don't use two different scales for same axis
- How does this plot change perception of information?



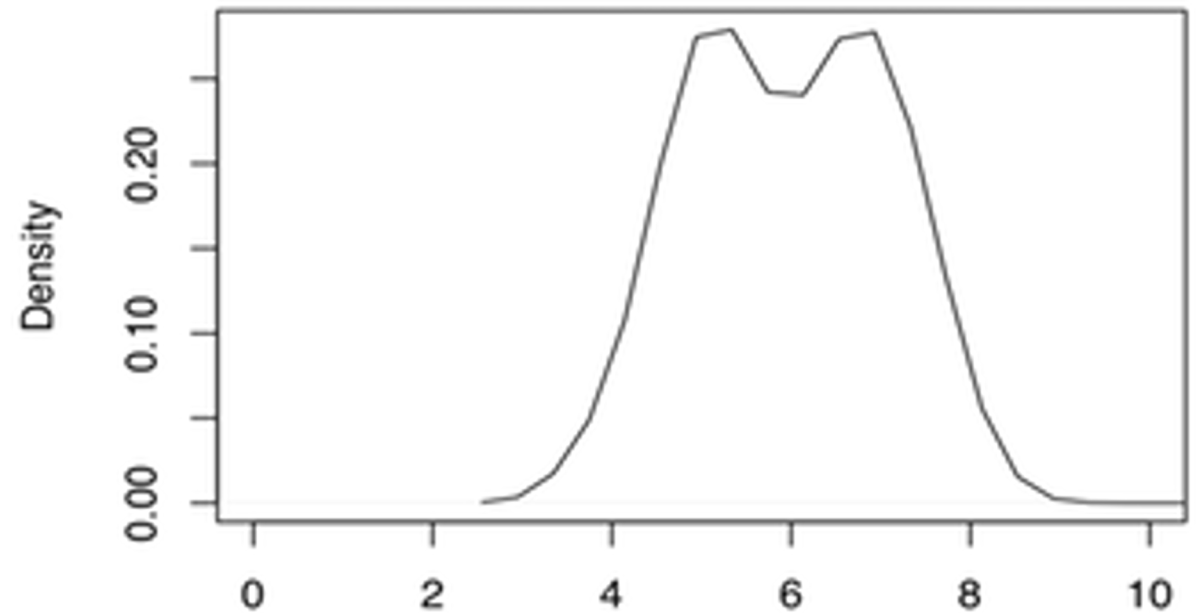
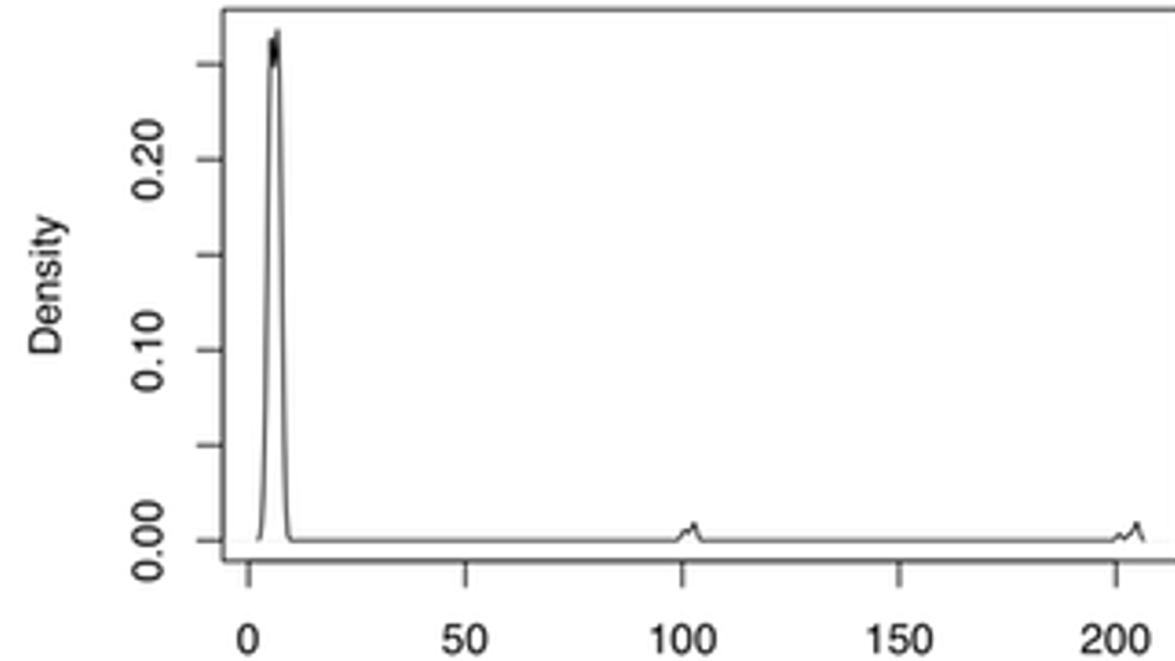
# Consider Scale of Data

- Scales of cancer screenings vs. abortions quite different
- Can plot percent change instead of raw counts

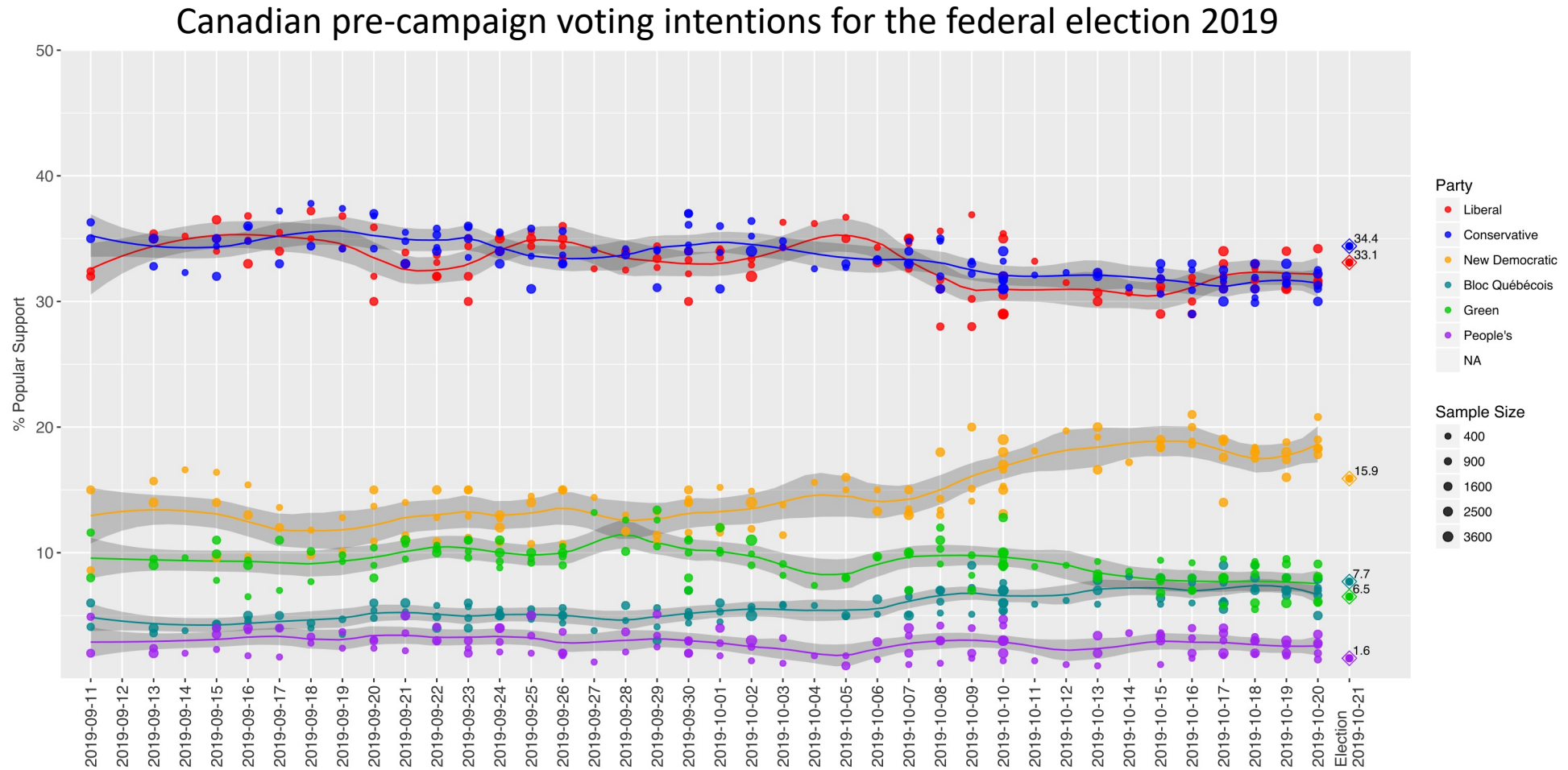


# Reveal the Data

- Choose axis limits to fill plot
- If necessary, zoom into region with most of data
- Can make separate plots for different regions



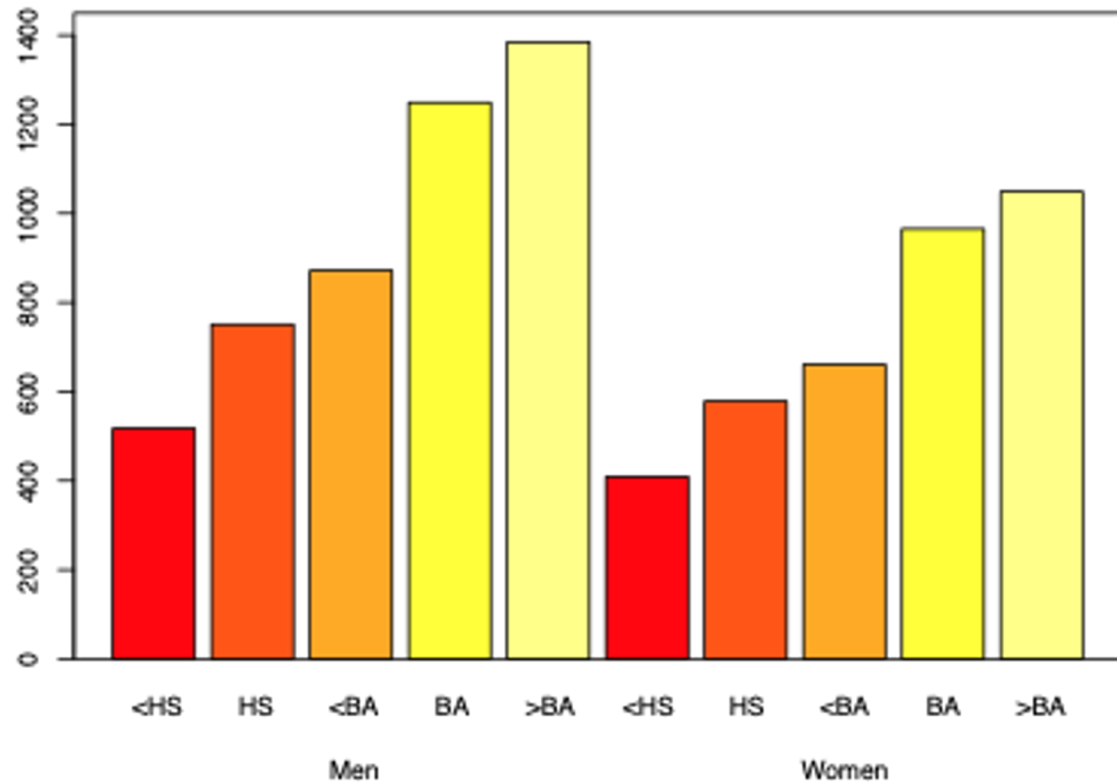
# Time Scale



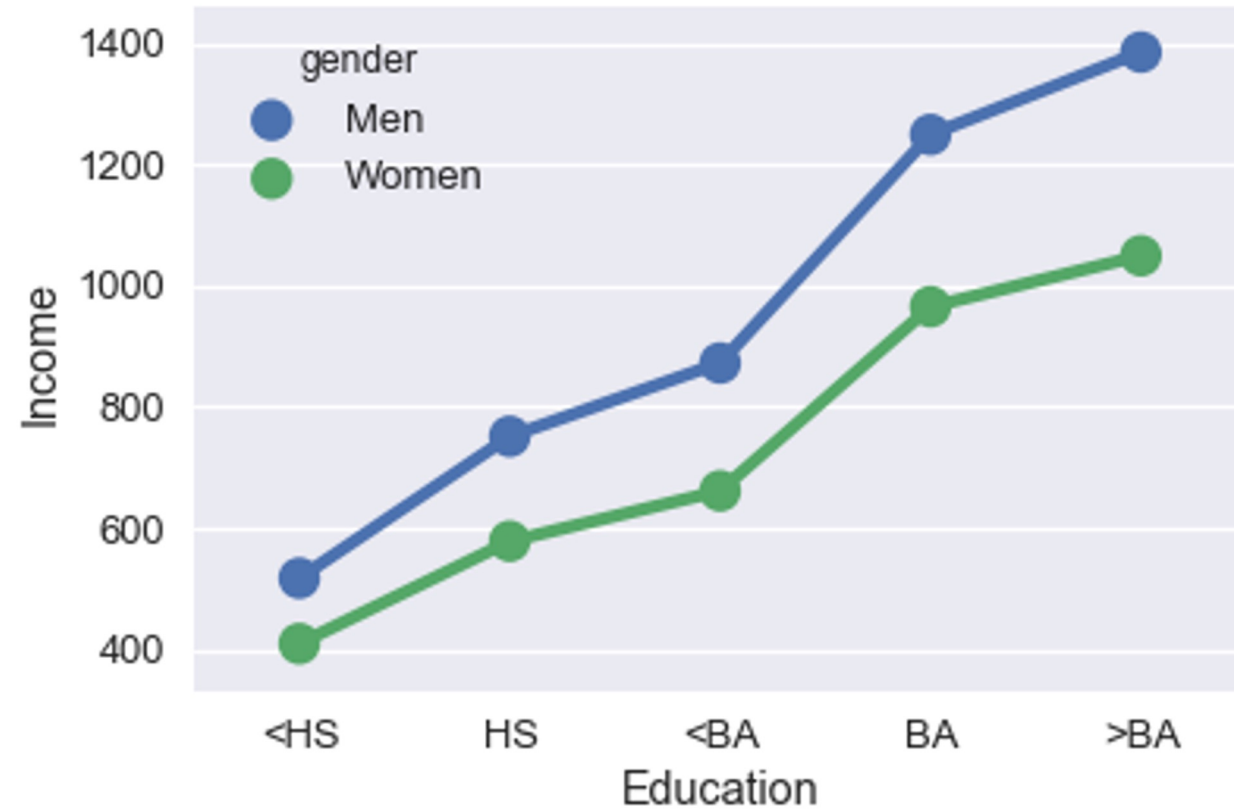
# **Principles of Conditioning**

# Conditioning

2014 Median Weekly Earnings  
Full-Time Workers over 25 years old

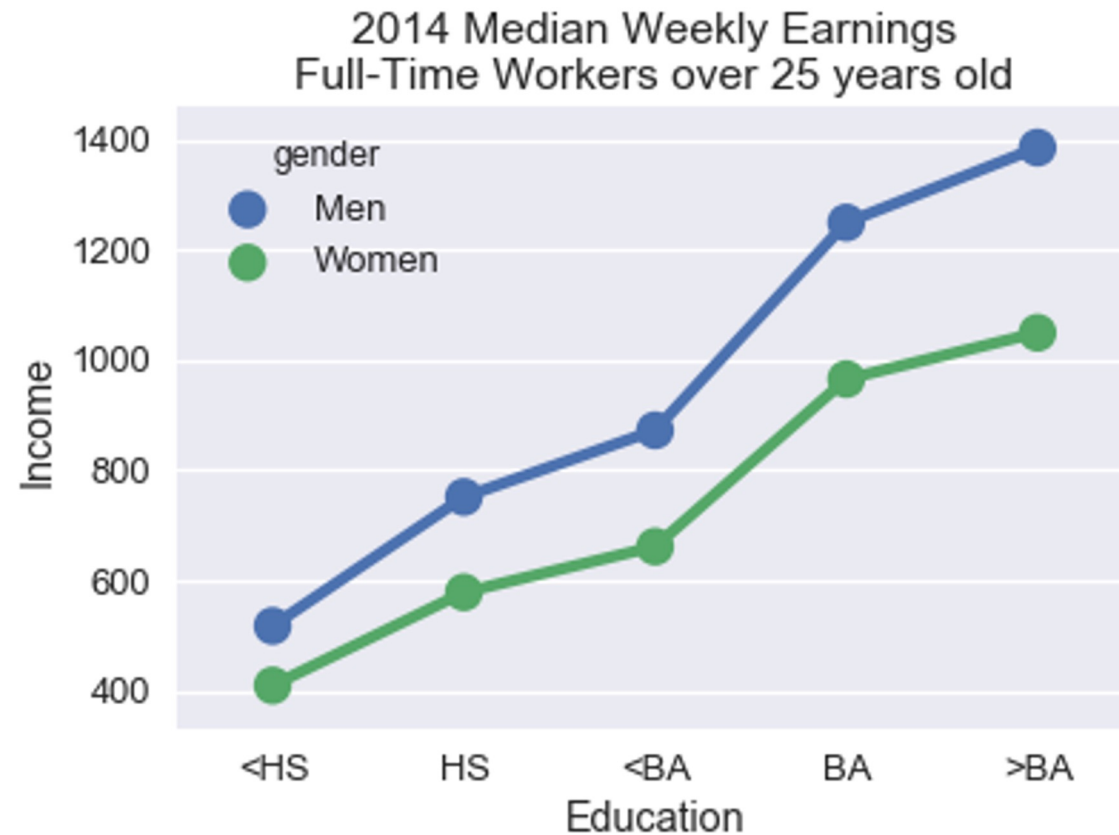


2014 Median Weekly Earnings  
Full-Time Workers over 25 years old



# Use Conditioning To Aid Comparison

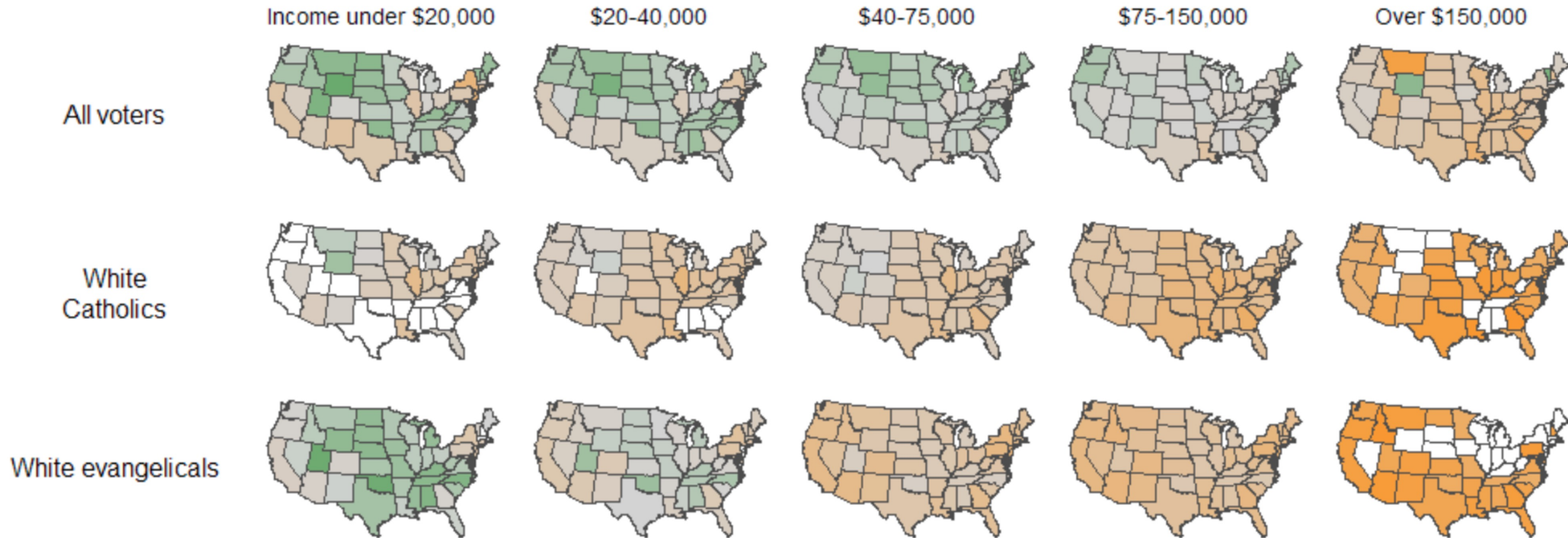
- Conditioning on male/female aligns points on x-axis
  - What does it reveal?
  - Why is this interesting?



