Learning goals:

- Understand the principles of Scale, Conditioning, Perception, Transformation, Context, and Smoothing

Data Visualization (Reading: 6.4-6.6)

UC Berkeley Data 100 Summer 2019
Sam Lau

(Slides adapted from Deb Nolan, Sandrine Dudoit, & Fernando Perez)

Announcements

- **Project 1 due today!**
- Small group tutoring is starting this week
  - Sign up: http://bit.ly/d100-tutor
- HW2 out Wednesday
  - Will be “officially” due Friday but we will take submissions without penalty until Tuesday (July 9)
- HW3 out Friday
  - Due the following Friday (July 12)

When Submitting Assignments...

- Run the last cell of the notebook
  - Ensure that OkPy link shows the latest version of the notebook
- **Then, submit the PDF to Gradescope!**
  - You must label each question with the pages it's on
  - If jassign breaks, you can use Print to PDF.

(Demo)

Data Visualization Principles

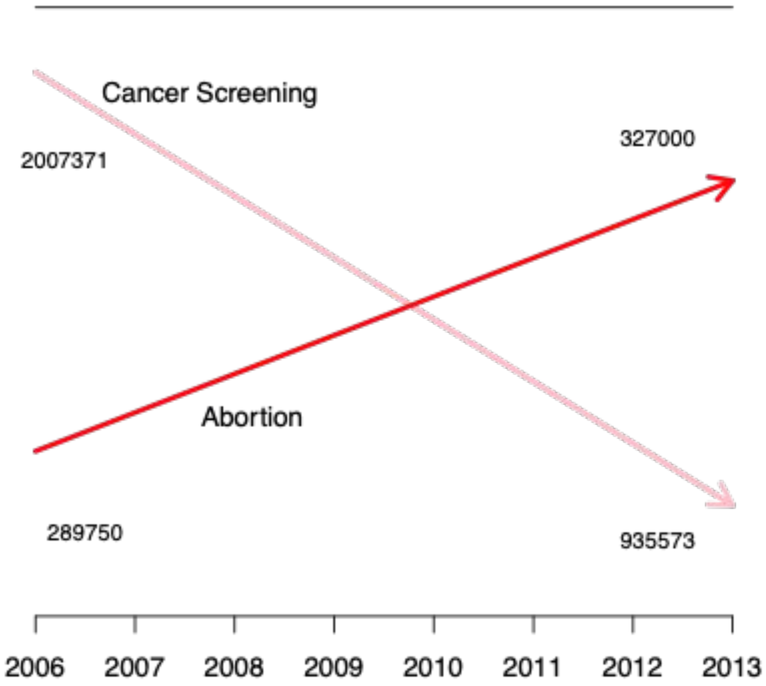
Six Principles Today

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

Explored via three case studies.

Case 1: Planned Parenthood 2015 Hearing

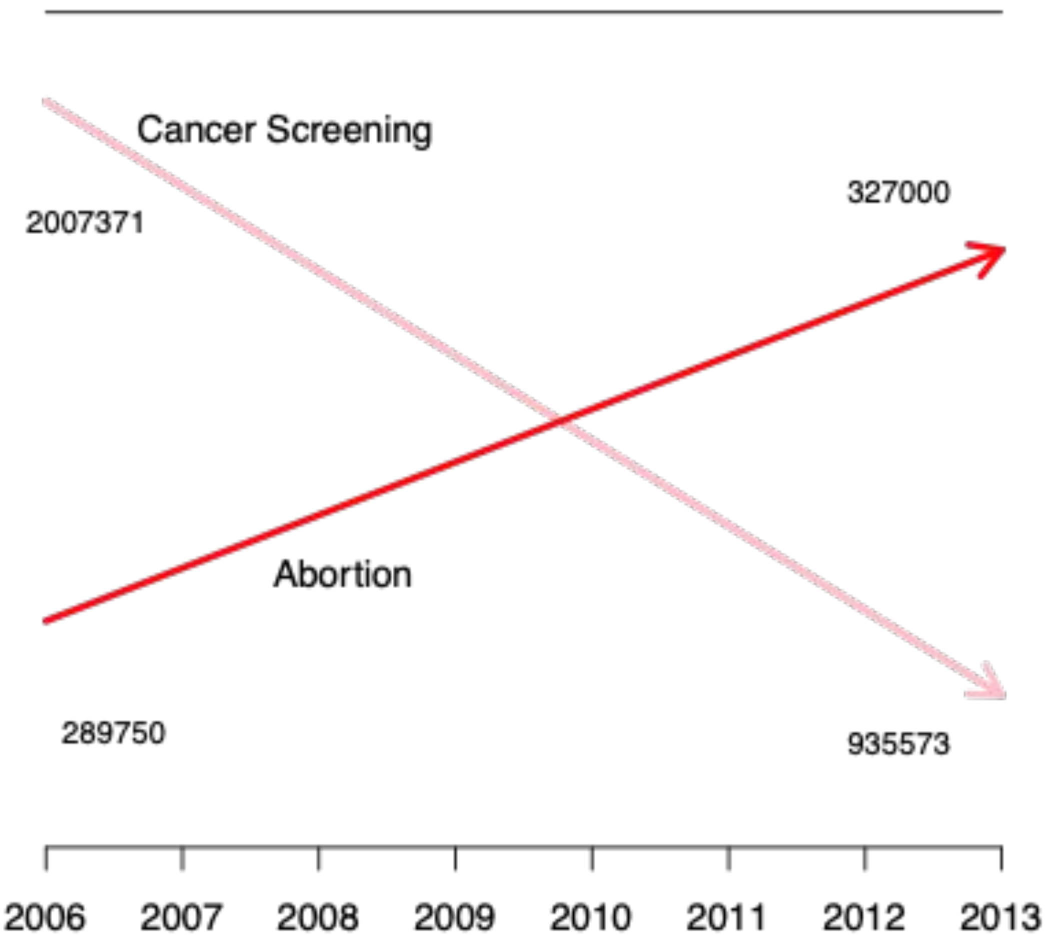
- Investigation of federal funding of Planned Parenthood in light of fetal tissue controversy
- Congressman Chaffetz (R-UT) showed plot which originally appeared in a report by Americans United for Life (http://www.aul.org/)



Full Report available at https://oversight.house.gov/interactivepage/plannedparenthood/ .

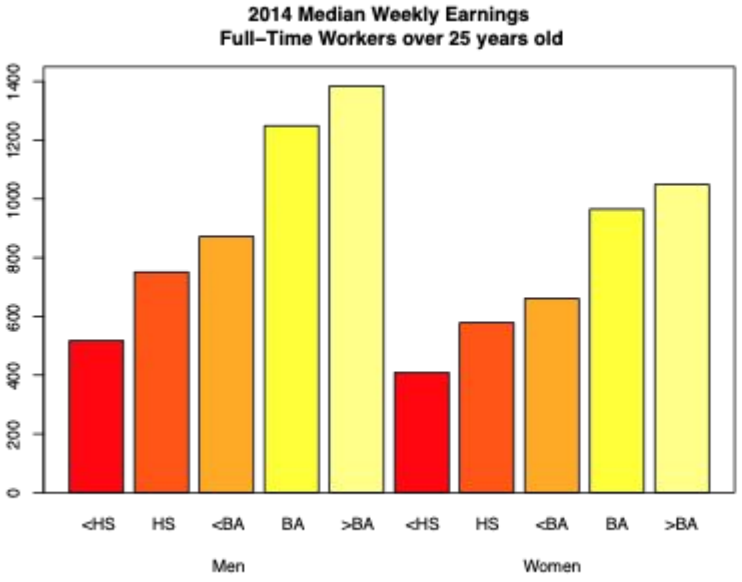
Case 1: Planned Parenthood 2015 Hearing

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



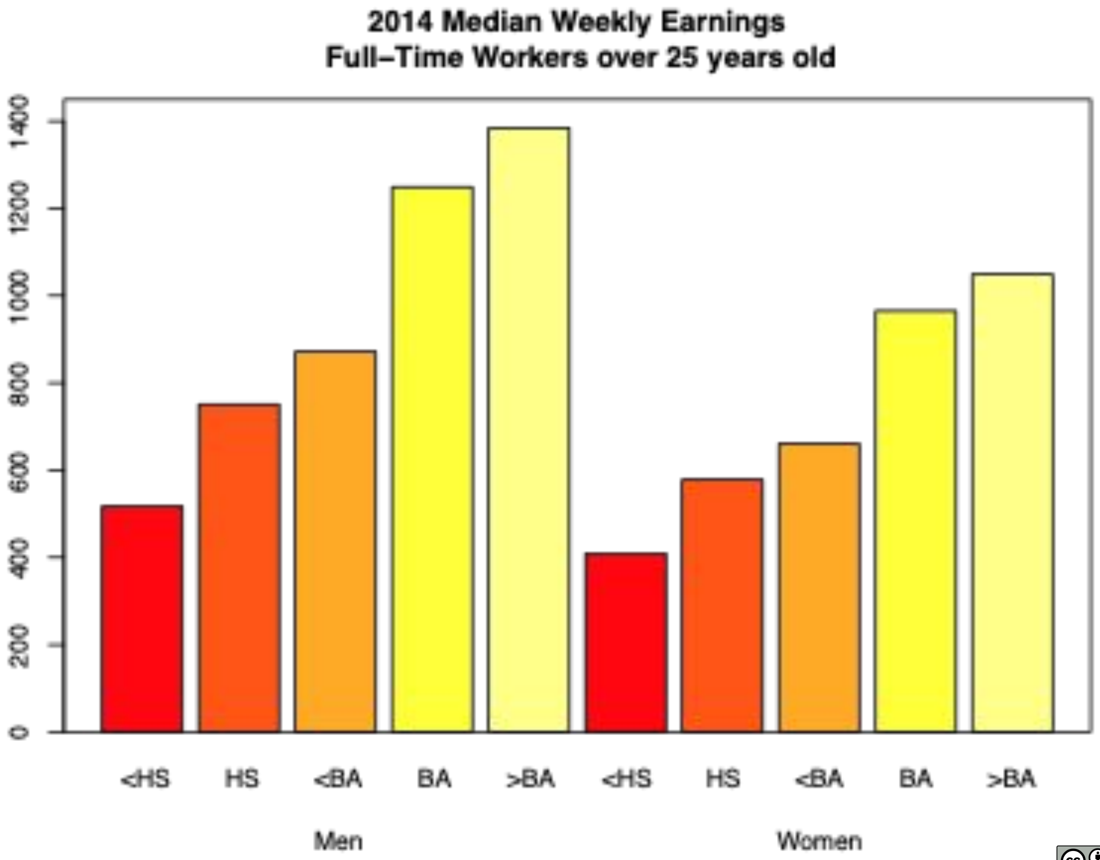
Case 2: Median Weekly Earnings

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



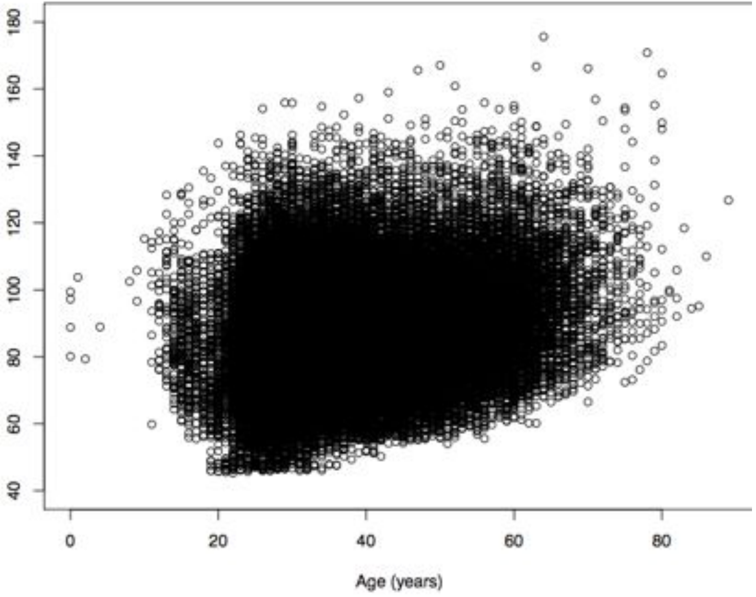
Case 2: Median Weekly Earnings

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



Case 3: Cherry Blossom Runners

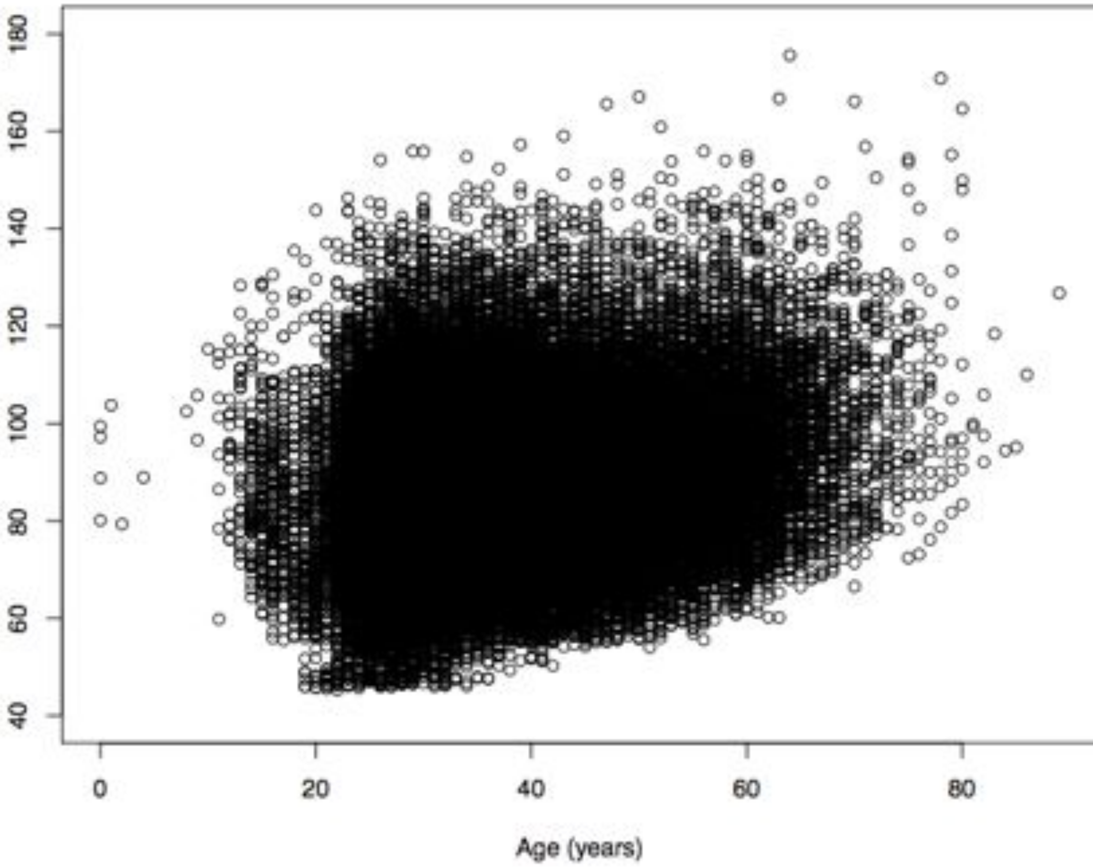
- 10 mi run in DC every April
- Results available from 1999-2019
- In 2019 over 17,000 runners
- Scatter plot of run time (min) against age (yrs)



