CMPT 733
Data Visualization (I)

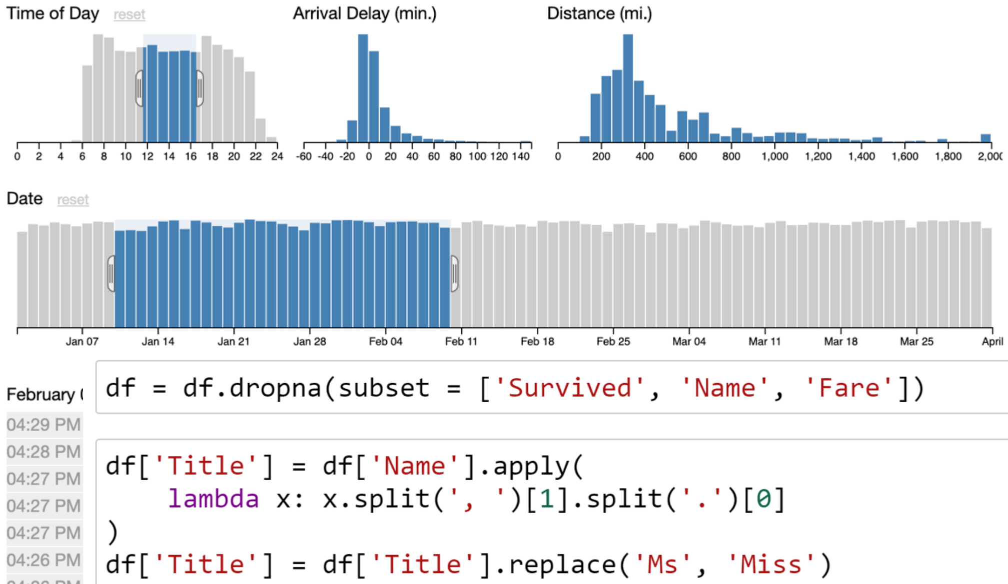
Instructor Zhengjie Miao

Course website https://coursys.sfu.ca/2025sp-cmpt-733-g1/pages/

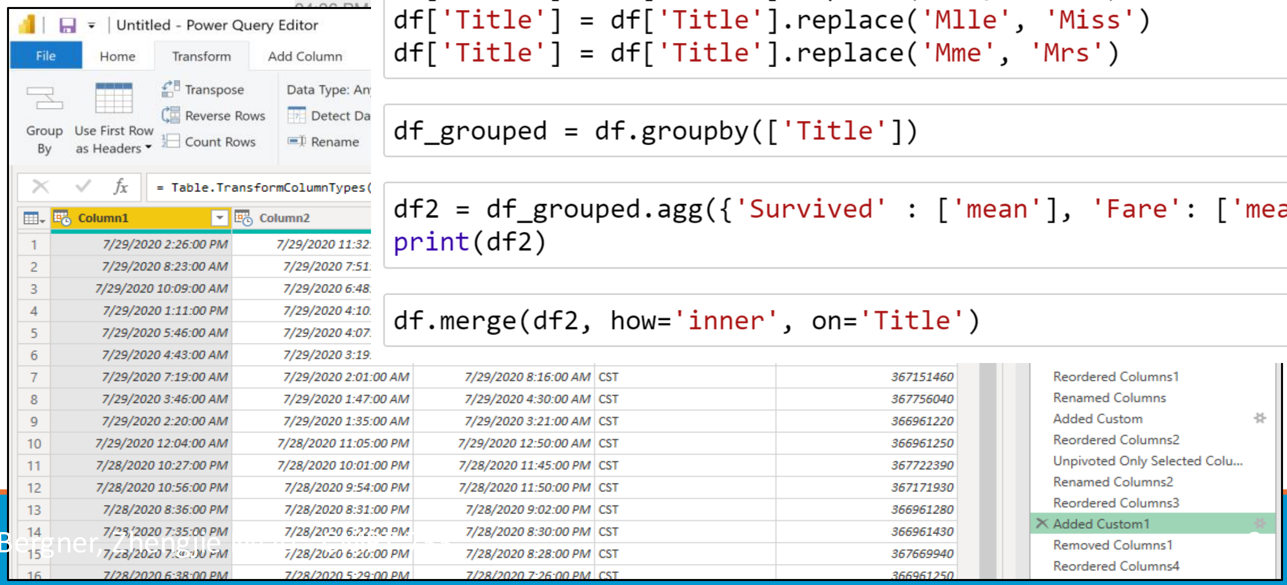
Outline

- Overview of Visualization: What, Why, and How?
- What & How: Examples
- Why: Goals
- How: Guidelines and techniques (in the second data vis lecture)

Recap: Human-Data Interface




```python
df = df.dropna(subset = ['Survived', 'Name', 'Fare'])

df['Title'] = df['Name'].apply(
    lambda x: x.split(', ')[1].split('.')[0]
)
df['Title'] = df['Title'].replace('Ms', 'Miss')
df['Title'] = df['Title'].replace('Mlle', 'Miss')
df['Title'] = df['Title'].replace('Mme', 'Mrs')

df_grouped = df.groupby(['Title'])

df2 = df_grouped.agg({'Survived' : ['mean'], 'Fare': ['mean']})
print(df2)

df.merge(df2, how='inner', on='Title')
```



Recap: Data Science Pipeline

What	When	Who	Goal
Computer Science	1950-	Software Engineer	Write software to make computers work

Plan → Design → Develop → Test → Deploy → Maintain

What	When	Who	Goal
Data Science	2010-	Data Scientist	Extract insights from data to answer questions

Collect→ Clean → Integrate → Analyze → Visualize → Communicate

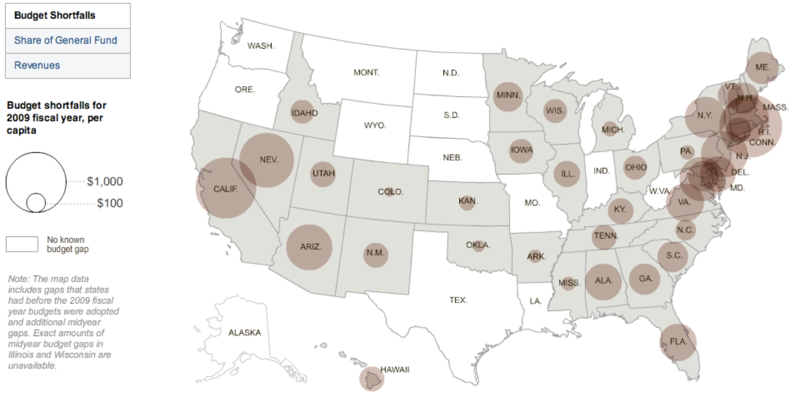
- What role does Visualization play?

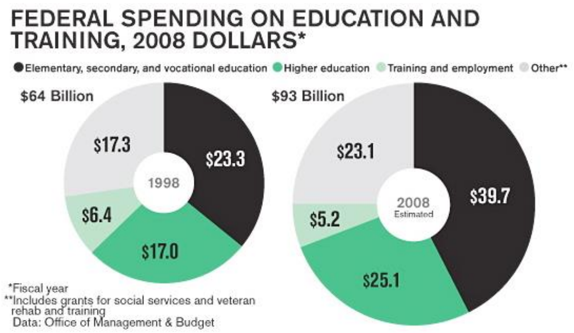
What is Data Visualization?

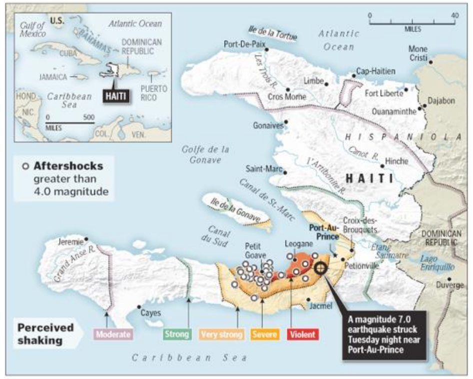
vi·su·al·ize [The American Heritage Dictionary]

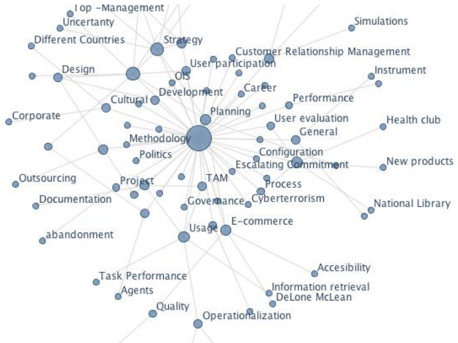
1. **To form a mental image**
2. **To make visible**

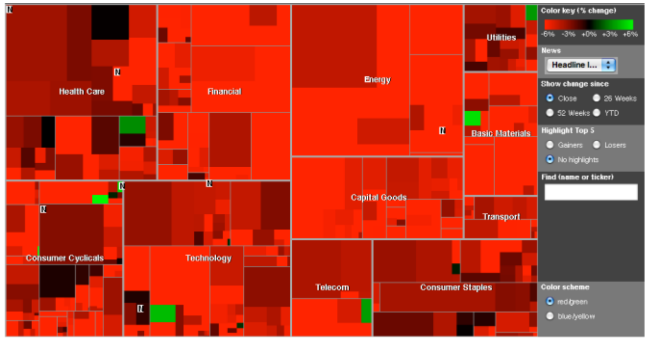
Visualization: To convey information through visual representations

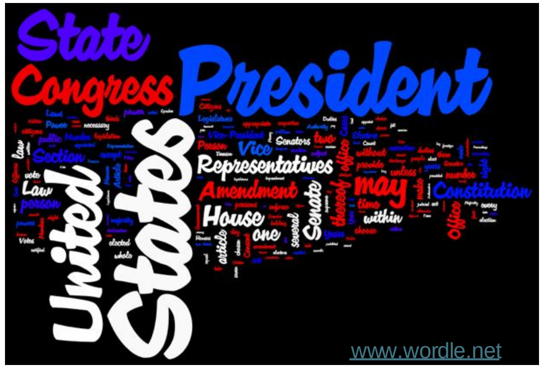








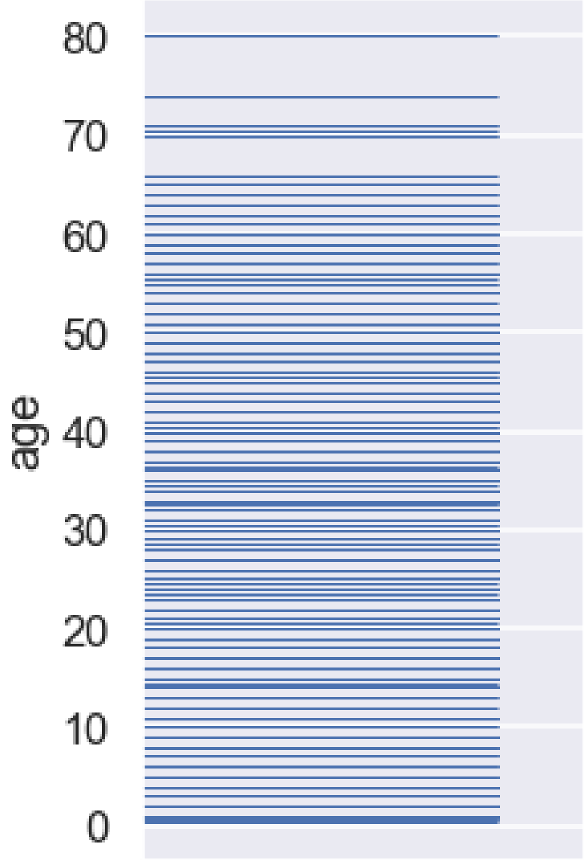




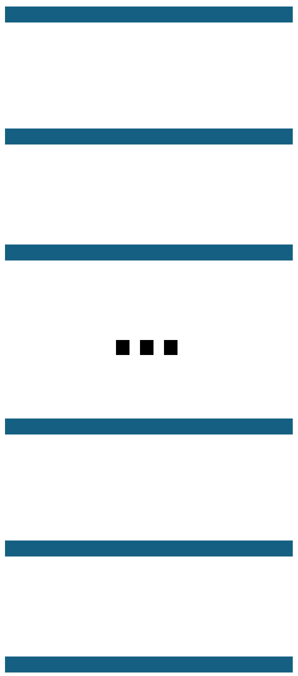
Computer Readable

	age
0	22.0
1	38.0
2	26.0
...	...
888	NaN
889	26.0
890	32.0

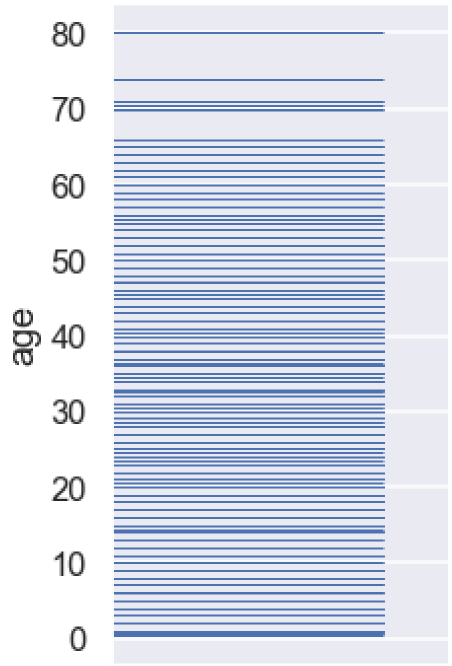
Human Readable



	age
0	22.0
1	38.0
2	26.0
...	...
888	NaN
889	26.0
890	32.0



10px
16px
11px
...
0px
11px
15px



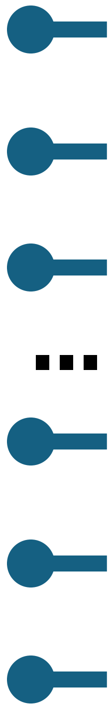
Mark

(Represents a datum)

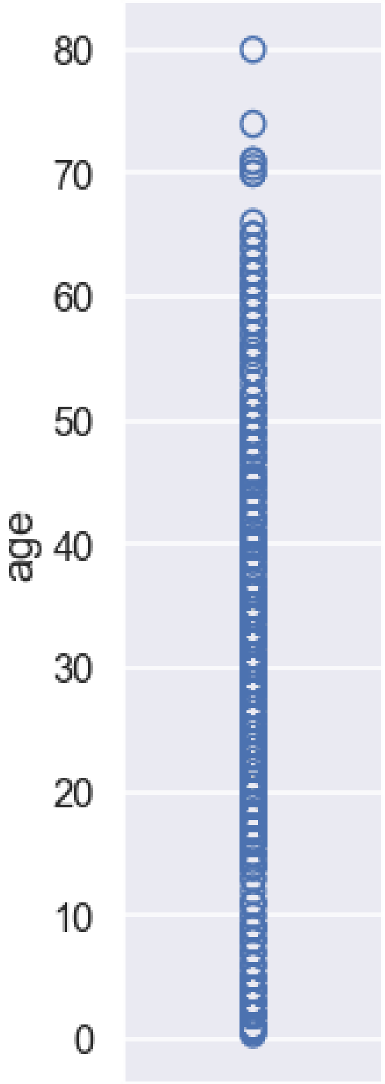
Encoding

(Maps datum to visual position)

	age
0	22.0
1	38.0
2	26.0
...	...
888	NaN
889	26.0
890	32.0



10px
16px
11px
...
0px
11px
15px



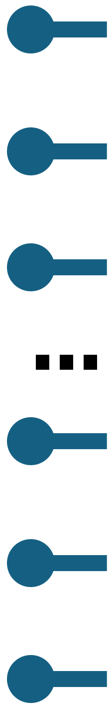
Mark

(Represents a datum)

Encoding

(Maps datum to visual position)

	age	fare
0	22.0	7.25
1	38.0	71.28
2	26.0	7.92
...	...	...
888	NaN	23.45
889	26.0	30.00
890	32.0	7.75



(10px, 7px)

(70px, 60px)

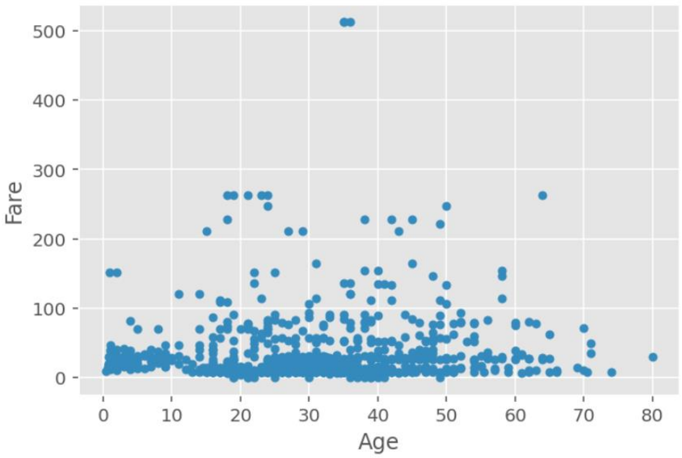
(45px, 9px)

...

(5px, 24px)

(45px, 37px)

(66px, 8px)



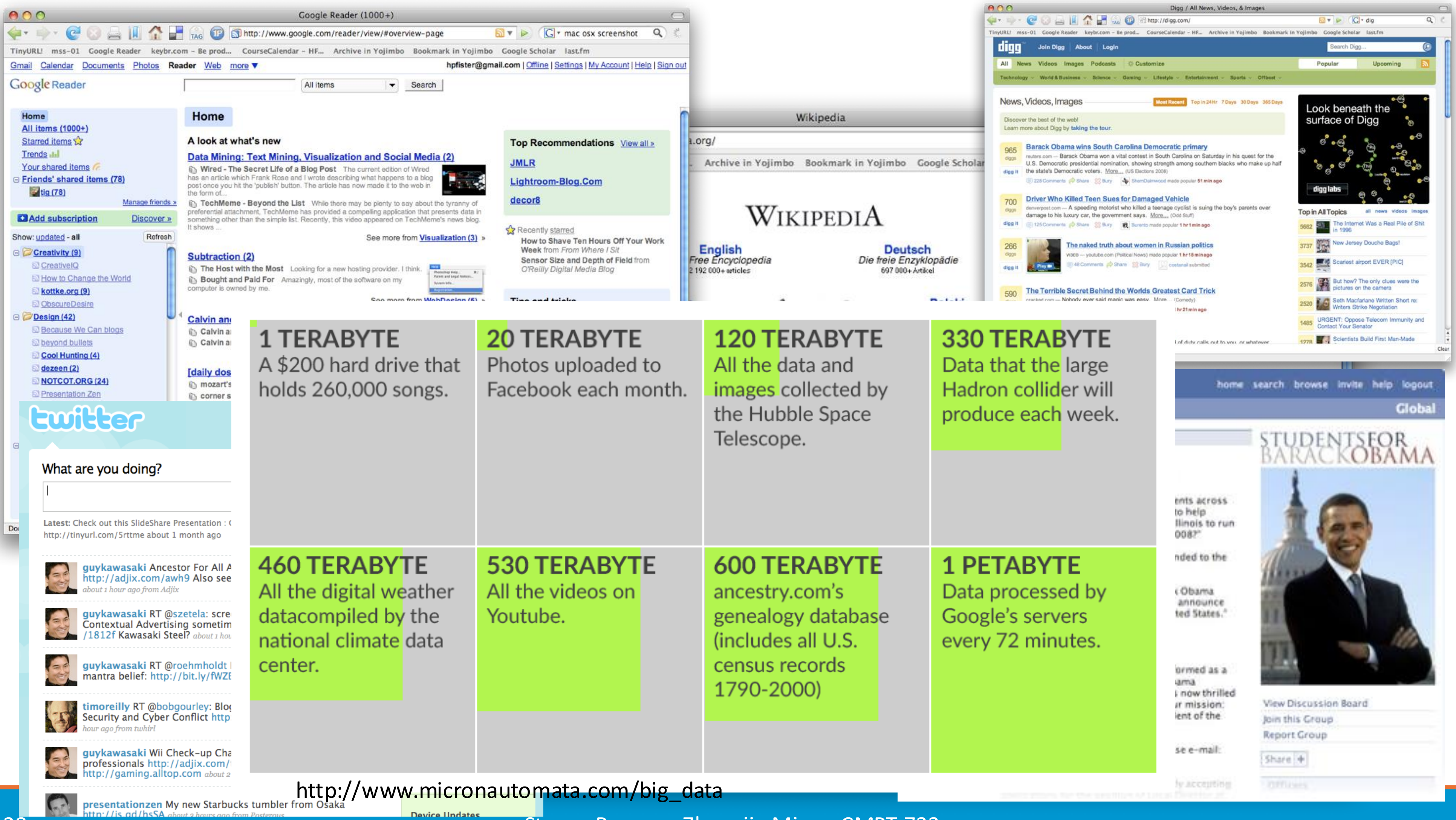
Mark

Encoding

Why Data Visualization?