# Use Small Multiples To Aid Comparison

- Faceted plots that share scales are easy to compare
- [https://statmodeling.stat.columbia.edu/2009/07/15/hard\\_sell\\_for\\_b/](https://statmodeling.stat.columbia.edu/2009/07/15/hard_sell_for_b/)

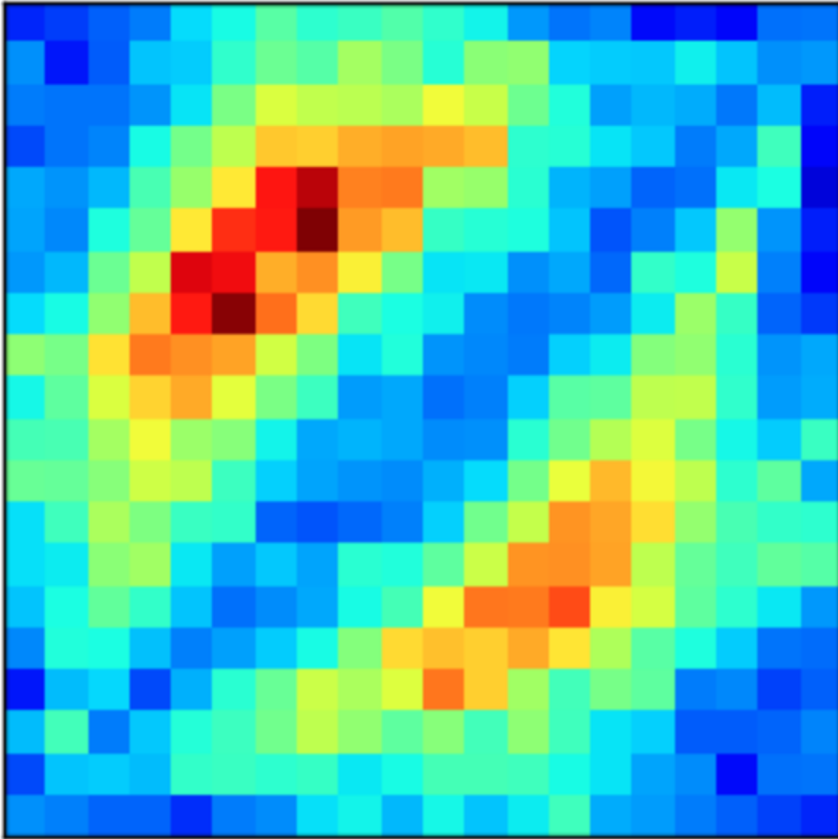
2000: State-level support (orange) or opposition (green) on school vouchers, relative to the national average of 45% support



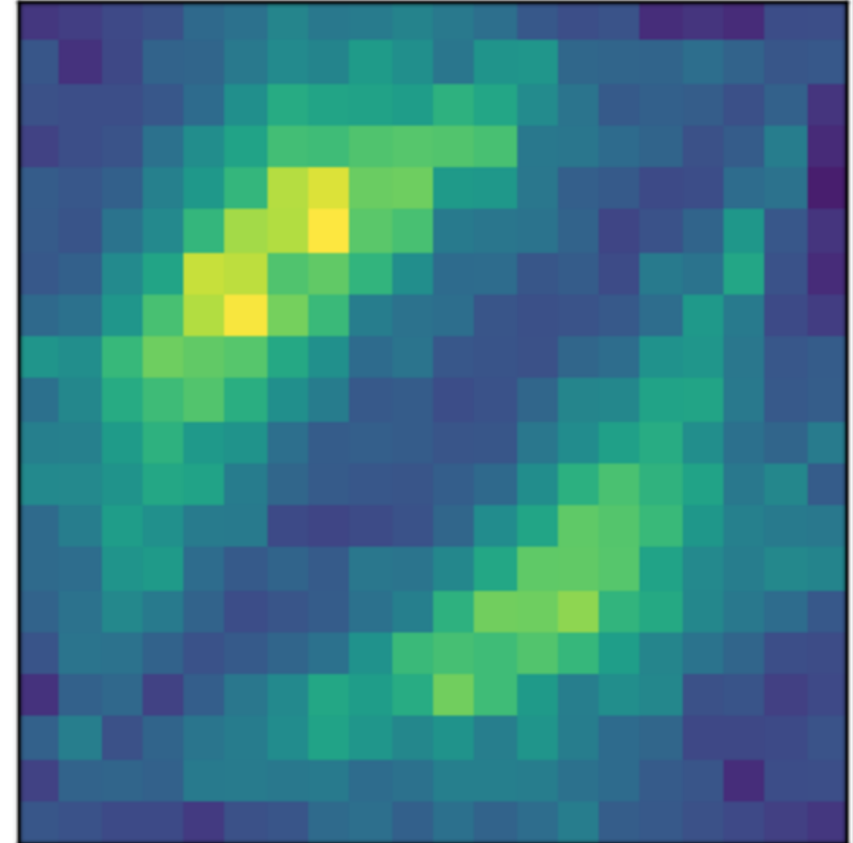
# **Principles of Perception**

# Color Choices Matter!

Jet Colormap



Viridis Colormap



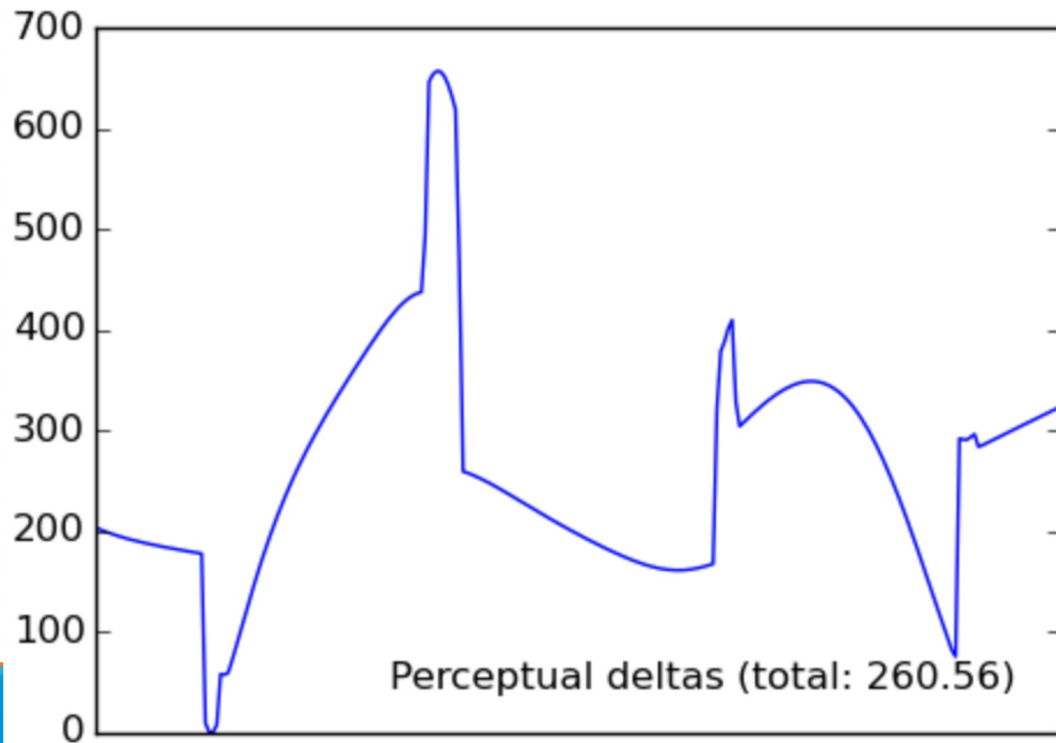
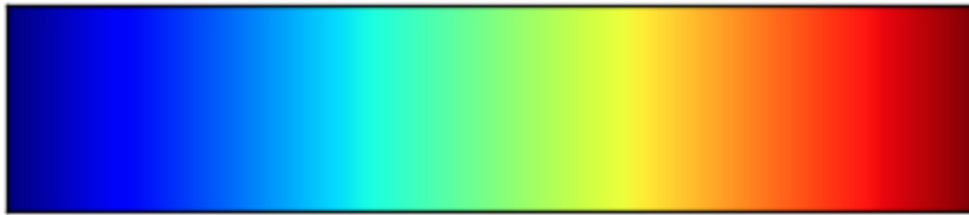
# Use a Perceptually Uniform Color Map

- Perceptually uniform:
  - Changing data from 0.1 to 0.2 appears similar to change from 0.8 to 0.9.
  - Measure by running experiments on people!
- Jet, the old matplotlib default, was far from uniform!
- Now fixed in MPL: <https://bids.github.io/colormap/> (Eric Firing et al.)
- Also, avoid red + green since many people are colorblind

# Use a Perceptually Uniform Color Map

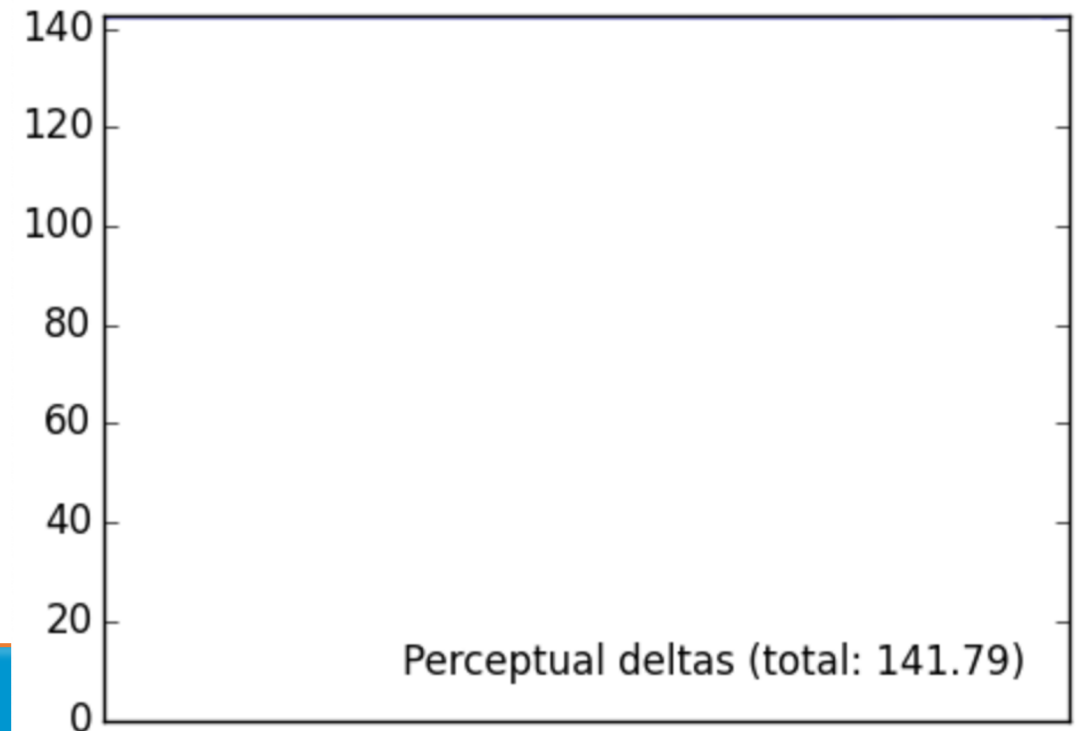
## Jet Colormap

The colormap in its glory



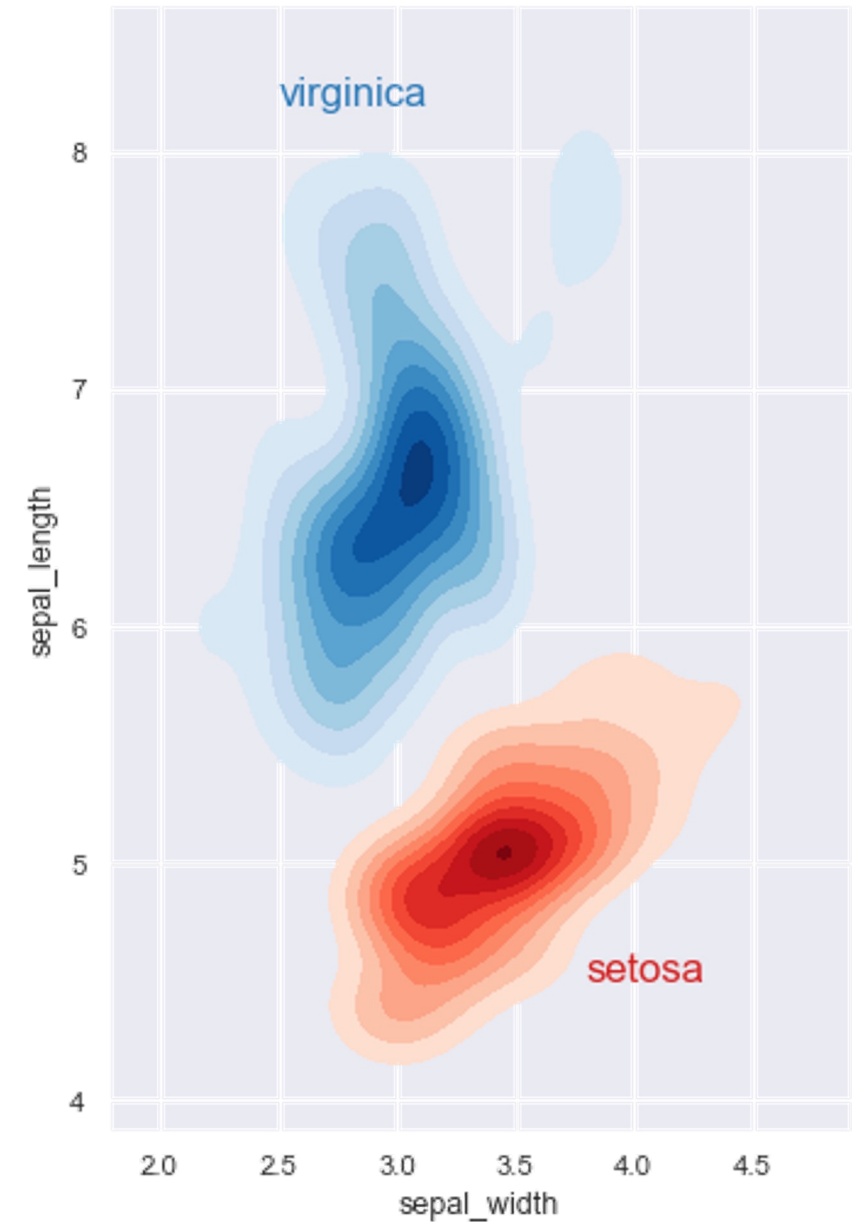
## Viridis Colormap

The colormap in its glory



# Use Color to Highlight Data Type

- Qualitative: Choose a qualitative scheme that makes it easy to distinguish between categories
- Quantitative: Choose a color scheme that implies magnitude.
- Plot on right has both!



# Use Color to Highlight Data Type

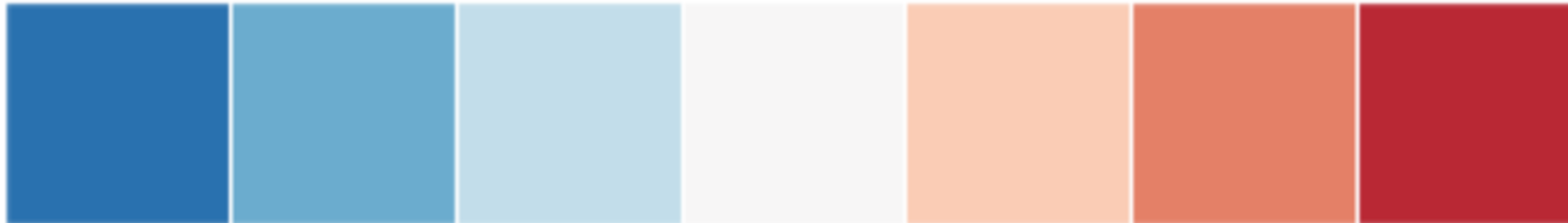
- Does the data progress from low to high?
- Use a sequential scheme where light colors are for more extreme values



# Use Color to Highlight Data Type

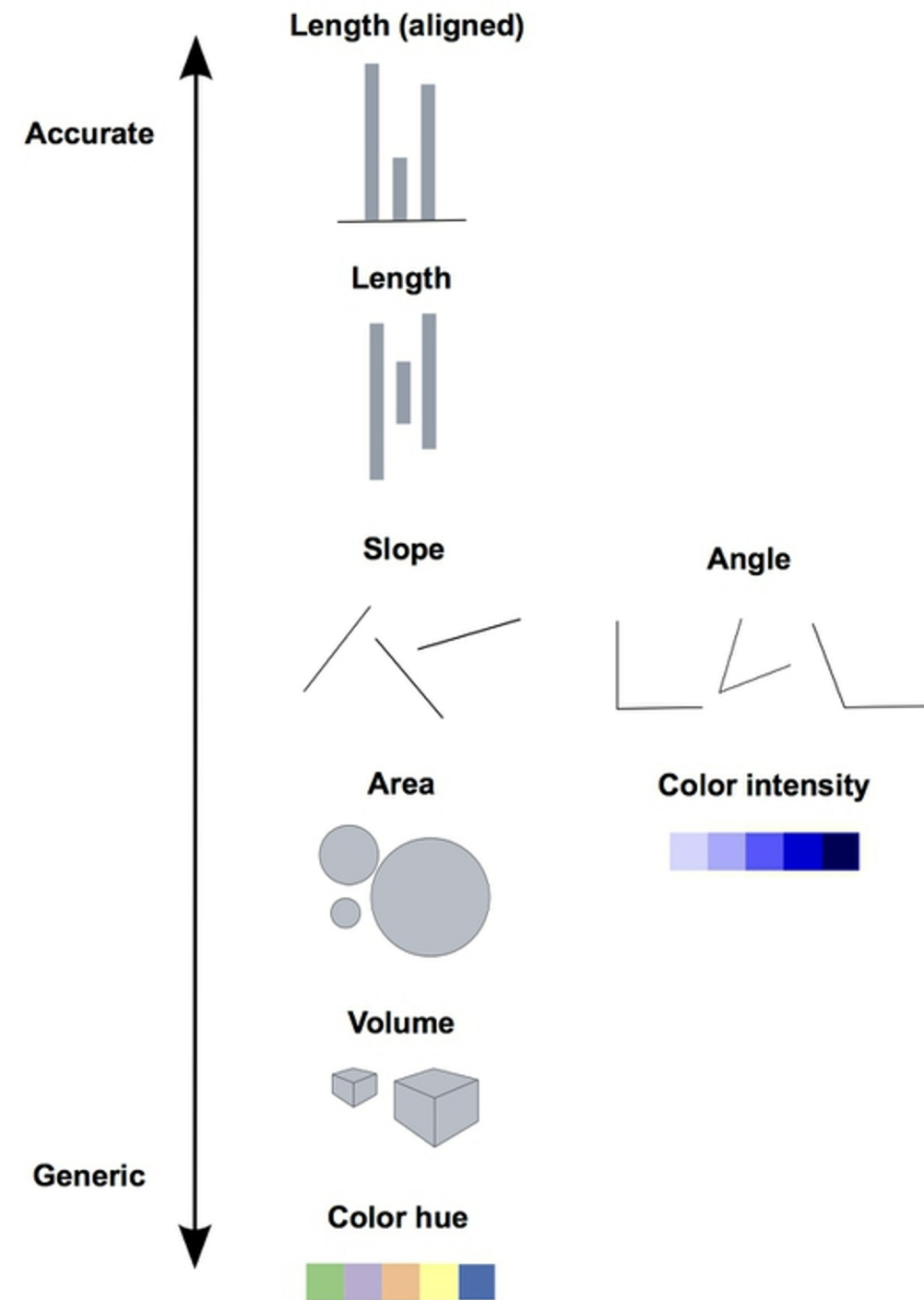
- Do both low and high value deserve equal emphasis? Use a diverging scheme where light colors represent middle values

```
sns.palplot(sns.color_palette("RdBu_r", 7))
```