http://www.cherryblossom.org/

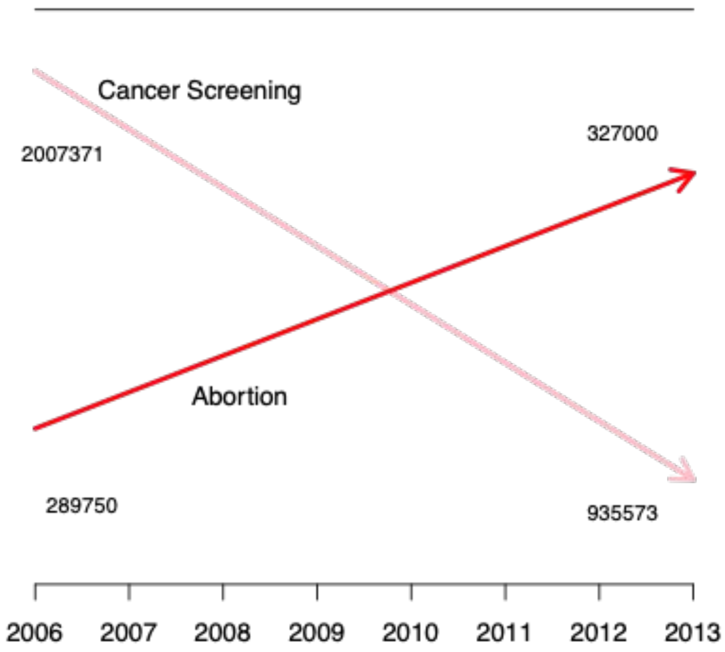
Case 3: Cherry Blossom Runners

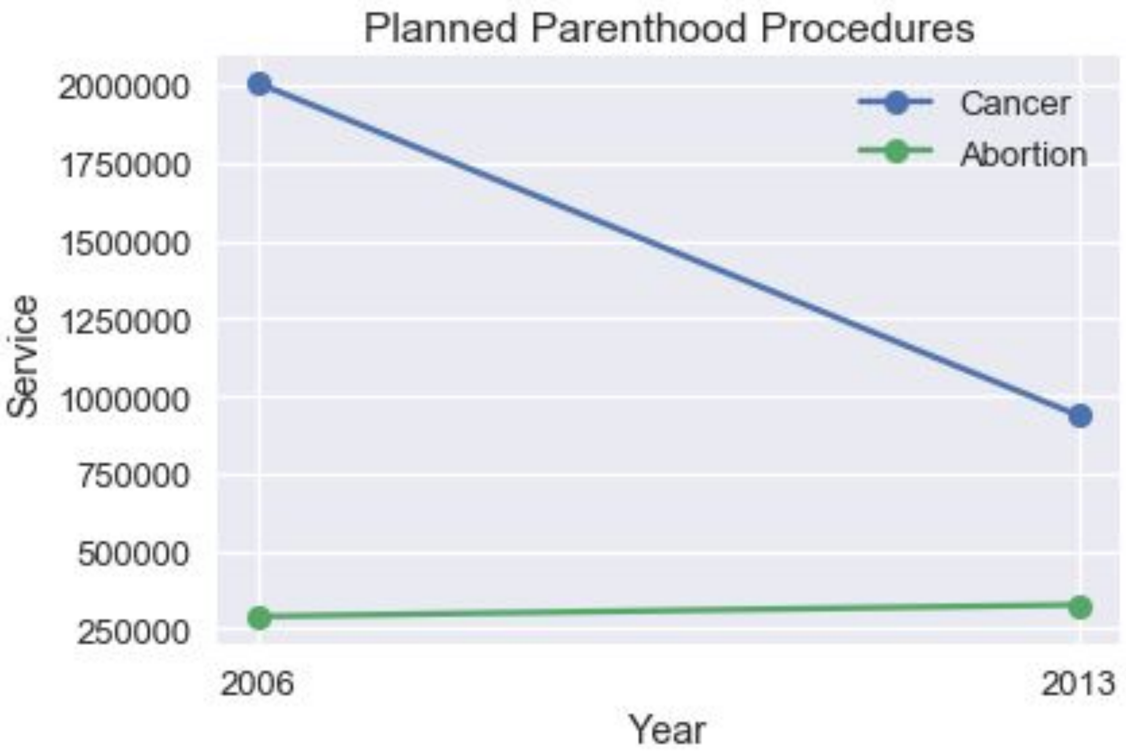
- 70,000+ points in the plot!
- What's the relationship between run time and age?



Principles of Scale

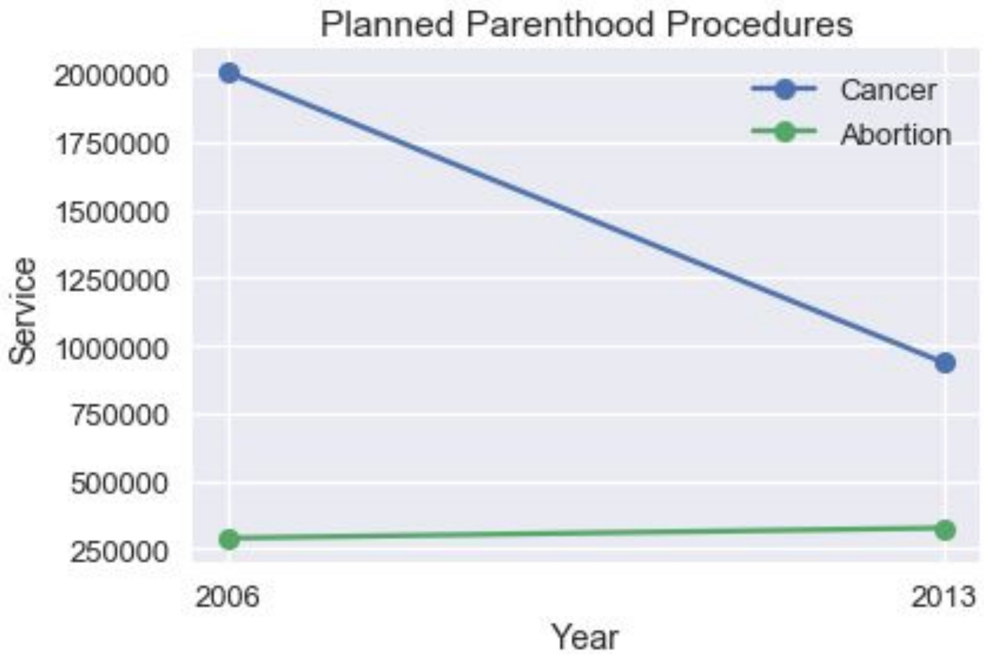
Scale


Cancer Screening
2007371
327000
Abortion
289750
935573
2006
2007
2008
2009
2010
2011
2012
2013


Planned Parenthood Procedures
Cancer
Abortion
Service
2000000
1750000
1500000
1250000
1000000
750000
500000
250000
2006
2013
Year

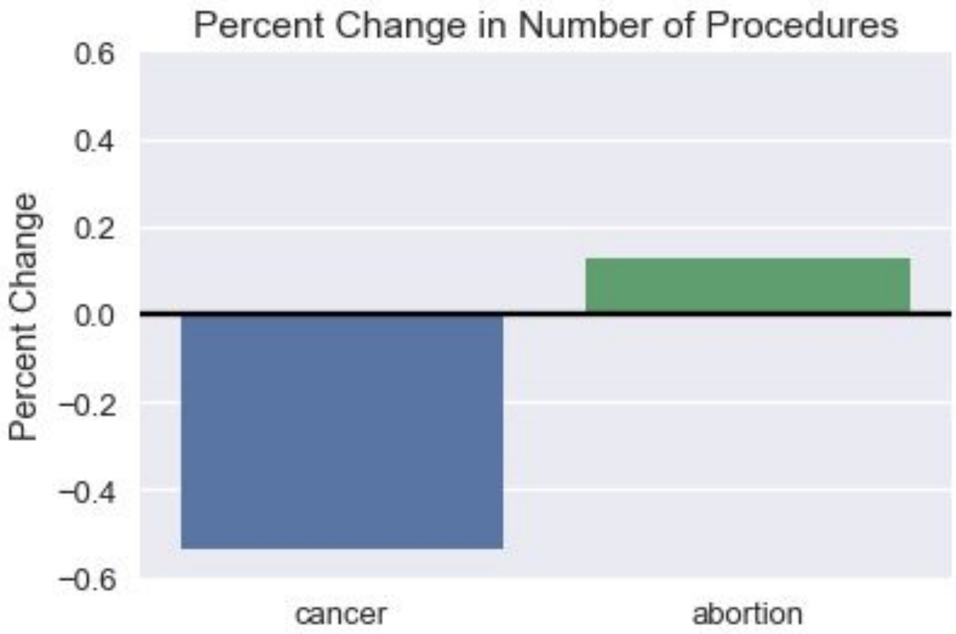
Keep consistent axis scales

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



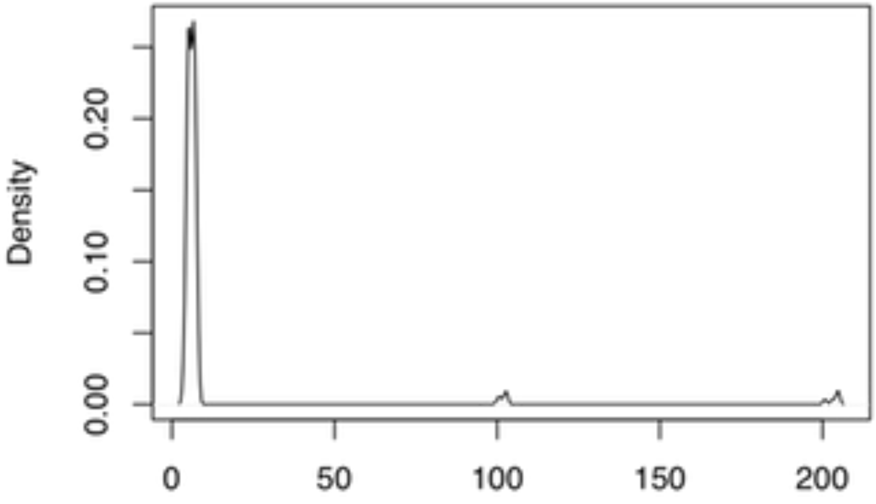
Consider Scale of Data

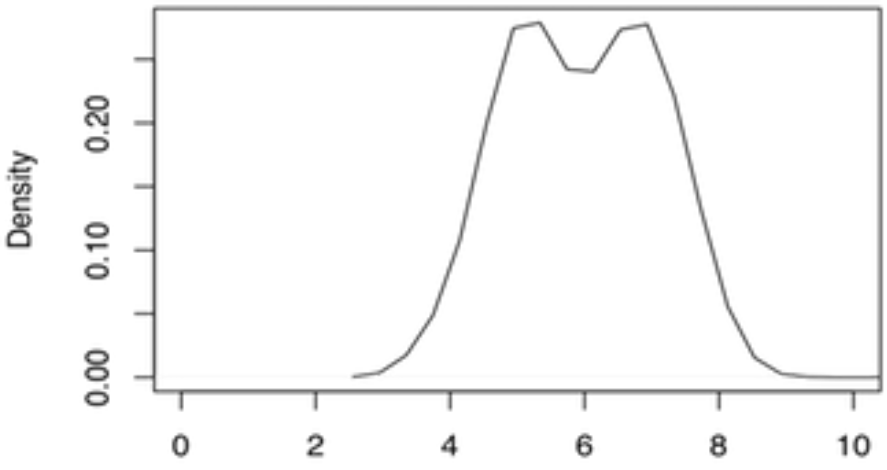
- Scales of cancer screenings vs. abortions quite different
- Can plot percent change instead of raw counts



Reveal the Data

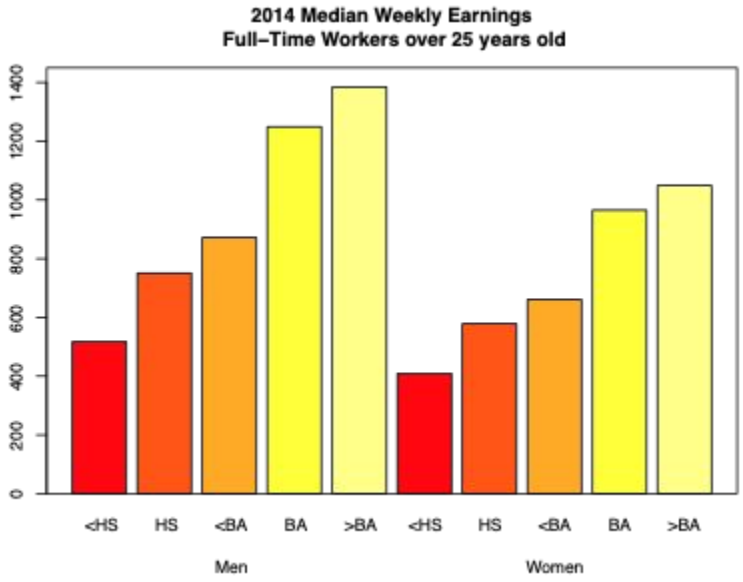
- Choose axis limits to fill plot
- If necessary, zoom into region with most of data
  - Can make separate plots for different regions