- Computer-based visualization systems provide visual representations of datasets designed to help people carry out tasks more effectively.
- Visualization is suitable when there is a need to augment human capabilities
  - Don't need vis when fully automatic solution exists and is trusted
  - Many analysis problems are ill-specified; visualization helps with: exploratory analysis, building trust, presentation of results

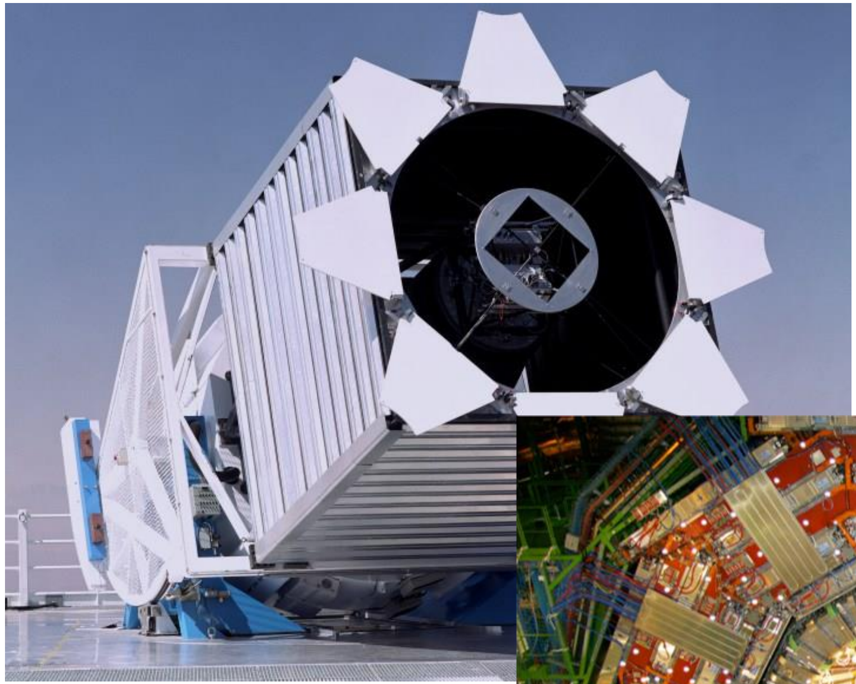
Information Explosion / Big Data



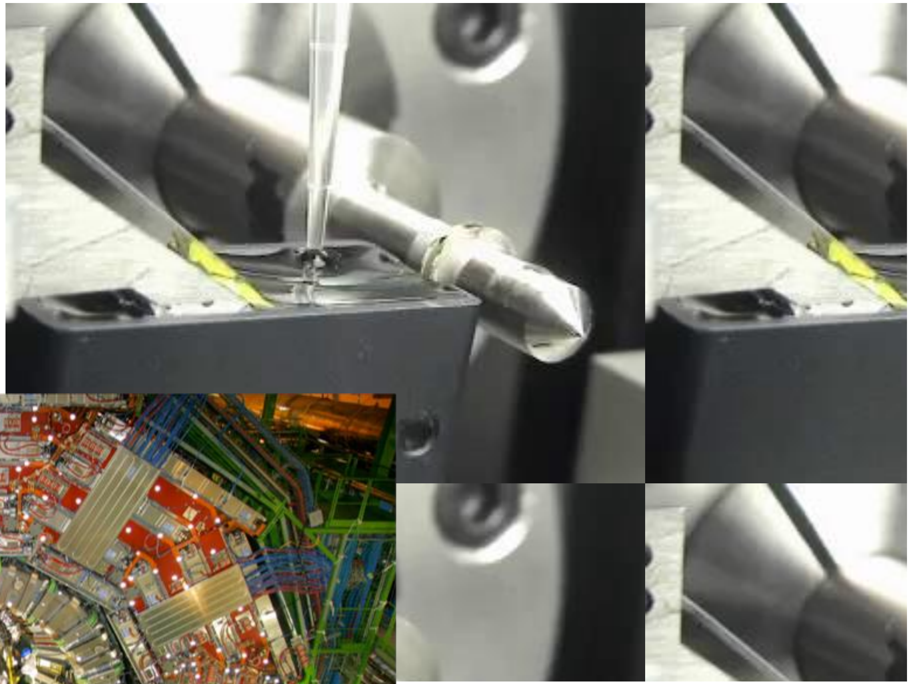
http://www.micronautomata.com/big_data

Instrument Data Explosion

Sloan Digital Sky Survey



ATLUM / Connectome Project



Maximilien Brice, © CERN

“The Industrial Revolution of Data”

Joe Hellerstein, UC Berkeley

Human attention becomes scarce

- “What information consumes is rather obvious: it consumes the attention of its recipients. Hence a wealth of information creates a poverty of attention, and a need to allocate that attention efficiently among the overabundance of information sources that might consume it.”

--- Herbert A. Simon (Nobel laureate economist)

Hal Varian, “How much will two bits be worth in the digital marketplace?”, Scientific American (1995)

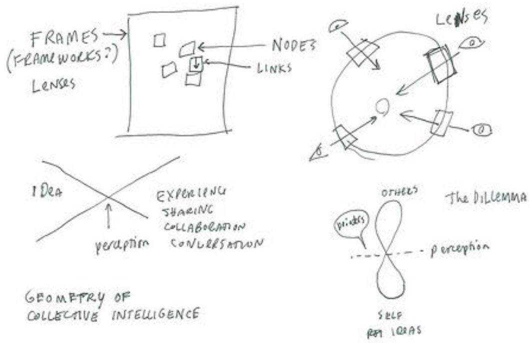
How to combat with scarcity in human attention?

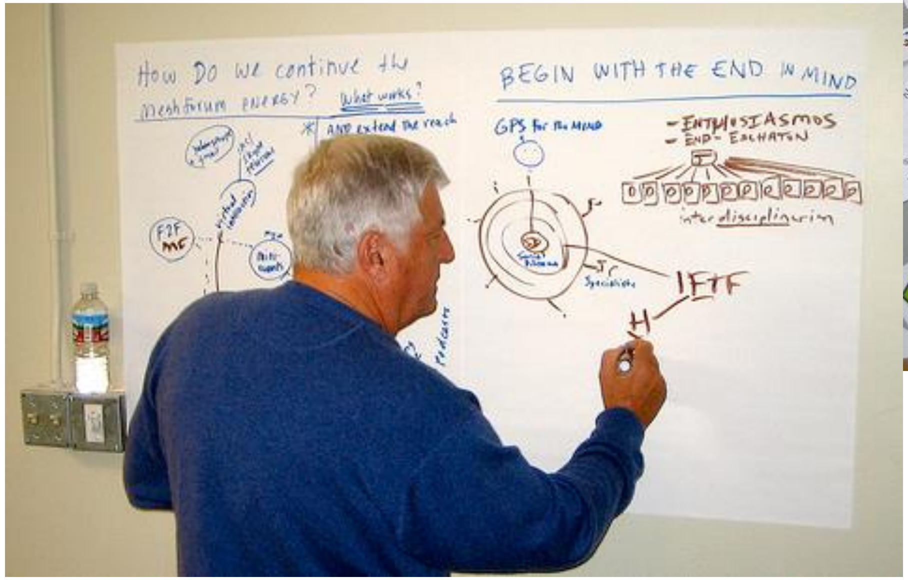
- Technology for producing and distributing information is useless without some way to locate, filter, organize and summarize it.
- A new profession of “information managers” will have to combine the skills of computer scientists, librarians, publishers and database experts to help us discover and manage information.

Visualization for easier human consumption!

Hal Varian, “How much will two bits be worth in the digital marketplace?”, Scientific American (1995)

“It is things that make us smart.”
Donald Norman





Visual Thinking Collection, Dave Grey

Idea Maps, by Jamie Nast

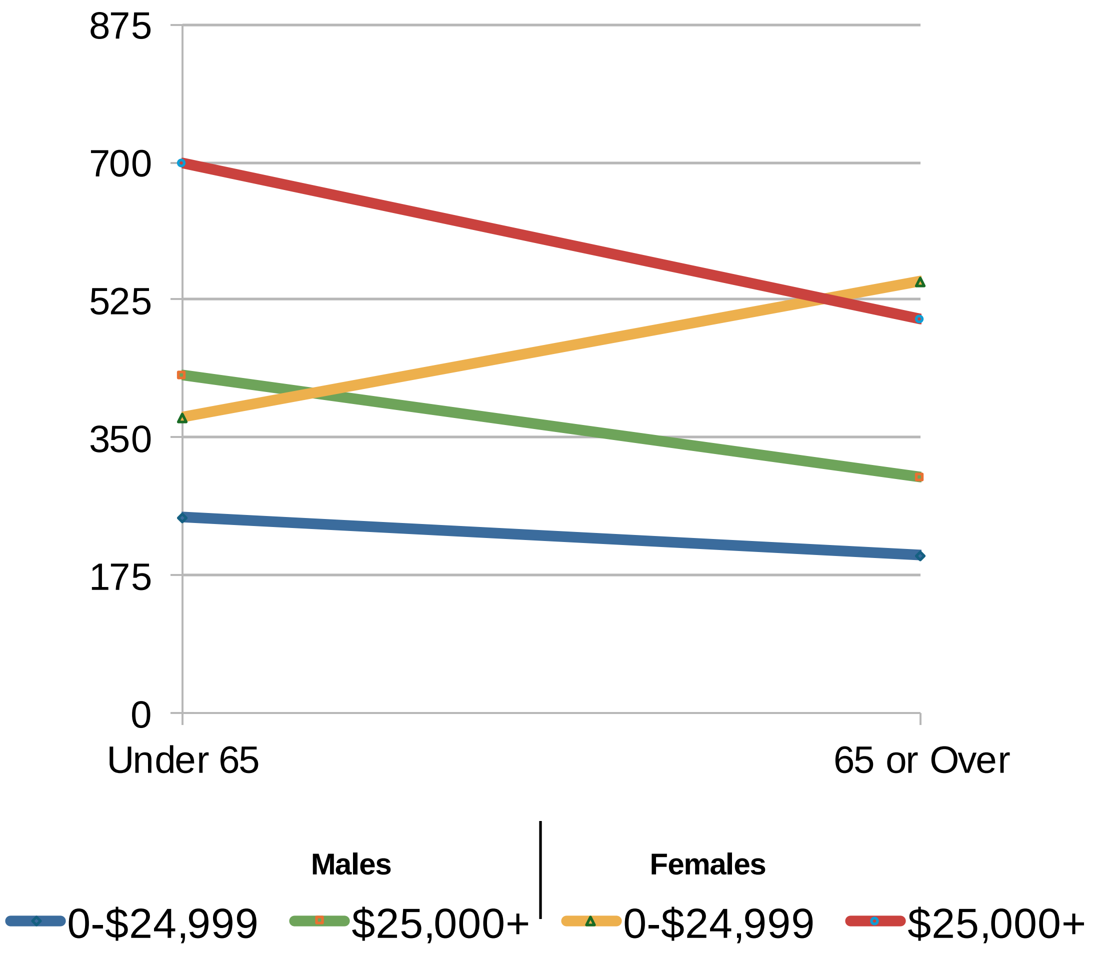
Mental Queries

Which gender or income level group shows different effects of age on triglyceride levels?

	Males		**Females**	
Income Group	Under 65	65 or Over	Under 65	65 or Over
0-$24,999	250	200	375	550
$25,000+	430	300	700	500

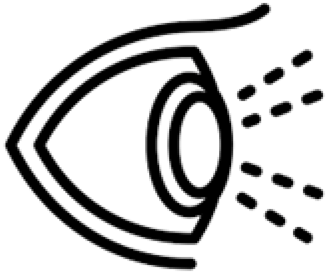
Slide after Stephen Kosslyn, Clear and to the Point

Visual Queries

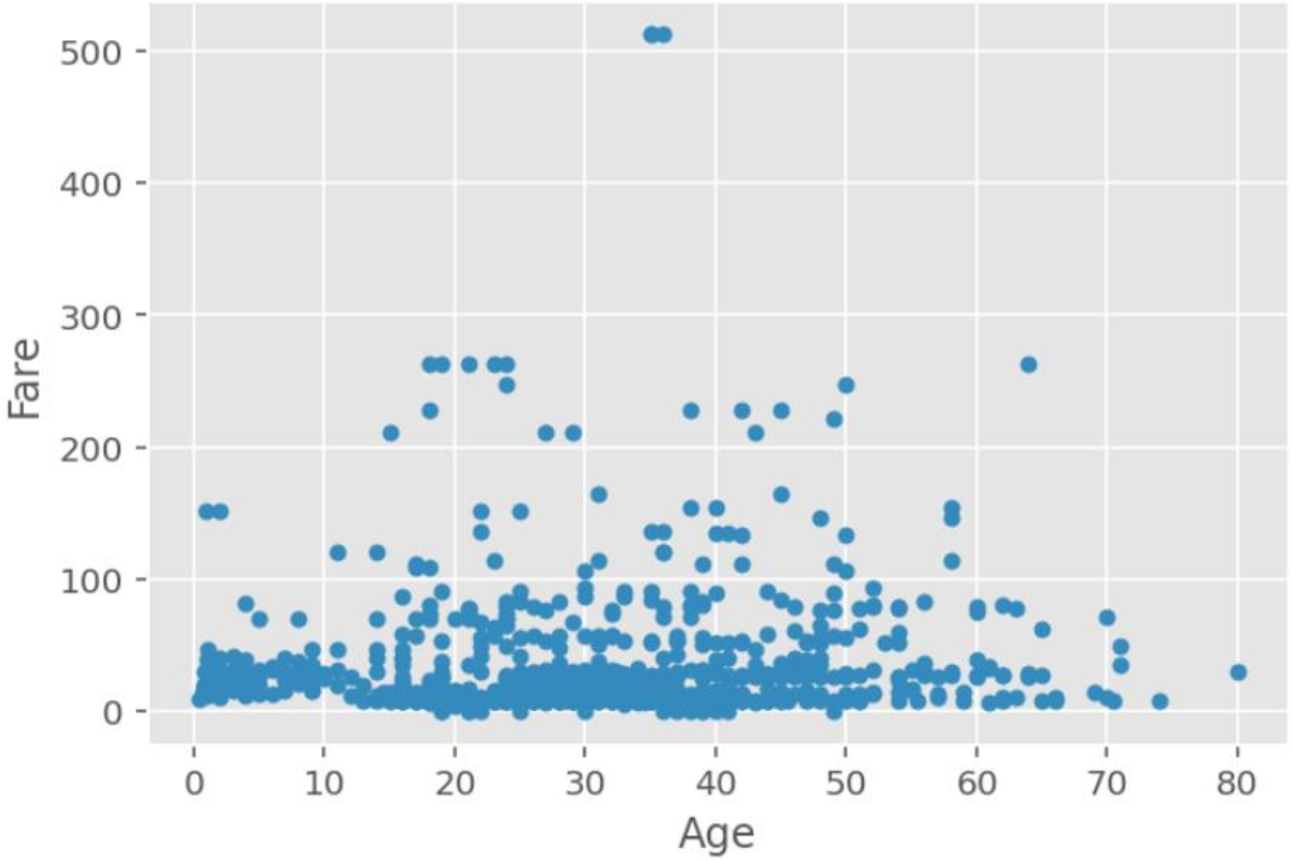


Slide after Stephen Kosslyn, Clear and to the Point

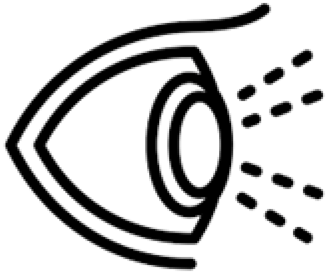
Visualizations are for Humans



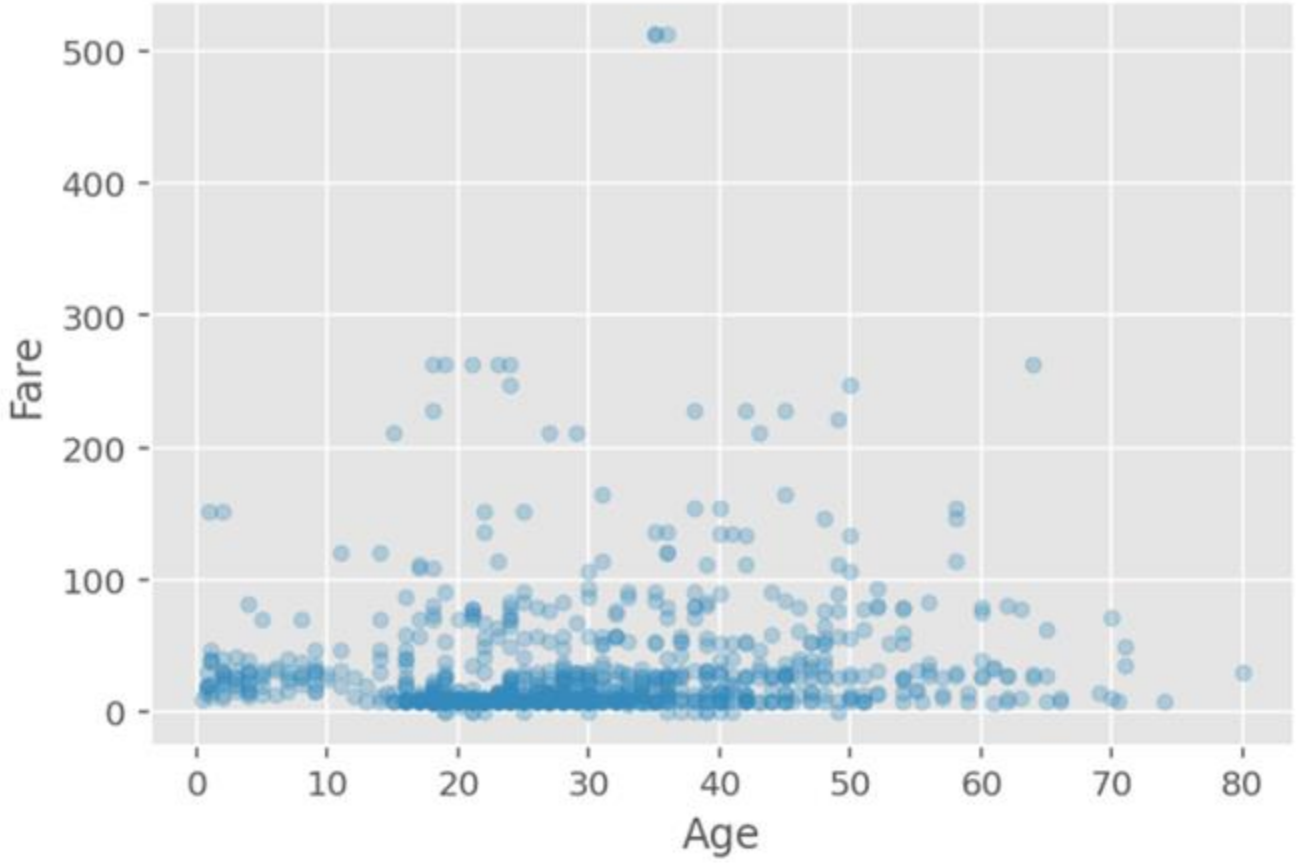
"Looks like older people didn't spend more than younger people."



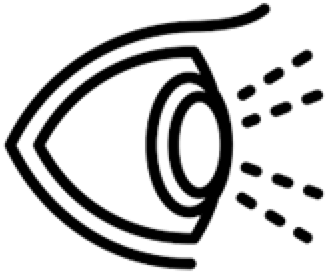
Visualizations are for Humans



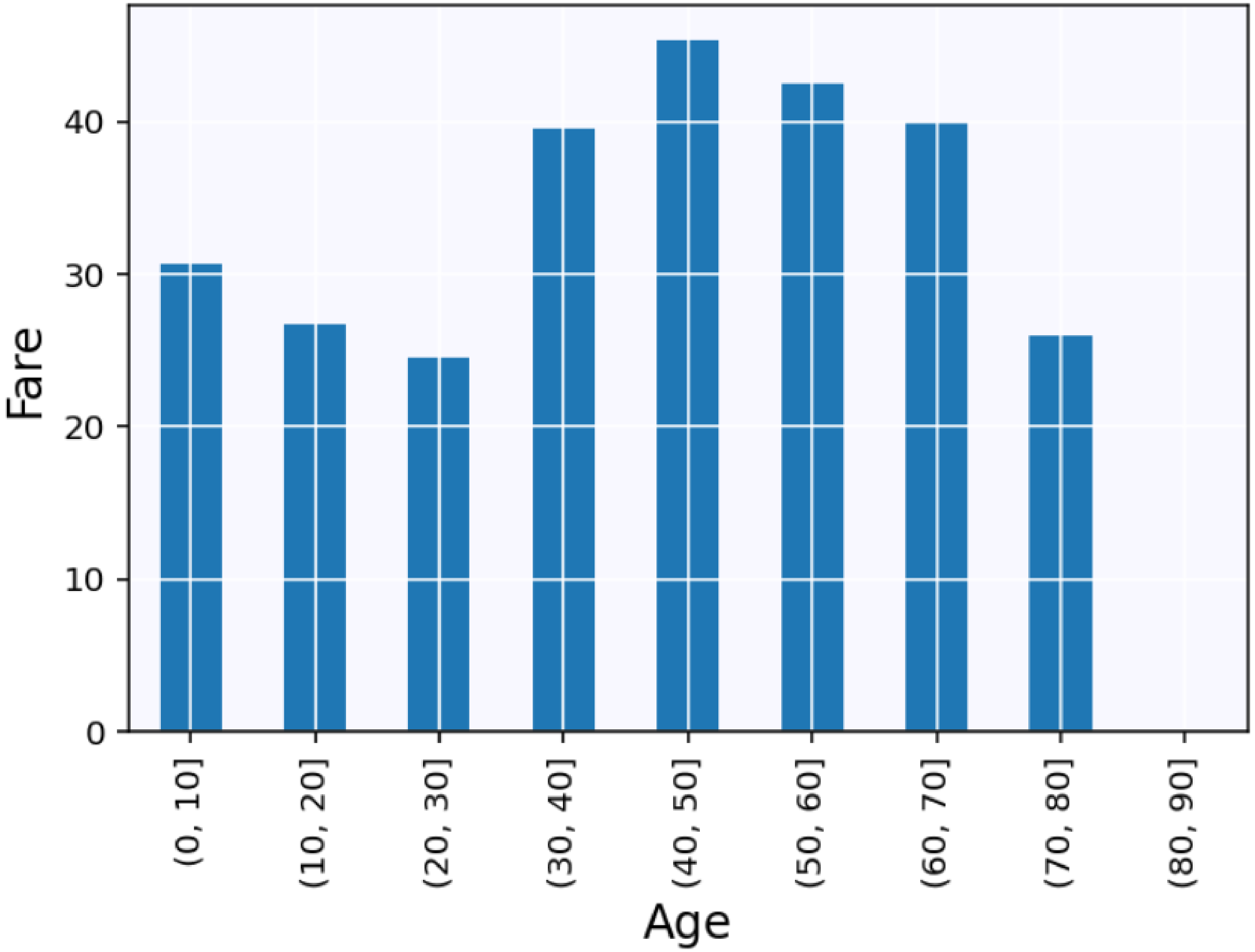
"Looks like older people didn't spend more than younger people."