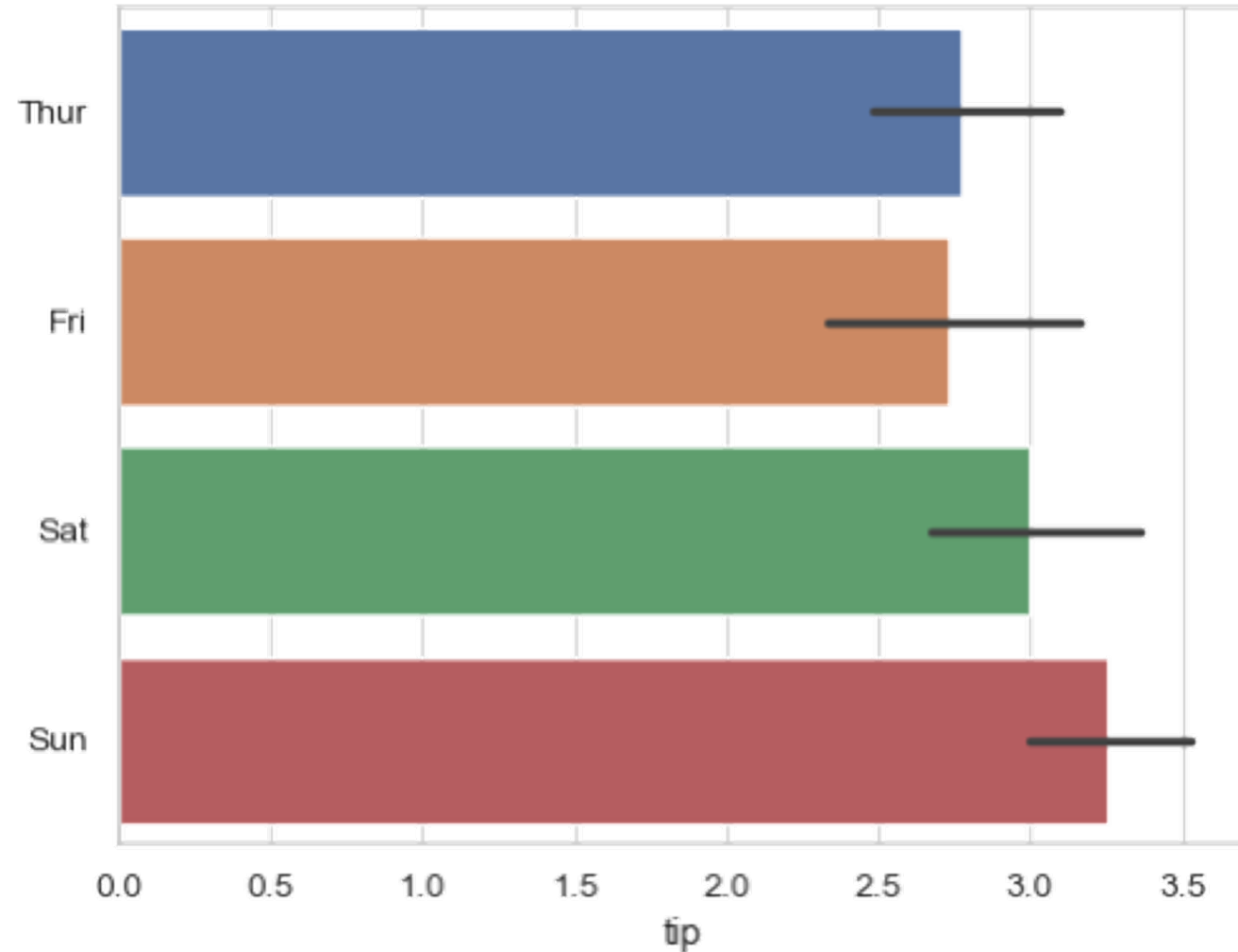
# Not All Marks Are Good!

- Accuracy of judgements depend on the type of mark
- Aligned lengths most accurate
- Color least accurate



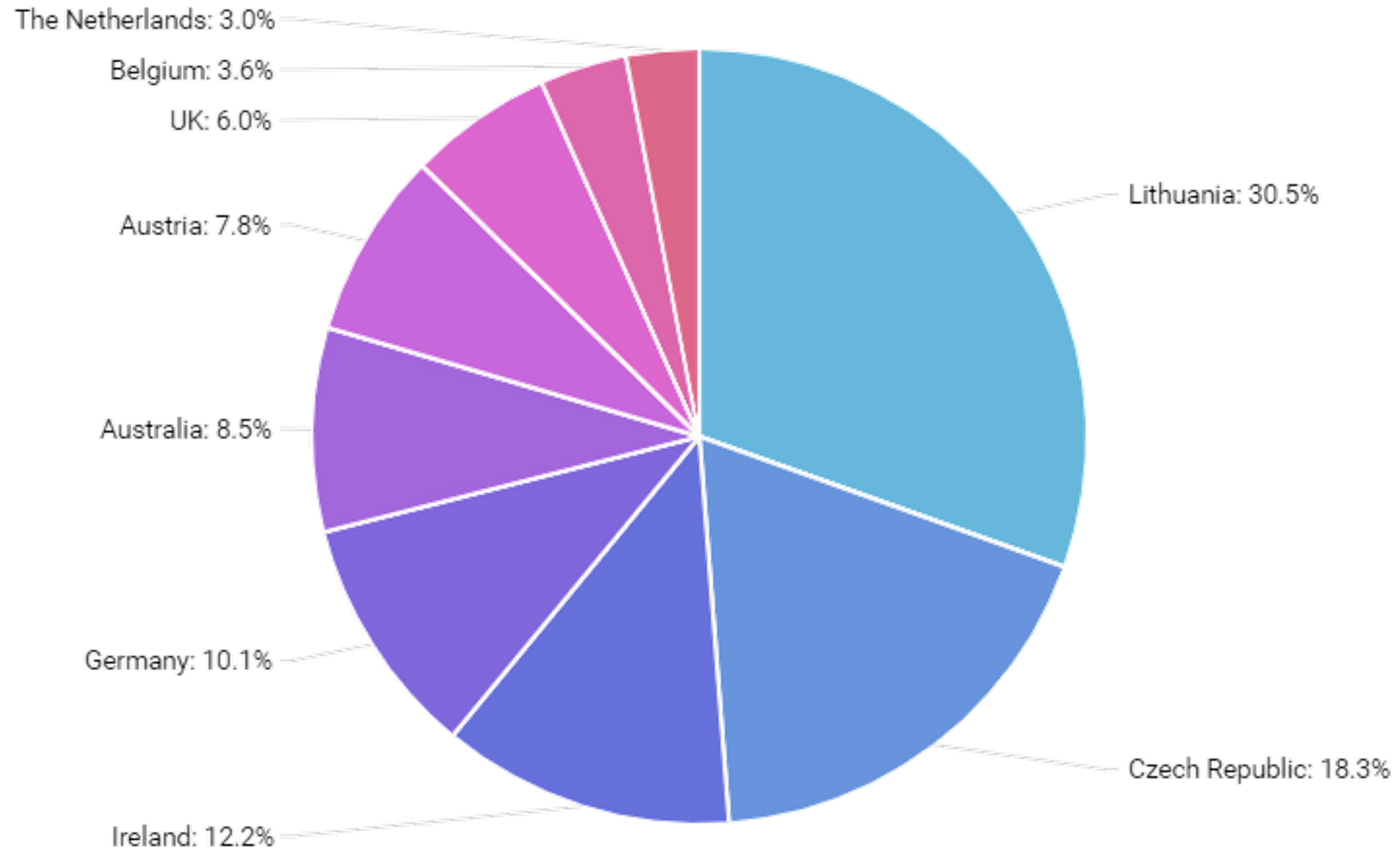
# Lengths are Easy to Understand

- People can easily distinguish two different lengths
- E.g. Heights of bars in bar chart



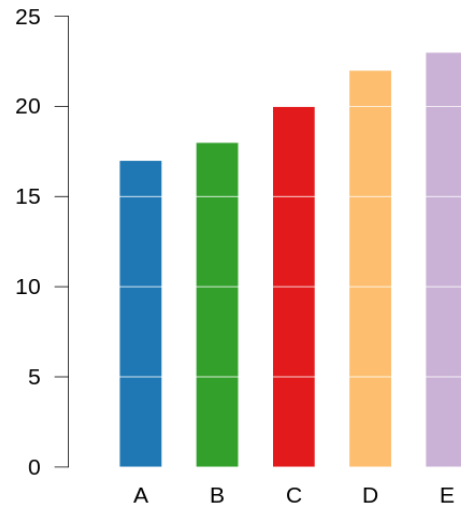
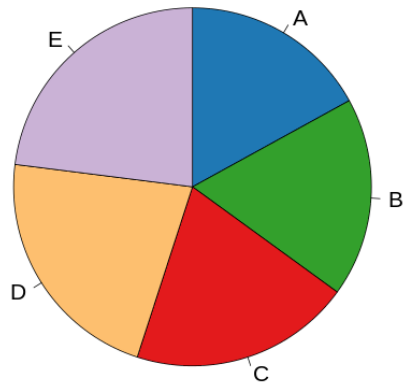
# Angles are Hard to Understand

- Avoid pie charts!
- Angle judgements are inaccurate
- In general, underestimate size of larger angle

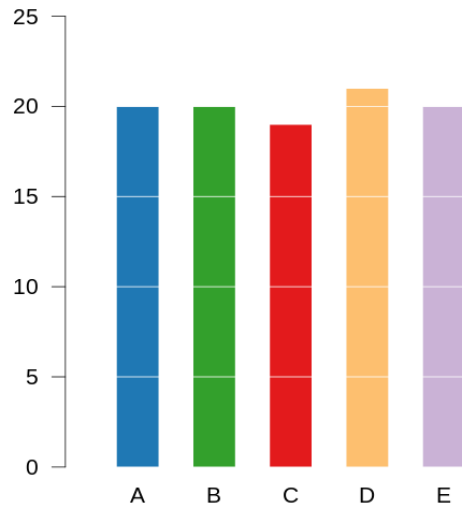
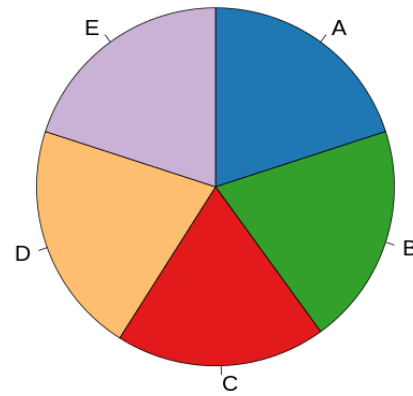


# Bar vs Pie Chart

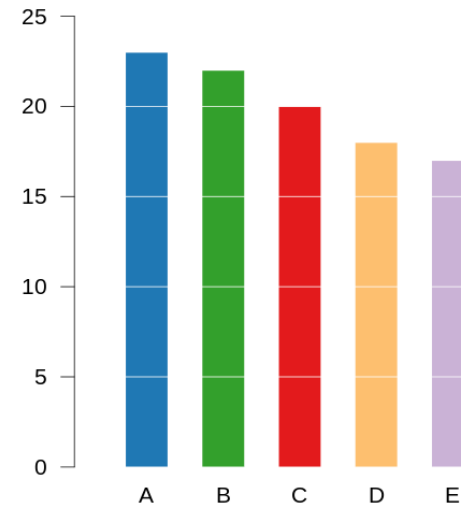
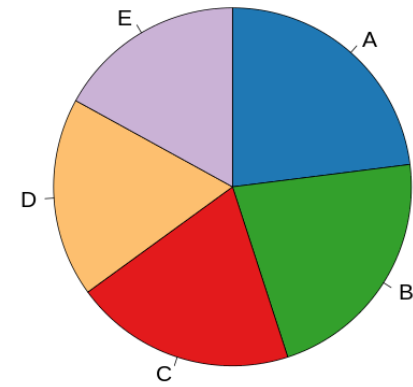
1st



2nd



3rd



# Areas are Hard to Understand

- Avoid area charts!
- Area judgements are inaccurate
- In general, underestimate size of larger area

## African Countries by GDP

### TOP COUNTRIES BY GDP IN U.S. \$ BILLIONS

Gross domestic product (GDP) refers to the market value of all final goods and services produced within a country in a given period (2005 – 2009).

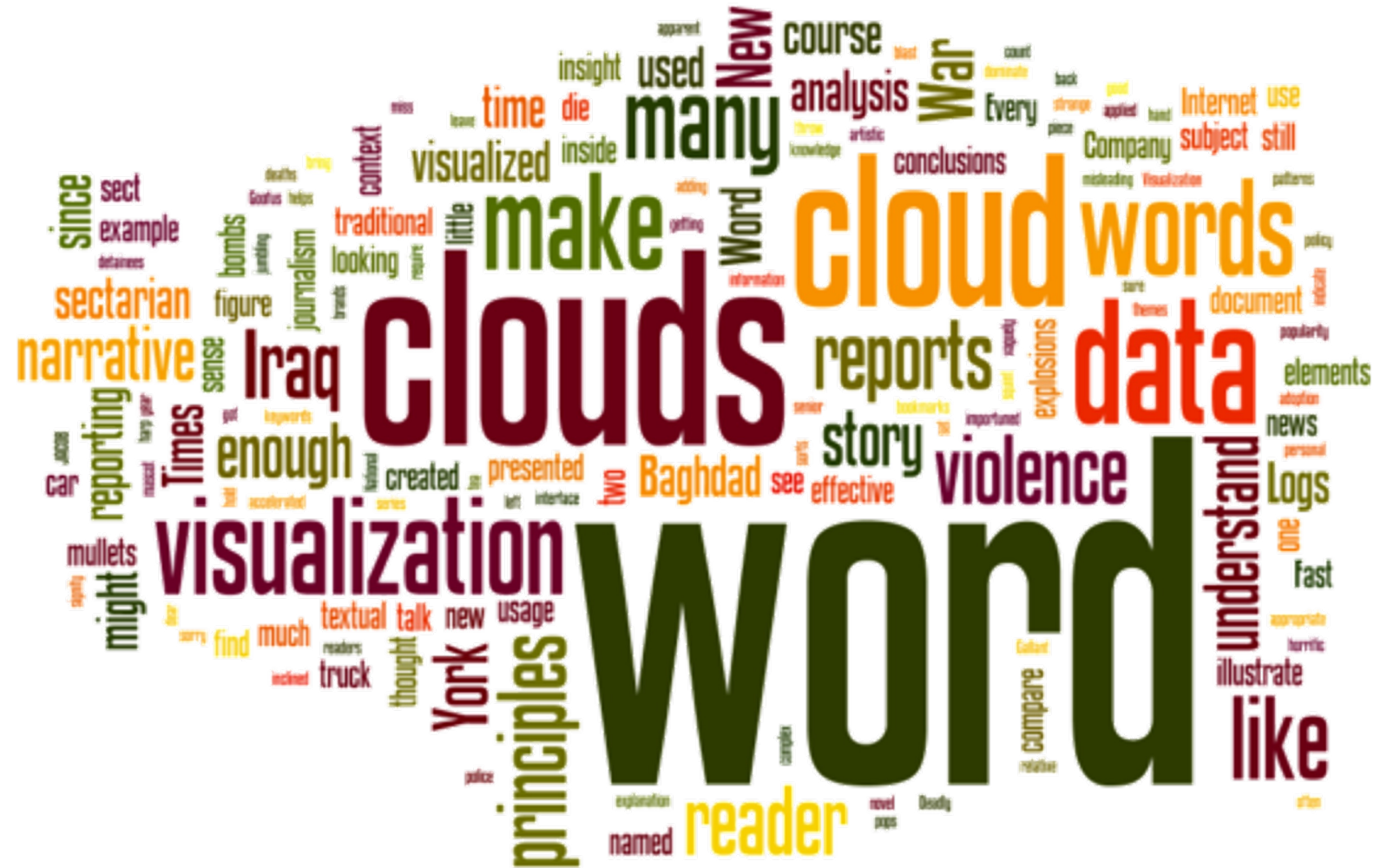
### GDP CALCULATION

private consumption + gross investment + government spending + (exports - imports)