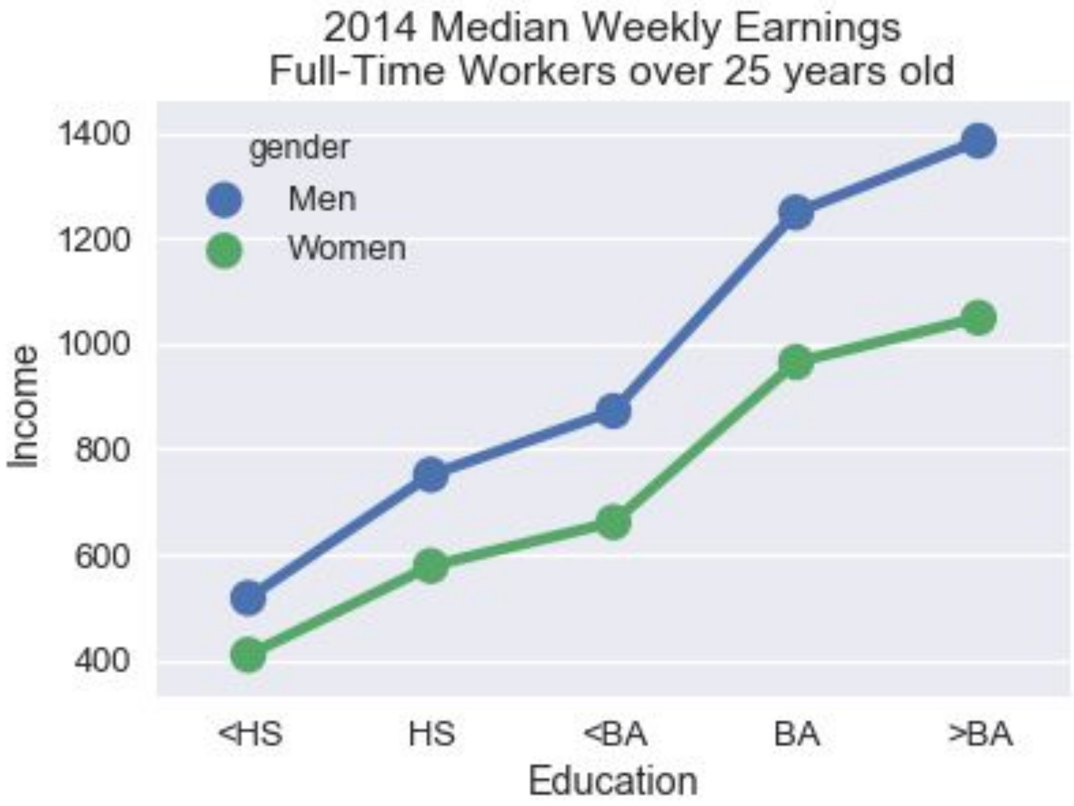




Principles of Conditioning

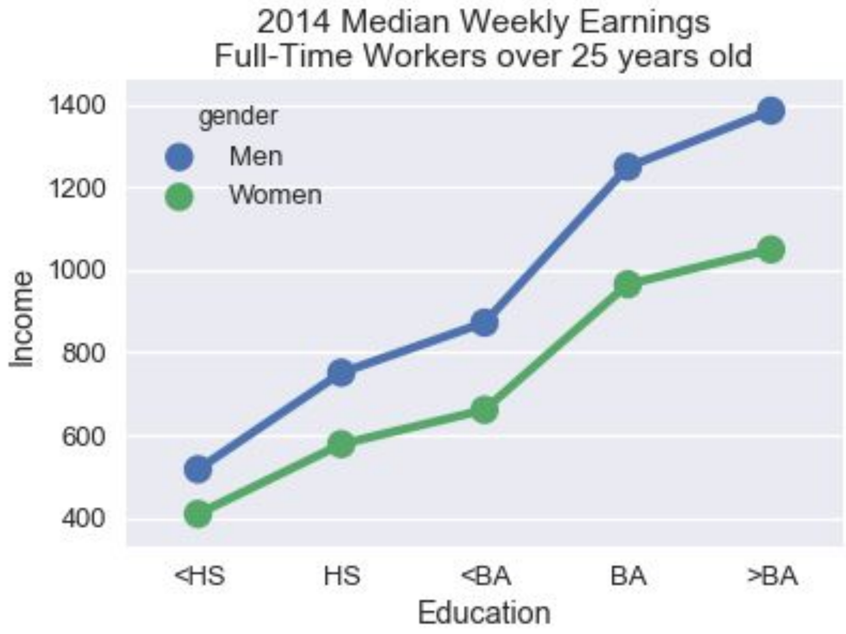
Conditioning





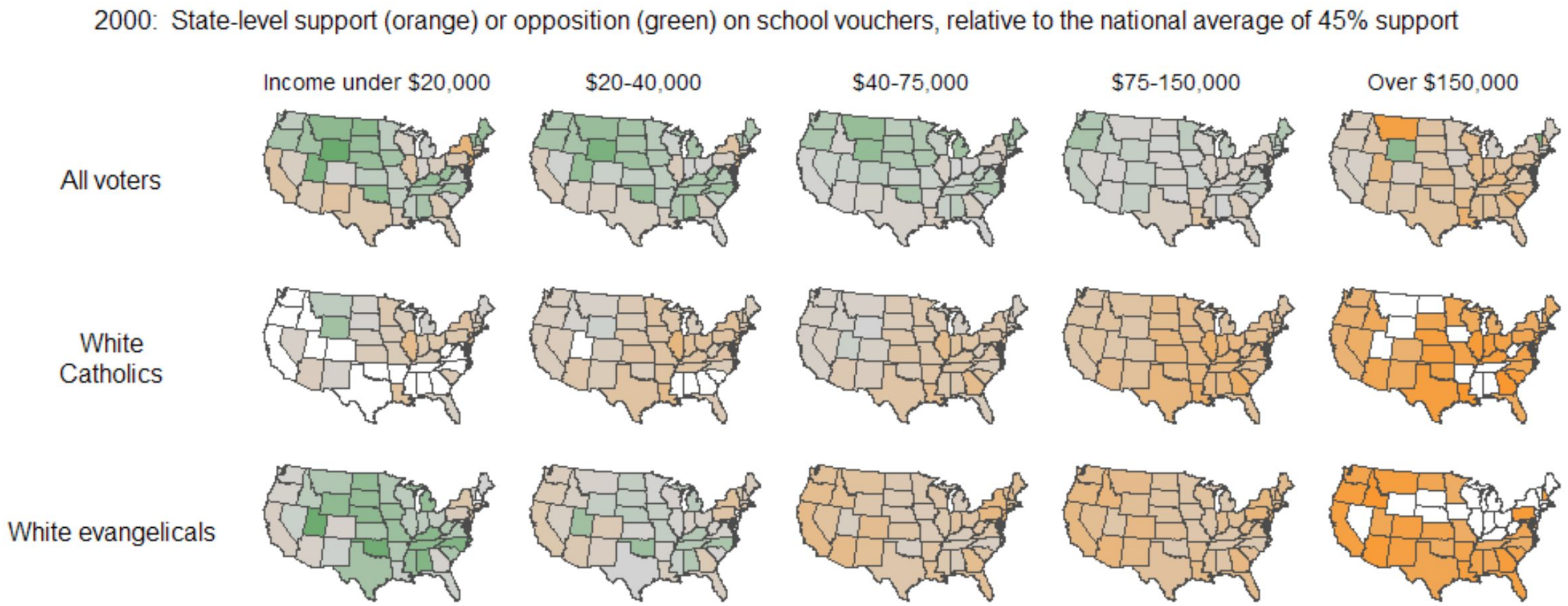
Use Conditioning To Aid Comparison

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



Use Small Multiples To Aid Comparison

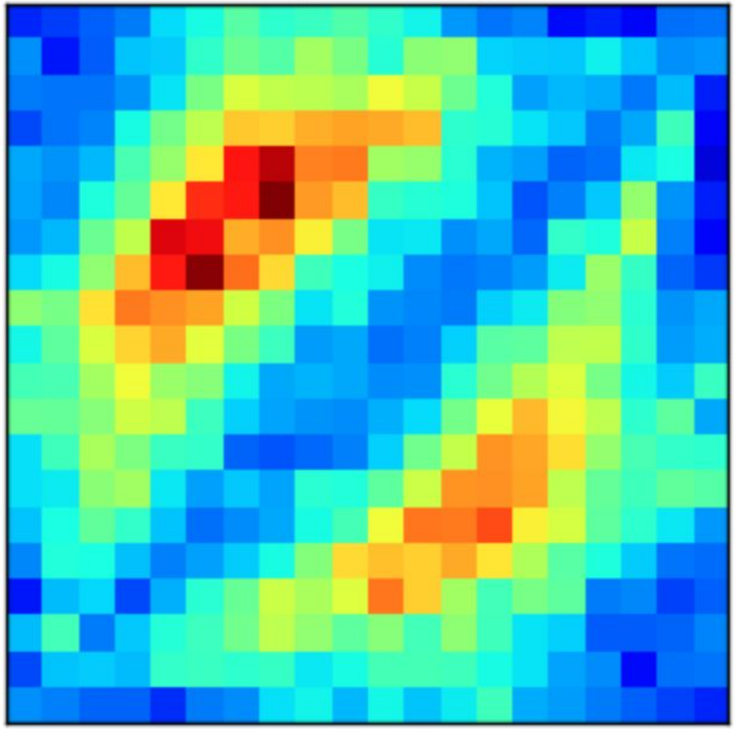
- Faceted plots that share scales are easy to compare
  - https://statmodeling.stat.columbia.edu/2009/07/15/hard_sell_for_b/



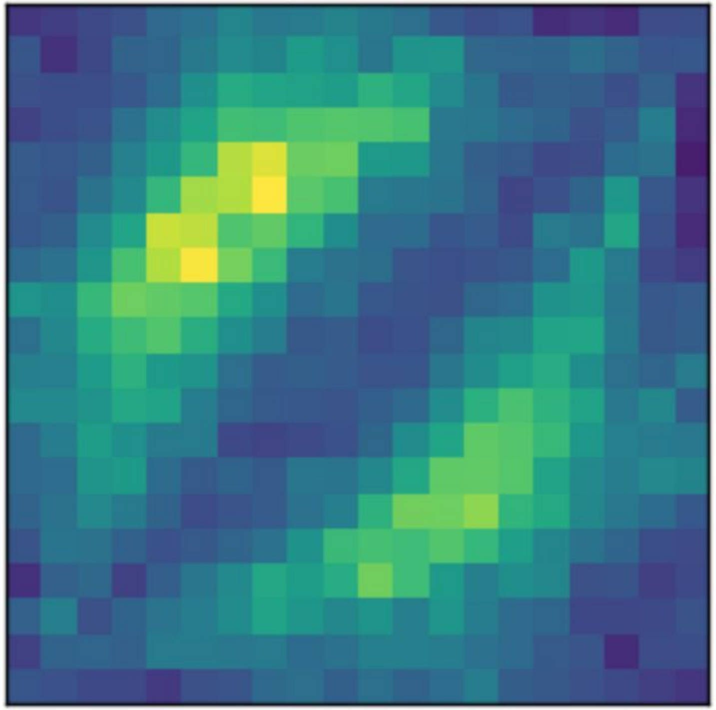
Principles of Perception

Color Choices Matter!

Jet Colormap



Viridis Colormap

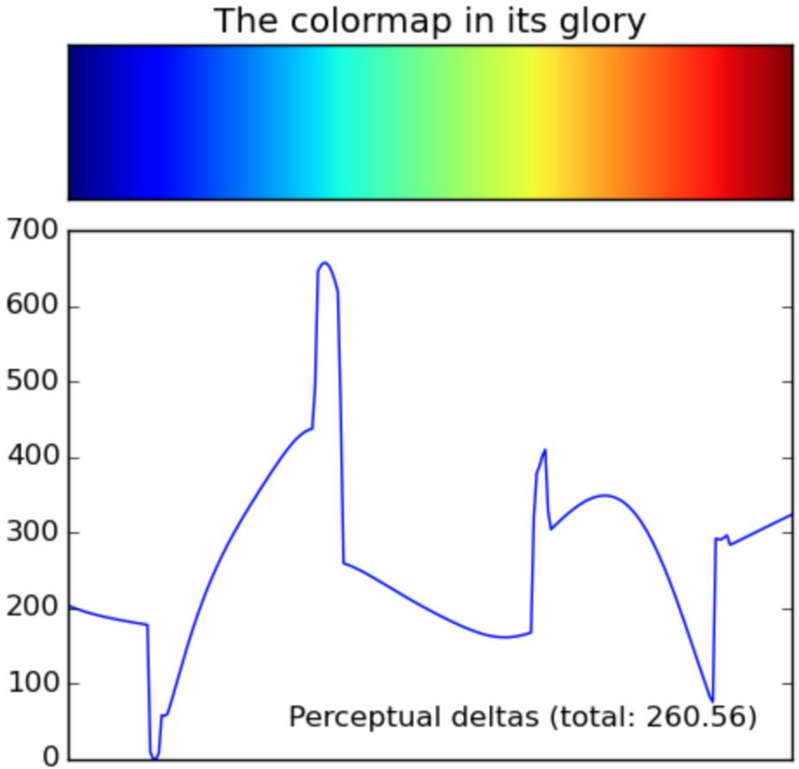


Use a Perceptually Uniform Color Map

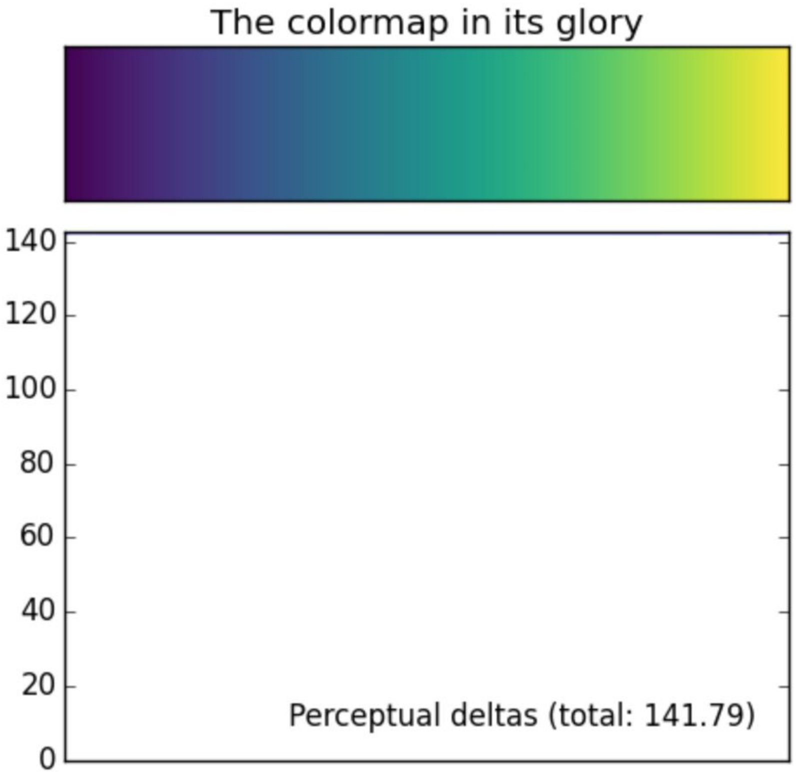
- Perceptually uniform: changing data from 0.1 to 0.2 appears similar to change from 0.8 to 0.9.
    - Measure by running experiments on people!
- Jet, the old matplotlib default, was far from uniform!
- Our own Stéfan van der Walt and Nathaniel Smith at the Berkeley Institute of Data Science fixed this :)
    - https://bids.github.io/colormap/
- Also, avoid red + green since many people are colorblind

Use a Perceptually Uniform Color Map

Jet Colormap

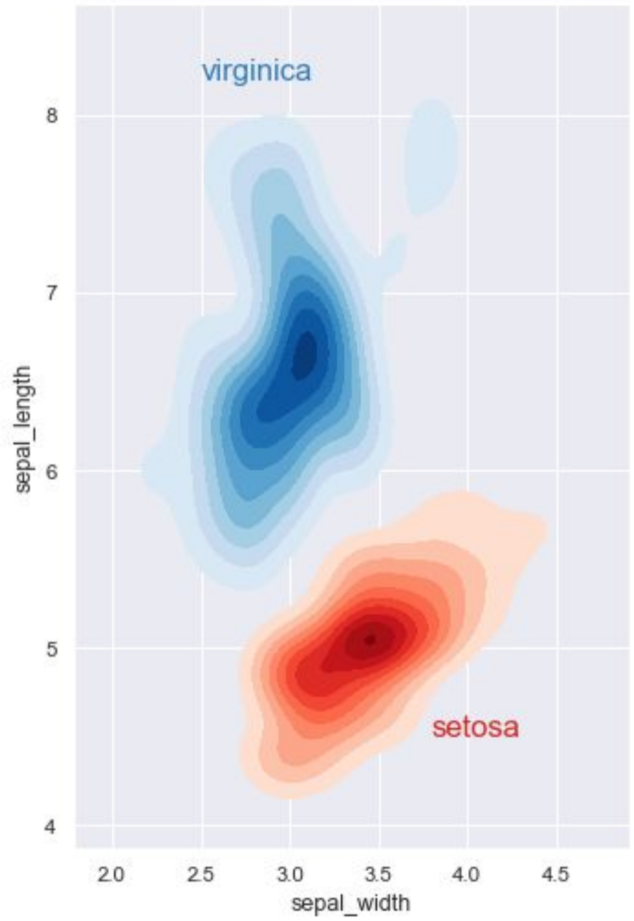


Viridis Colormap



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!



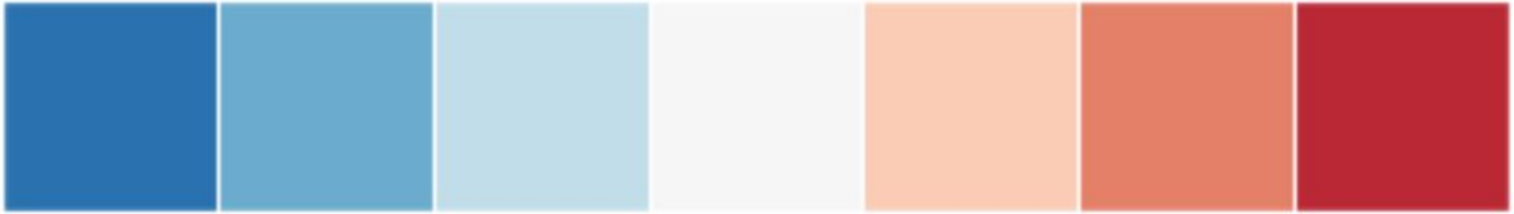
Use Color to Highlight Data Type

- 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

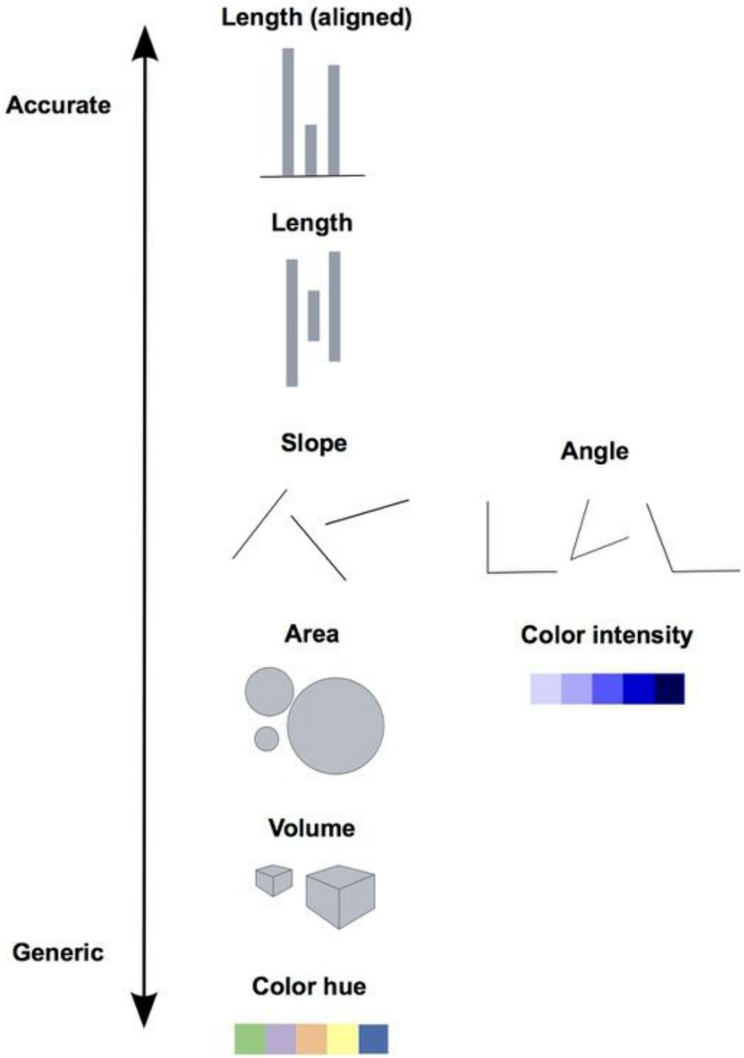
- Do both low and high value deserve equal emphasis? Use a diverging scheme where light colors represent middle values

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



Not All Marks Are Good!