Visualizations are for Humans



“Looks like older people didn’t spend more than younger people.”



Visualizations are for Humans

x	y
10.0	8.04
8.0	6.95
13.0	7.58
9.0	8.81
11.0	8.33
14.0	9.96
6.0	7.24
4.0	4.26
12.0	10.84
7.0	4.82
5.0	5.68

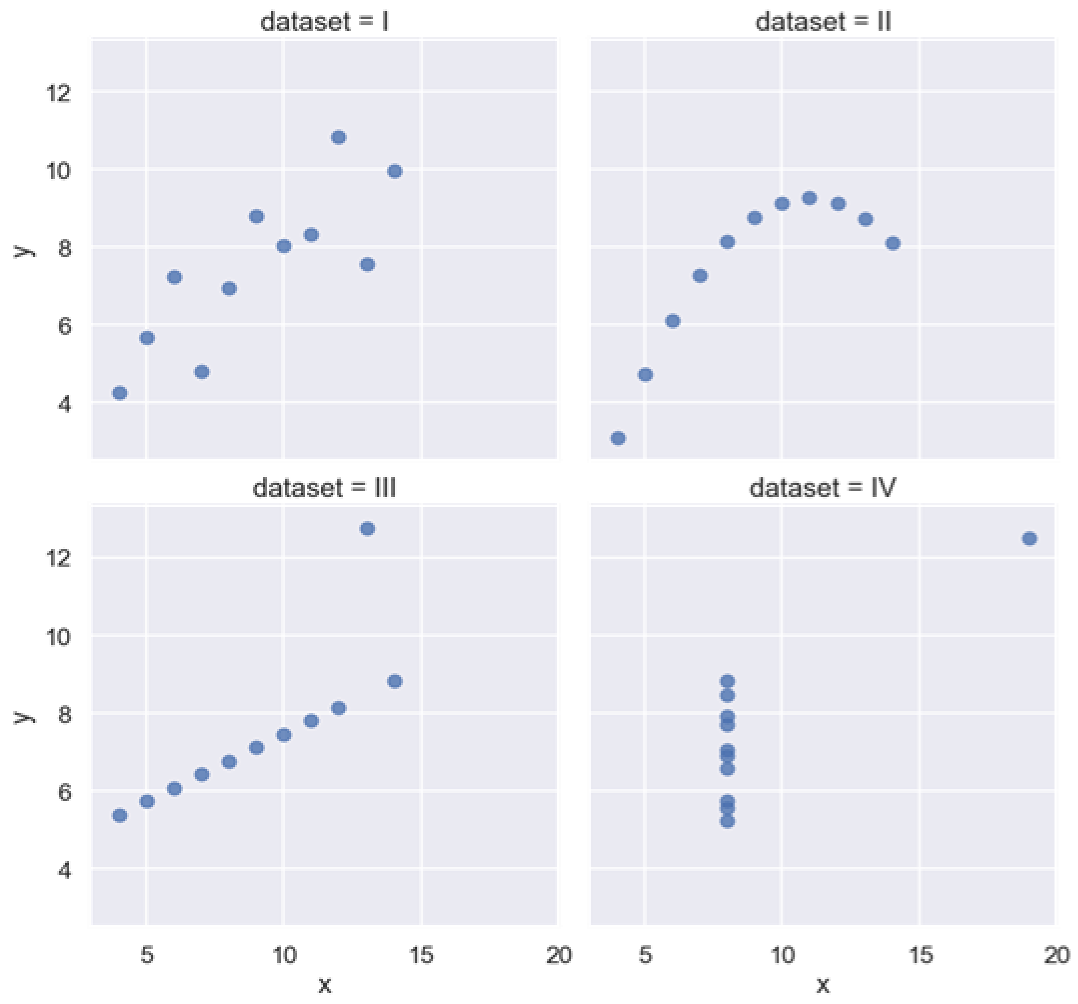
x	y
10.0	9.14
8.0	8.14
13.0	8.74
9.0	8.77
11.0	9.26
14.0	8.10
6.0	6.13
4.0	3.10
12.0	9.13
7.0	7.26
5.0	4.74

x	y
10.0	7.46
8.0	6.77
13.0	12.74
9.0	7.11
11.0	7.81
14.0	8.84
6.0	6.08
4.0	5.39
12.0	8.15
7.0	6.42
5.0	5.73

x	y
8.0	6.58
8.0	5.76
8.0	7.71
8.0	8.84
8.0	8.47
8.0	7.04
8.0	5.25
19.0	12.50
8.0	5.56
8.0	7.91
8.0	6.89

https://en.wikipedia.org/wiki/Anscombe%27s_quartet

Visualizations are for Humans



Human eyes good at seeing visual patterns!

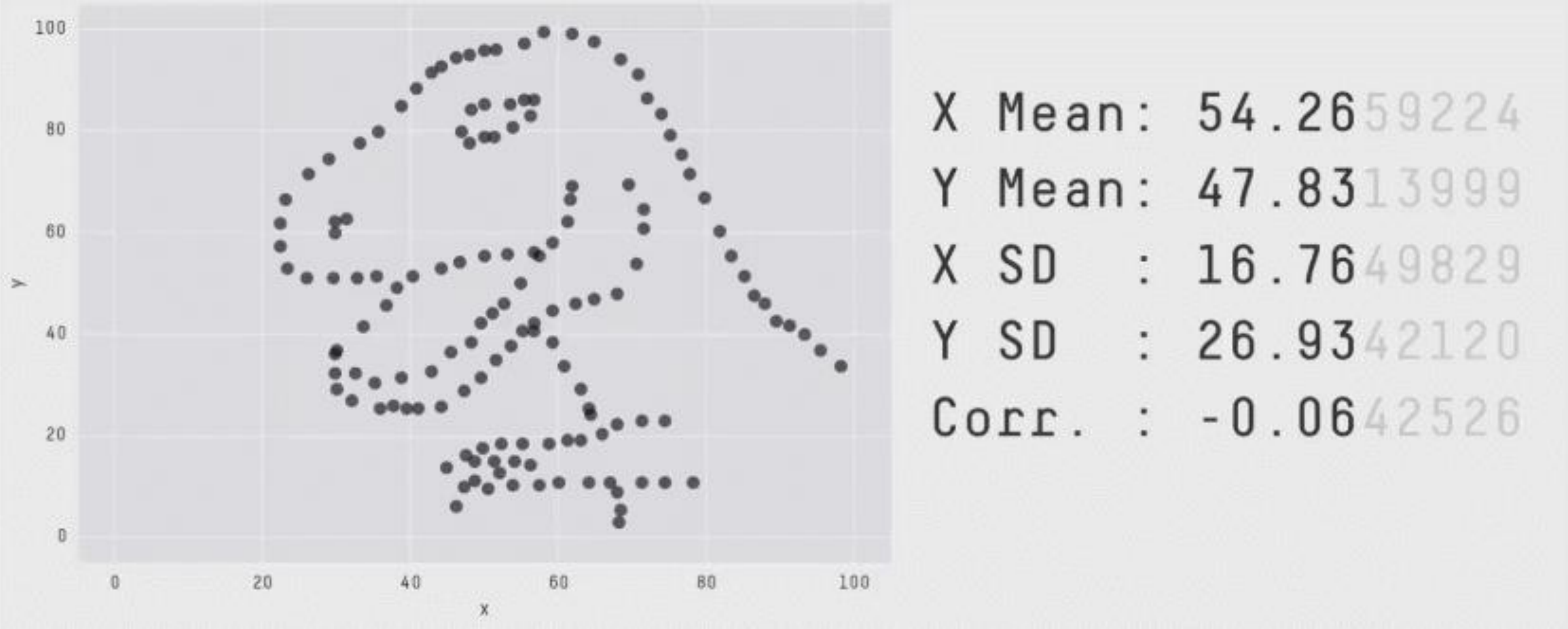
Anscombe's Quartet

Identical statistics	
x mean	9
x variance	10
y mean	7.5
y variance	3.75
x/y correlation	0.816

https://en.wikipedia.org/wiki/Anscombe%27s_quartet

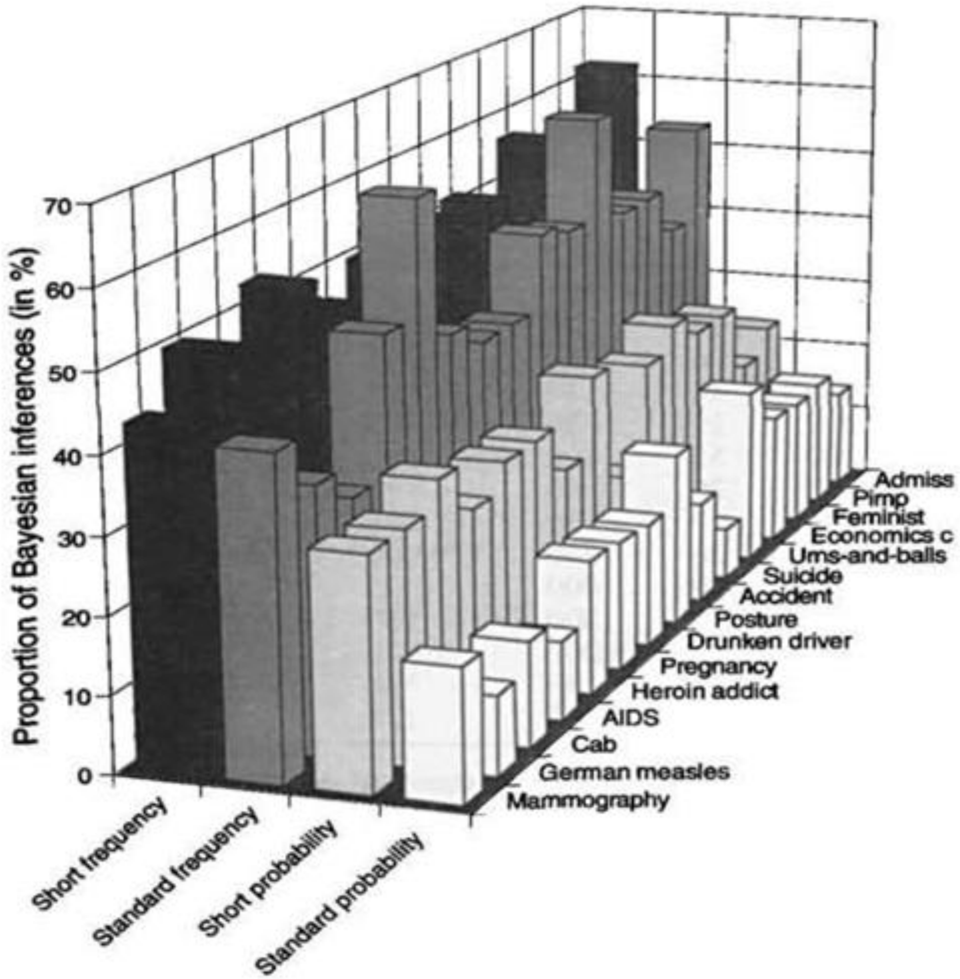
Visualizations are for Humans

Human eyes good at seeing visual patterns!



https://blog.revolutionanalytics.com/2017/05/the-datasaurus-dozen.html

Visualizations are for Humans



Human eyes good at seeing visual patterns!...

Sometimes.

Why Data Visualization?

- One goal of data science is to inform human decisions
  - Excellent plots directly address this goal
  - Sometimes the most useful results from data analysis are the visualizations!

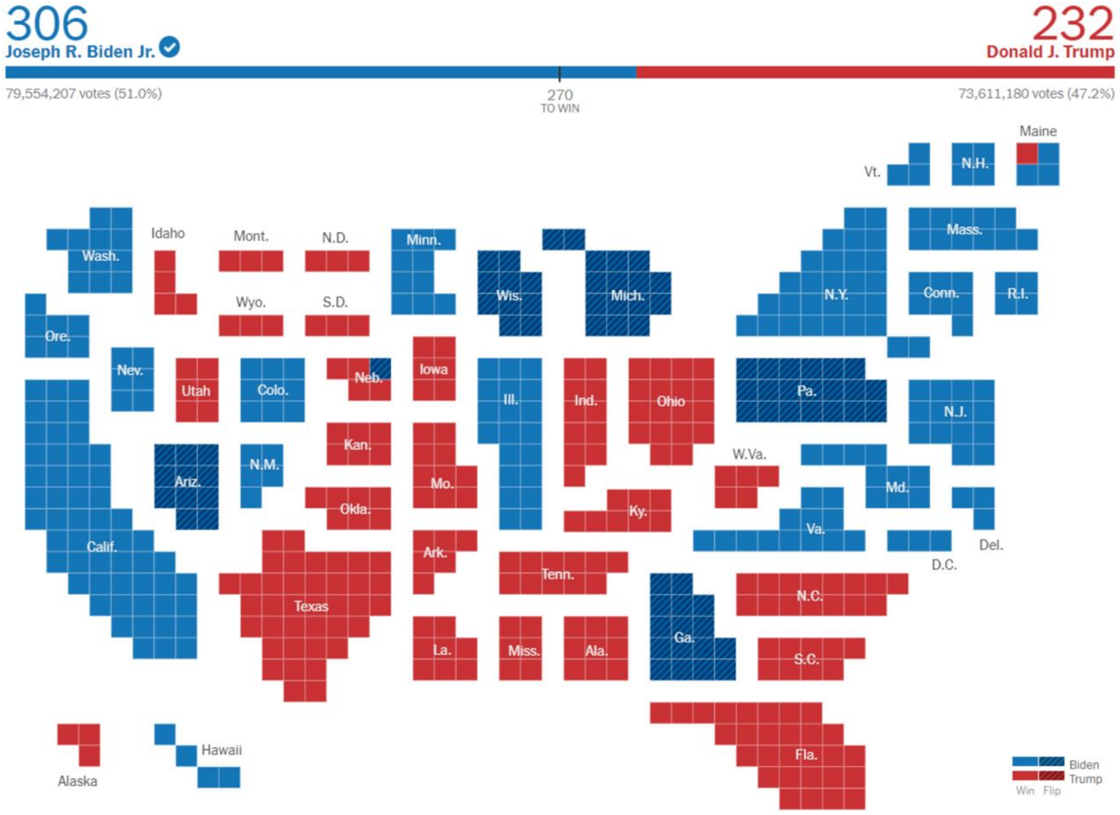
- Data viz is not as simple as calling plot()
  - Many plots possible, but only a few are useful
  - Every visualization has tradeoffs

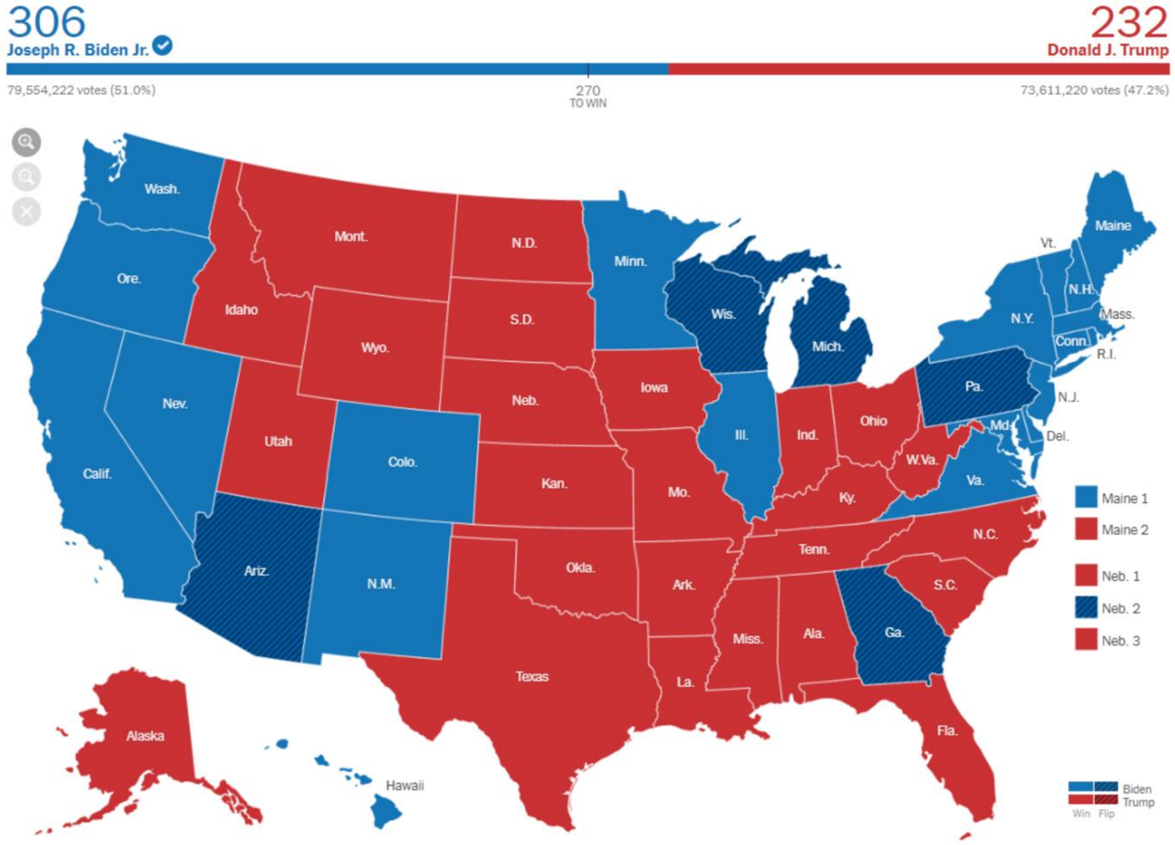
Tradeoffs

https://www.nytimes.com/2020/11/19/learning/whats-going-on-in-this-graph-2020-presidential-election-maps.html

What's Going On in This Graph? | 2020 Presidential Election Maps

These two maps show the results of the 2020 presidential election. Which map does a better job of visualizing the outcome of the election?