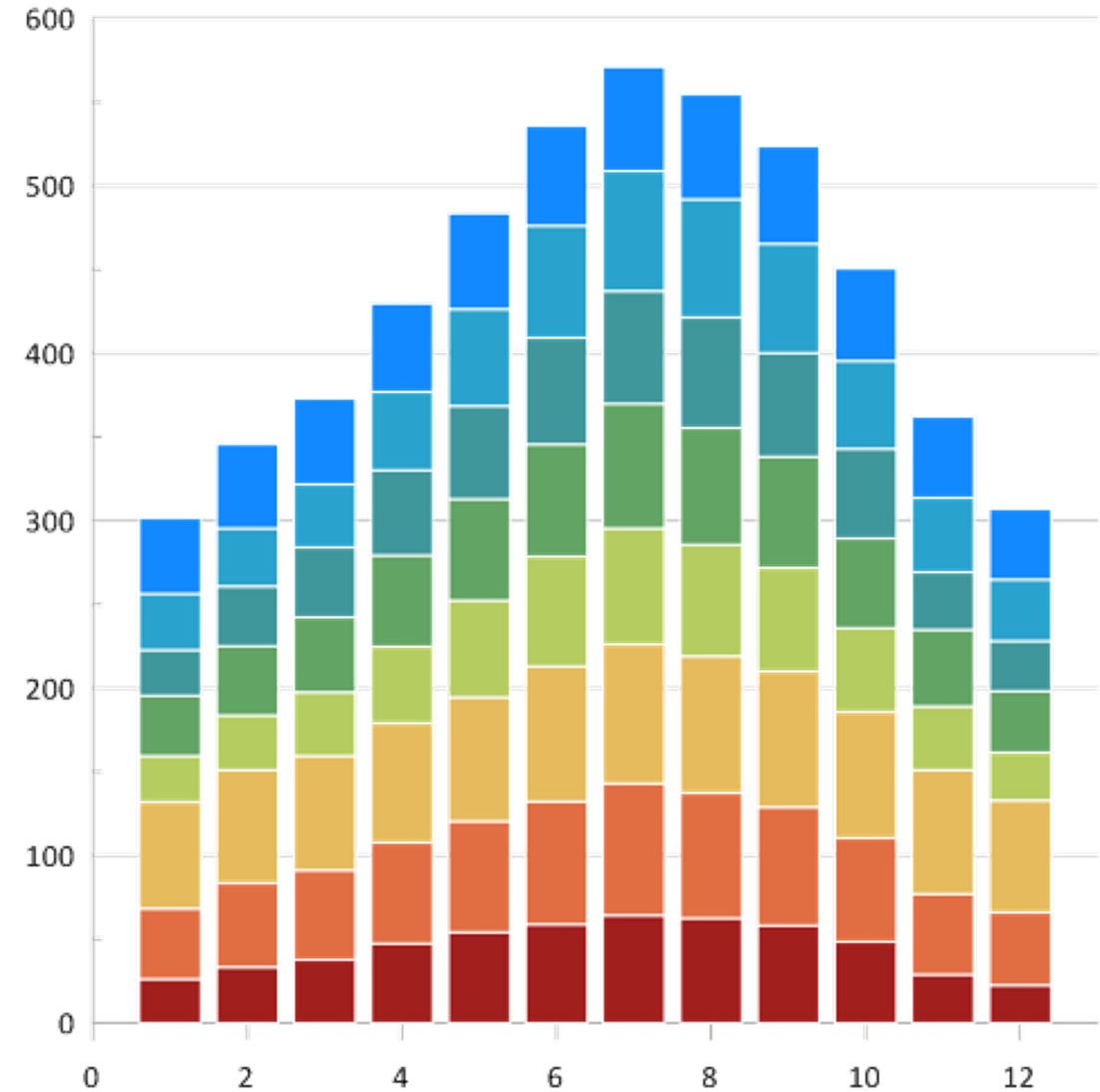
# Areas are Hard to Understand

- Avoid word clouds!
- Hard to tell the “area” taken up by a word

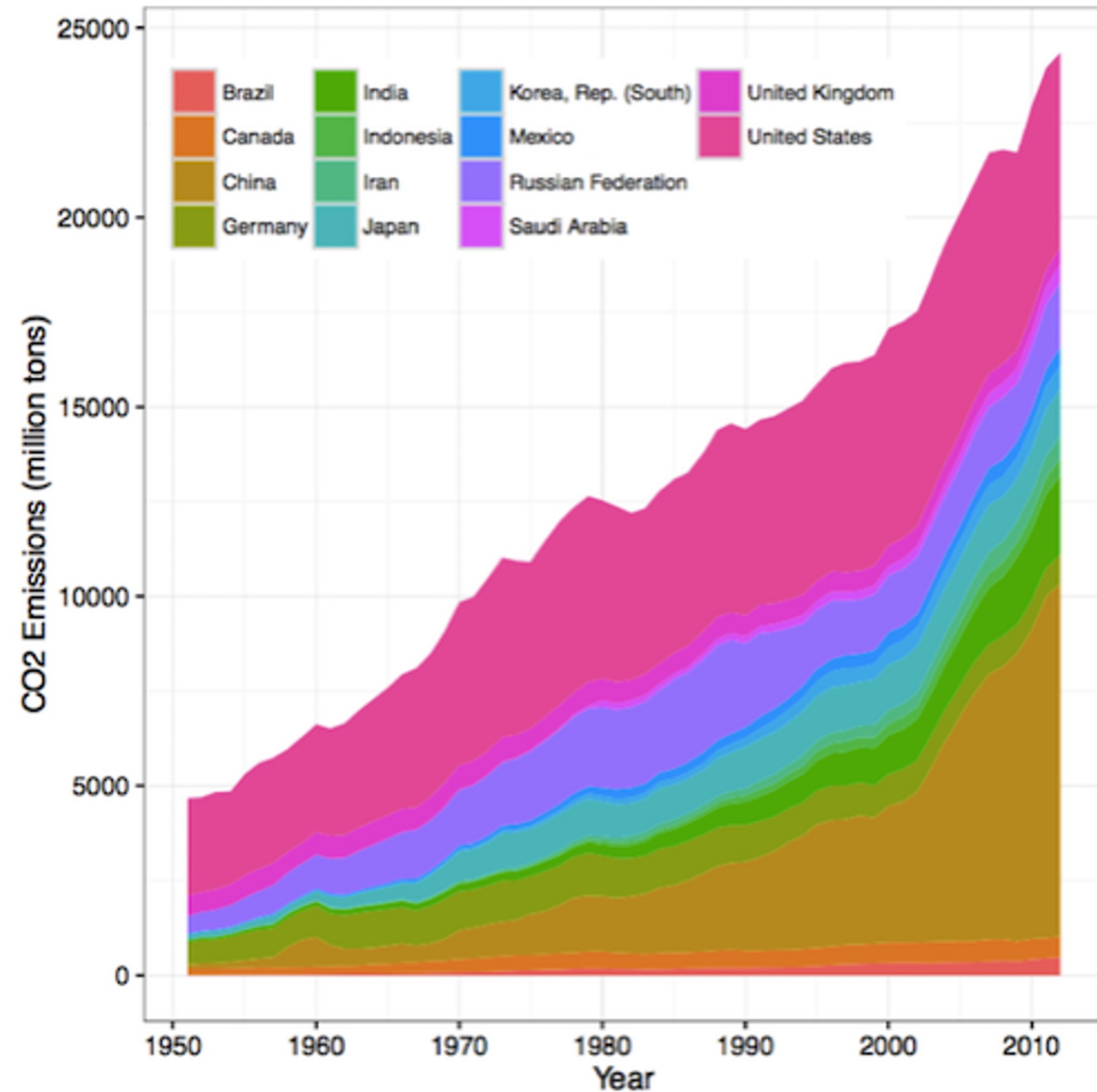


# Avoid Jiggling Baseline

- Stacked bar charts / histograms hard to read because baseline moves
- Notice that top bars are all about the same height

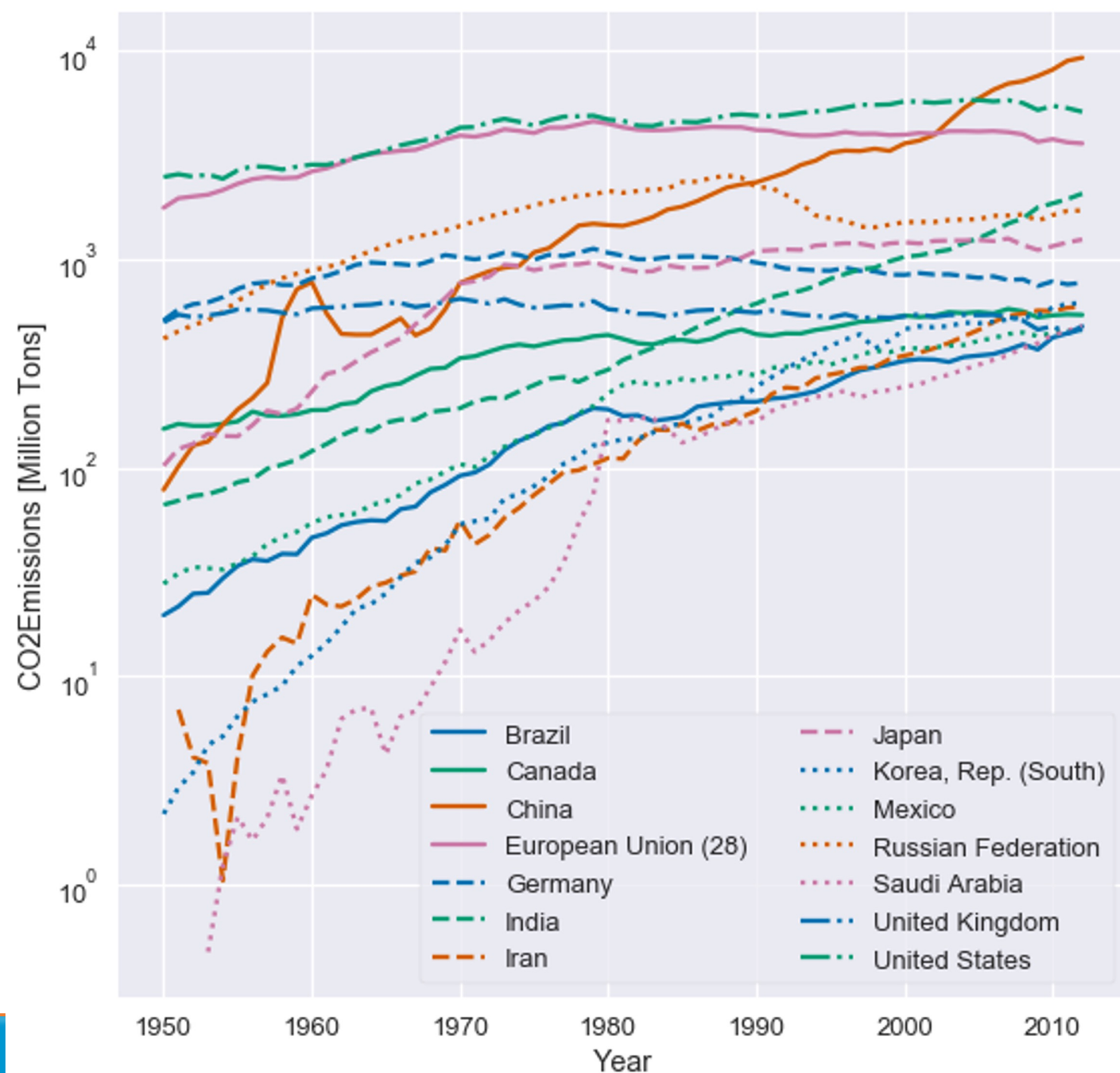


- Stacked area charts hard to read because baseline moves



# Avoid Jiggling Baseline

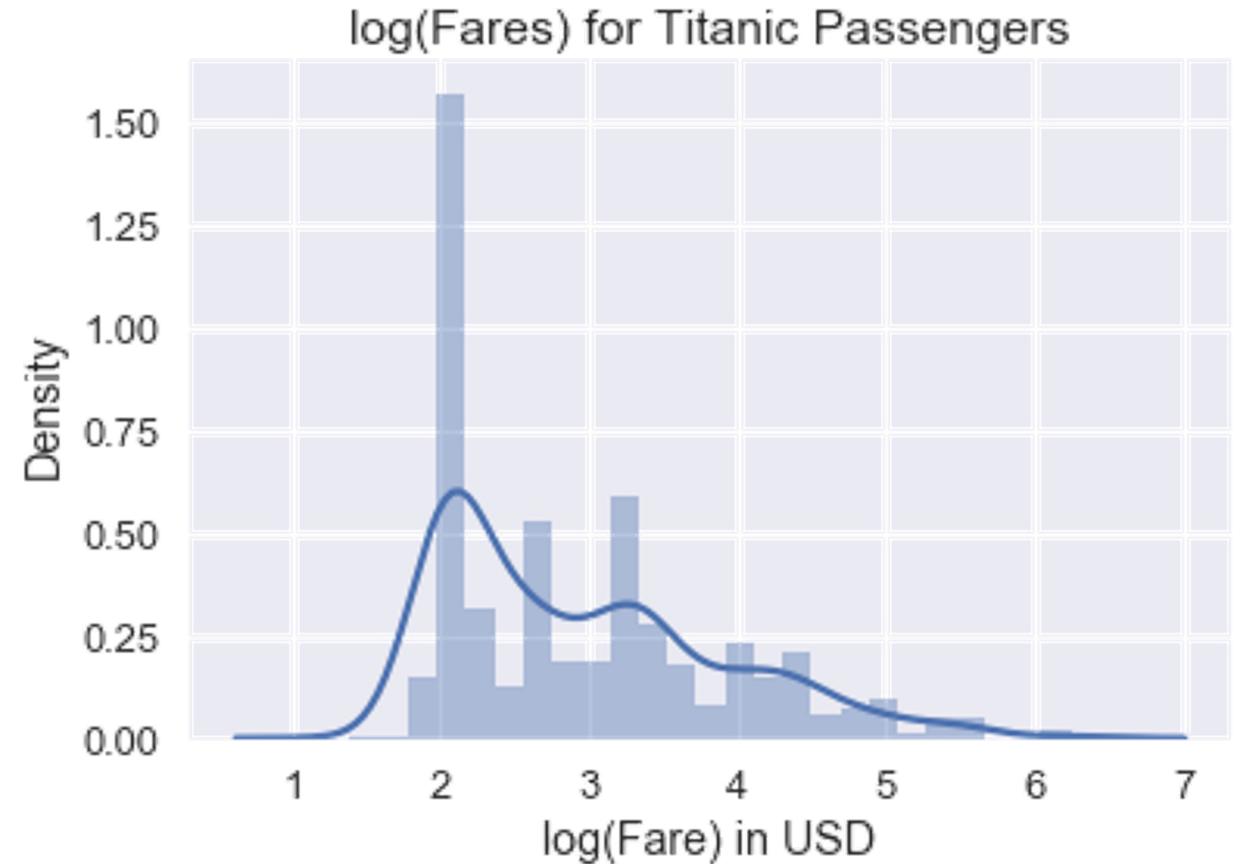
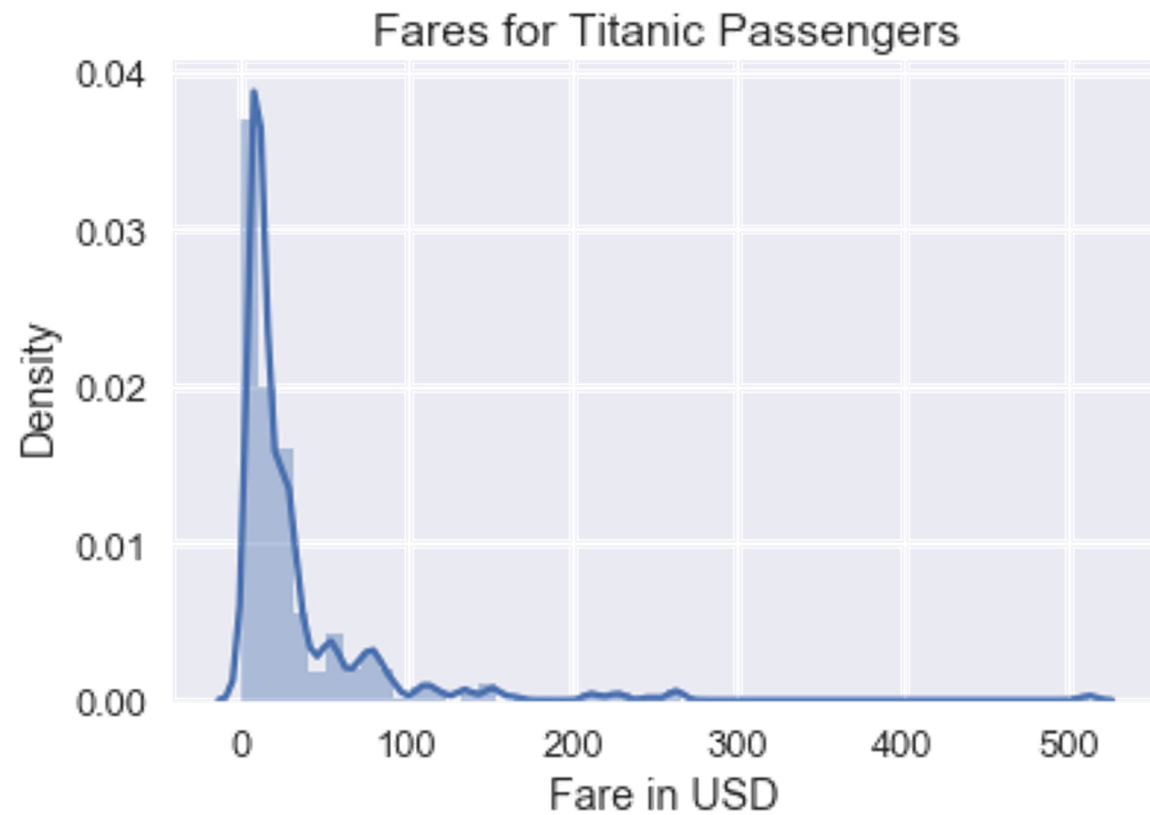
- Instead, plot lines themselves