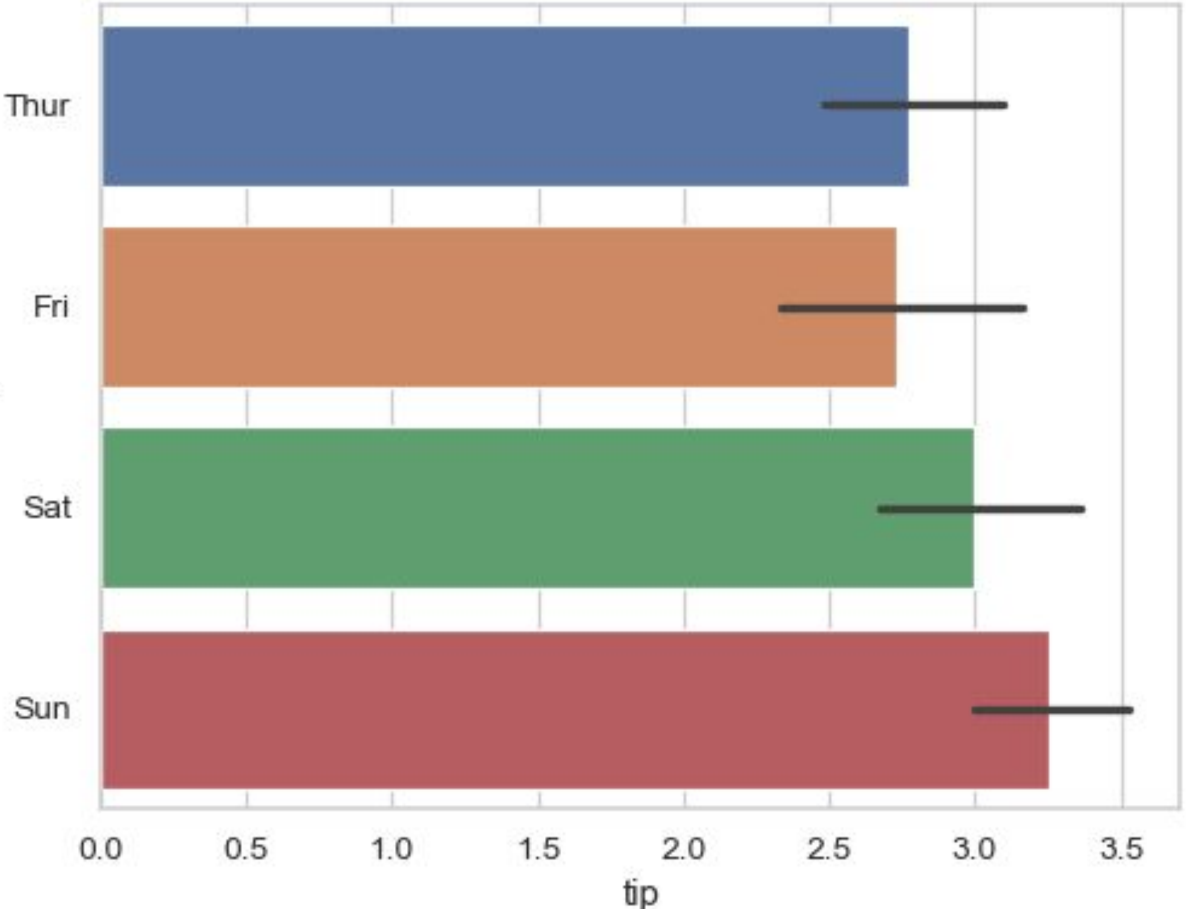
- Accuracy of judgements depend on the type of mark.
- Aligned lengths most accurate
- Color least accurate



Lengths are Easy to Understand

People can easily distinguish two different lengths

E.g. Heights of bars in bar chart

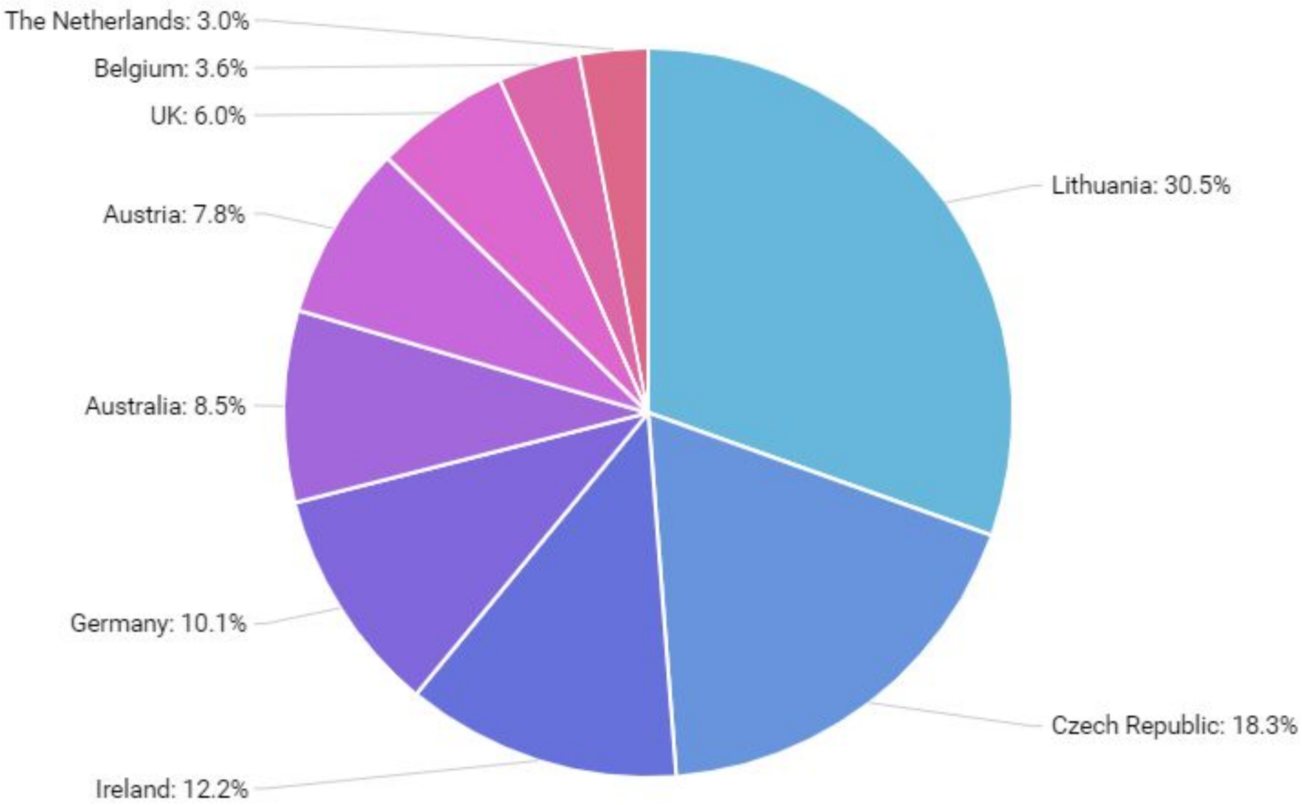


Angles are Hard to Understand

Avoid pie charts!

Angle judgements are inaccurate

In general, underestimate size of larger angle.

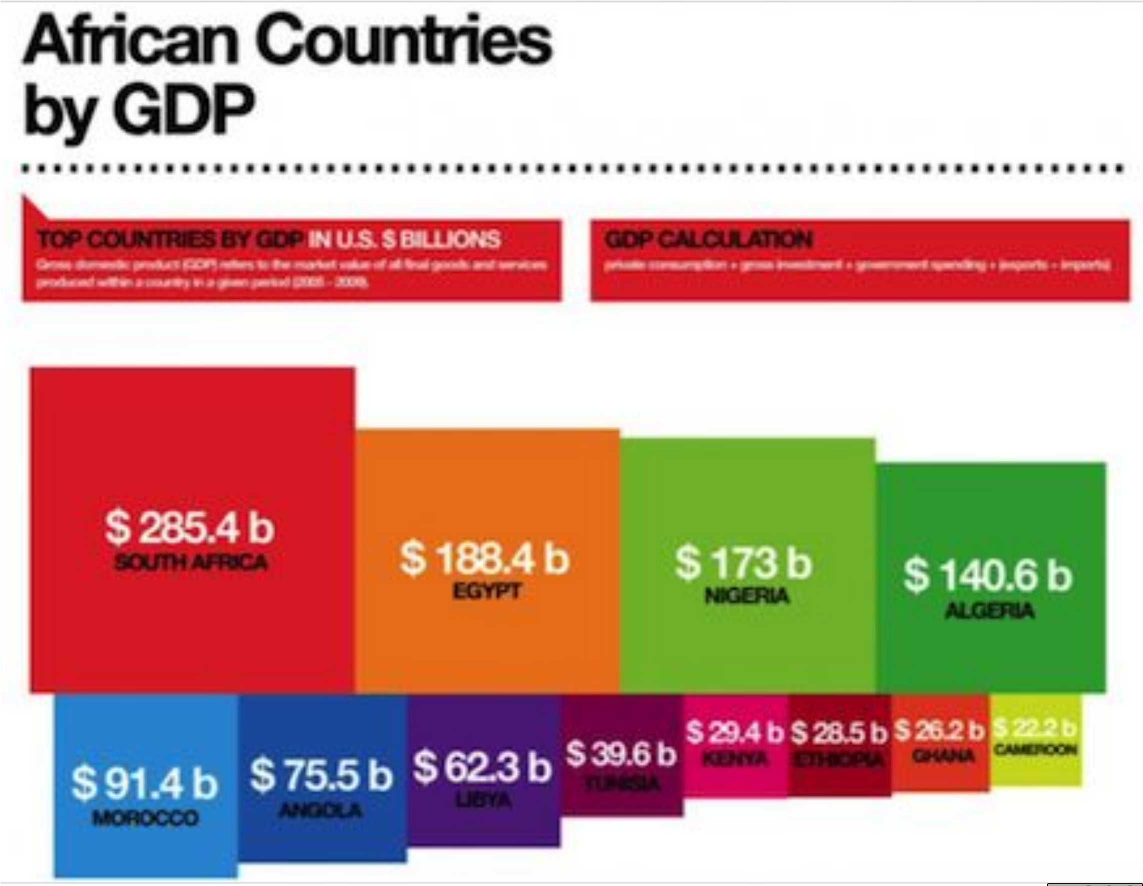


Areas are Hard to Understand

Avoid area charts!

Area judgements are inaccurate

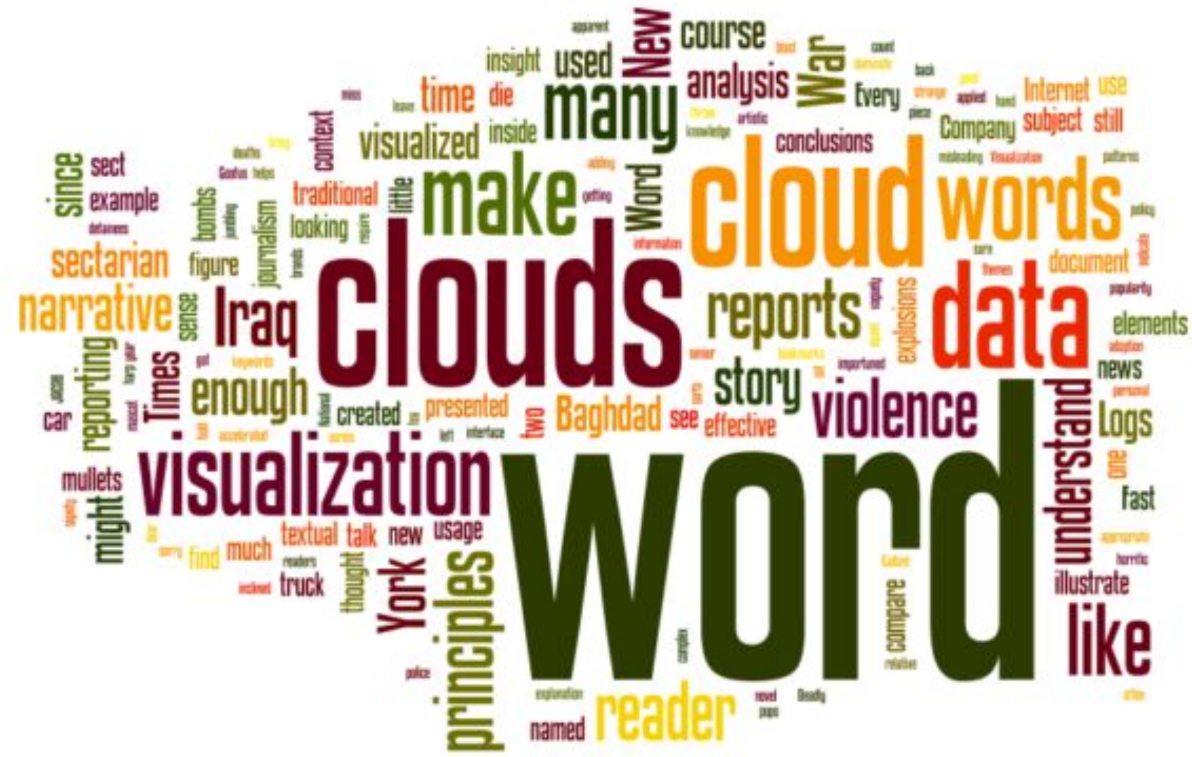
In general, underestimate size of larger area



Areas are Hard to Understand

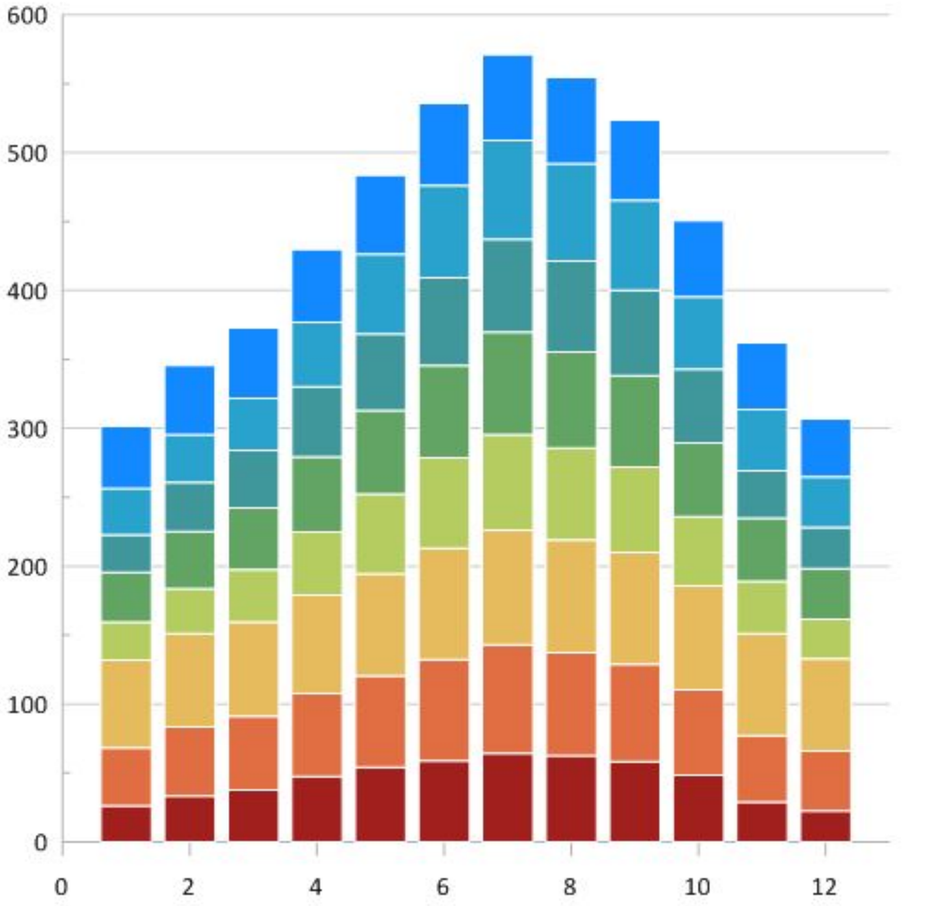
Avoid word clouds!