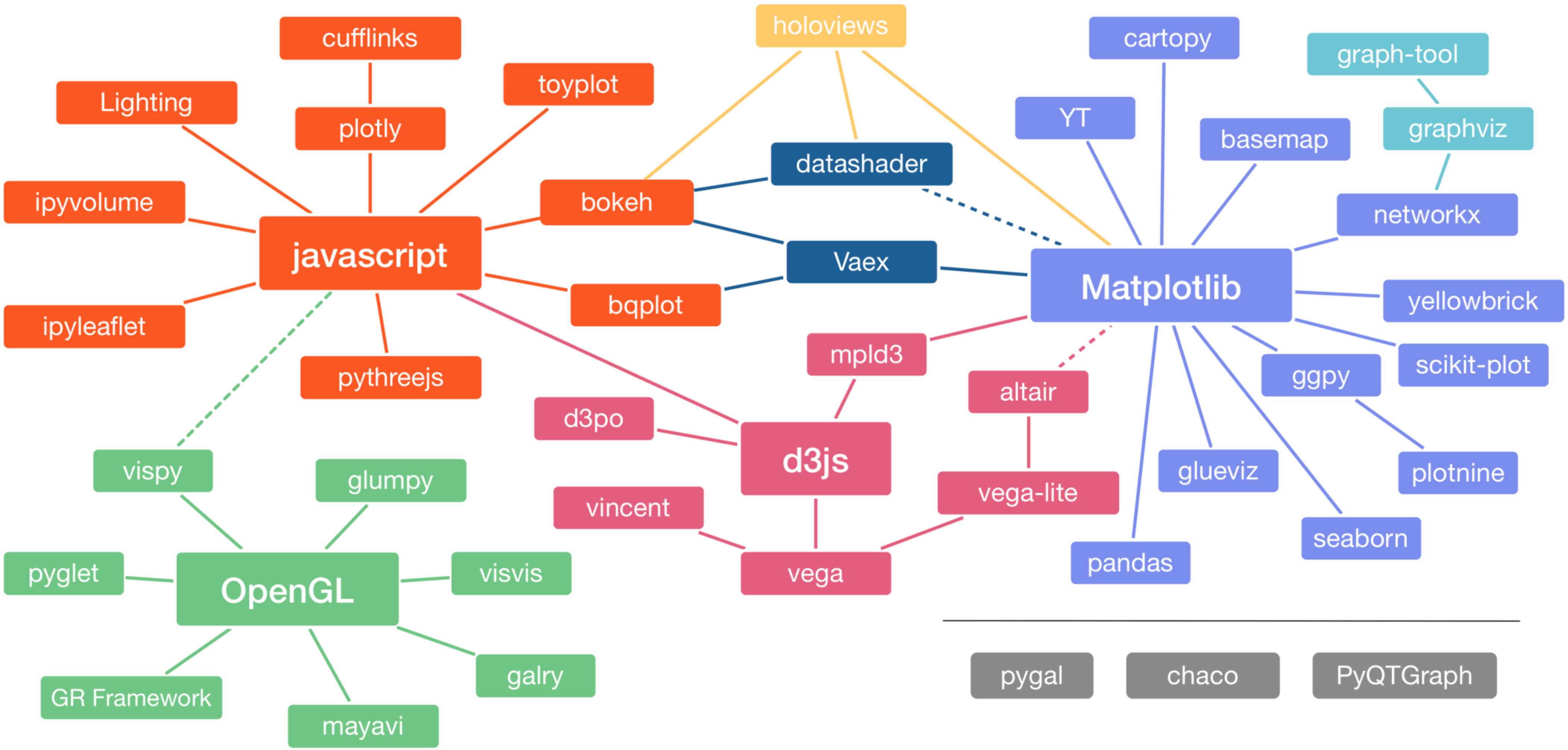
Outline

- Overview of Visualization: What, Why, and How?
- **What & How: Examples**
- Why: Goals
- How: Guidelines and Techniques

Visualization Tools

- Python Libraries
  - Seaborn
  - Matplotlib (especially Pyplot)
  - Plotly
  - …
- Other tools
  - Microsoft Excel
  - Tableau
  - Ggplot
  - …

Python Visualization Landscape



Adaptation of Jake VanderPlas' graphic about the Python visualization landscape, by Nicolas P. Rougier

Python examples

seaborn

- Declarative API
- Best used with tidy (aka long-form) data
  - Seaborn will perform groupby automatically

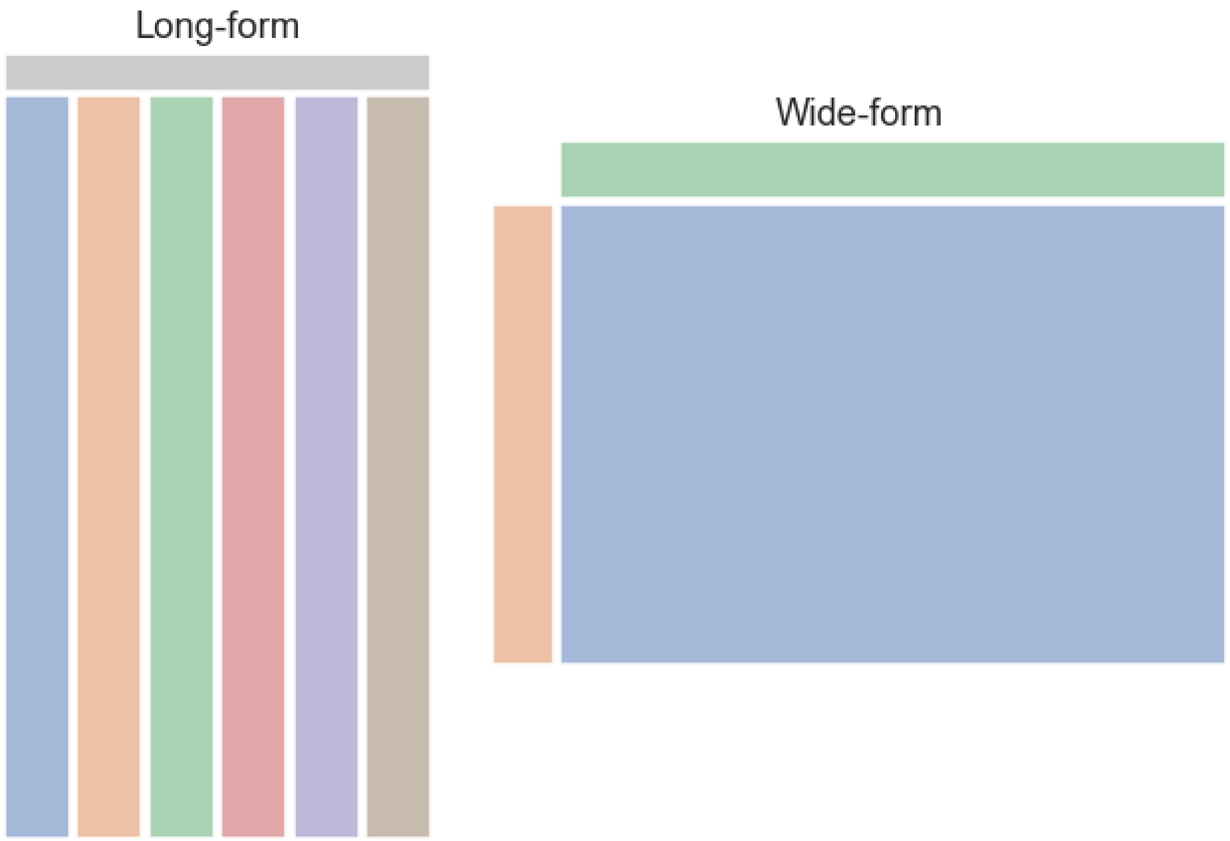
Typical usage:

```
sns.someplot(x='...', y='...', data=...)
```

Demo/Tutorial: https://seaborn.pydata.org/tutorial.html

Long form and wide form data

- Long-form:
  - Each column is a *variable* (feature)
  - Each row is an *observation* (data point)
- Wide form:
  - Each column is a variable
  - *Each row is a variable*
  - Each value is an observation



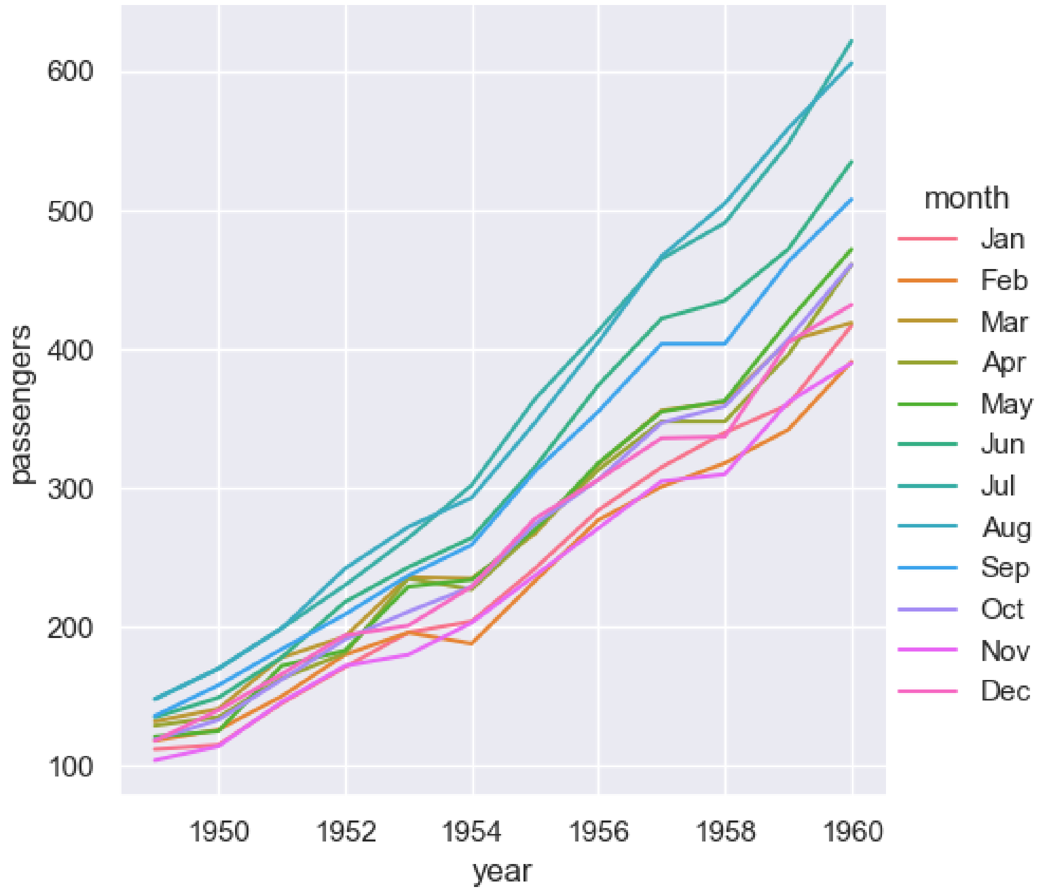
Long form example

```python
flights = sns.load_dataset("flights")
flights.head()
```

✓ 0.9s

	year	month	passengers
0	1949	Jan	112
1	1949	Feb	118
2	1949	Mar	132
3	1949	Apr	129
4	1949	May	121

```python
sns.relplot(data=flights, x="year", y="passengers",
            hue="month", kind="line")
```



Wide form example

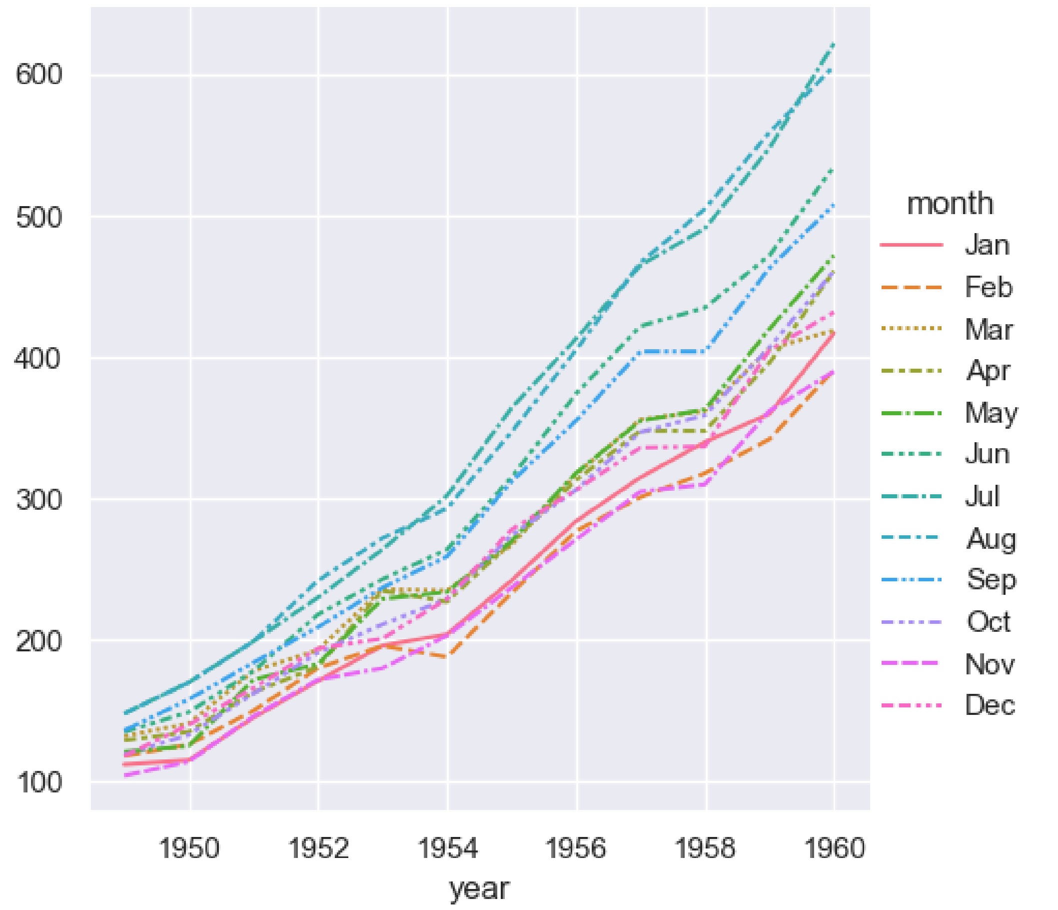
```python
flights_wide = flights.pivot(index="year", columns="month", values="passengers")
flights_wide.head()
```

✓ 0.0s

month year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1949	112	118	132	129	121	135	148	148	136	119	104	118
1950	115	126	141	135	125	149	170	170	158	133	114	140
1951	145	150	178	163	172	178	199	199	184	162	146	166
1952	171	180	193	181	183	218	230	242	209	191	172	194
1953	196	196	236	235	229	243	264	272	237	211	180	201

```python
sns.relplot(data=flights_wide, kind="line")
```

✓ 0.2s



Note

Seaborn treats the argument to `data` as wide form when neither `x` nor `y` are assigned.

Messy data

- One major reason we need data prep.!

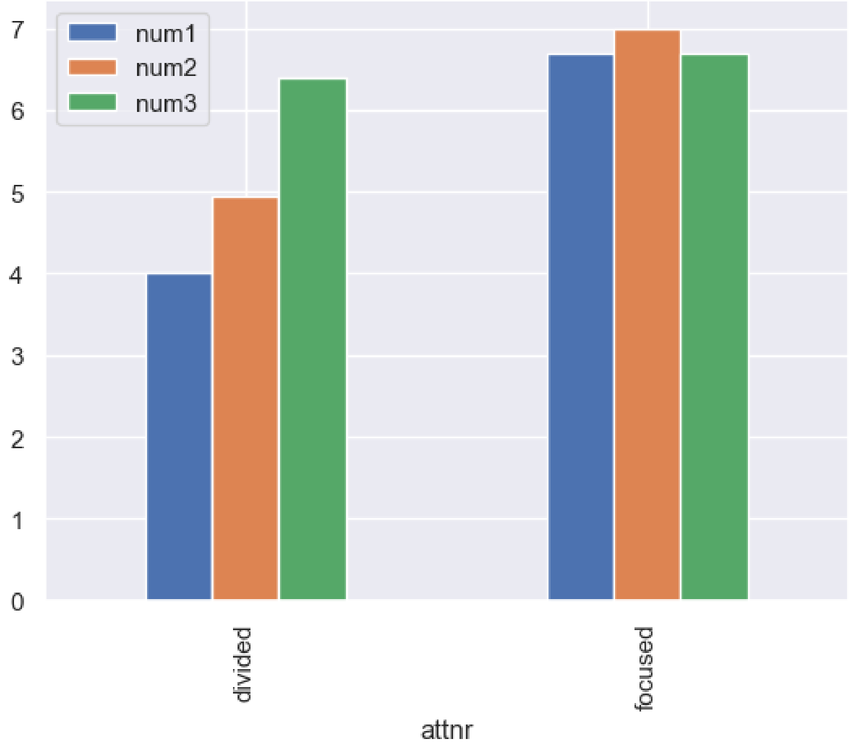
```python
anagrams = sns.load_dataset("anagrams")
anagrams
```

✓ 0.6s

	subidr	attnr	num1	num2	num3
0	1	divided	2	4.0	7
1	2	divided	3	4.0	5
2	3	divided	3	5.0	6
3	4	divided	5	7.0	5
4	5	divided	4	5.0	8

```python
anagrams_aggr = anagrams.groupby('attnr')[['num1', 'num2', 'num3']].mean()
anagrams_aggr.plot(kind = 'bar')
```

✓ 0.0s



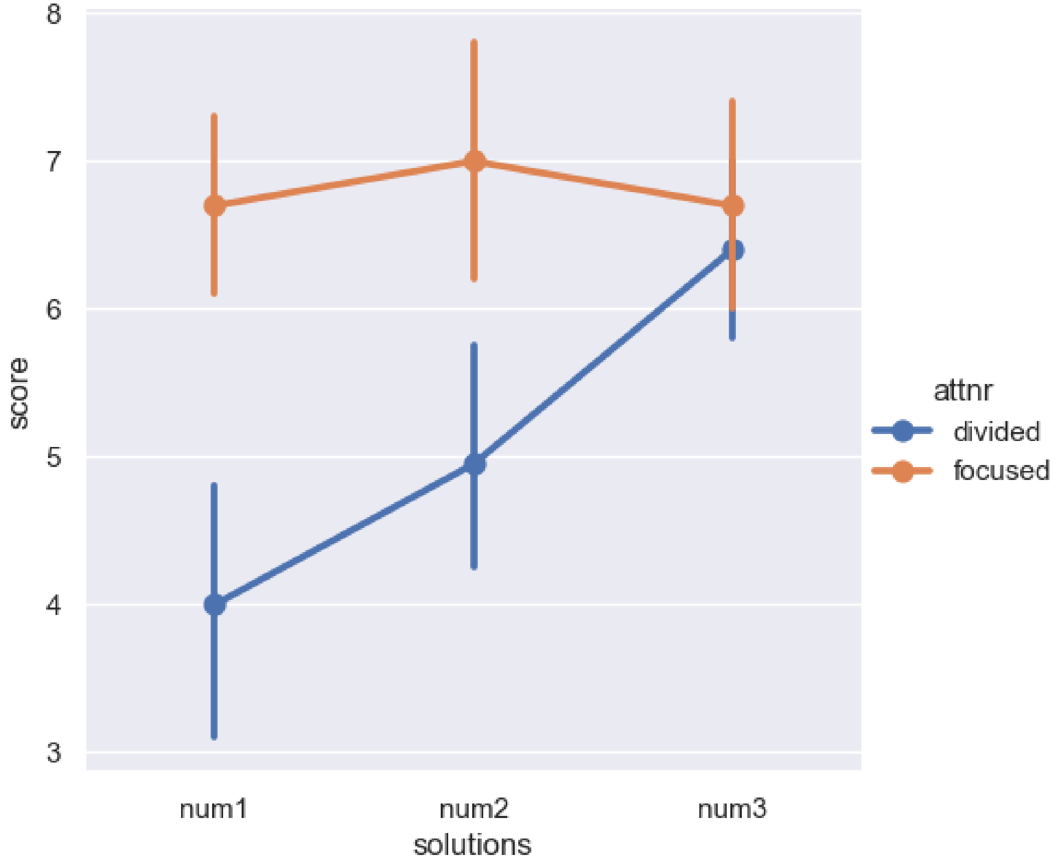
Messy data

```
anagrams_long = anagrams.melt(
    id_vars=["subidr", "attnr"],
    var_name="solutions", value_name="score")
anagrams_long
```

✓ 0.0s

	subidr	attnr	solutions	score
0	1	divided	num1	2.0
1	2	divided	num1	3.0
2	3	divided	num1	3.0
3	4	divided	num1	5.0
4	5	divided	num1	4.0

```
sns.catplot(data=anagrams_long, x="solutions", y="score",
            hue="attnr", kind="point")
```



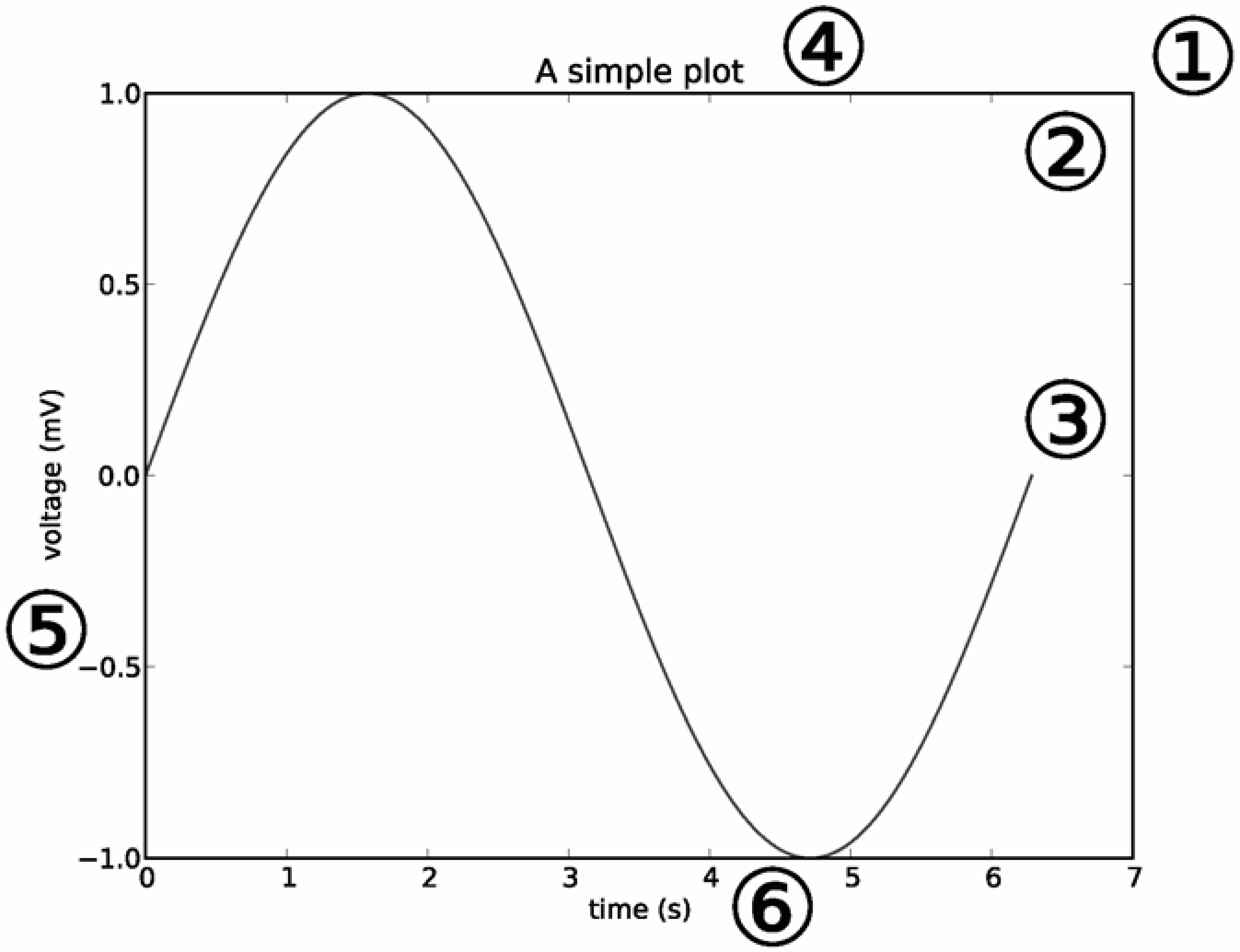
matplotlib

- Underlying library for seaborn, pandas, and most other Python plotting libraries
- A Figure contains several Axes. Each Axes contains a plot.
- When creating a plot, a new figure + axes is created if not already initialized.
  - Matplotlib remembers that axes for the duration of the cell (hidden state!)
- Note: Axes = one chart within a larger Figure
  - Axis = x or y-axis within a chart (sorry!)