# **Principles of Transformation**

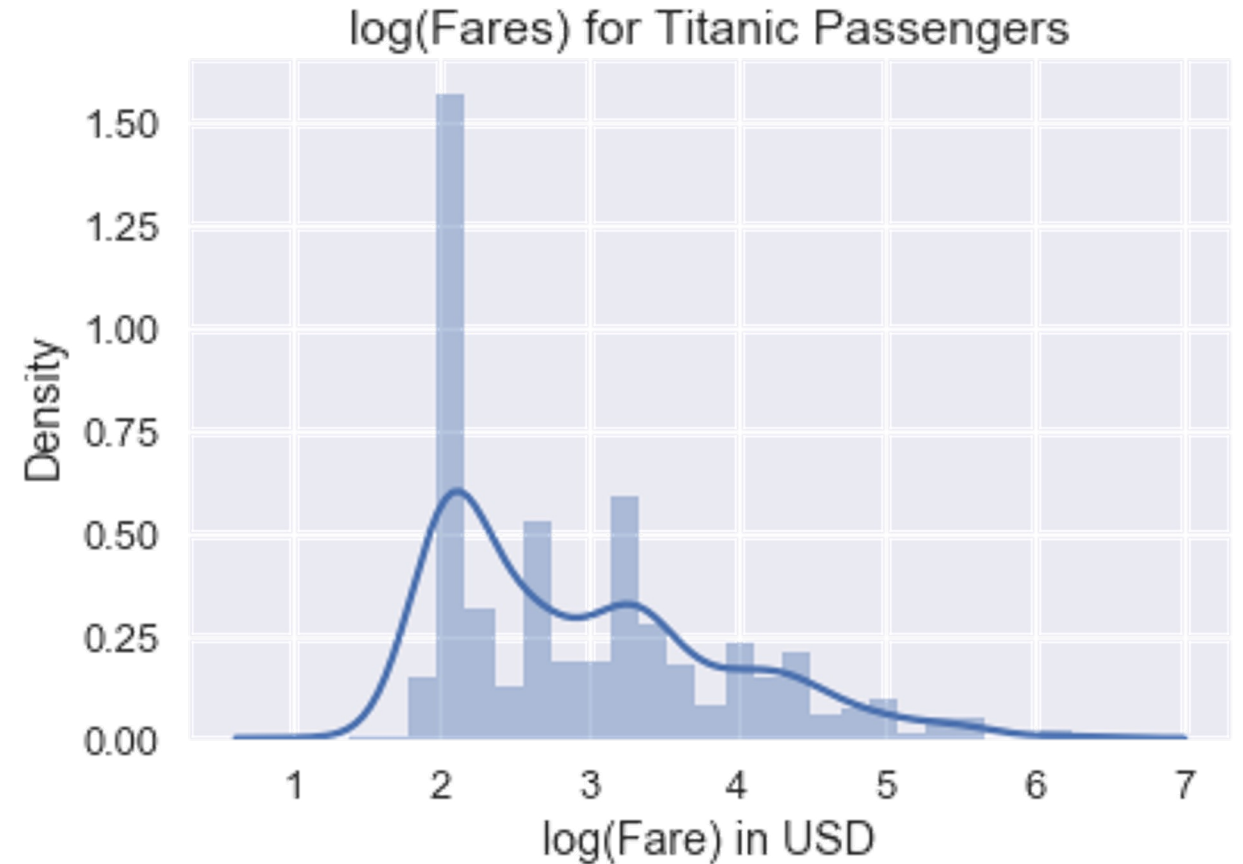
# Transforming Data Can Reveal Patterns

- When data are heavy tailed, useful to take the log and replot



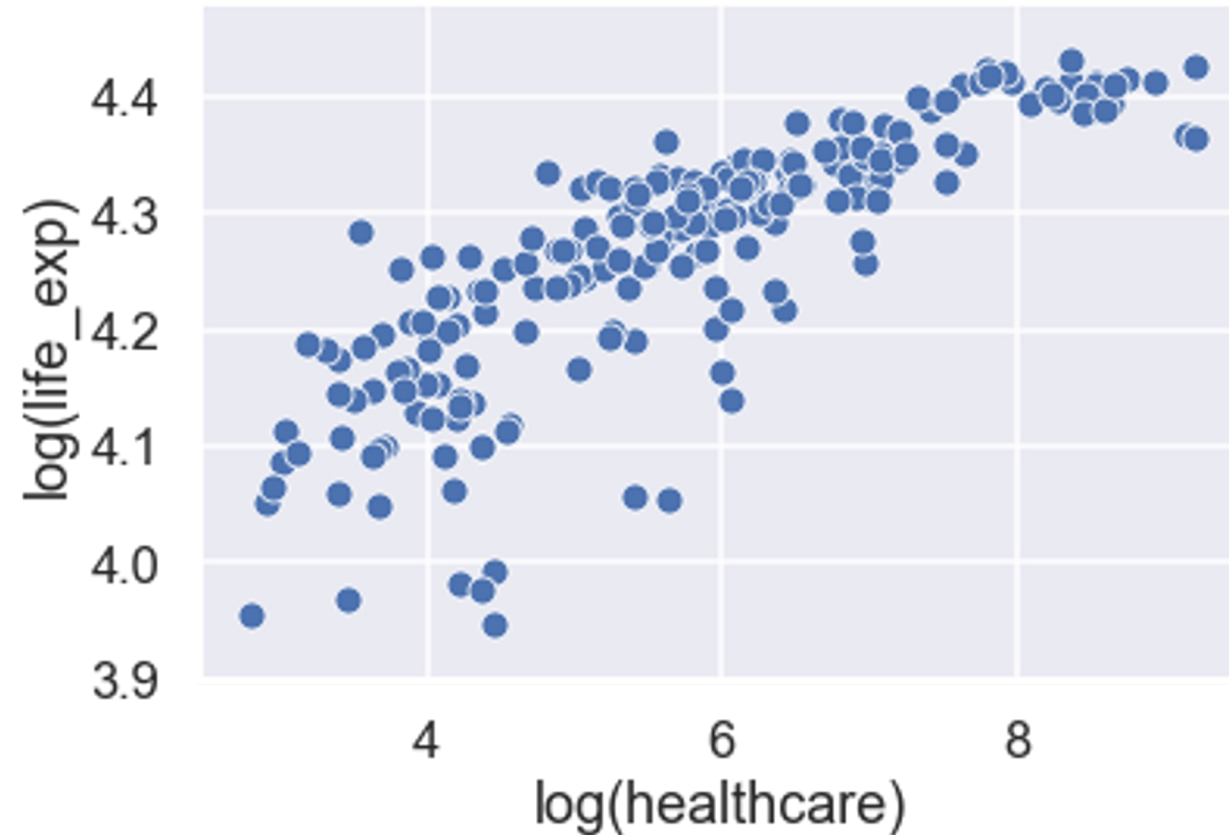
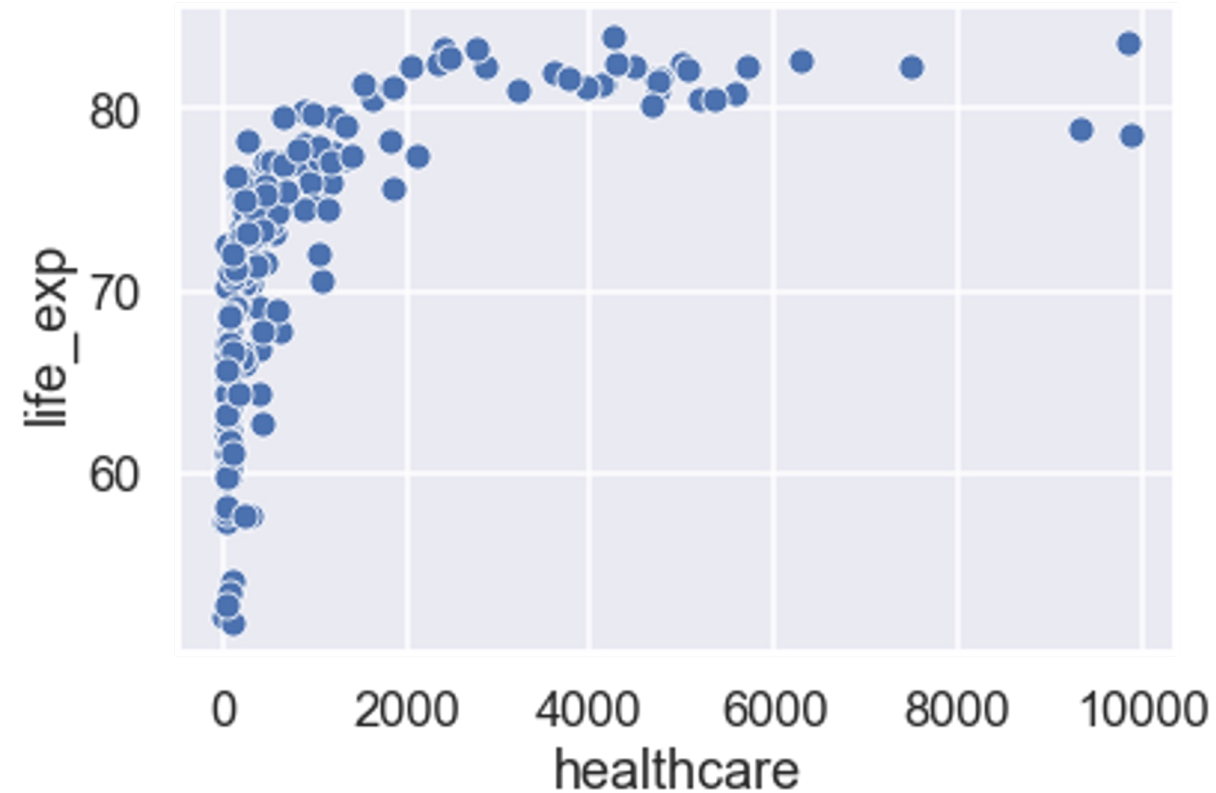
# Transforming Data Can Reveal Patterns

- Shows a mode when  $\log(\text{fare}) = 2$  and a smaller mode at 3.4.
- What do these correspond to in actual dollars?
- $\exp(2) = \$7.4$
- $\exp(3.4) = \$30$



# Transforming Data Can Reveal Patterns

- Log of nonlinear data can reveal pattern in scatter plot!



# Log of y-values

- Fit line to log of y-values:

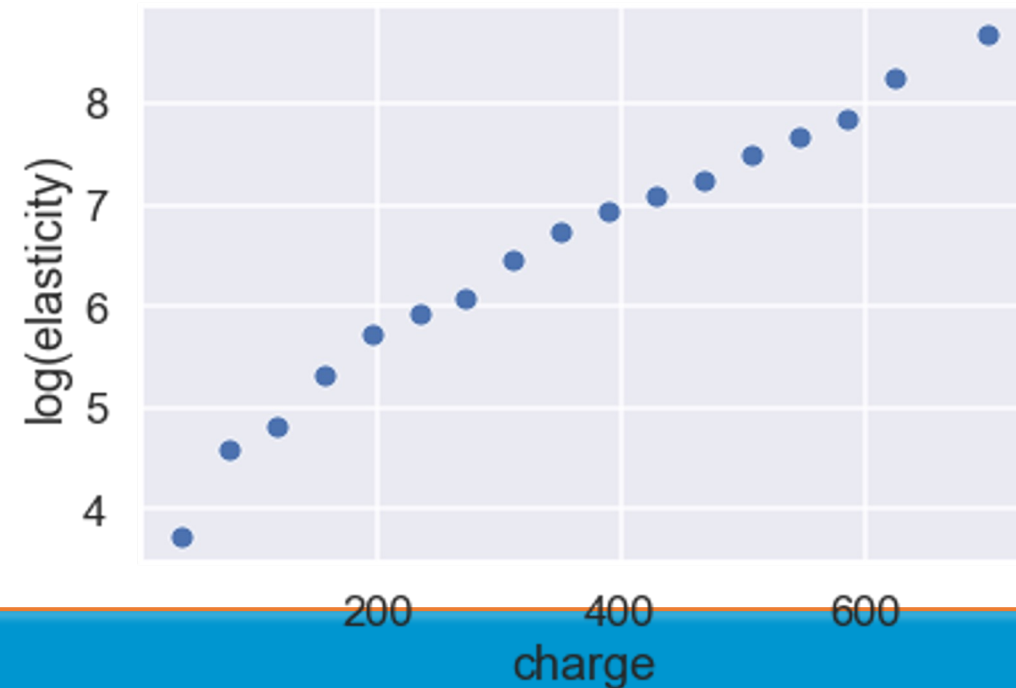
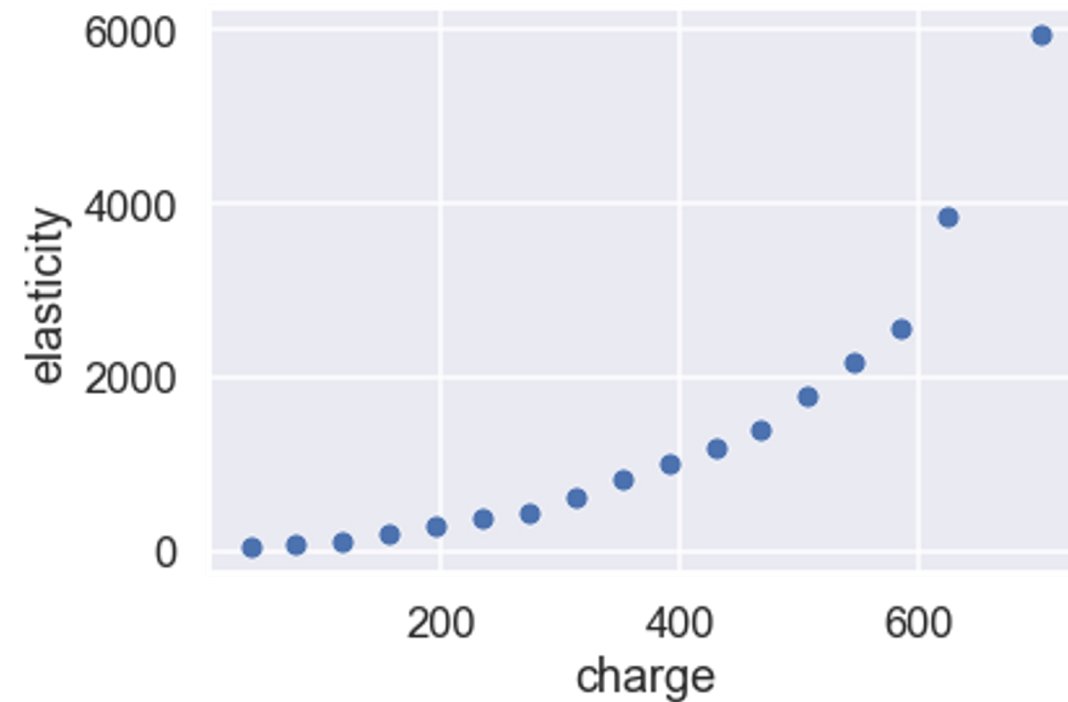
$$\log y = ax + b$$

$$y = e^{ax+b}$$

$$y = e^{ax} e^b$$

$$y = Ce^{ax}$$

- Linear relationship after log of y-values implies exponential model for original plot



# Log of both x and y-values

- Fit line to log of x and y-values:

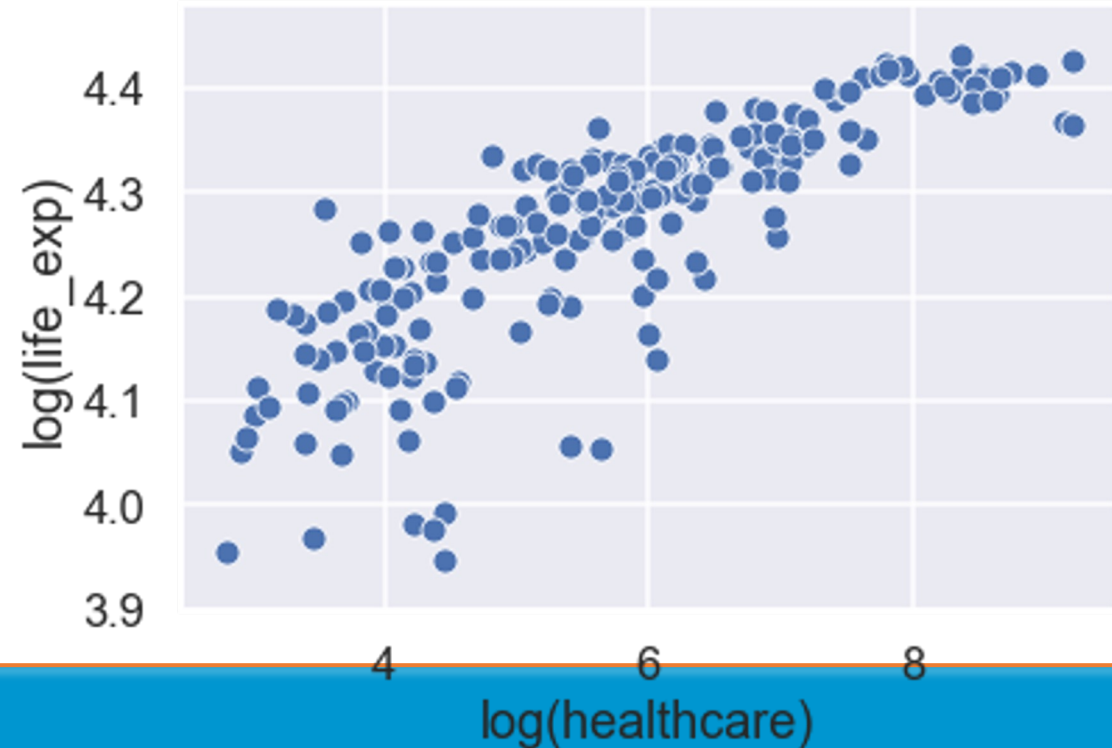
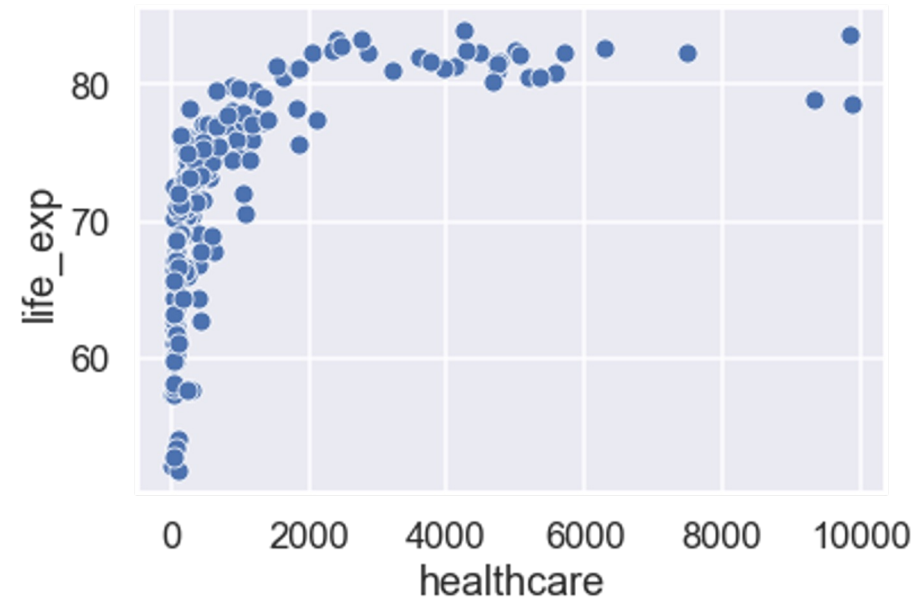
$$\log y = a \cdot \log x + b$$

$$y = e^{a \cdot \log x + b}$$

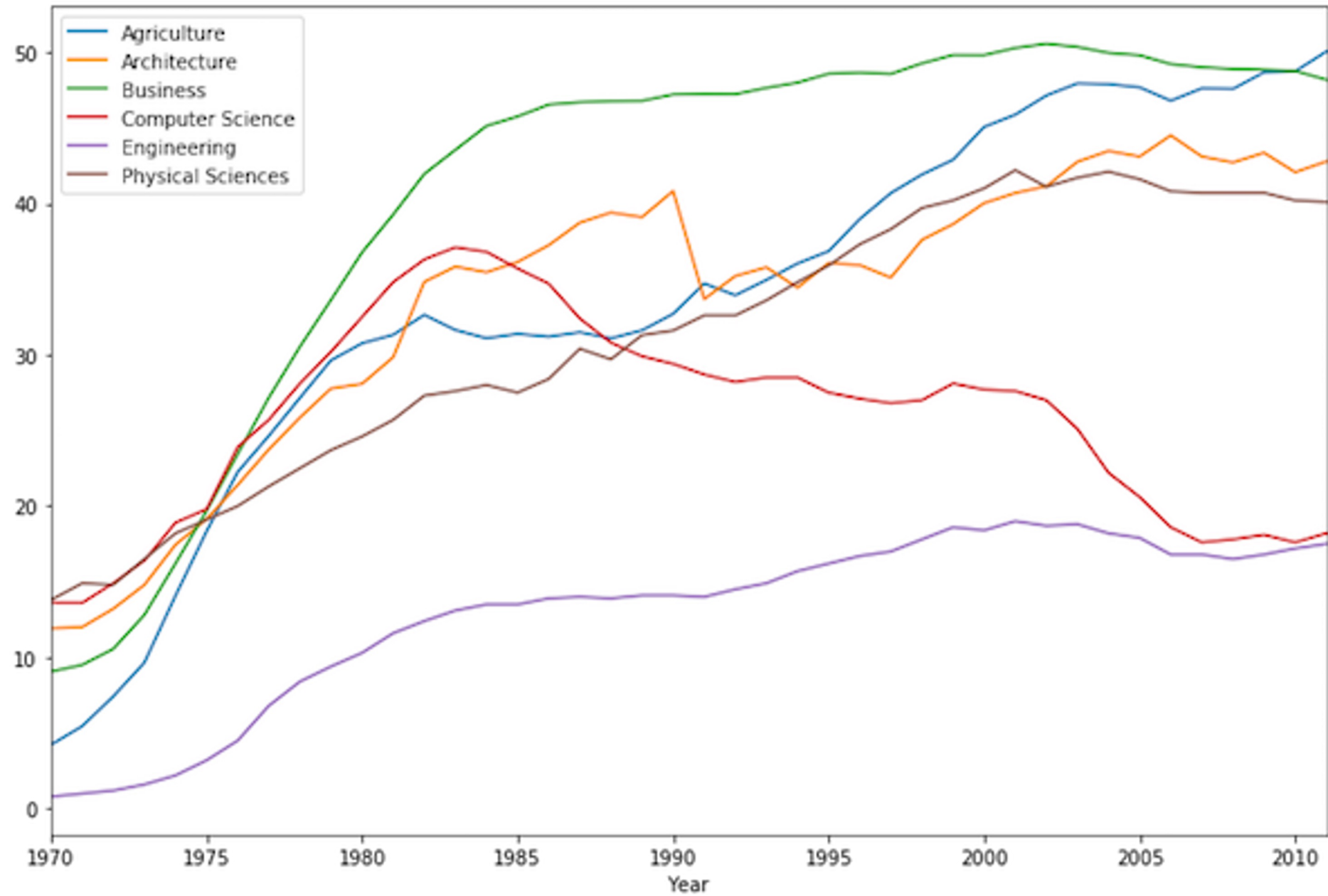
$$y = C e^{a \cdot \log x}$$

$$y = C x^a$$

- Linear relationship after log of x and y-values implies polynomial model for original plot