Hard to tell the “area” taken up by a word



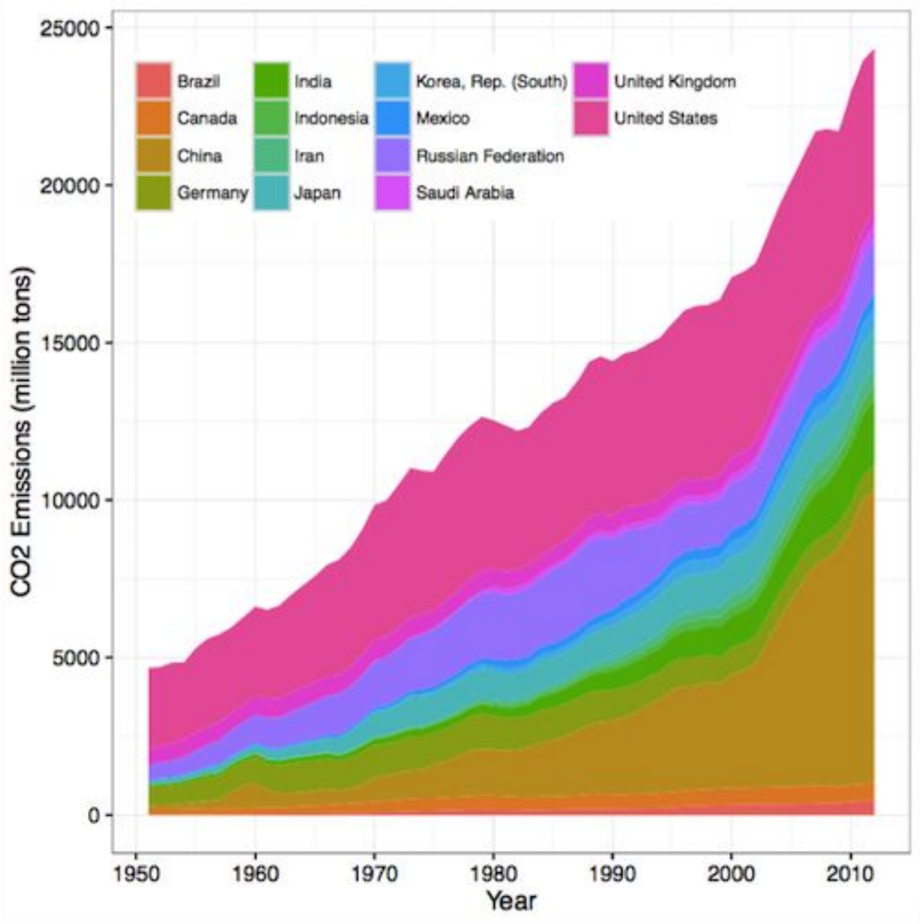
Avoid Jiggling Baseline

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



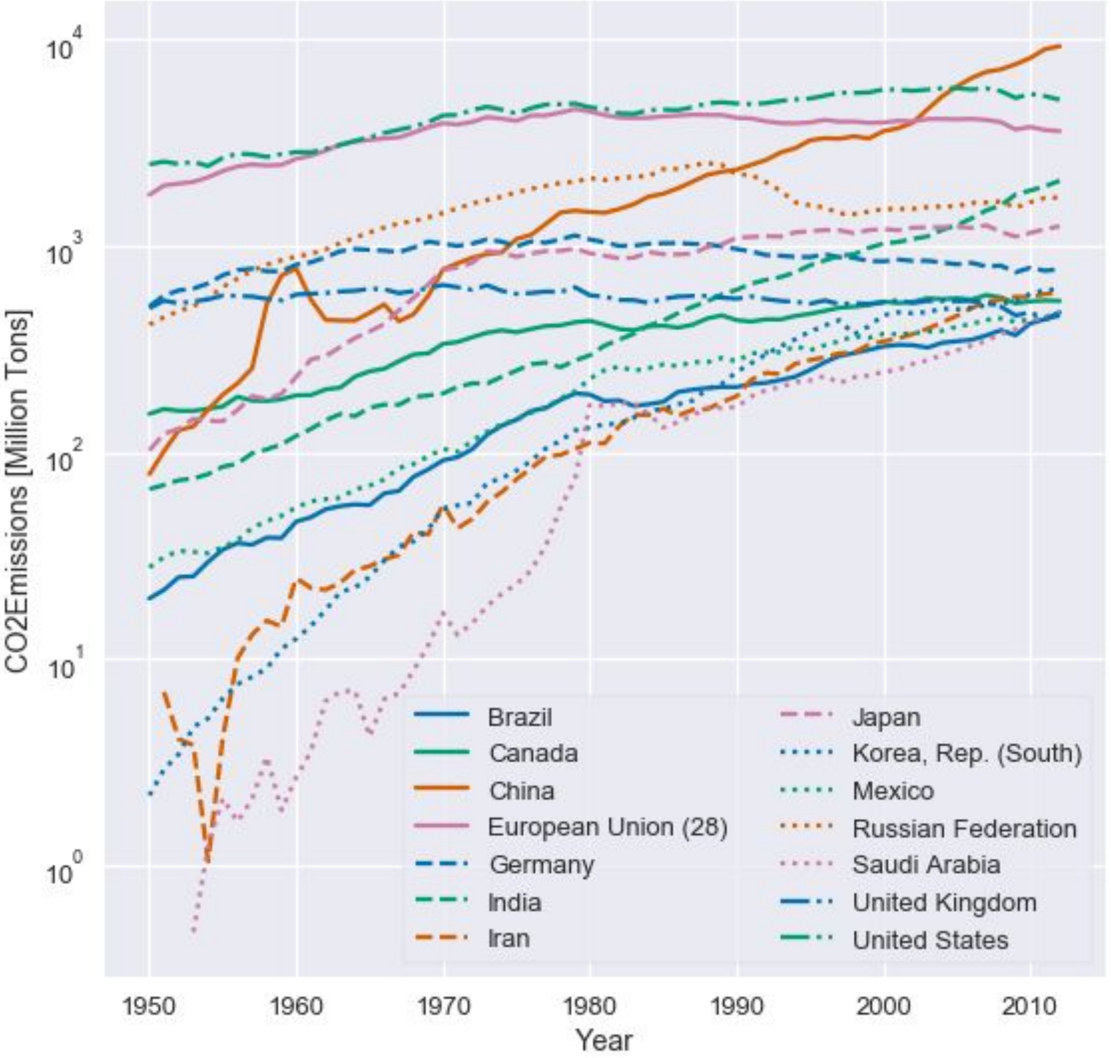
Avoid Jiggling Baseline

- Stacked area charts hard to read because baseline moves



Avoid Jiggling Baseline

Instead, plot lines themselves



Break!

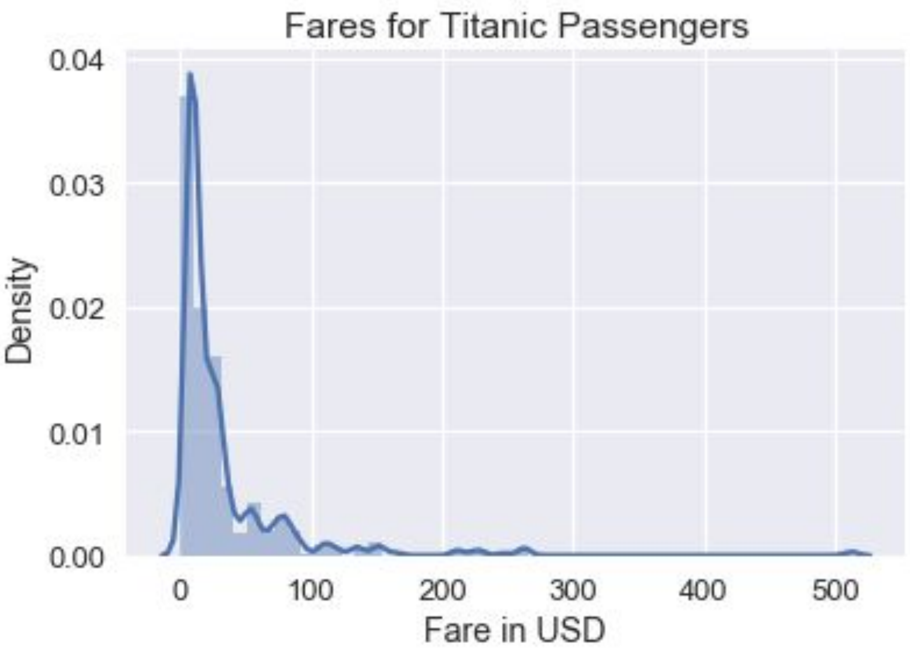
Fill out Attendance:

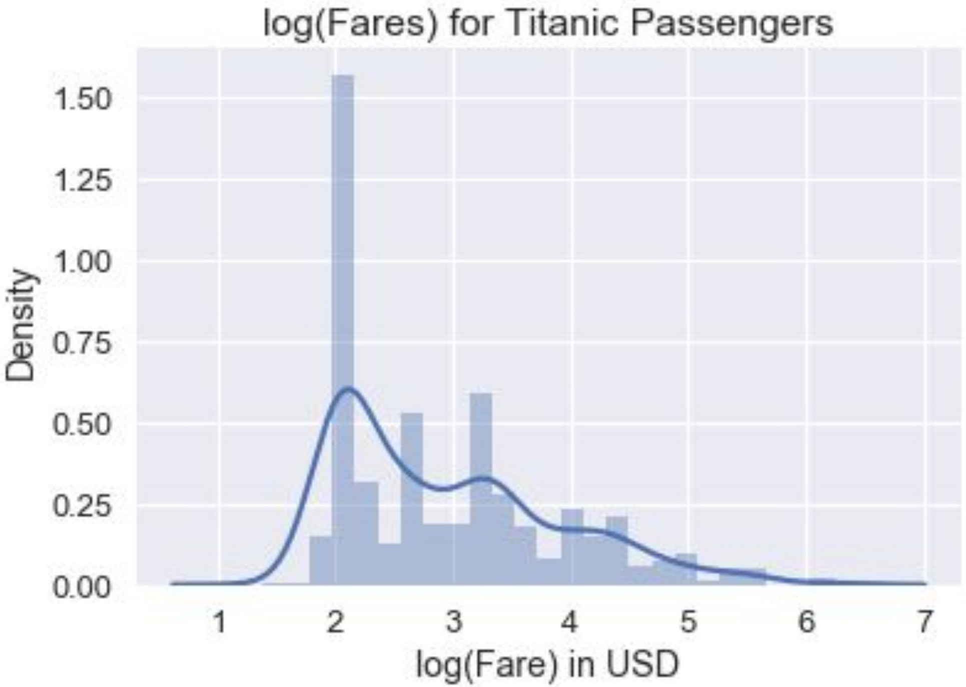
http://bit.ly/at-d100

Principles of Transformation

Transforming Data Can Reveal Patterns

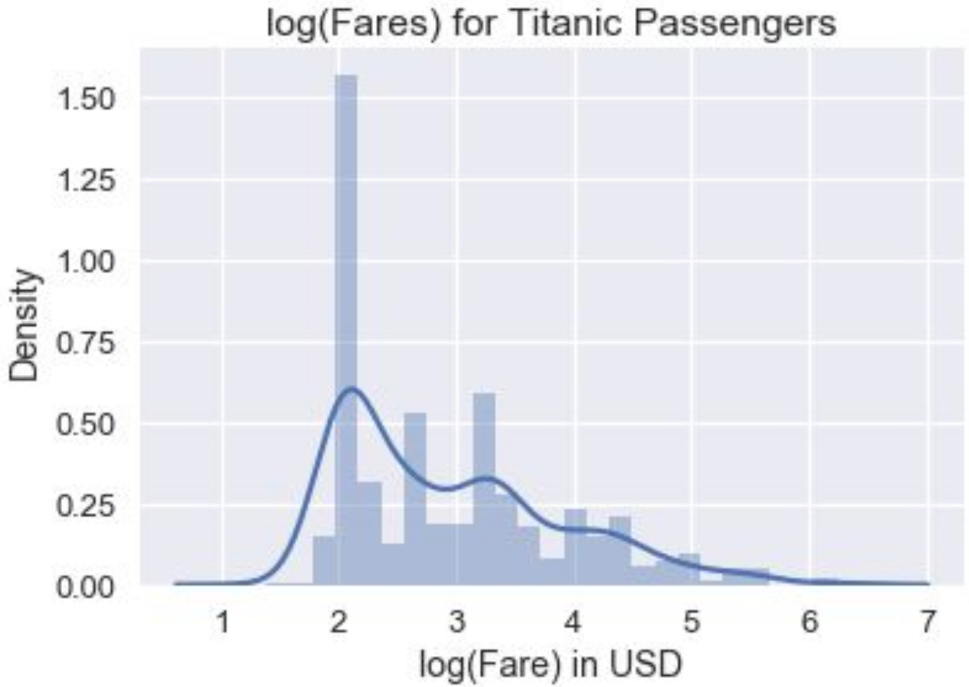
- When data are heavy tailed, useful to take the log and replot





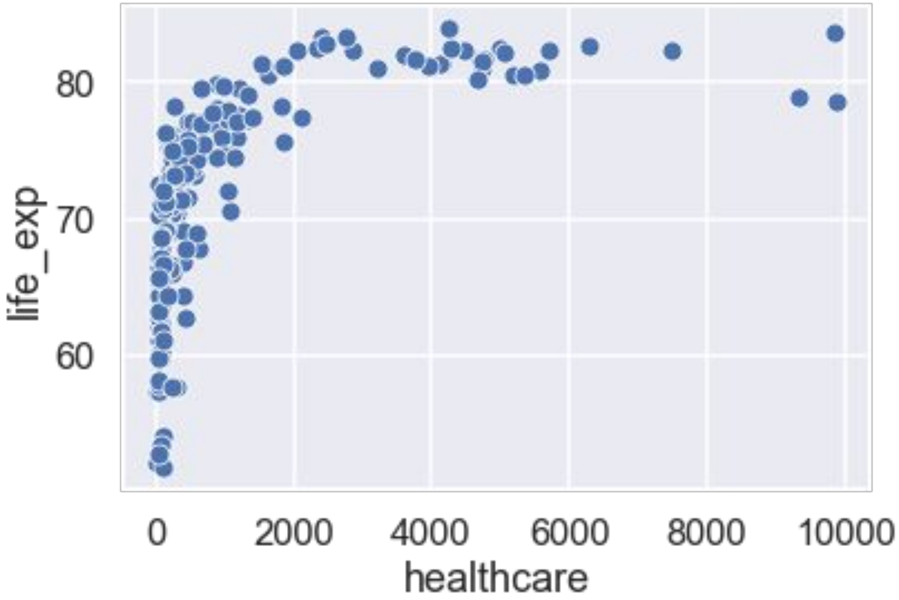
Transforming Data Can Reveal Patterns

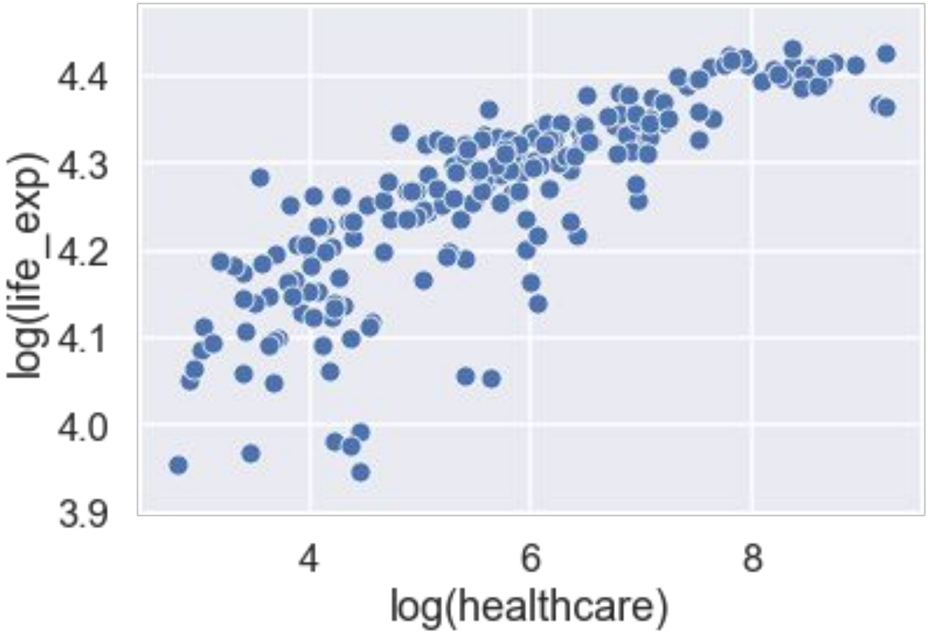
- Shows a mode when log(fare) = 2 and a smaller mode at 3.4.
- What do these correspond to in actual dollars?
- exp(2) = $7.4
- exp(3.4) = $30