Demo/Tutorial: https://matplotlib.org/tutorials/introductory/pyplot.html

matplotlib

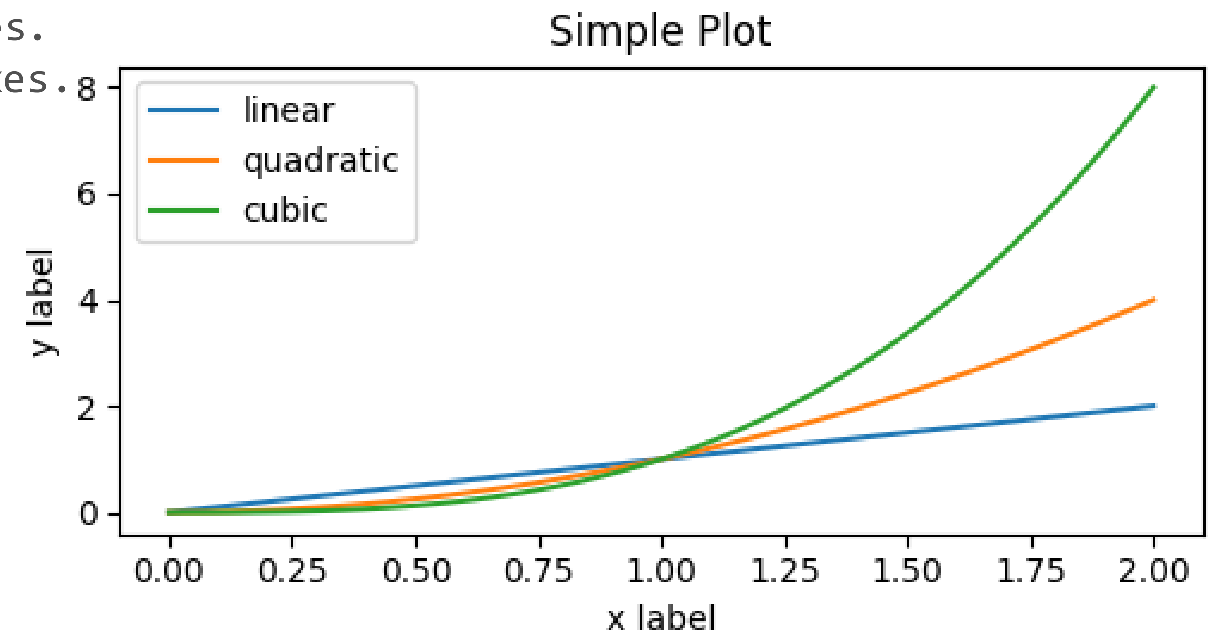
1. Figure
2. Axes
3. Line
4. Title
5. YAxis
6. XAxis



matplotlib

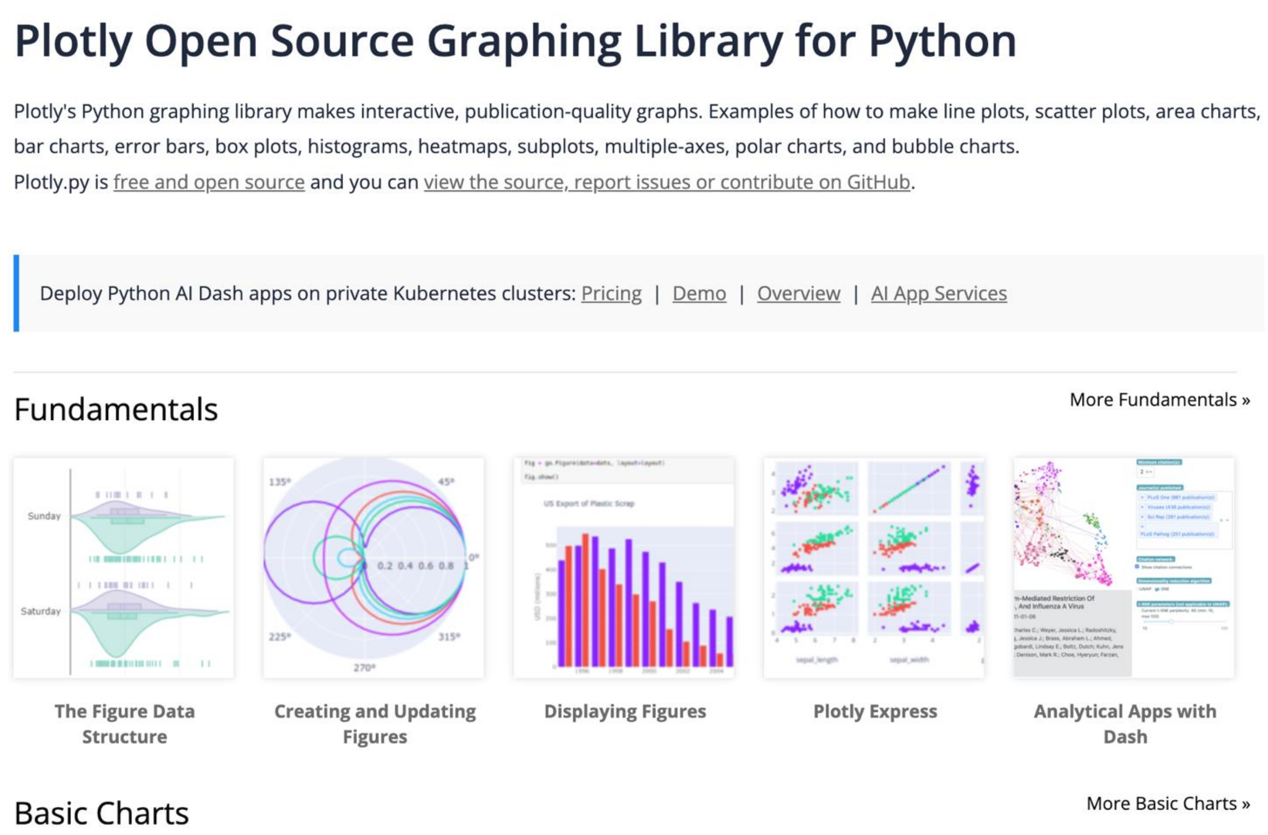
Typical usage:

```
x = np.linspace(0, 2, 100) # Sample data.
fig, ax = plt.subplots(figsize=(5, 2.7), layout='constrained’)
ax.plot(x, x, label='linear') # Plot some data on the Axes.
ax.plot(x, x**2, label='quadratic') # Plot more data on the Axes...
ax.plot(x, x**3, label='cubic') # ... and some more.
ax.set_xlabel('x label') # Add an x-label to the Axes.
ax.set_ylabel('y label') # Add a y-label to the Axes.
ax.set_title("Simple Plot") # Add a title to the Axes.
ax.legend() # Add a legend.
```



Plotly

- For interactive web visualizations



Demo/Tutorial: https://plotly.com/python/

Typical Workflow

- Start with seaborn plot
  - Get as close to desired result as possible
- Fine-tune with matplotlib, e.g:
  - Changing title, axis labels
  - Annotating interesting points
- Publication-ready plots take lots of fine-tuning!

Seaborn API

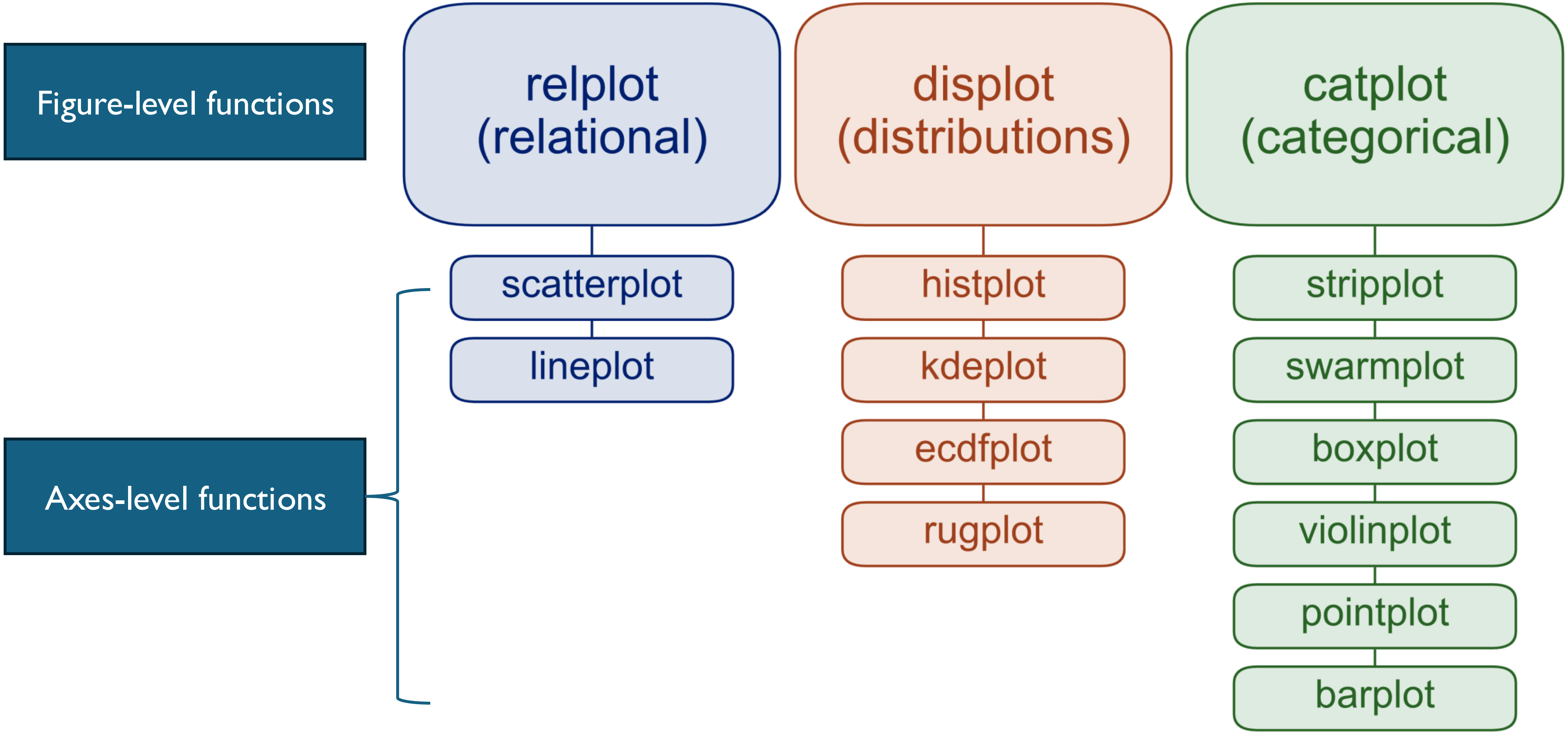


Figure-level functions
Axes-level functions
relplot
(relational)
scatterplot
lineplot
displot
(distributions)
histplot
kdeplot
ecdfplot
rugplot
catplot
(categorical)
stripplot
swarmplot
boxplot
violinplot
pointplot
barplot

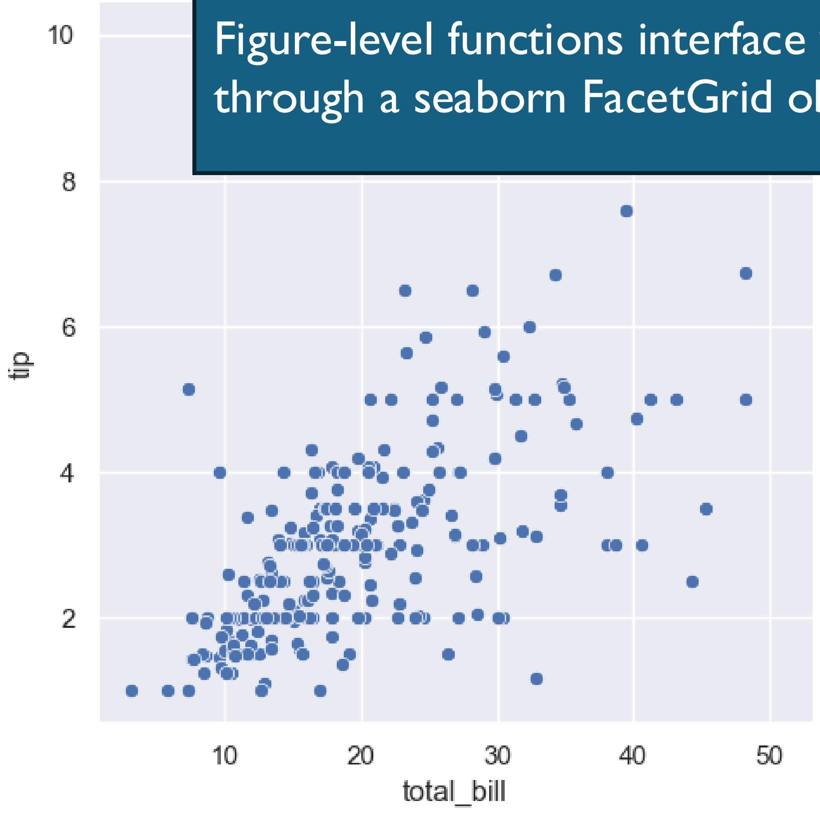
Figure-level vs. axes-level

```python
sns.relplot(data=tips, x="total_bill",
            y="tip", kind="scatter")
```

✓ 0.1s

```
<seaborn.axisgrid.FacetGrid at 0x1242596d0>
```

Figure-level functions interface with matplotlib through a seaborn FacetGrid object




```python
tips = sns.load_dataset("tips")
tips.head()
```

✓ 0.0s

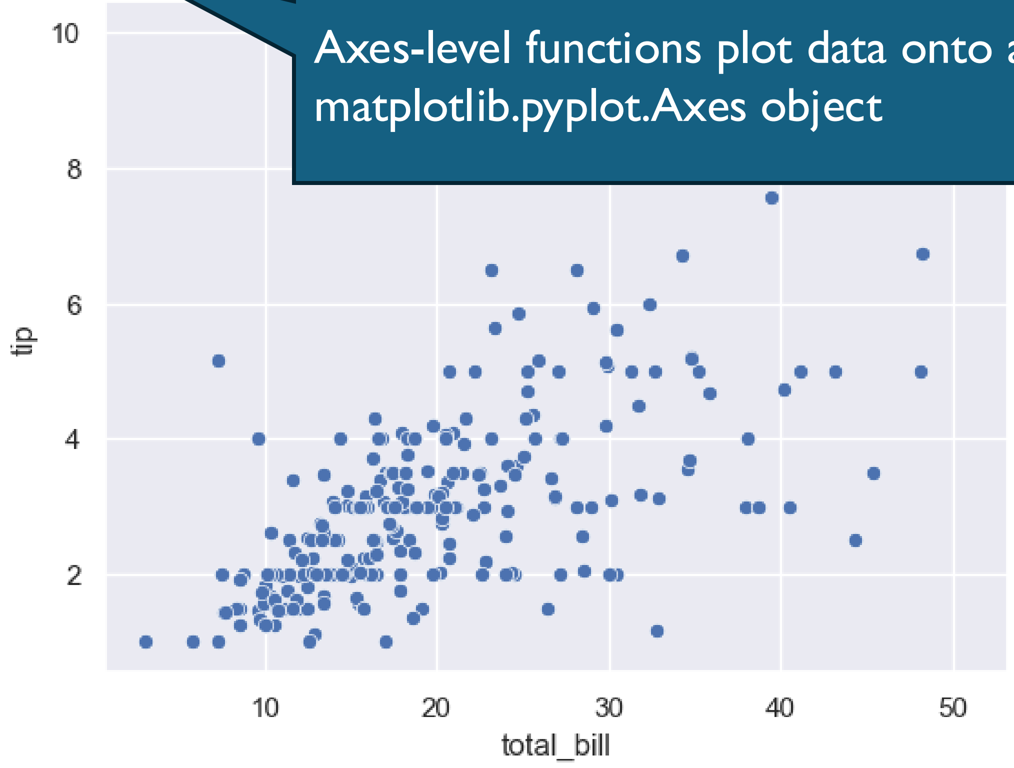
	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2

```python
sns.scatterplot(data=tips, x="total_bill", y="tip")
```

✓ 0.0s

```
<Axes: xlabel='total_bill', ylabel='tip'>
```

Axes-level functions plot data onto a matplotlib.pyplot.Axes object



Summary of Figure-level functions

Advantages	Drawbacks
Easy faceting by data variables	Many parameters not in function signature
Legend outside of plot by default	Cannot be part of a larger matplotlib figure
Easy figure-level customization	Different API from matplotlib

Plot types

- Dot plot, Rug plot
- Jitter plot
- Error bar plot
- Box plot
- Histogram
- Kernel density estimate
- Cumulative distribution function

Common Visualizations for One Quantitative Variable

Histograms

Data divided into a set of discrete bins

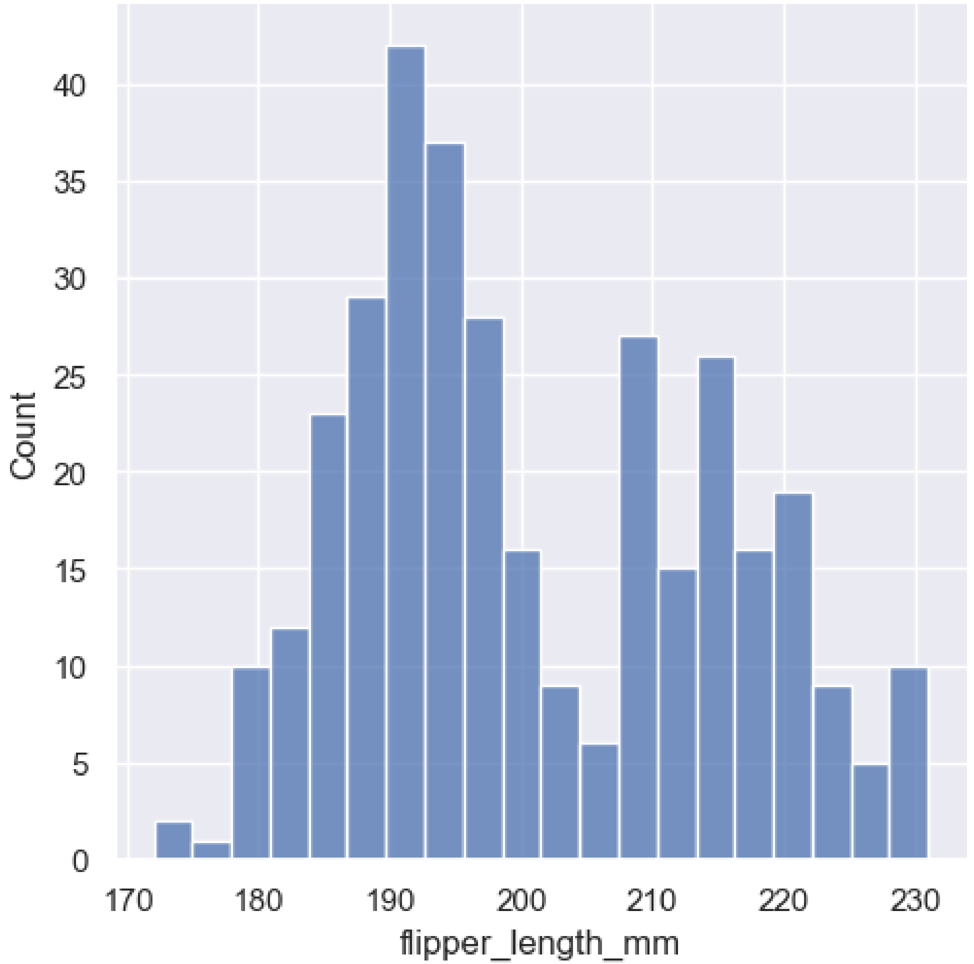
- Height: the count of observations falling within each bin (left inclusive, right exclusive)
- Deciding on number of bins is hard! Trial-and-error process
- Can also set “binwidth”

```python
penguins = sns.load_dataset("penguins")
penguins.head()
```