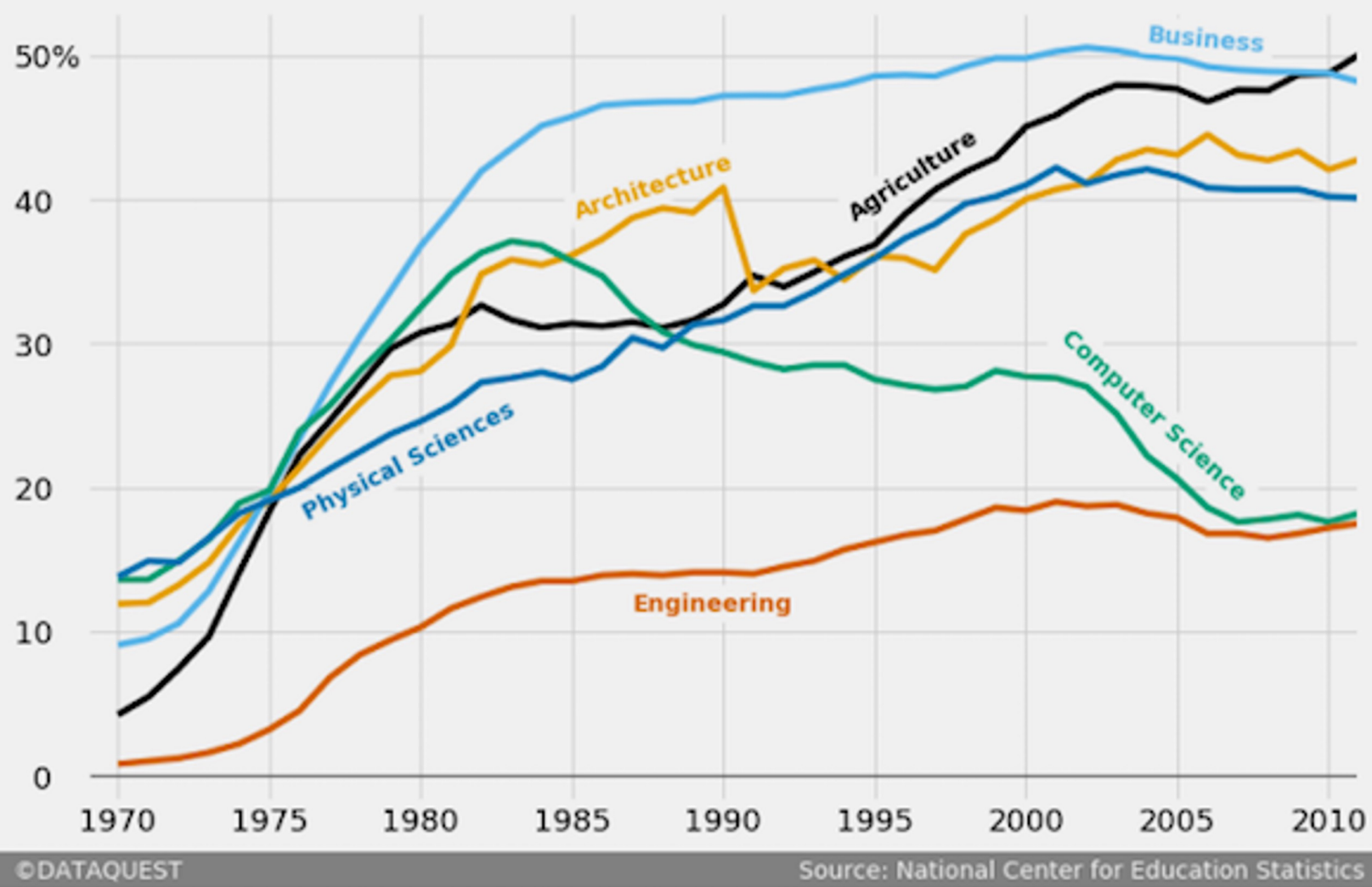


# Principles of Context



## The gender gap is transitory - even for extreme cases

Percentage of Bachelors conferred to women from 1970 to 2011 in the US for extreme cases where the percentage was less than 20% in 1970



# Add Context Directly to Plot

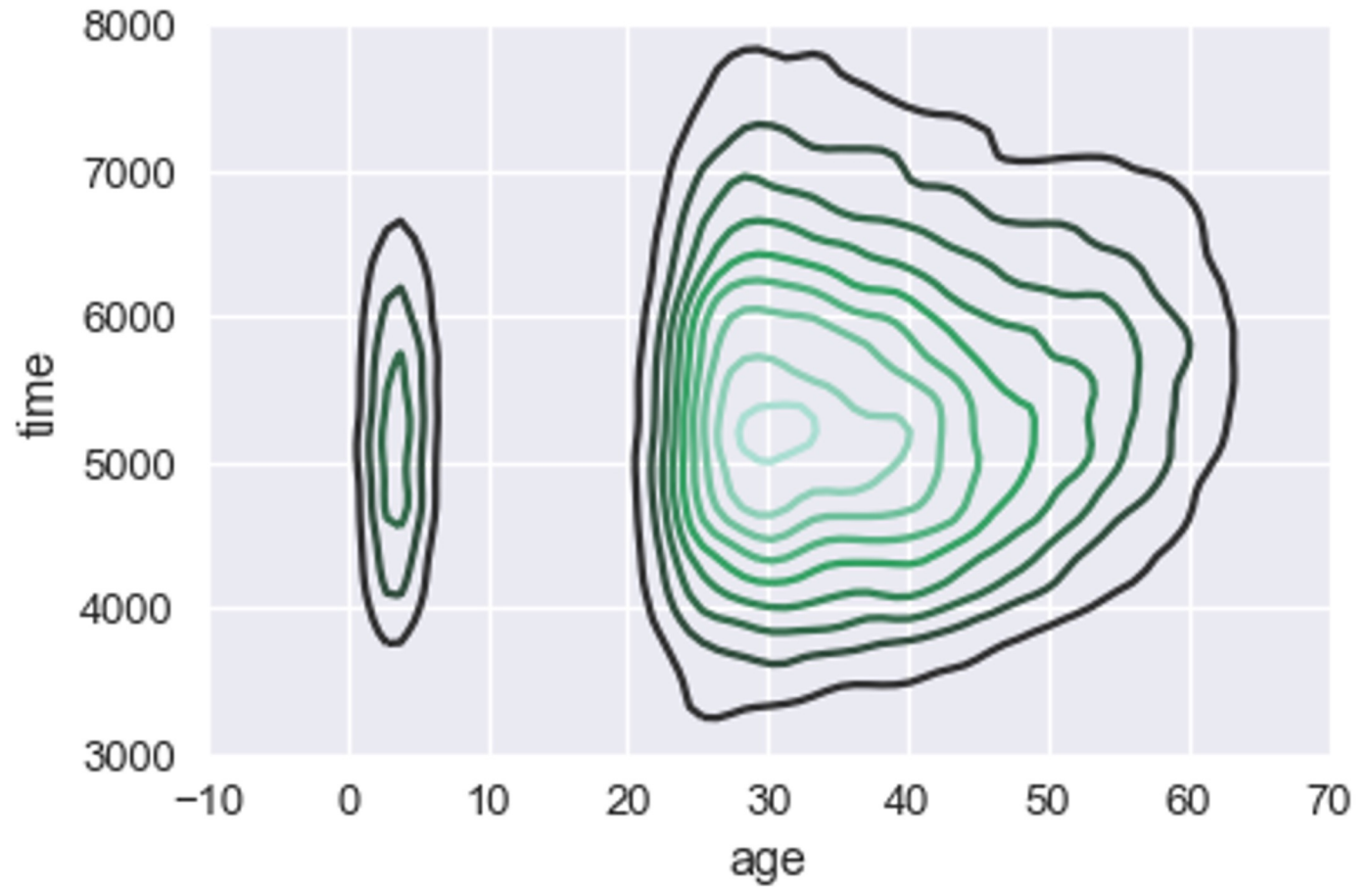
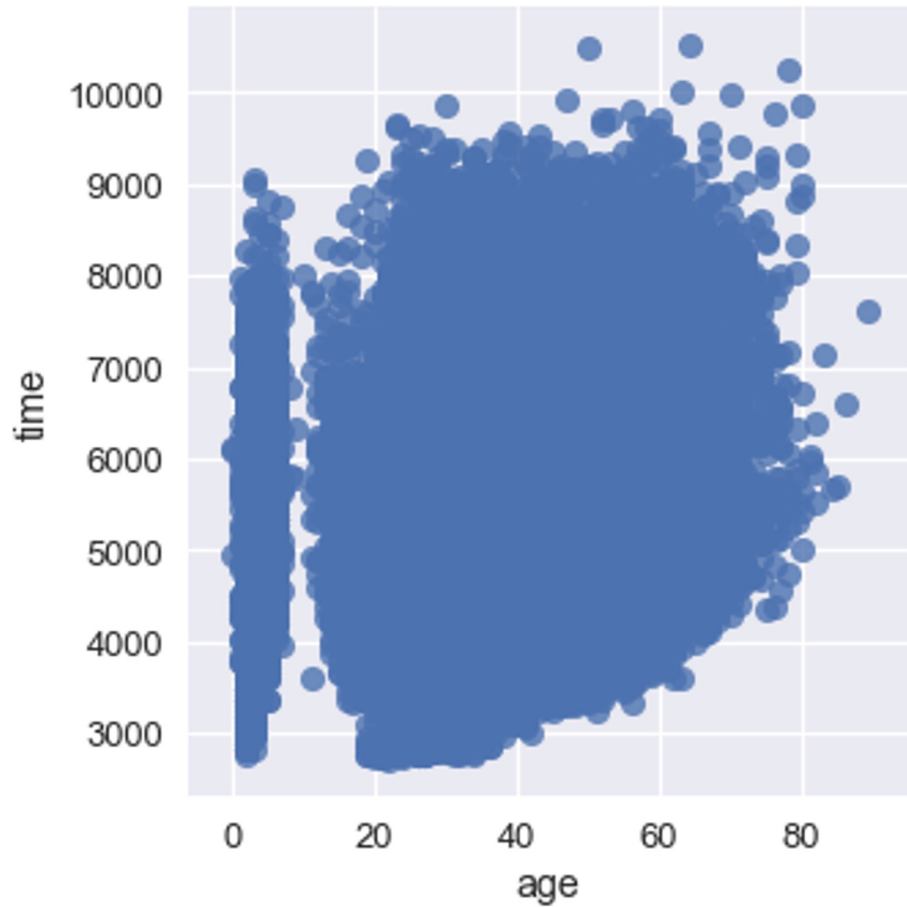
## A publication-ready plot needs:

- Informative title (takeaway, not description)
  - “Older passengers spend more on plane tickets” instead of “Scatter plot of price vs. age”.
- Axis labels
- Reference lines and markers for important values
- Labels for unusual points
- Captions that describe data

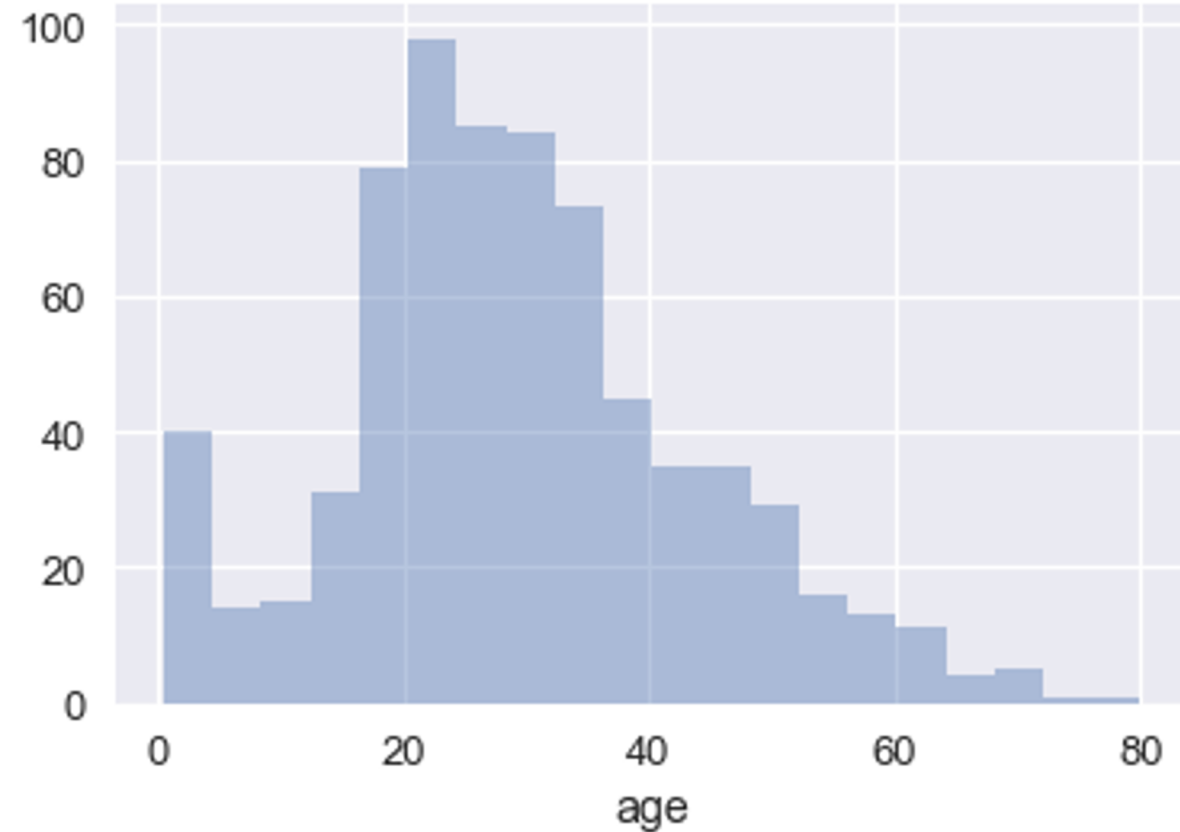
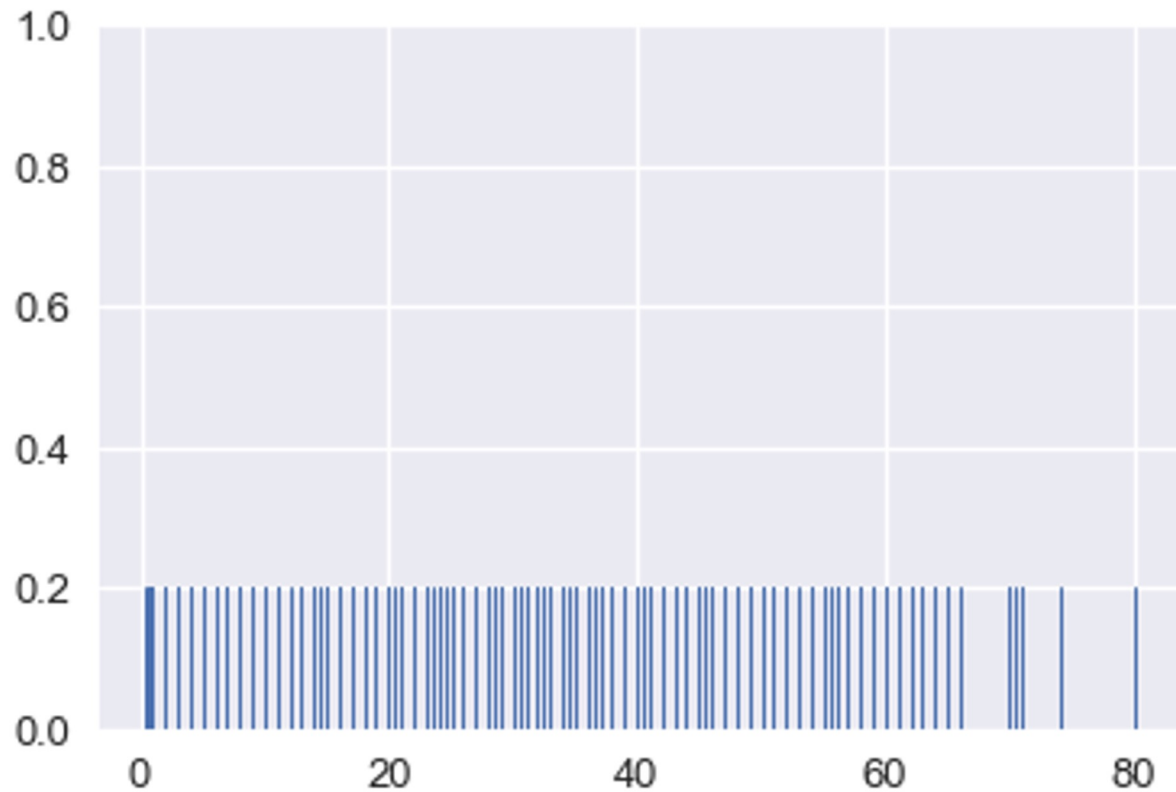
# **Principles of Smoothing**