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:

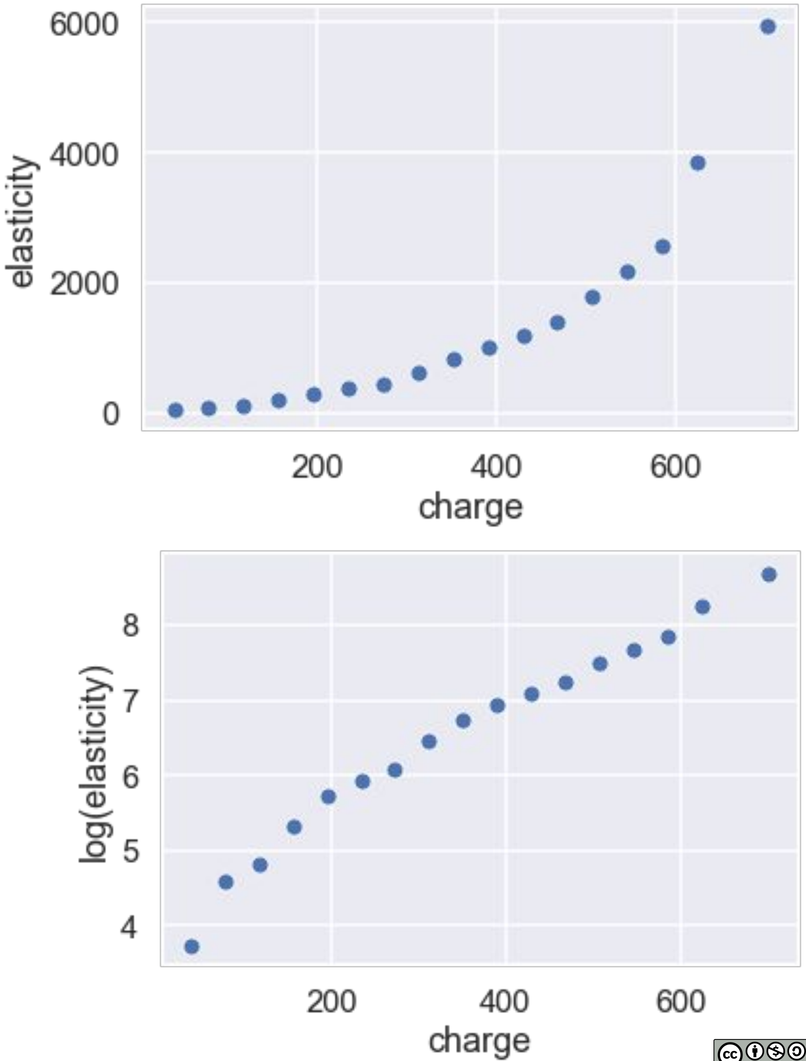
$$\log y = ax + b$$

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

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

$$y = Ce^{ax}$$

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

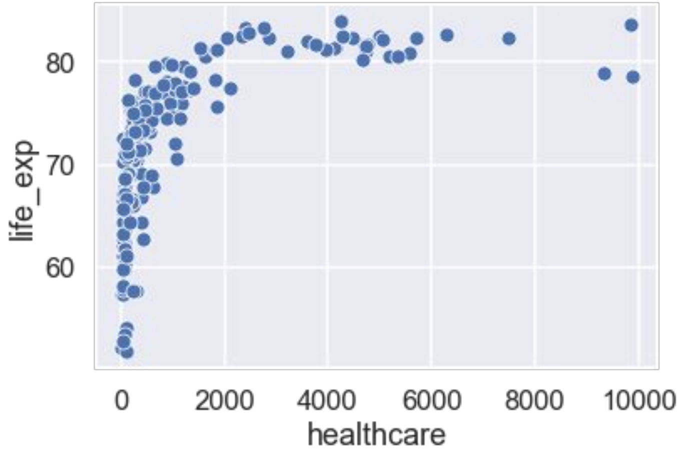


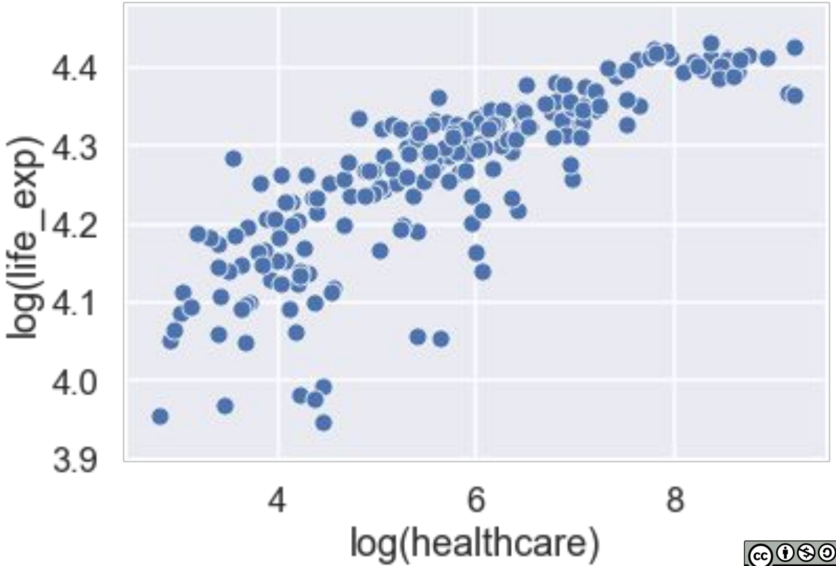
Log of both x and y-values

Fit line to log of x and y-values:

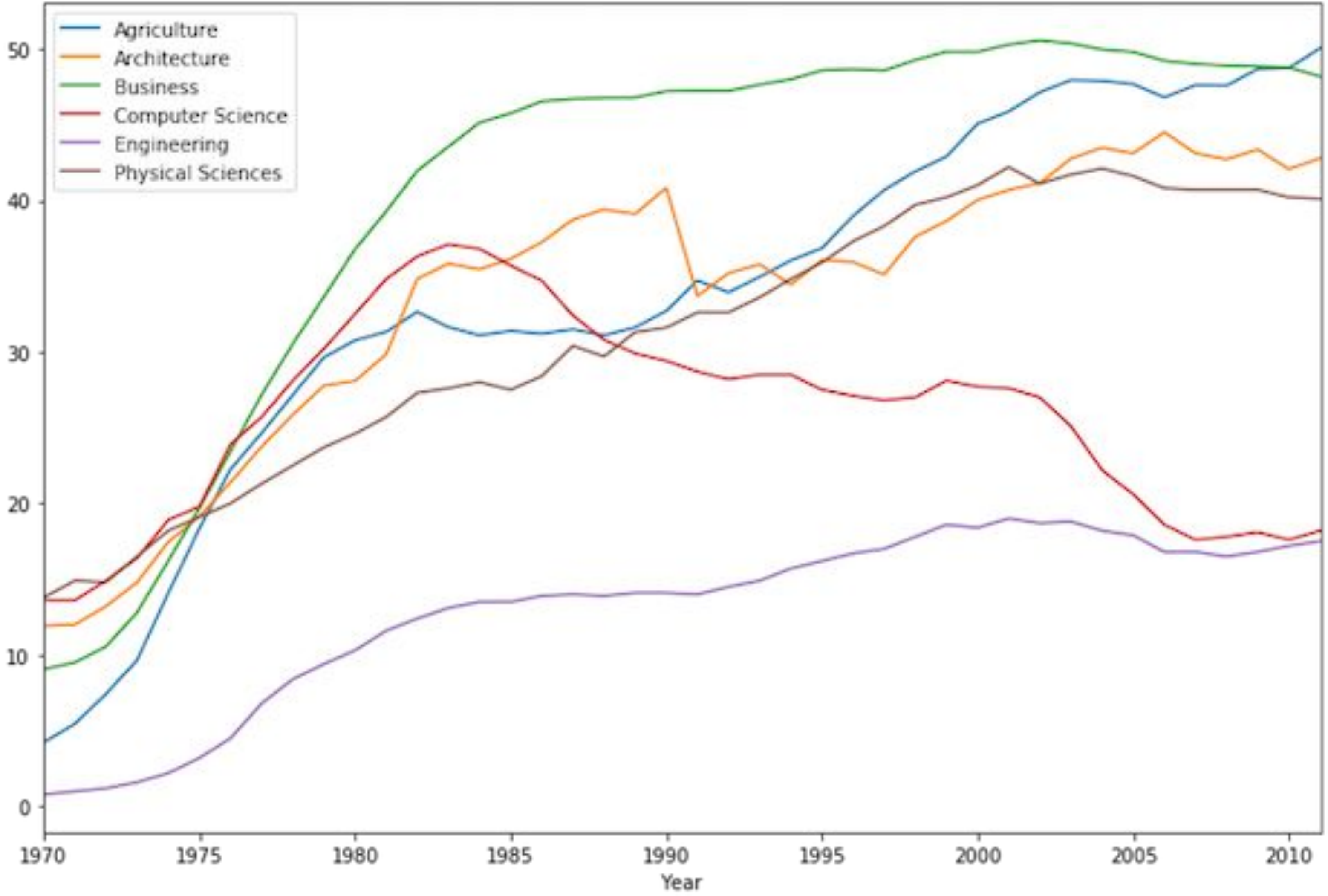
$$\log y = a \cdot \log x + b$$
$$y = e^{a \cdot \log x + b}$$
$$y = Ce^{a \cdot \log x}$$
$$y = Cx^a$$

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

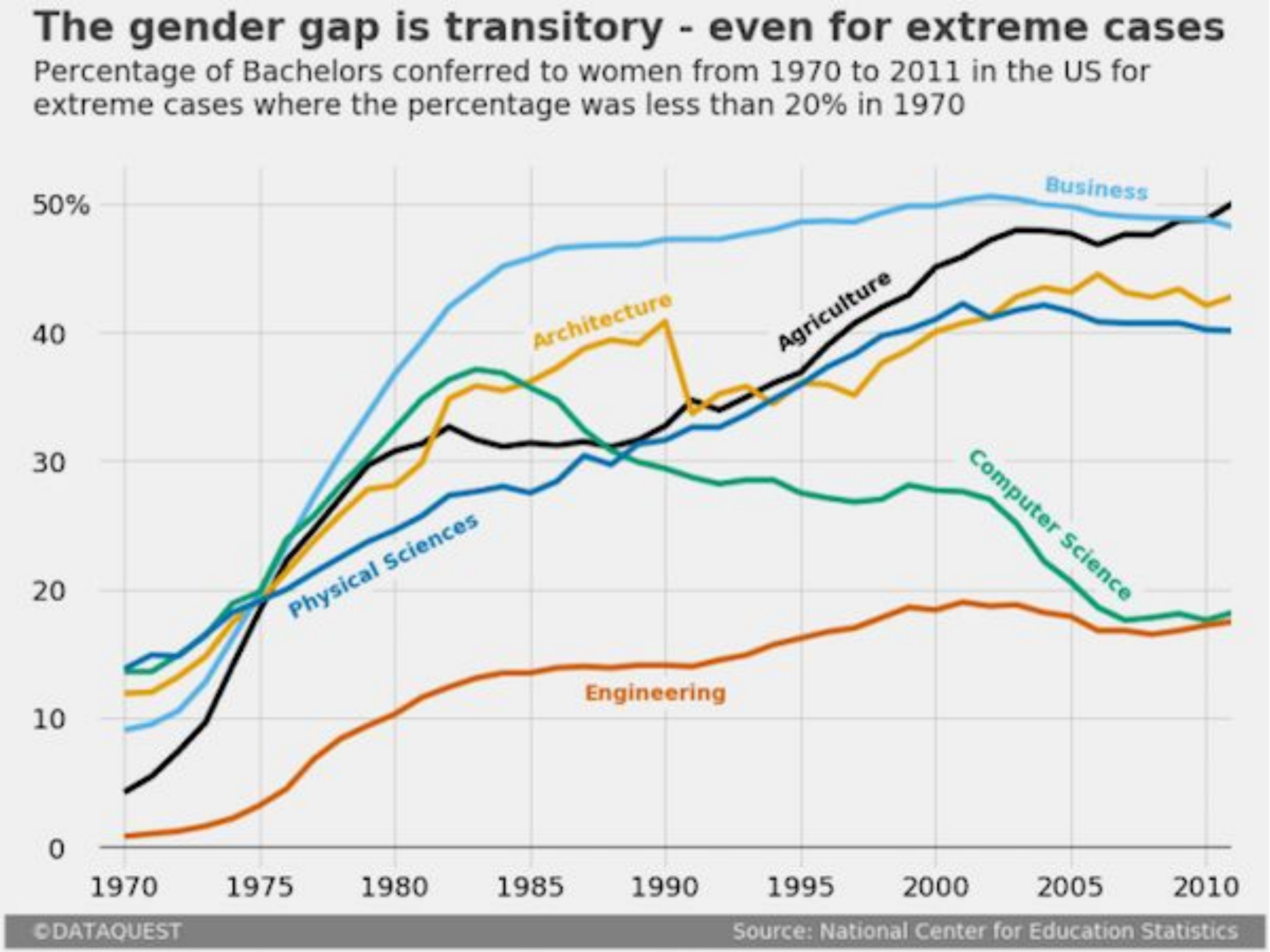




Principles of Context



Agriculture
Architecture
Business
Computer Science
Engineering
Physical Sciences
50
40
30
20
10
0
1970
1975
1980
1985
1990
1995
2000
2005
2010
Year


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
50%
40
30
20
10
0
1970
1975
1980
1985
1990
1995
2000
2005
2010
Business
Architecture
Agriculture
Physical Sciences
Computer Science
Engineering
©DATAQUEST
Source: National Center for Education Statistics