✓ 0.0s

	species	island	bill_length_mm	bill_depth_mm	flip
0	Adelie	Torgersen	39.1	18.7	

```python
sns.displot(penguins,
  x="flipper_length_mm", bins=20)
```

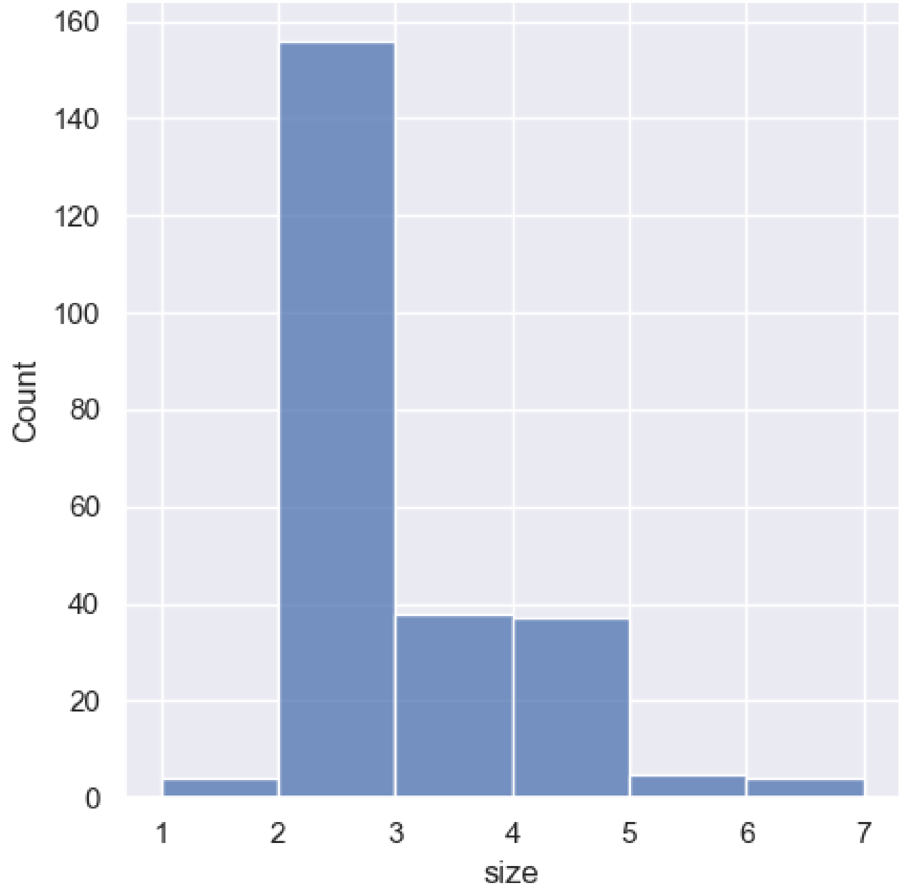


Histograms

Data divided into a set of discrete bins

- Deciding on number of bins is hard! Trial-and-error process
- Can also specify precise bin breaks

```
sns.displot(tips, x="size",
  bins=[1, 2, 3, 4, 5, 6, 7])

sns.displot(tips, x="size",
  bins=[1, 2, 3, 4, 5])
```

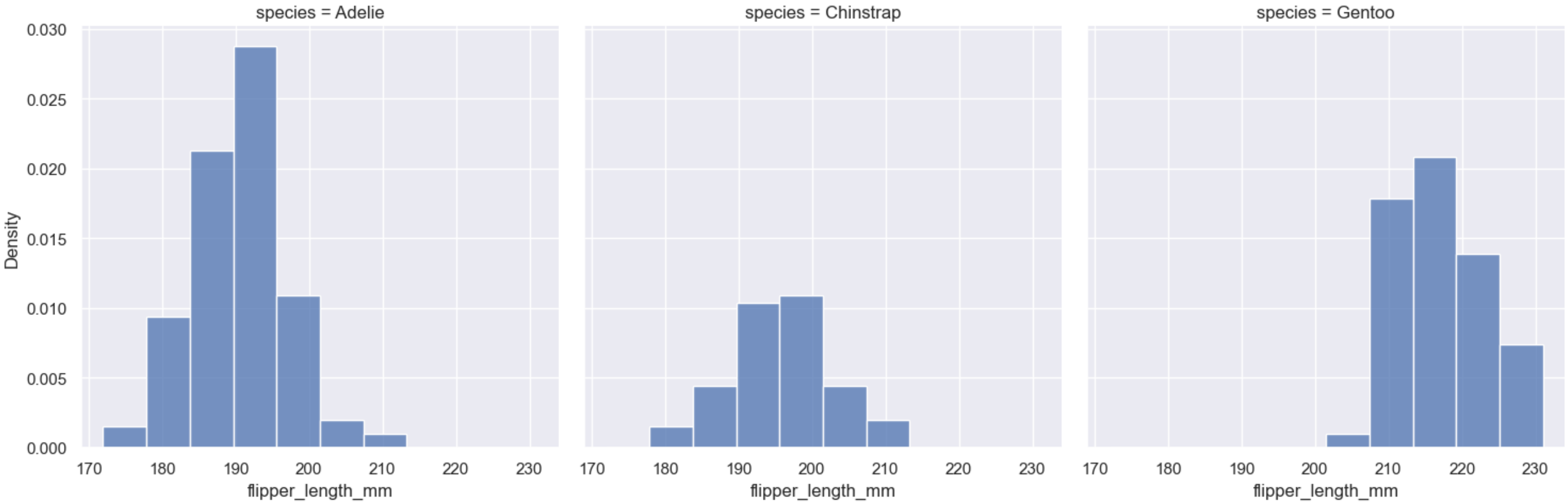


Normalized Histograms

Always have proportion per unit on y-axis

- Stat="density" divides everything by totals; total area = 1
- Stat="probability"; sum of heights = 1

```
sns.displot(penguins,
  x="flipper_length_mm", col="species",
stat="density"/stat="probability")
```



Density Plots

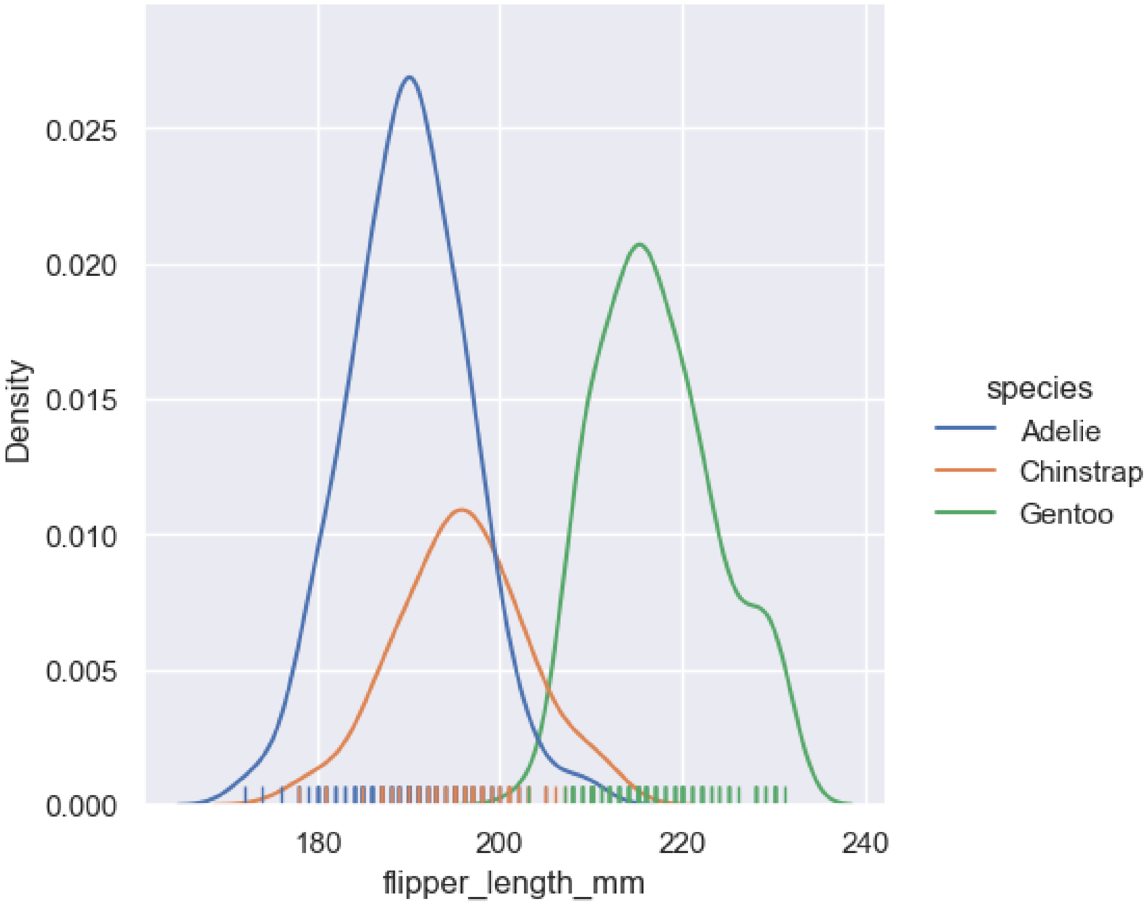
Density plots similar to a “smoothed” histogram

- Be careful of interpretation!

Rug plots put a tick at each data point

- Used to show all points

```
sns.displot(penguins,
  x="flipper_length_mm", kind="kde",
  hue=”species", rug=True)
```



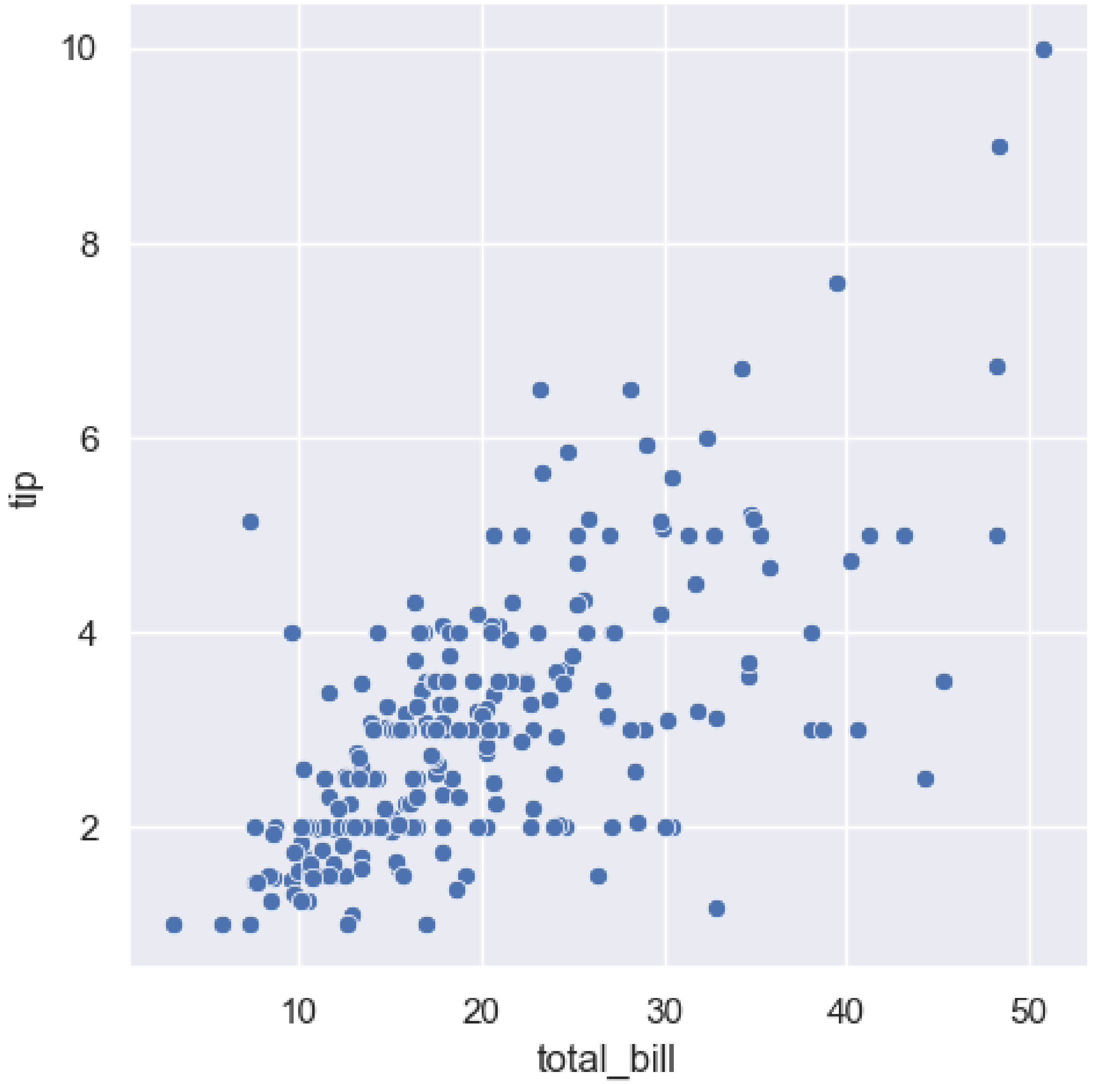
Common Visualizations for Two Quantitative Variables

Scatter Plots

Used to reveal relationships between pair of variables

- Susceptible to overplotting
  - Points overlap!

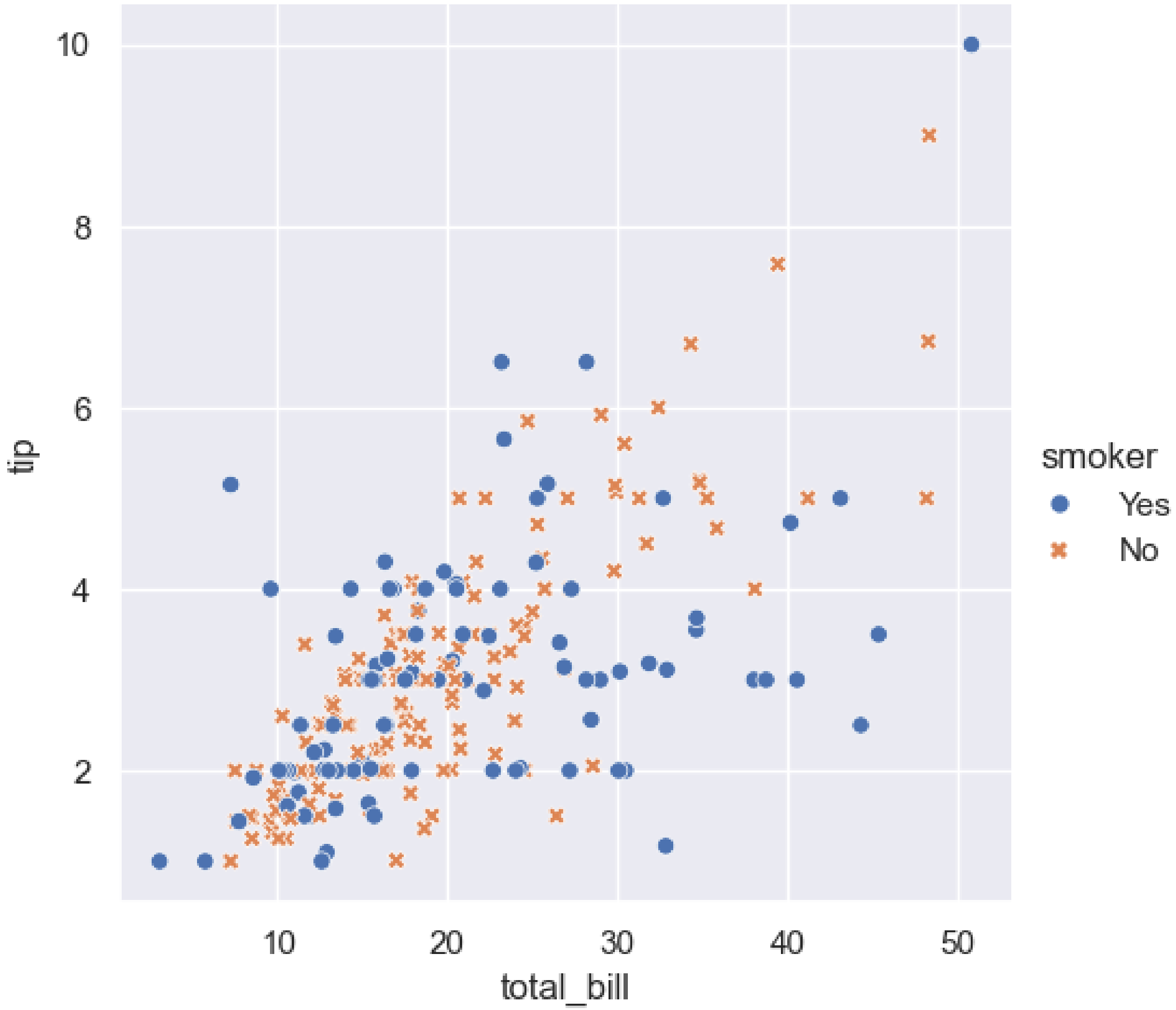
```
sns.relplot(
  data=tips, x="total_bill", y="tip"
)
```



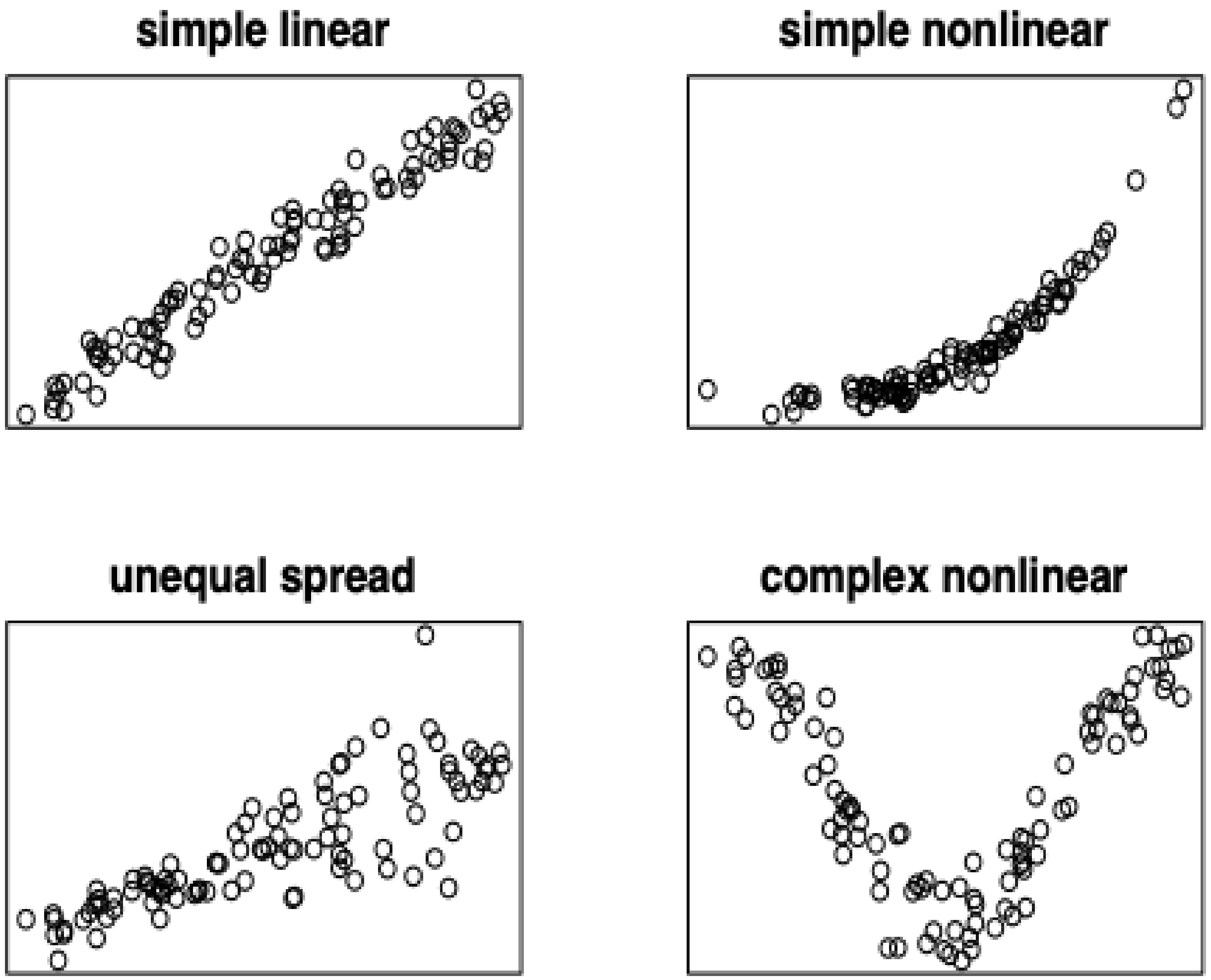
Scatter Plots

- Another dimension can be added to the plot by
  - coloring the points
  - or using a different marker styles

```
sns.relplot(
  data=tips, x="total_bill", y="tip",
  hue="smoker", style="smoker"
)
```