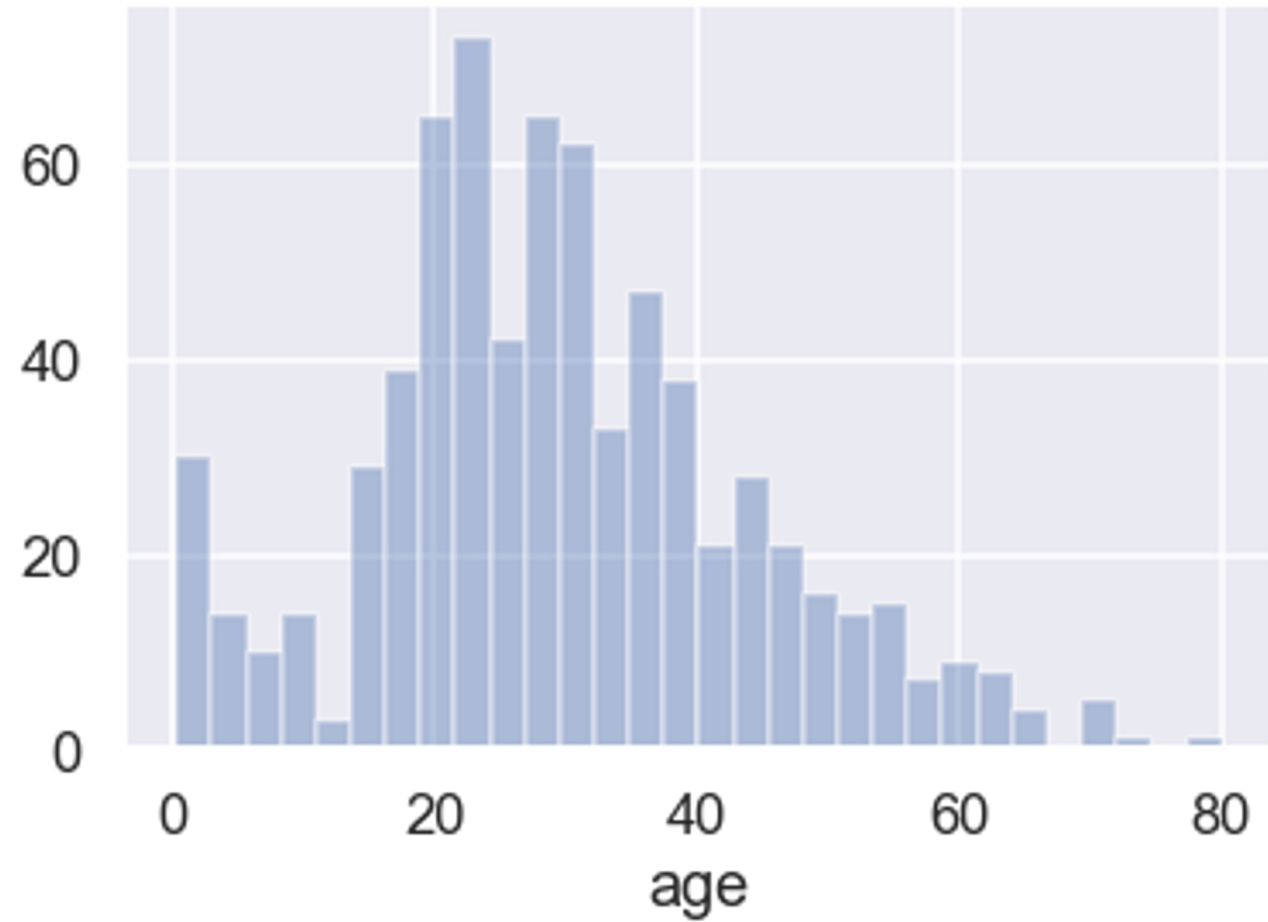
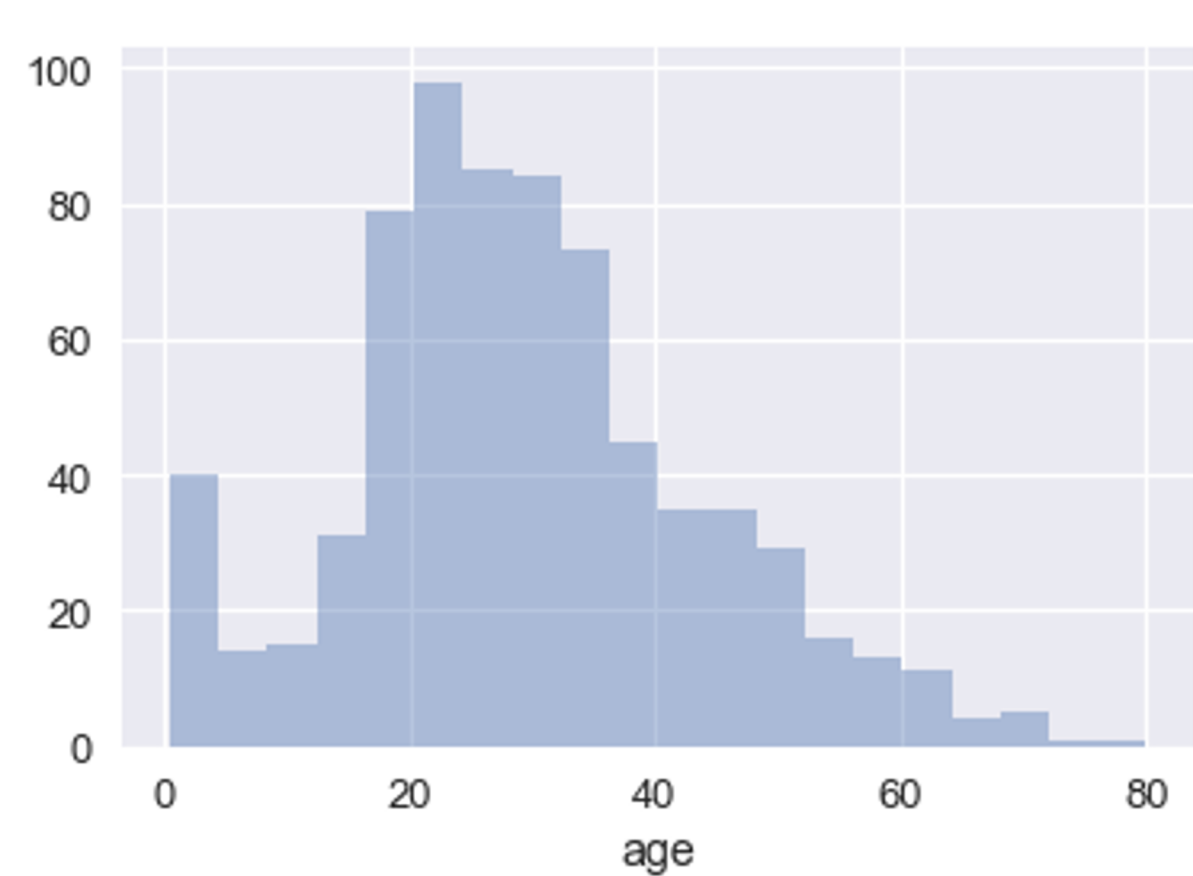
# Apply Smoothing for Large Datasets



# A Histogram is a Smoothed Rug Plot

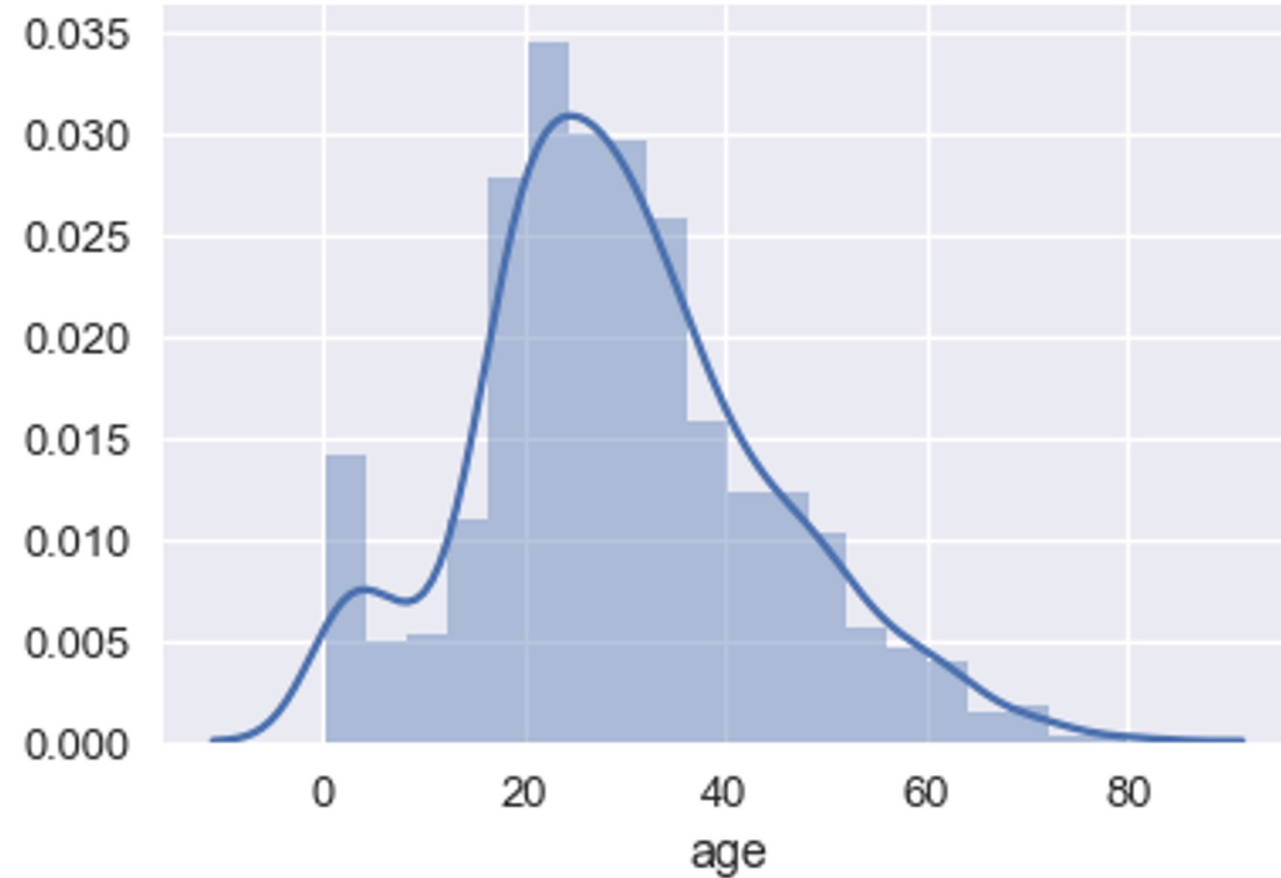


# Smoothing Needs Tuning



# Kernel Density Estimation (KDE)

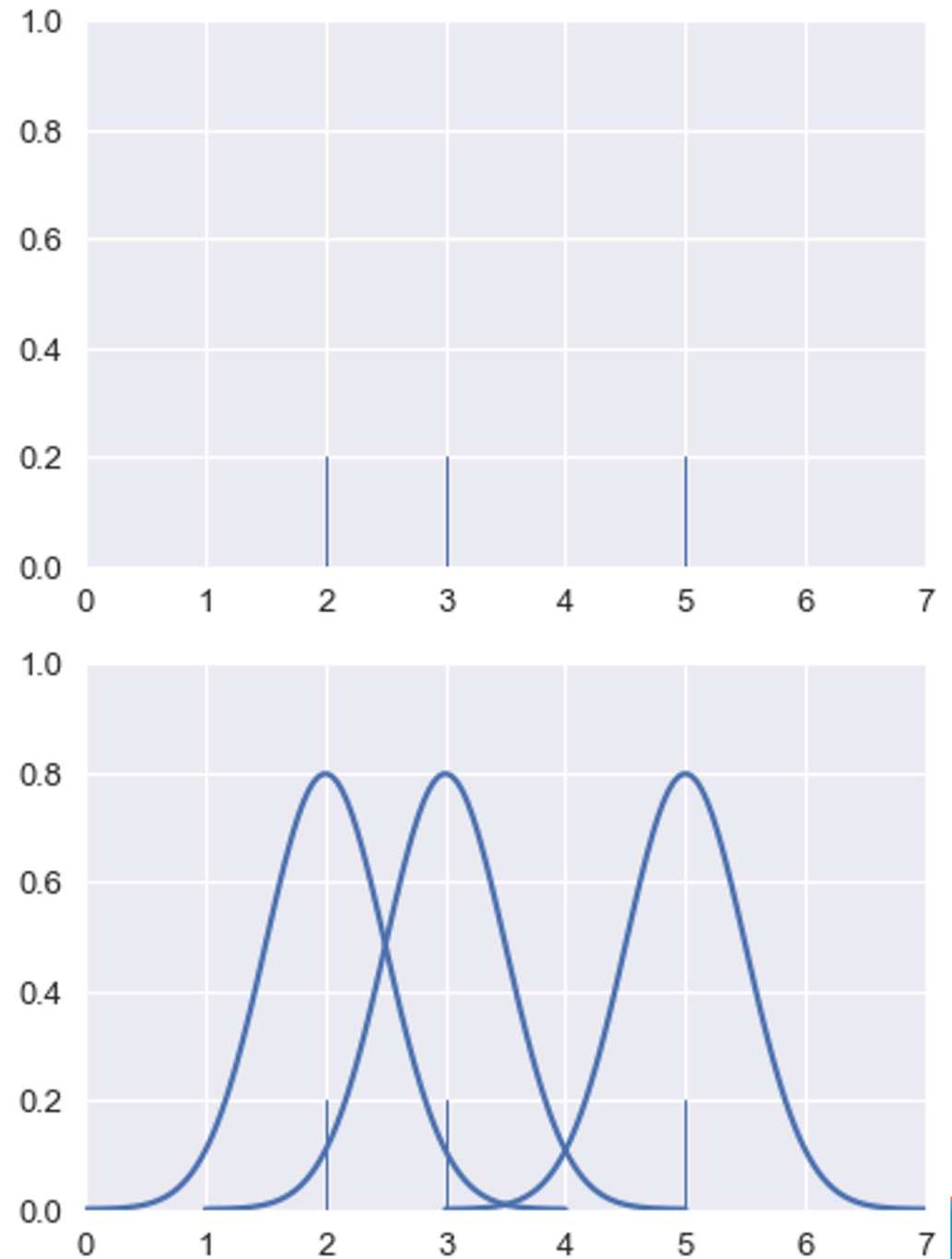
- Sophisticated smoothing technique
- Used to estimate a probability density function from a set of data



# Kernel Density Estimation

Intuition:

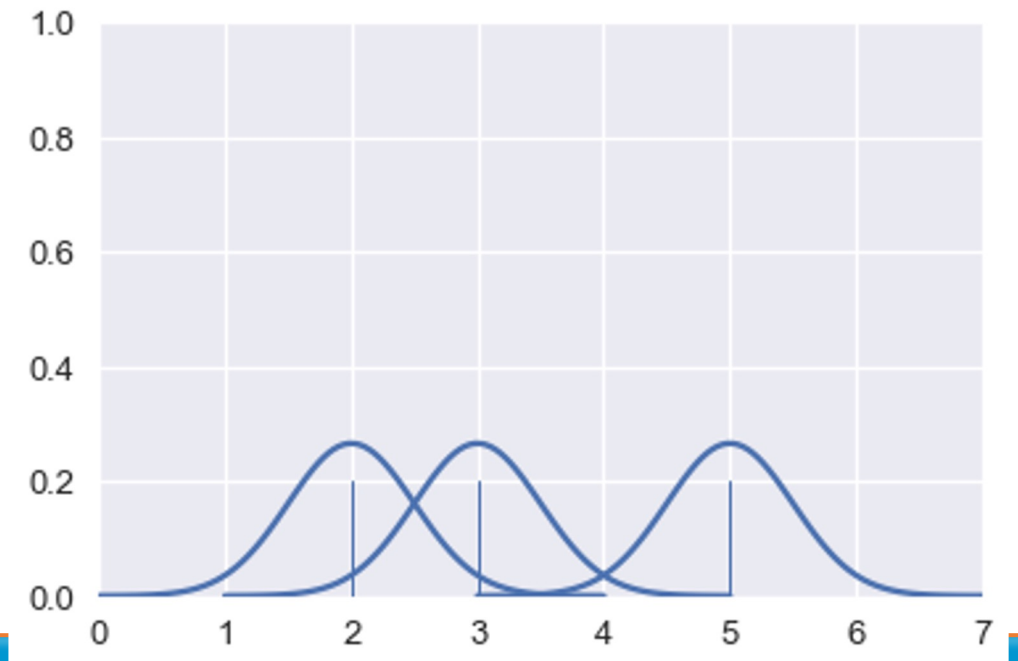
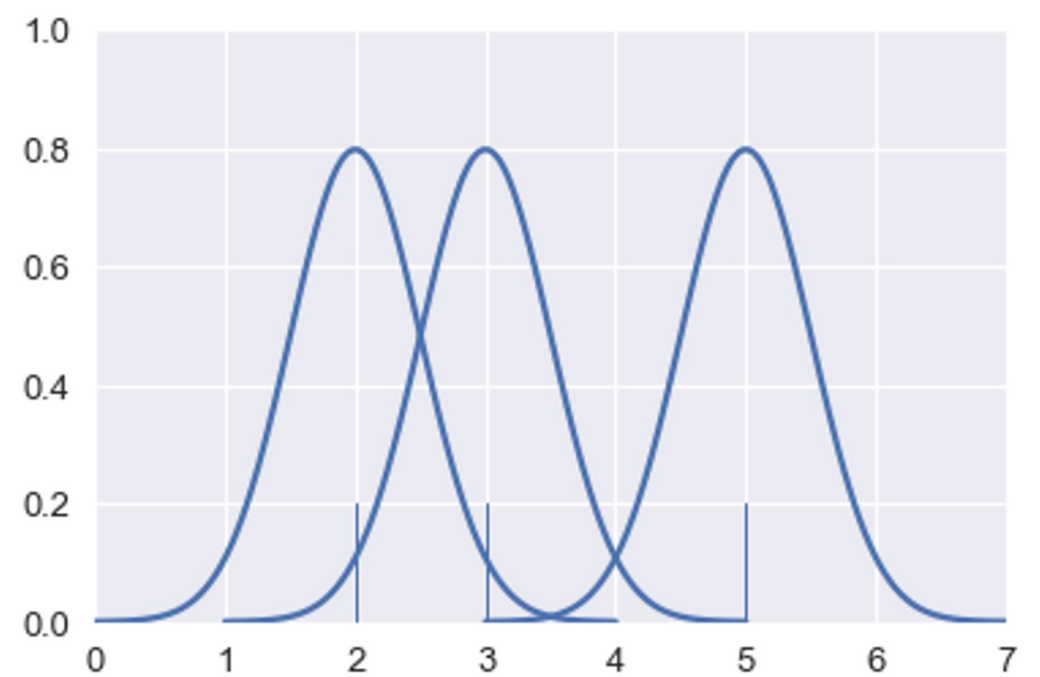
1. Place a “kernel” at each data point



# Kernel Density Estimation

Intuition:

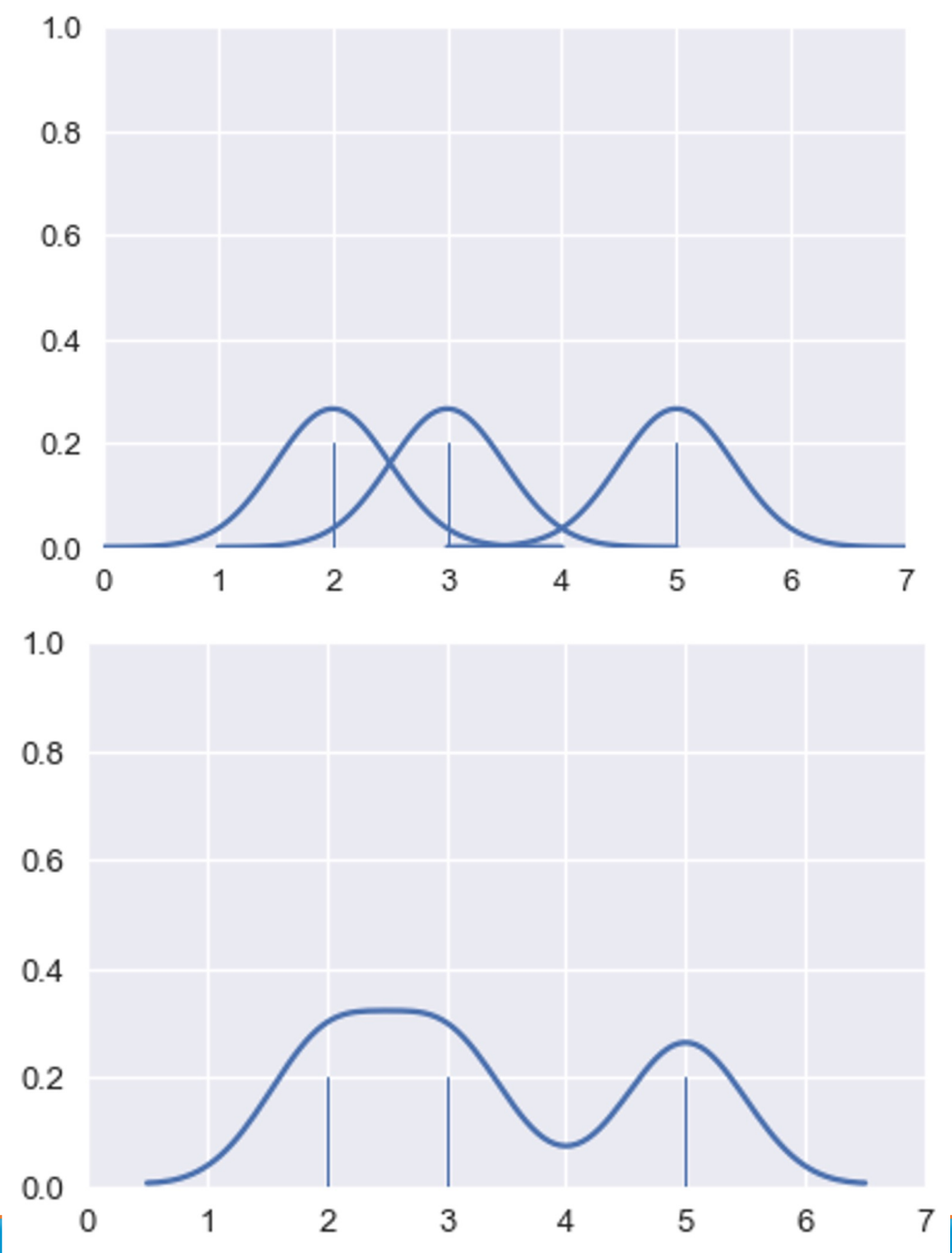
1. Place a “kernel” at each data point
2. Normalize kernels so that total area = 1