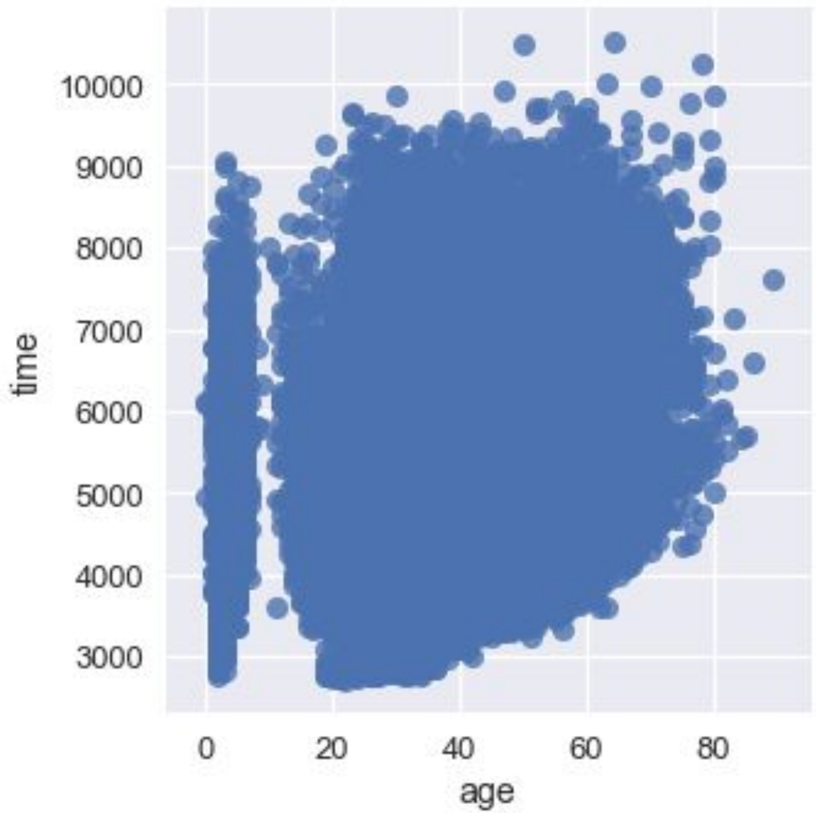
Add Context Directly to Plot

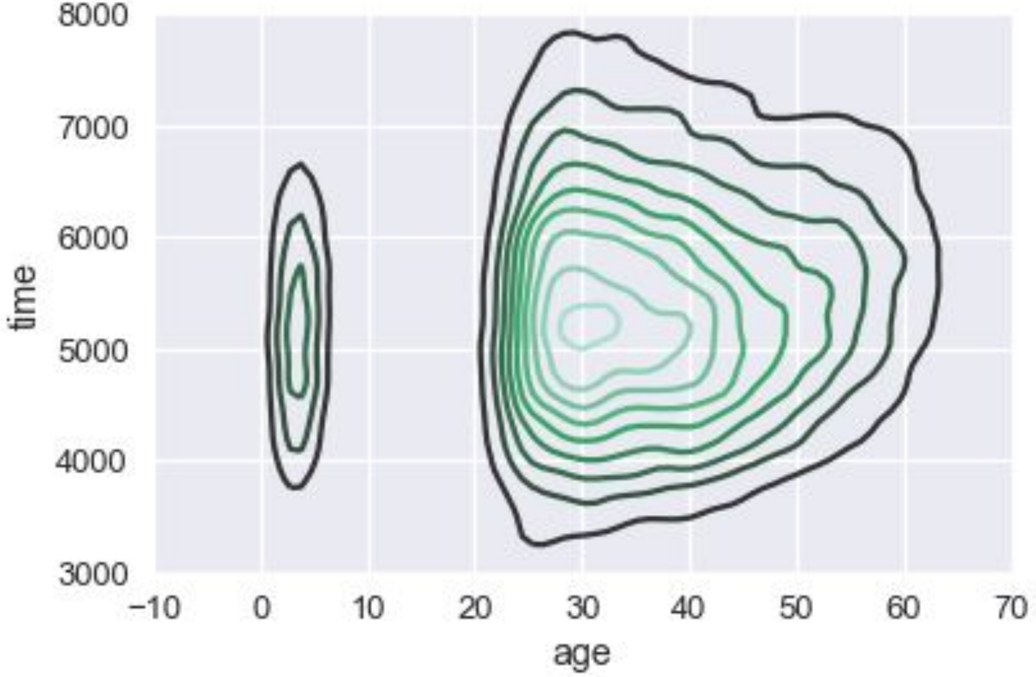
A publication-ready plot **needs**:

- 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

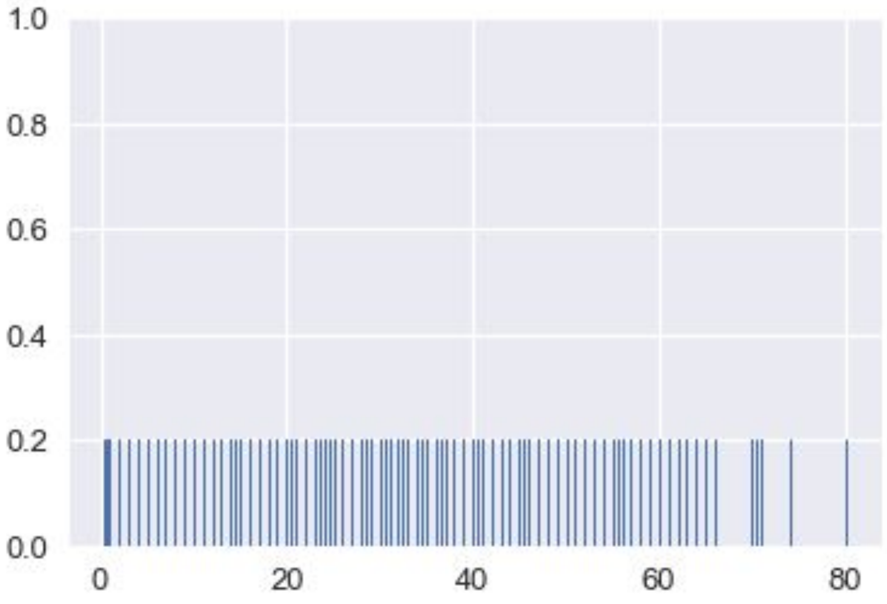
Principles of Smoothing

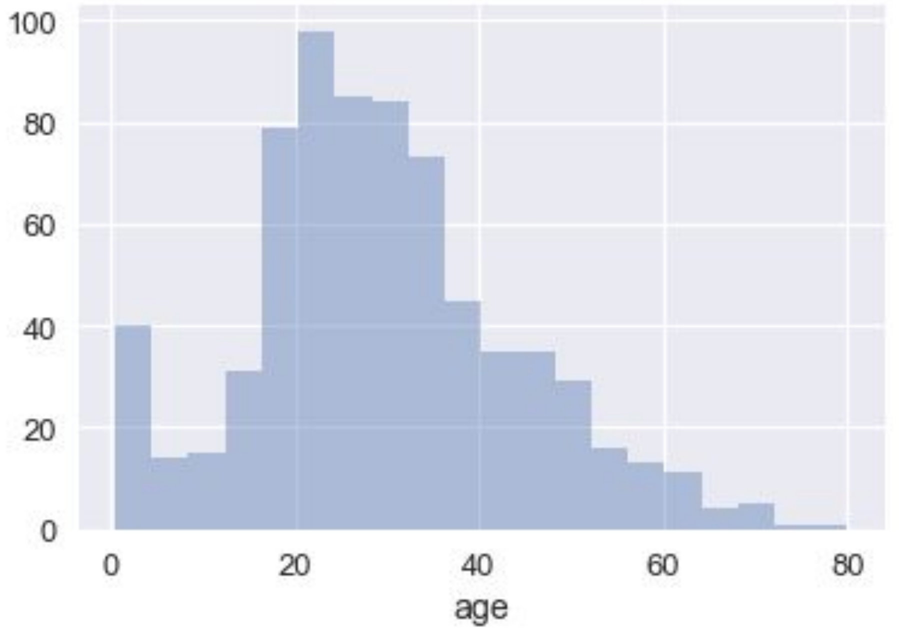
Apply Smoothing for Large Datasets



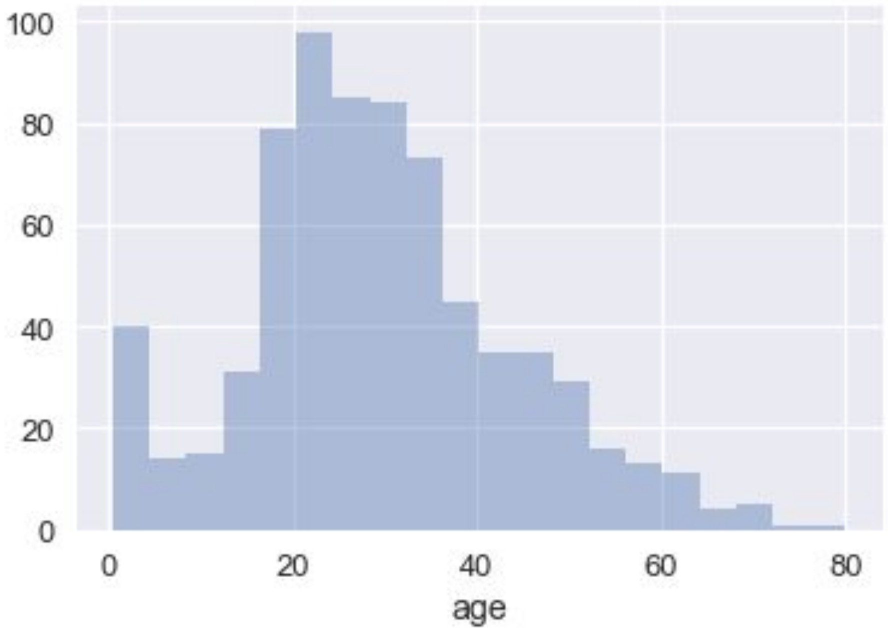


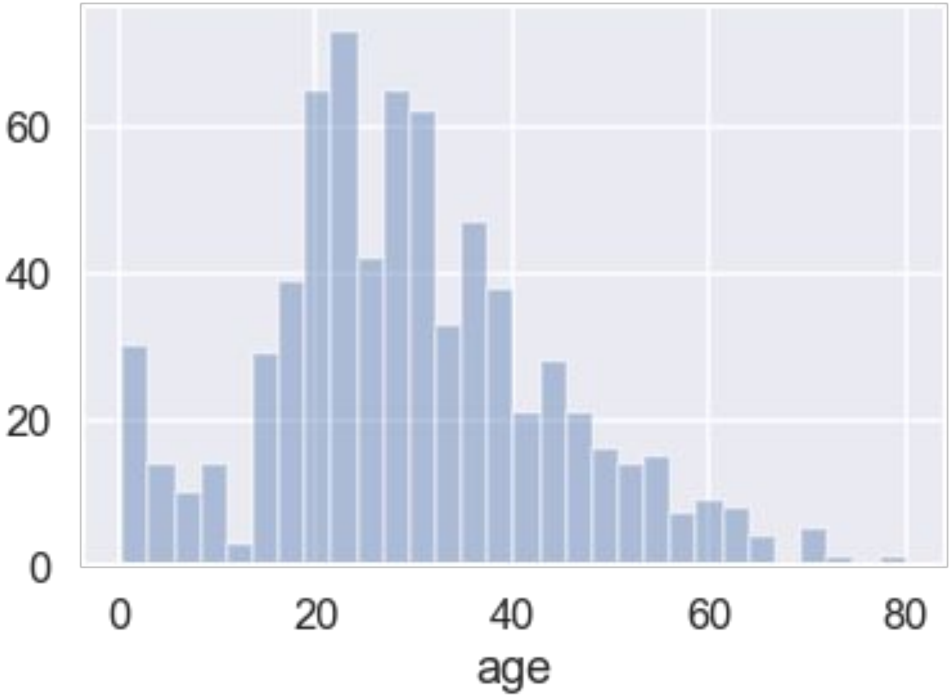
A Histogram is a Smoothed Rug Plot





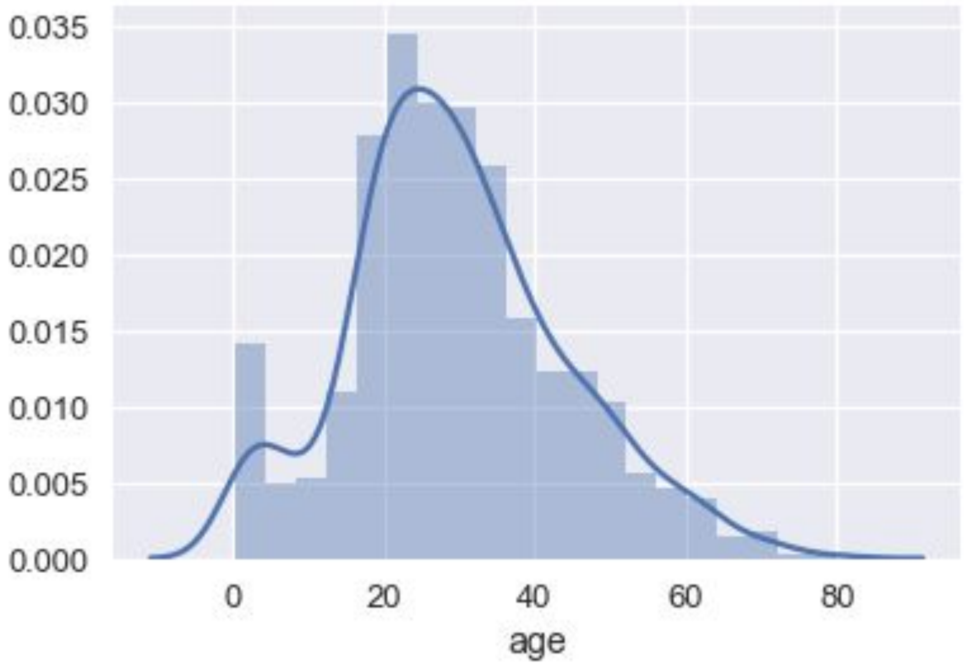
Smoothing Needs Tuning





Kernel Density Estimation (KDE)

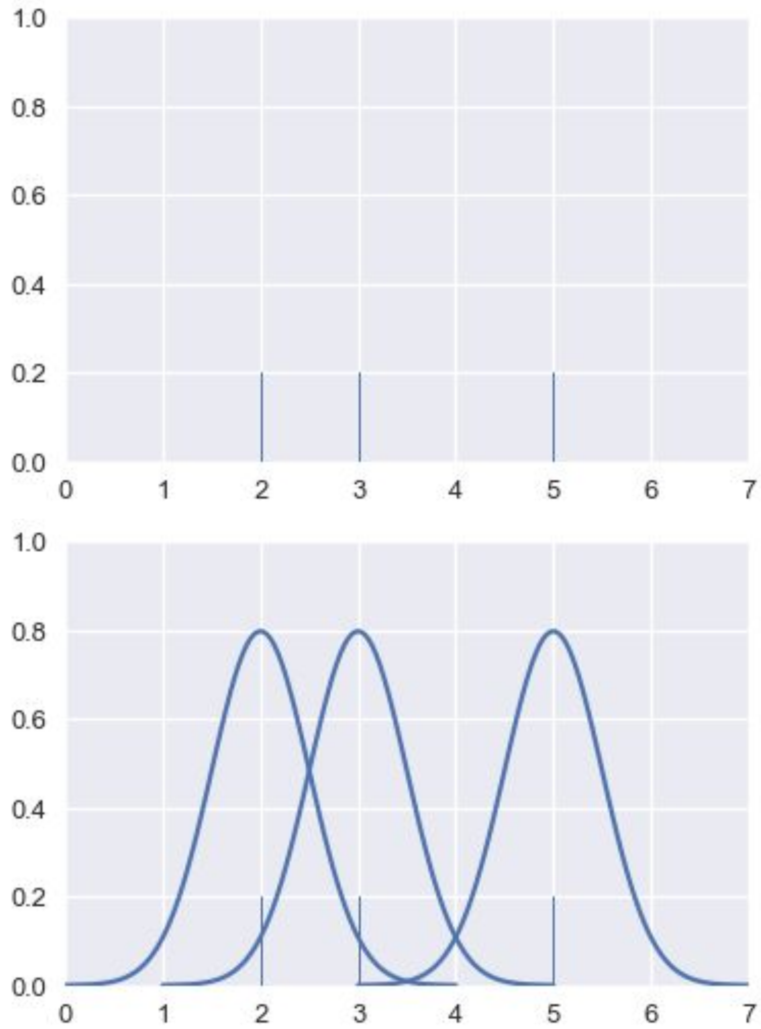
- Sophisticated smoothing technique
- Used to estimate a probability density function from a set of data



Kernel Density Estimation

Intuition:

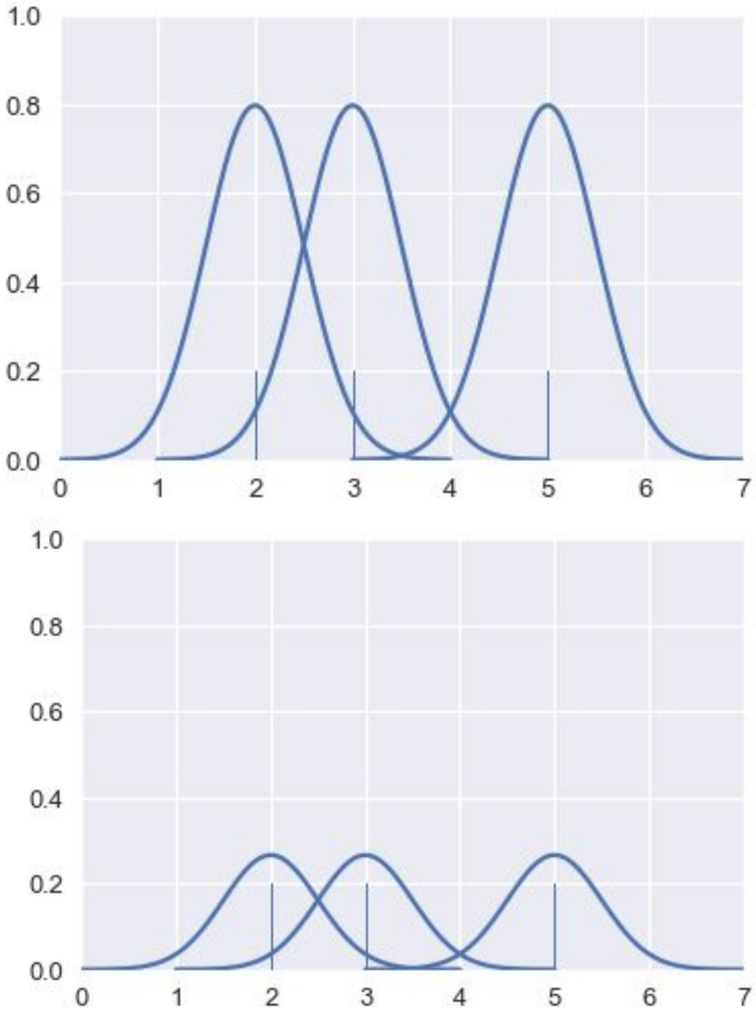
1. Place a “kernel” at each data point



Kernel Density Estimation

Intuition:

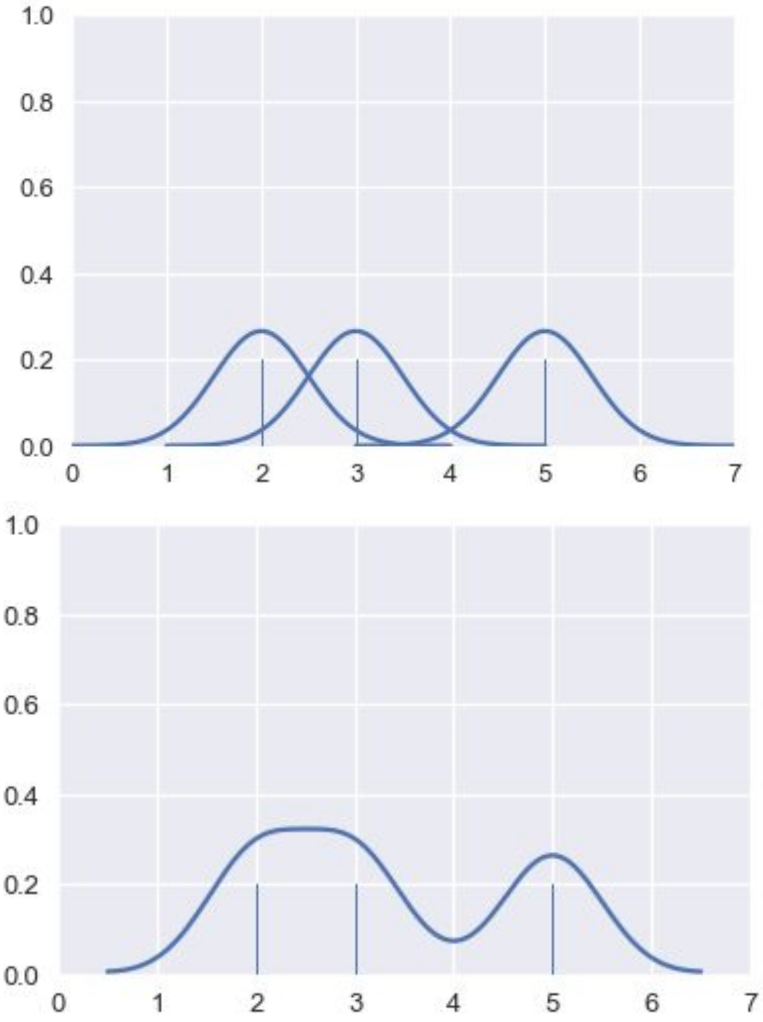
1. Place a “kernel” at each data point
2. Normalize kernels so that total area = 1



Kernel Density Estimation

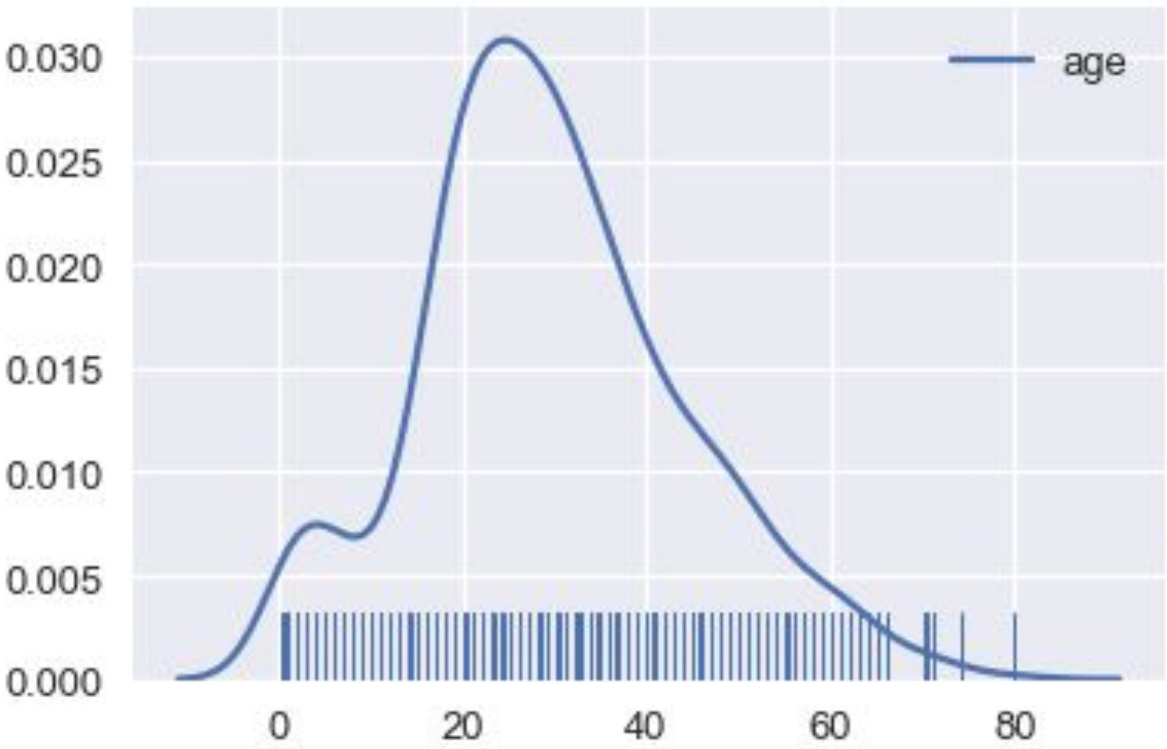
Intuition:

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



Kernel Density Estimation

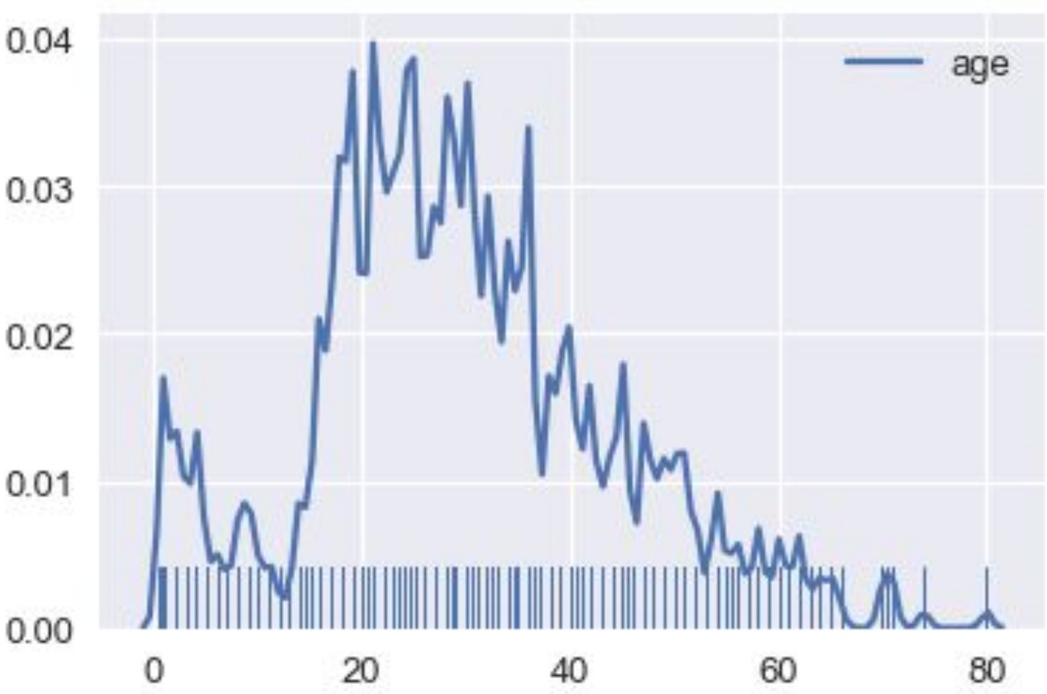
Gaussian kernel most common (default for seaborn).



Kernel Density Estimation

Changing width of each kernel = changing bandwidth

Narrow bandwidth is analogous to narrow bins for histogram

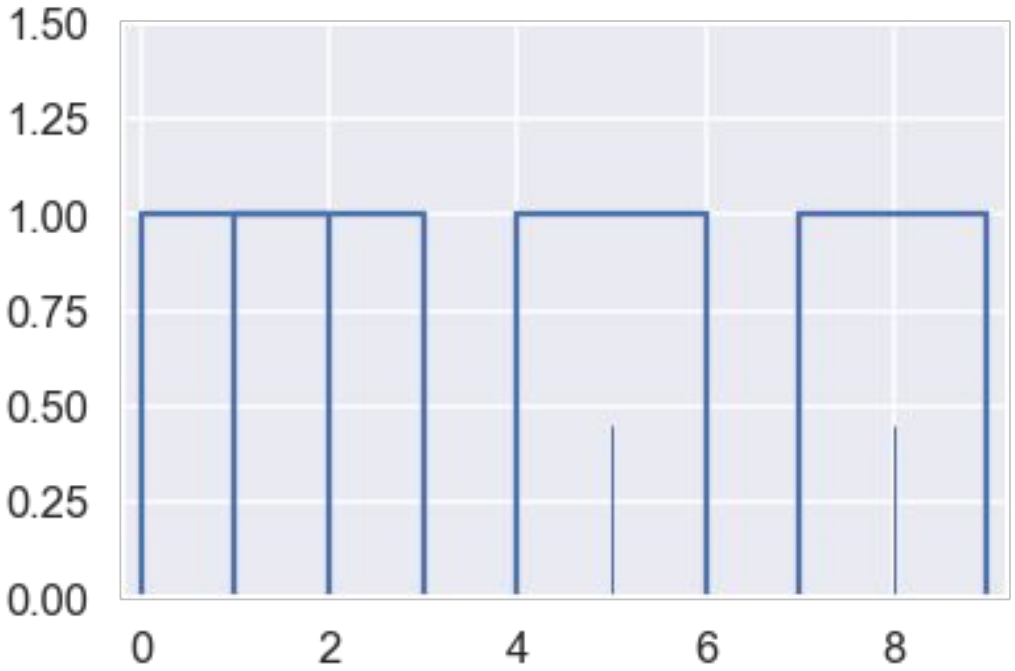


KDE Example — Uniform Kernel

Uniform kernel with bandwidth of 2.

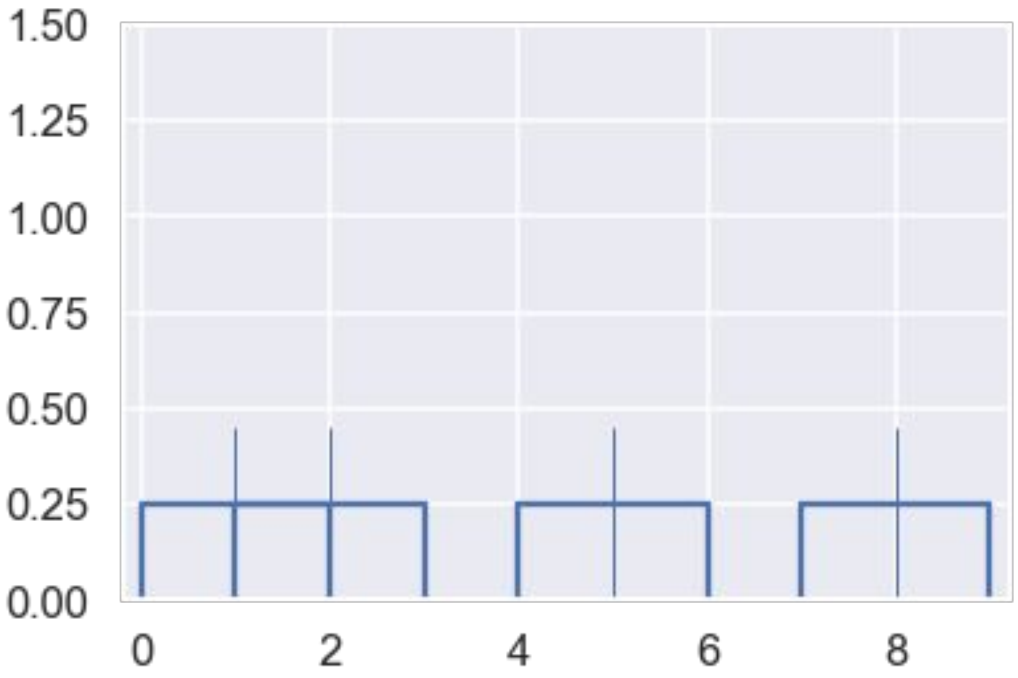
Data points at: $x = [1, 2, 5, 8]$

Kernel at each x:



KDE Example — Uniform Kernel

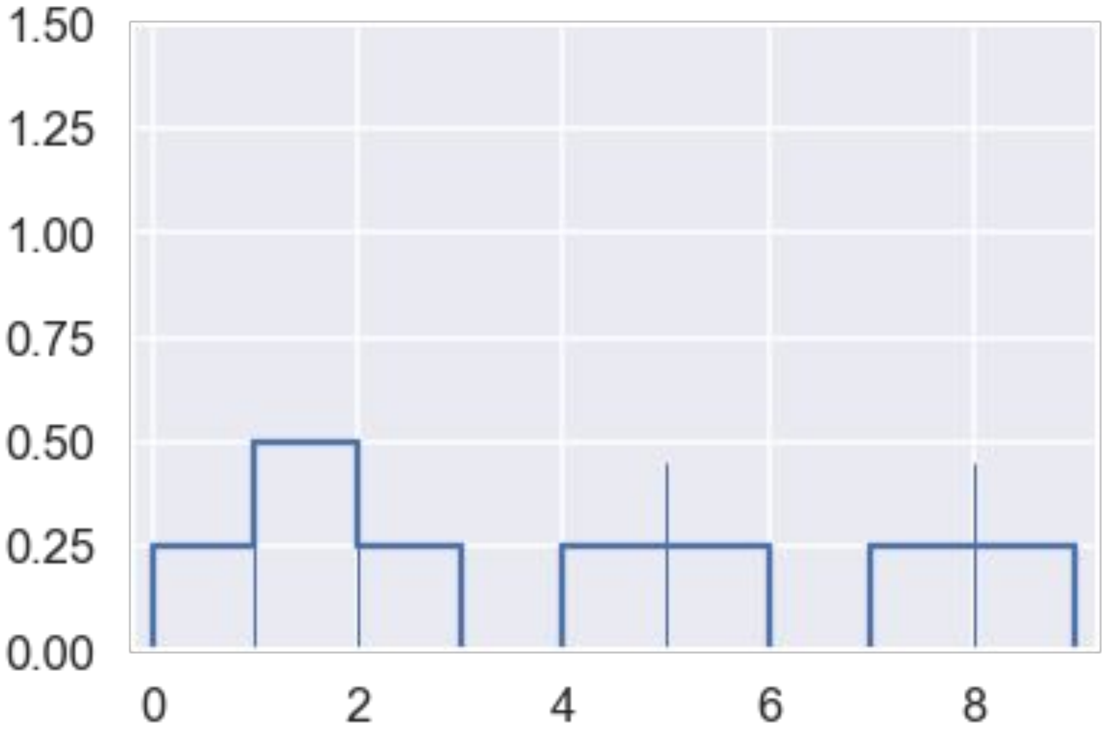
Scale each kernel by 1/4 since there are four points:



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