Scatter Plots



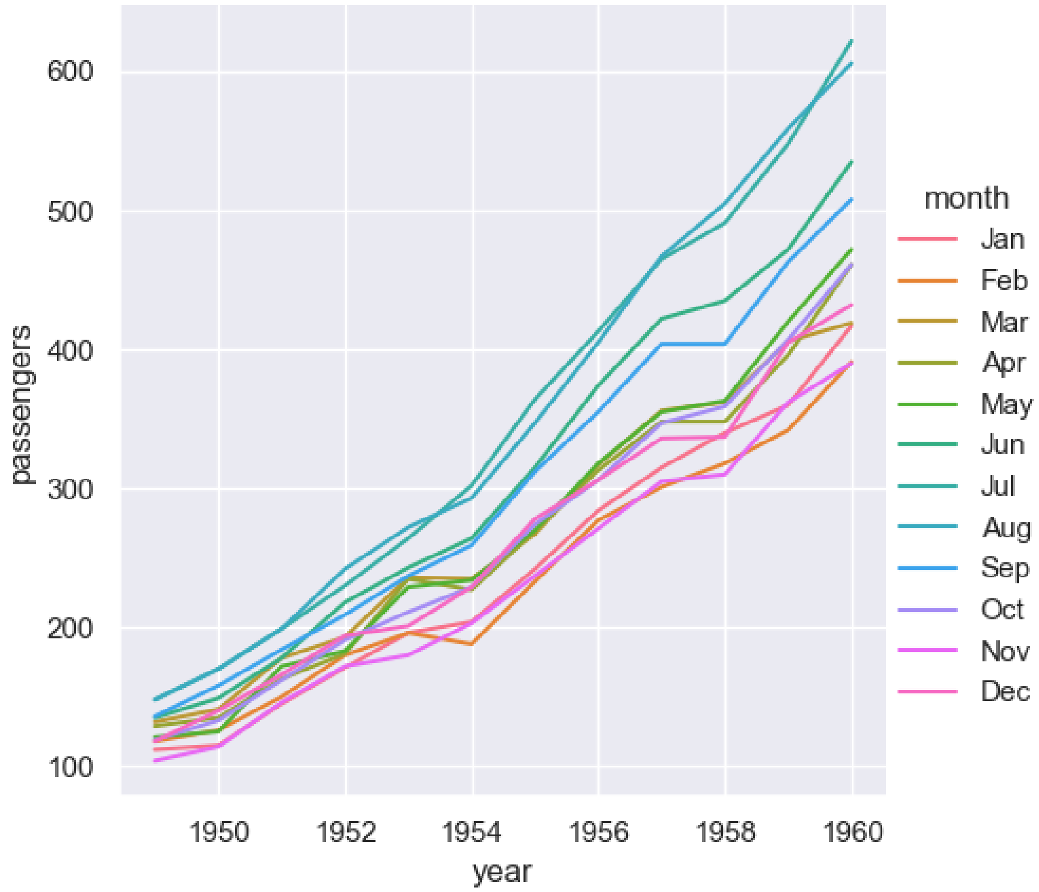
Used to inform model choices

- E.g. simple linear model requires linear trend and equal spread.

Line Plots

Understand changes in one variable as a function of an ordered variable (e.g., time)

```
sns.relplot(data=flights,
  x="year", y="passengers",
  hue="month", kind="line")
```

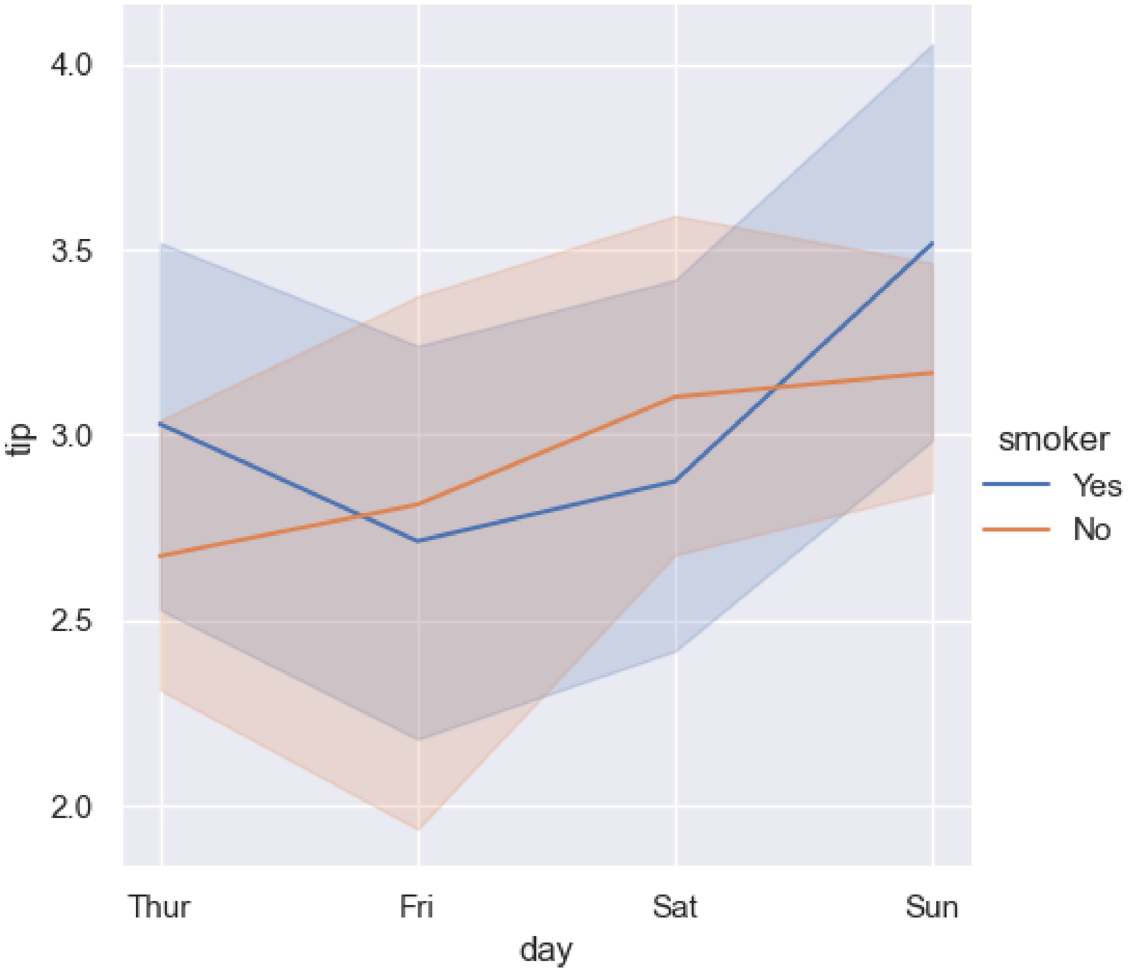


Line Plots

When there are multiple y for the same x, the default behavior in seaborn is to aggregate the y for each x value

- Line: mean
- Shaded region: the 95% confidence interval

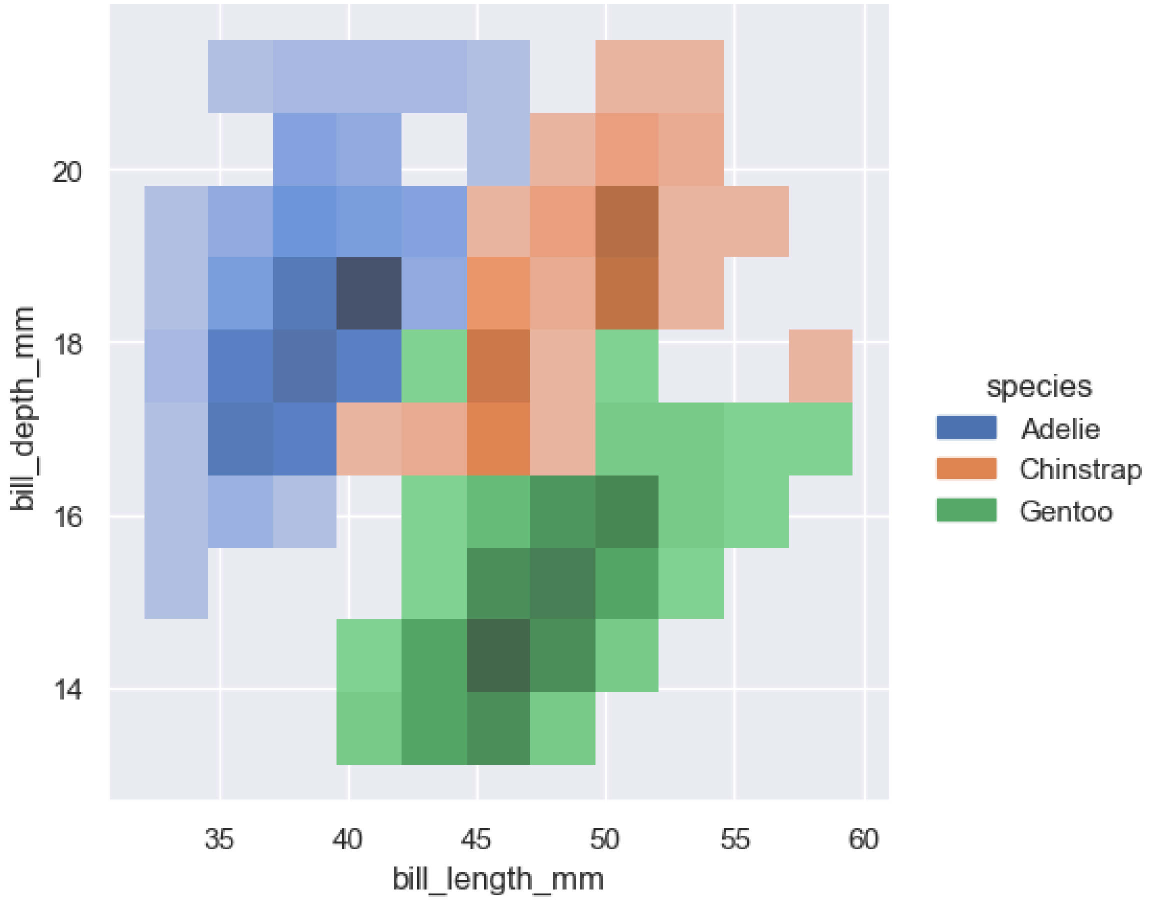
```
sns.relplot(
  data=tips, x="day", y="tip",
  hue="smoker", kind="line"
)
```



Heatmap/Hex Plots

- Equivalent of histogram in two dimensions
- Shaded rectangles/hexagons usually correspond to more points

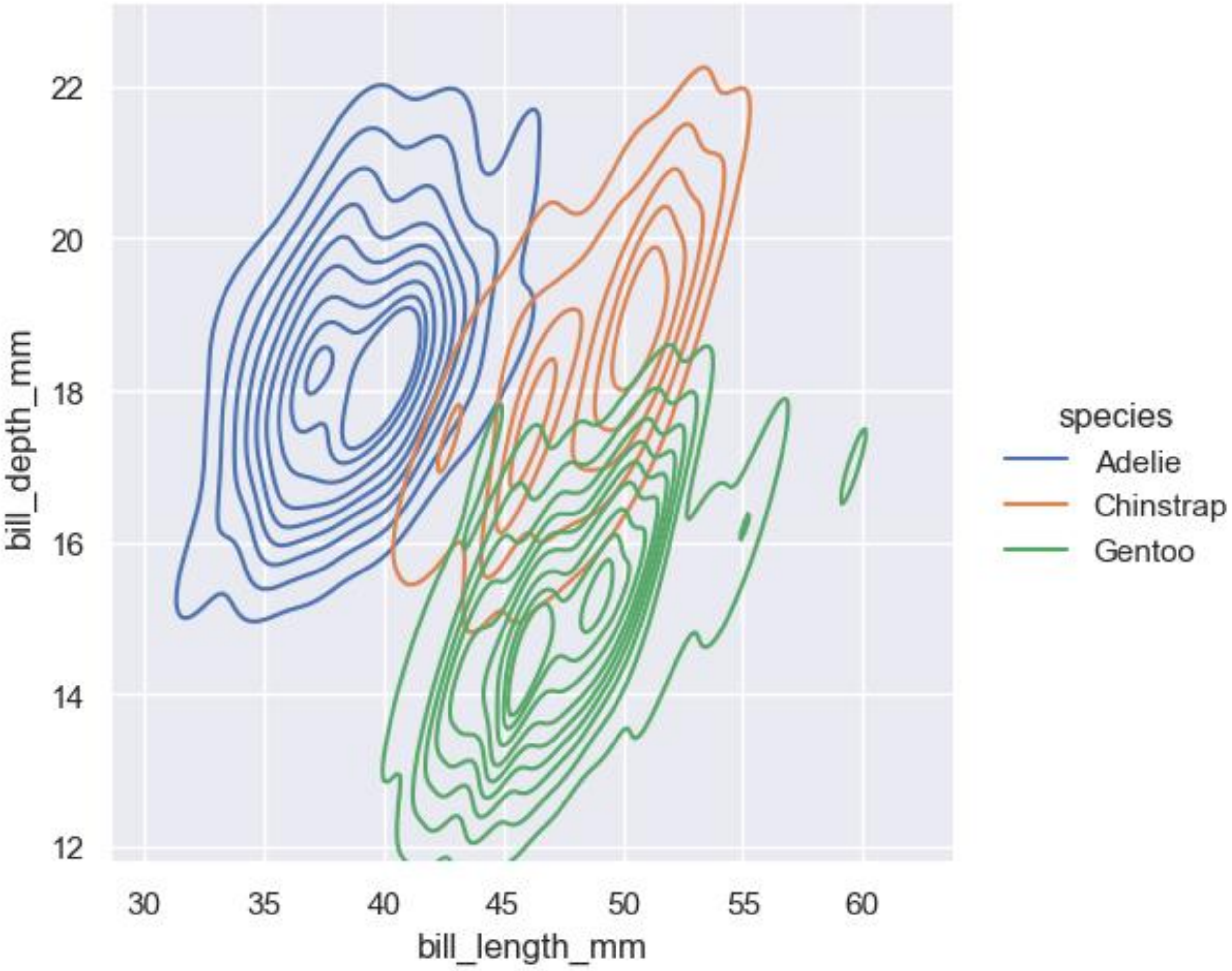
```
sns.displot(x='bill_length_mm',
  y='bill_depth_mm',data=penguins,
  kind='hist', hue='species')

sns.jointplot(x='bill_length_mm',
  y='bill_depth_mm', data=penguins,
  kind='hex')
```



2D Density Plots

- Density plots also work in two dimensions!

```
sns.displot(penguins,
  x="bill_length_mm",
  y="bill_depth_mm",
  hue="species", kind="kde"
)
```

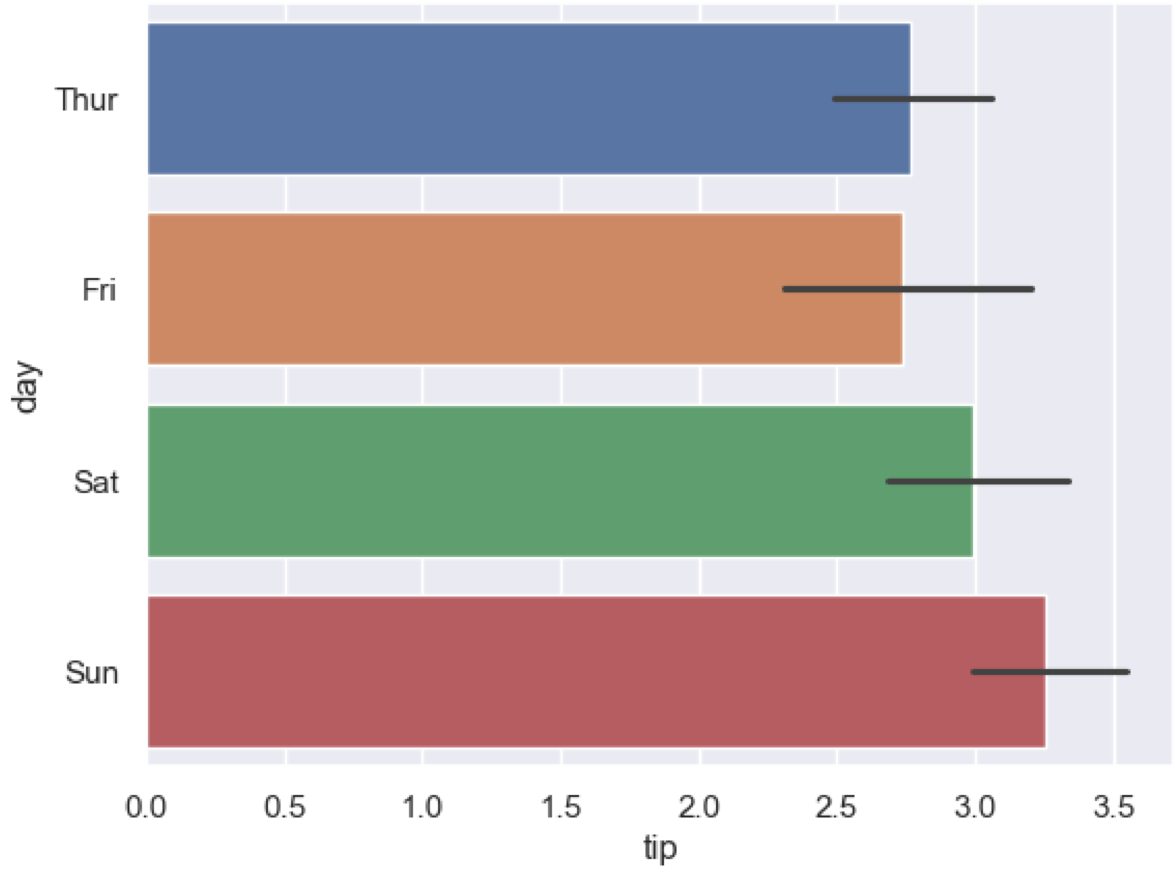


Common Visualizations for Qualitative + Quantitative Variable

Bar Plots

- Typically use horizontal bars to avoid label overlap
- Can also plot confidence intervals on bars if appropriate

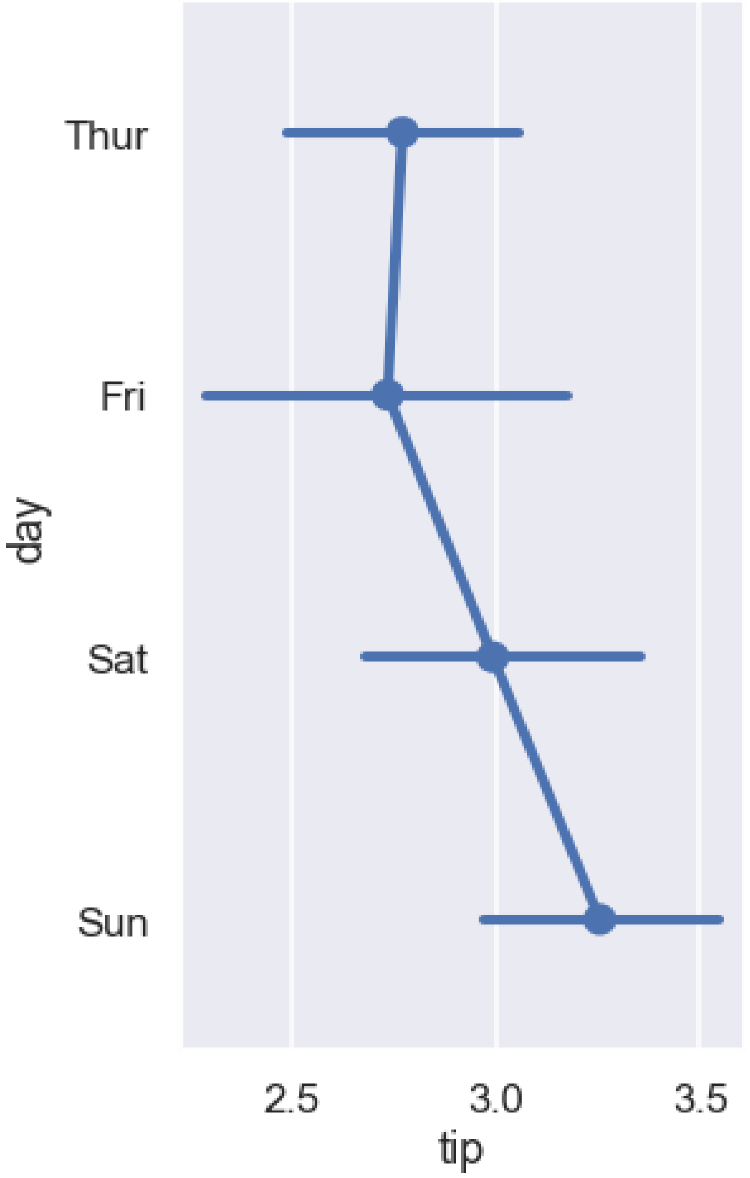
```
sns.catplot(tips, x="tip", y="day", hue="day",
kind="bar")
```