# Kernel Density Estimation

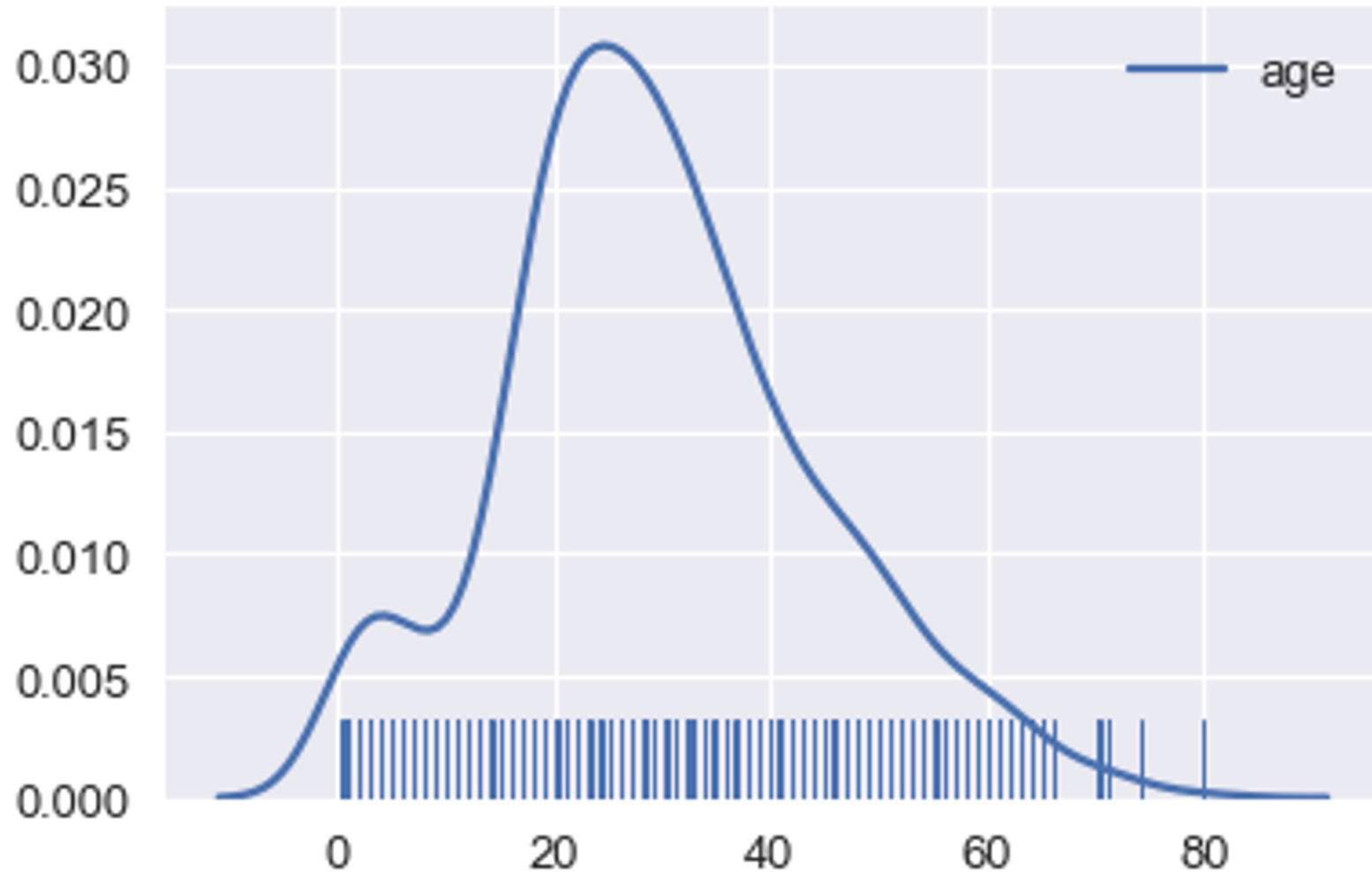
Intuition:

1. Place a “kernel” at each data point
2. Normalize kernels so that total area = 1
3. Sum all kernels together



# Kernel Density Estimation

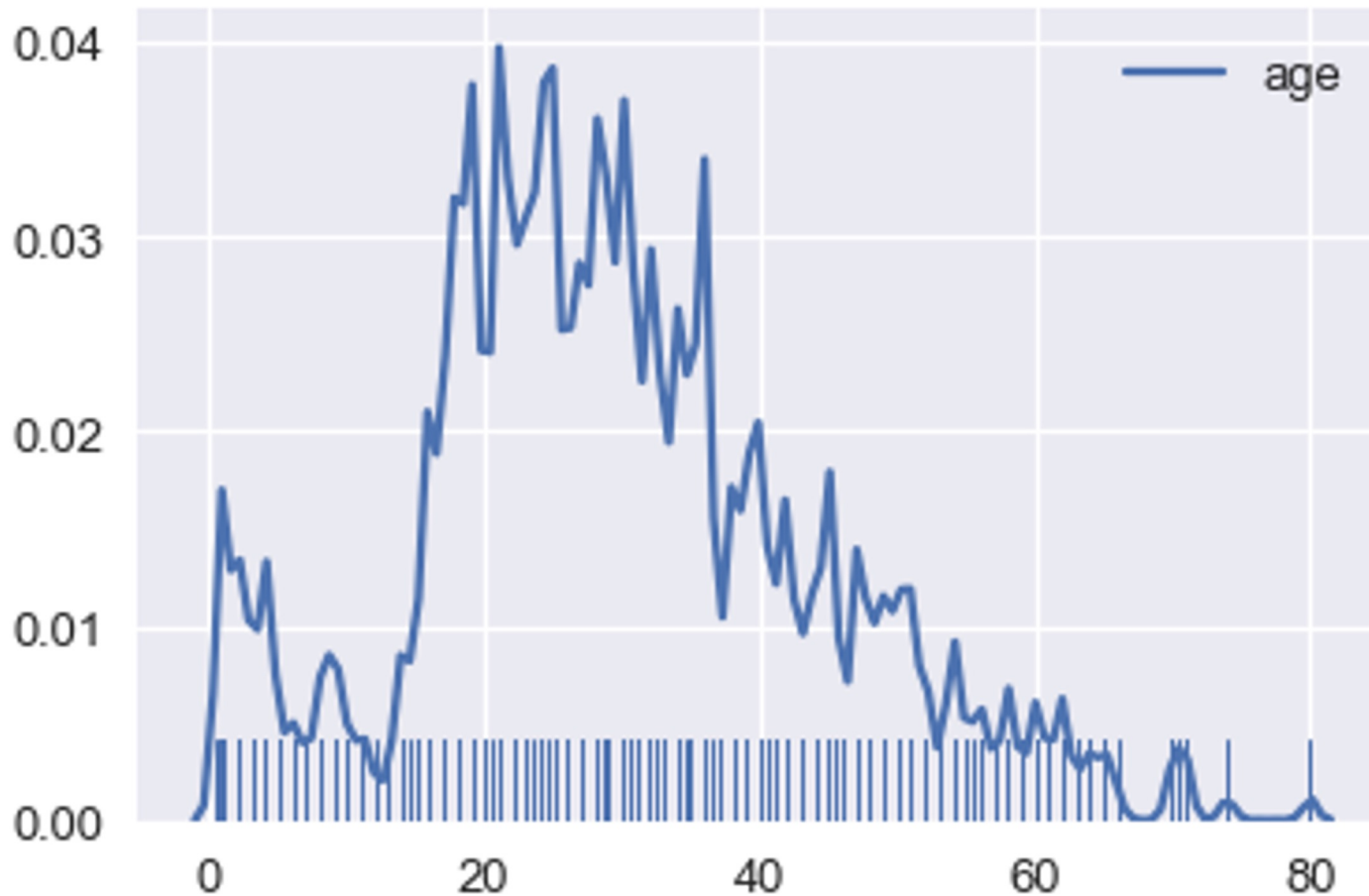
Gaussian kernel most common (default for seaborn).



# Kernel Density Estimation

Changing width of each kernel = changing bandwidth

Narrow bandwidth is analogous to narrow bins for histogram

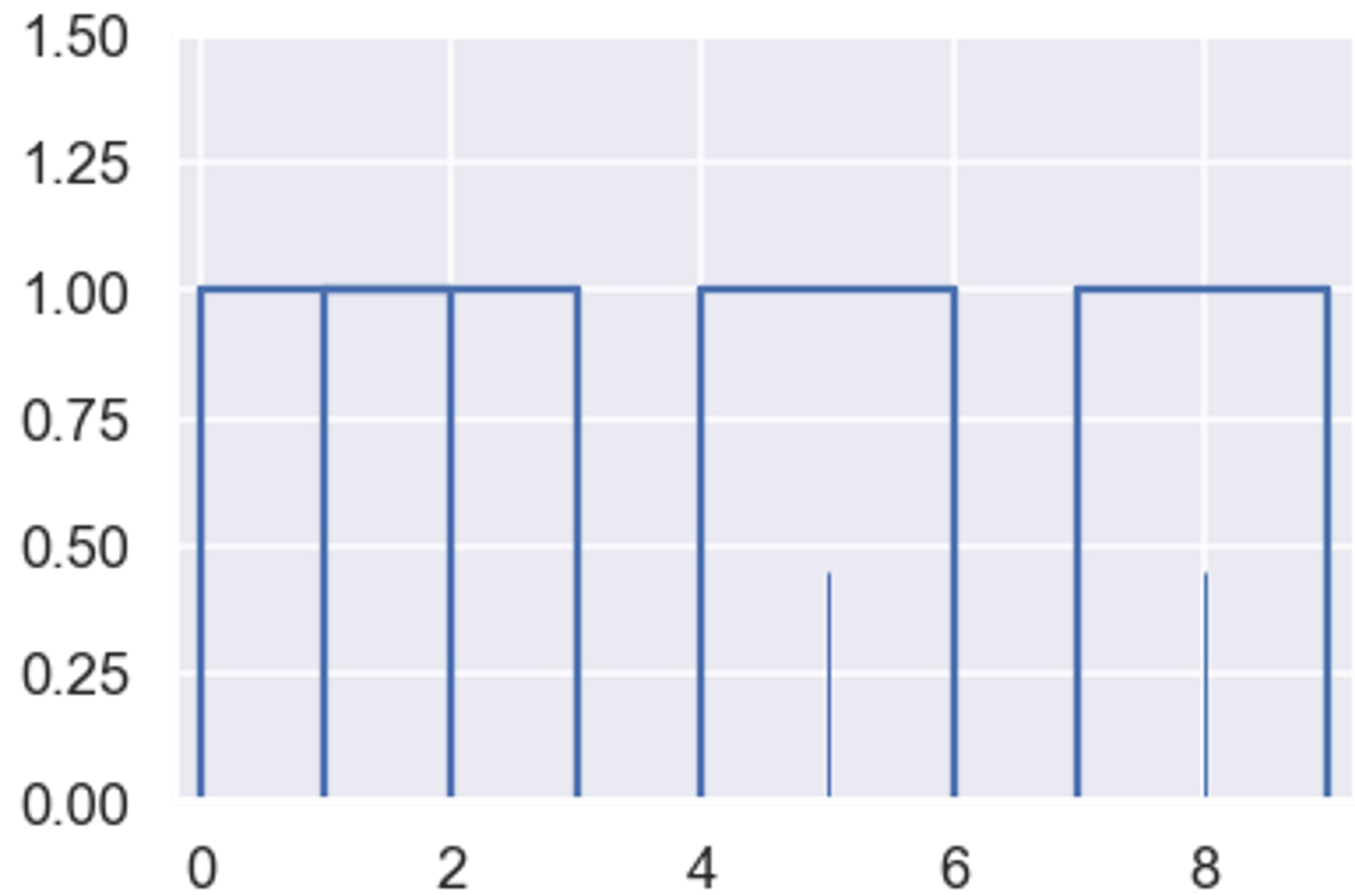


# KDE Example — Uniform Kernel

Uniform kernel with bandwidth of 2.

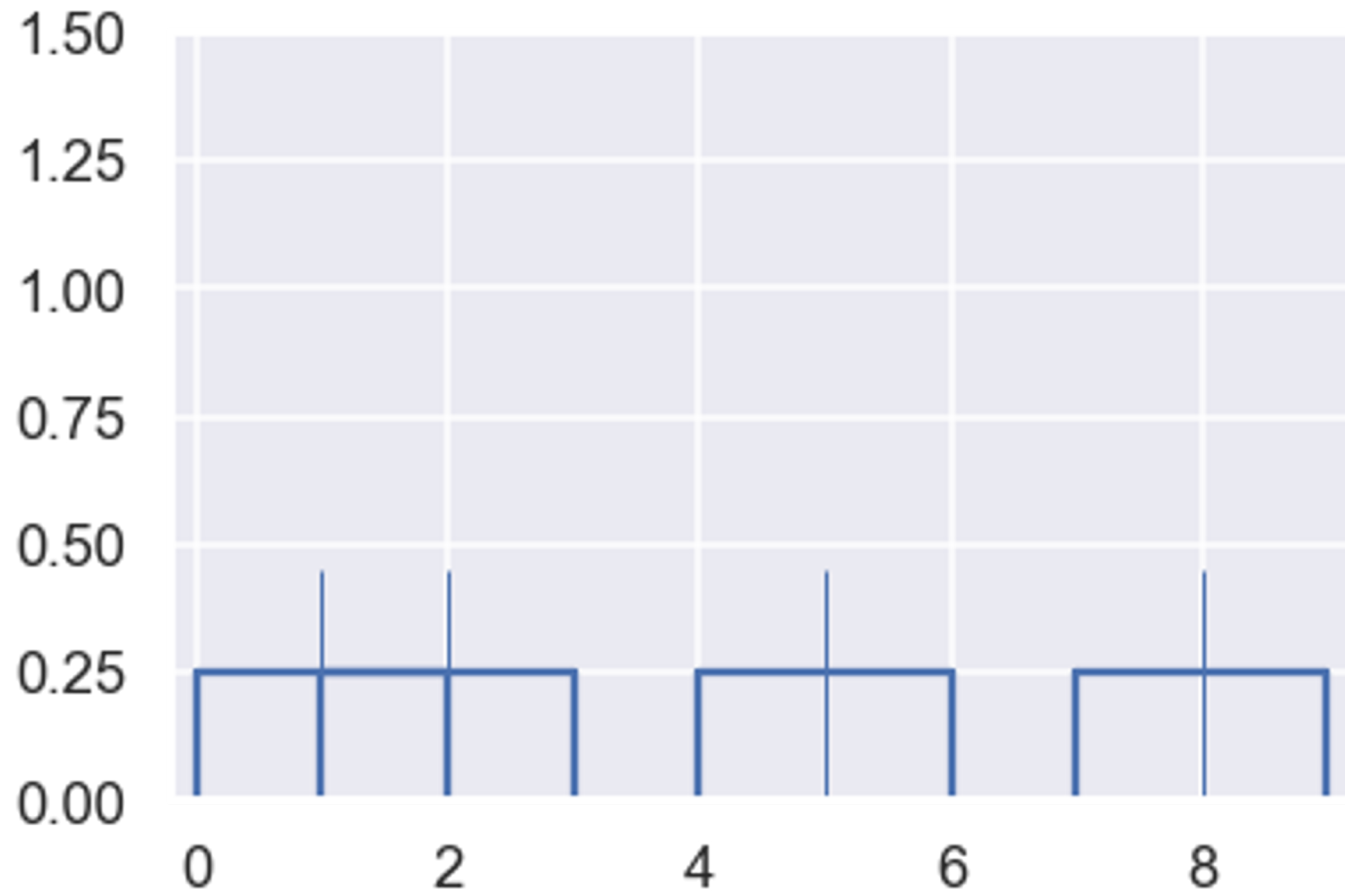
Data points at:

Kernel at each  $x$ :  $x = [1, 2, 5, 8]$



# KDE Example — Uniform Kernel

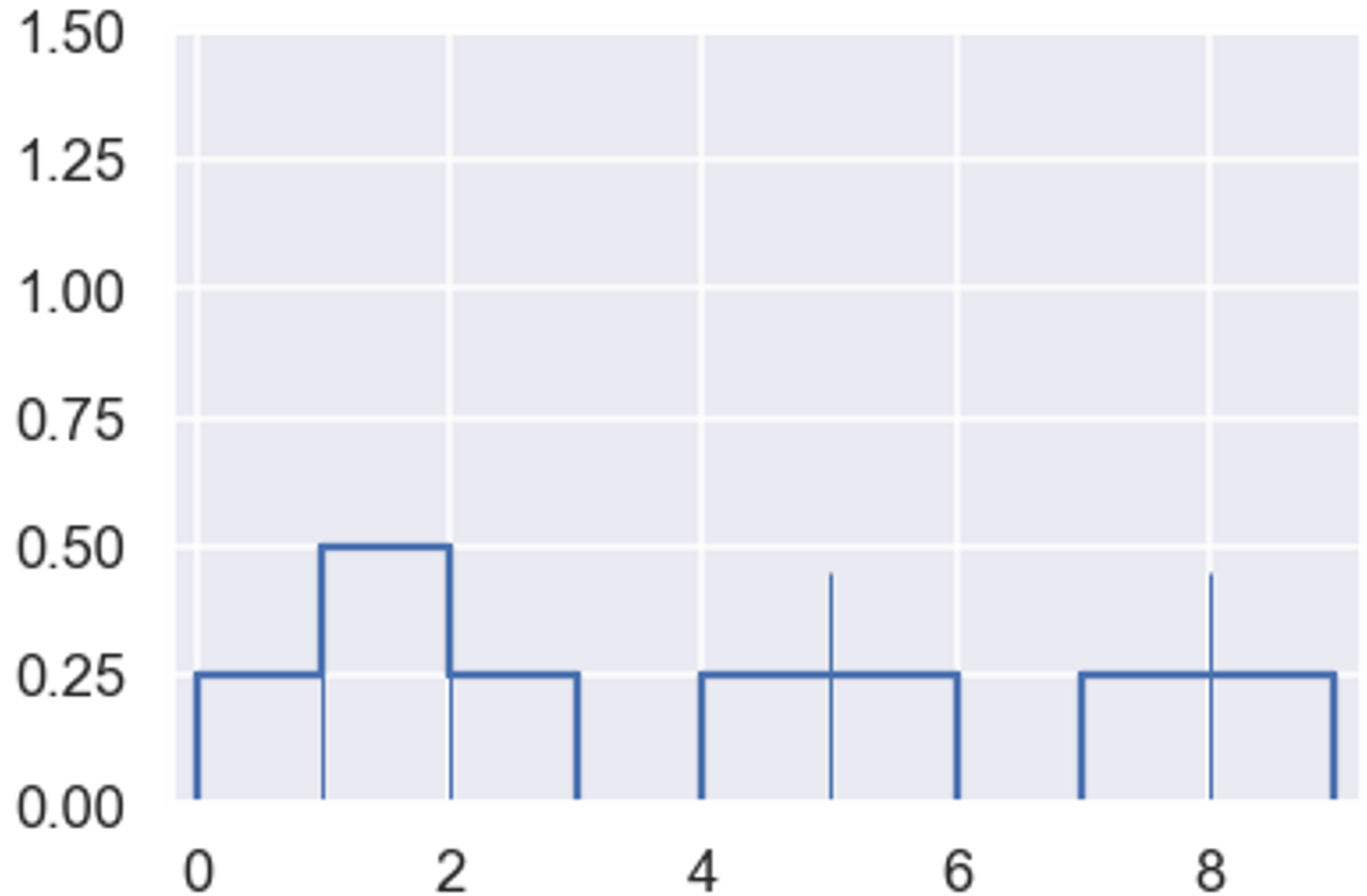
Scale each kernel by  $1/4$  since there are four points:



# KDE Example – Uniform Kernel

Add kernels together:

Height at 1.5? 0.5



# Summary

- When choosing a visualization, consider the principles of Scale, Conditioning, Perception, Transformation, Context, and Smoothing!
- In general: show the data!
  - Maximize data-ink ratio: cut out everything that isn't data-related