Point Plots / Dot Plots

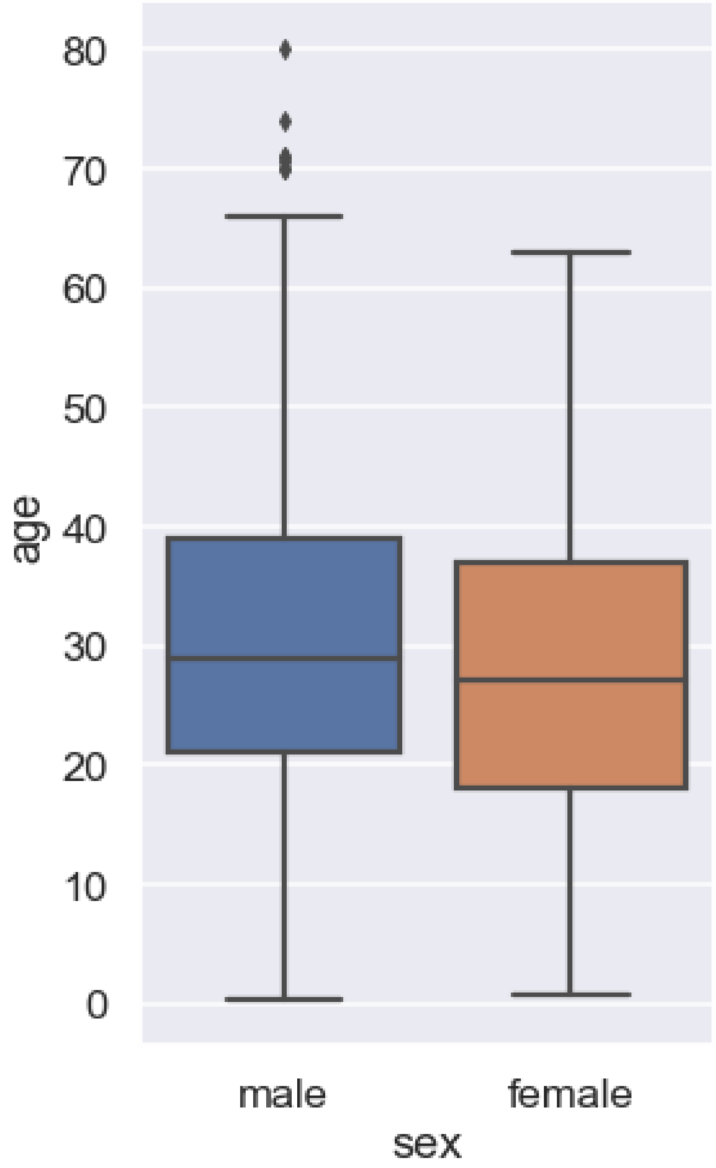
- Minimal cousin of the bar plot
- Some prefer point plots since the bar widths in a bar plot have no meaning

```
sns.catplot(data=tips, x="tip",
  y="day", kind="point")
```



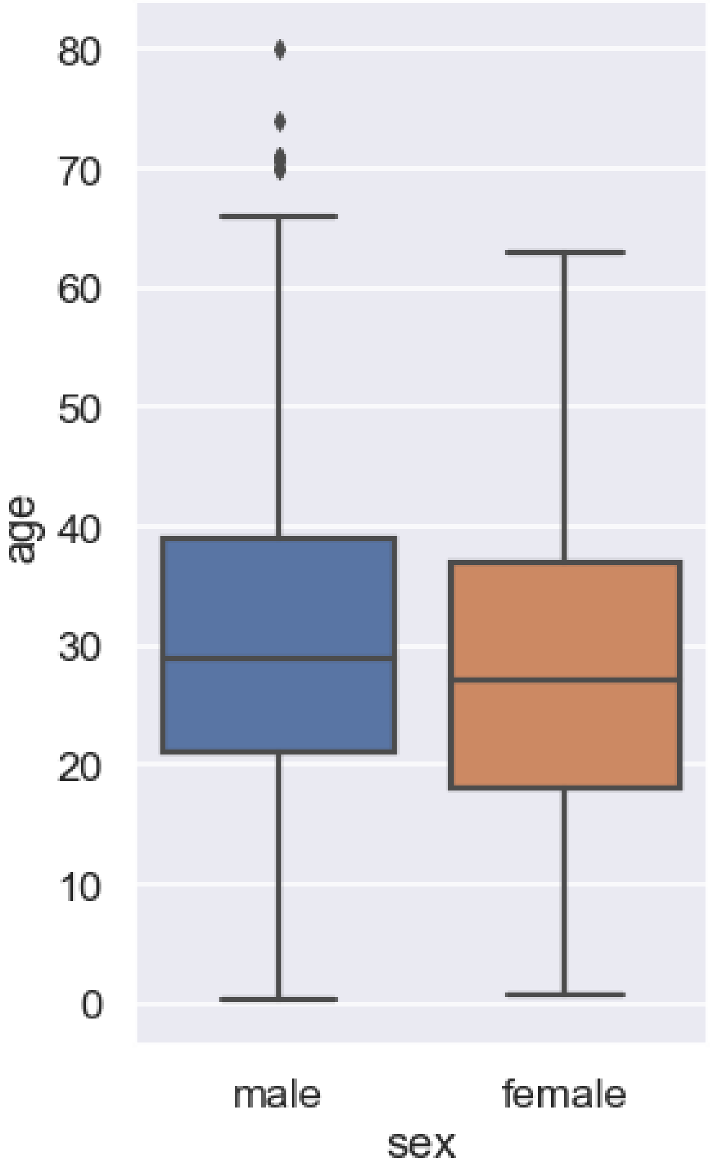
Box Plots

- Used to compare distributions
- Uses quartiles
  - Q1: 25th percentile
  - Q2 (median): 50th
  - Q3: 75th
- Middle line = median
- Box shows 1st and 3rd quartile
- Whiskers show rest of data



Box Plots

- Outliers plotted beyond whiskers
- Interquartile range IQR = Q3 - Q1
- Outliers are defined as:
  - 1.5 * IQR beyond Q1 or Q3
- Example for male ages:
  - Q1 = 21; Q2 = 29; Q3 = 39
  - IQR = 18; 1.5*IQR = 27
  - Outliers are:
    - Above Q3 + 1.5*IQR = 66
    - Below Q1 - 1.5*IQR = -6



Summary

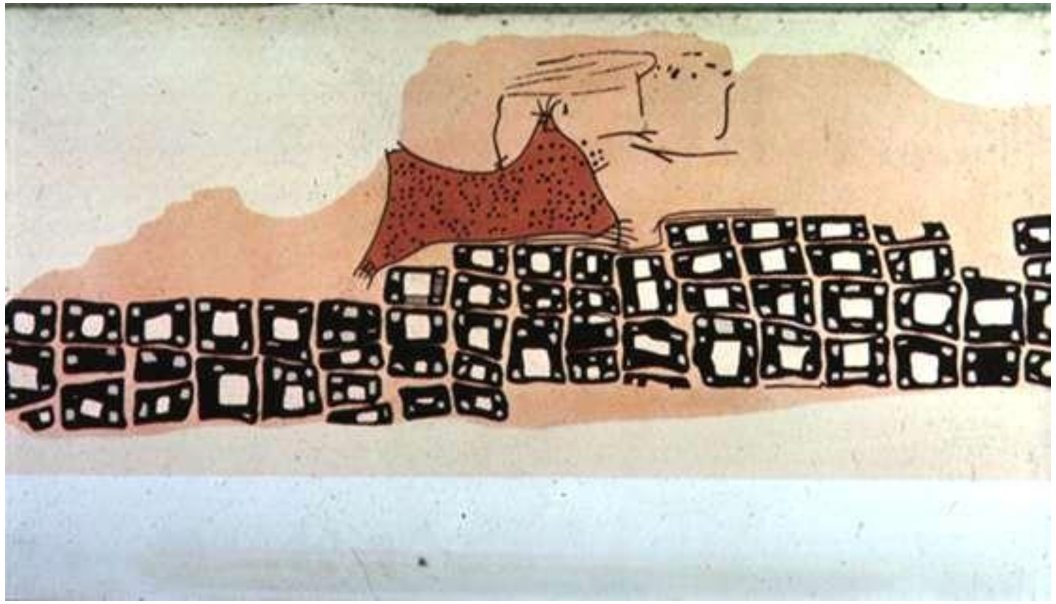
- Use seaborn + matplotlib
  - Pandas also has basic built-in plotting methods

- Types of variables constrain the charts you can make
  - Single quantitative: histogram, density plot
  - 2+ quantitative: scatter plot, 2D density plot
  - Quantitative + qualitative: bar plot, point plot, box plot

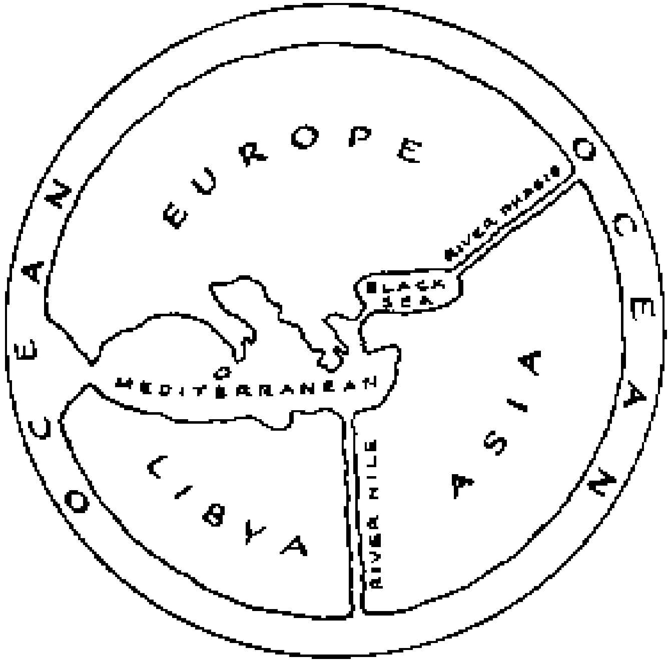
Demo/Tutorial: https://seaborn.pydata.org/tutorial.html

Visualization Goals

Map



Konya town map, Turkey, c. 6200 BC

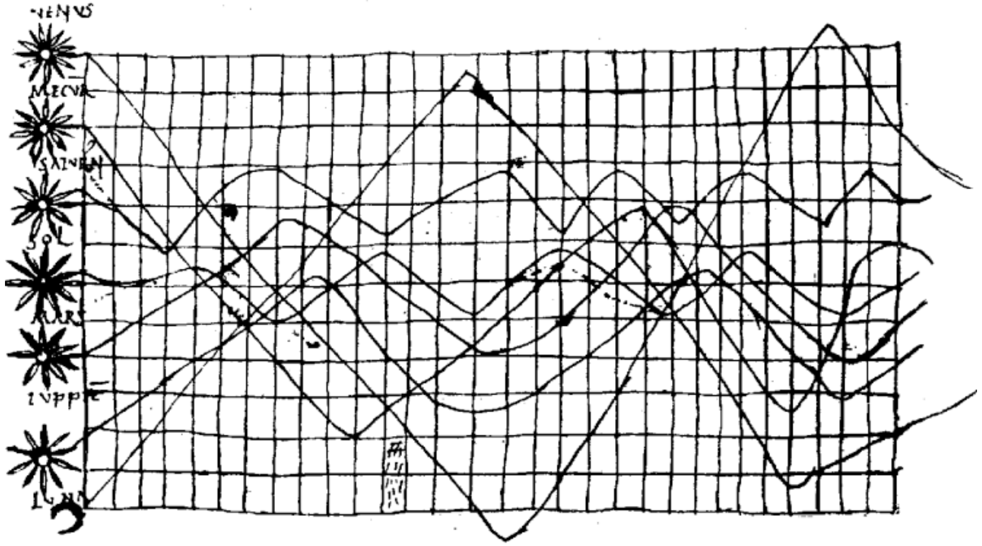


Anaximader's Map of the World

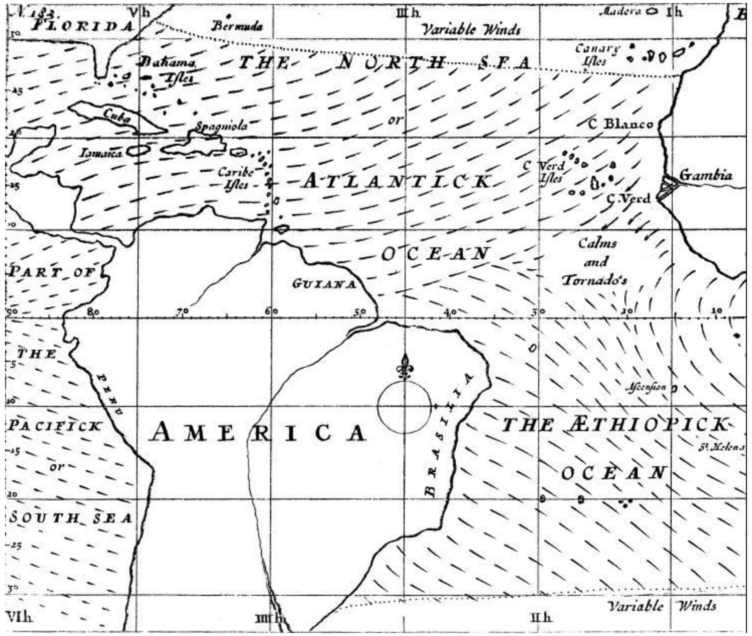
Anaximander of Miletus, c. 550 BC

Milestones Project, Michael Friendly, York U
http://datavis.ca/milestones/index.php?group=pre-1600

Map

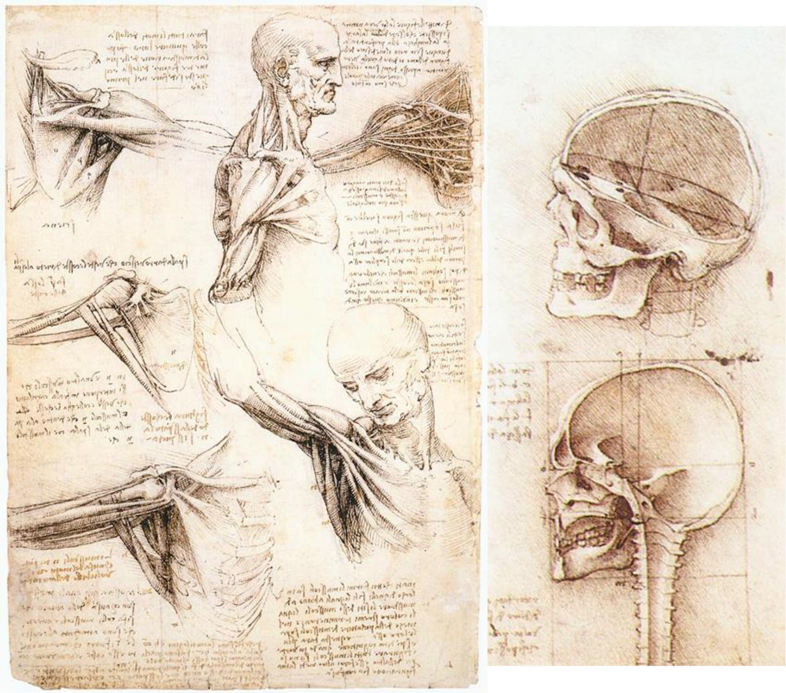


Planetary Movement Diagram, c. 950



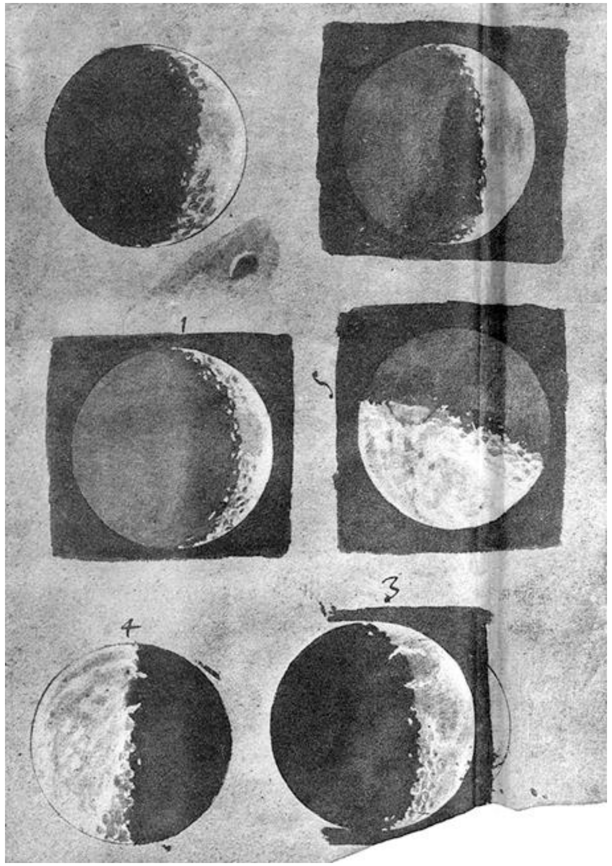
Halley's Wind Map, 1686

Record



Leonardo Da Vinci, ca. 1500

William Curtis (1746-1799)

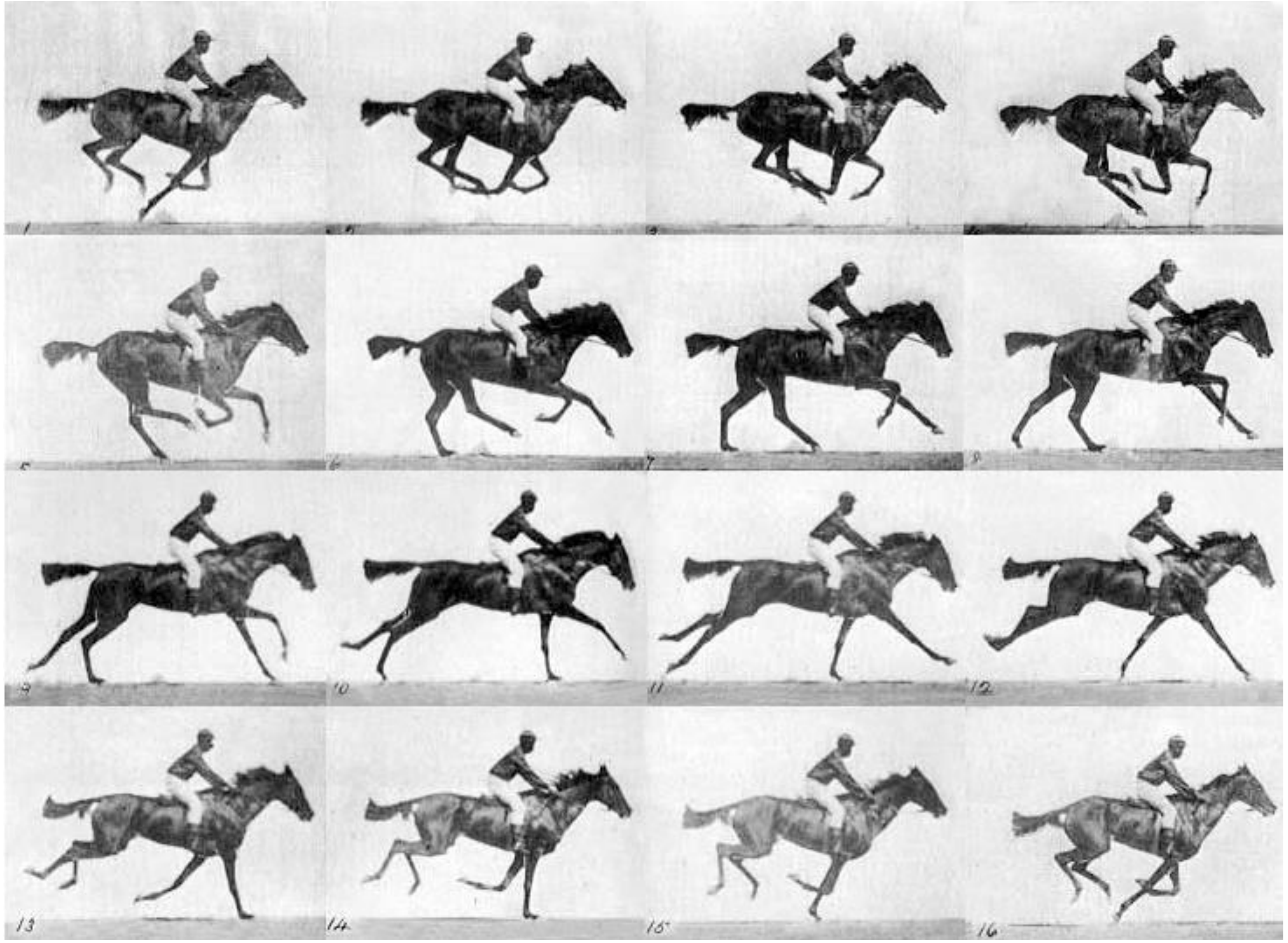


Galileo Galilei, 1616

The History of Visual Communication

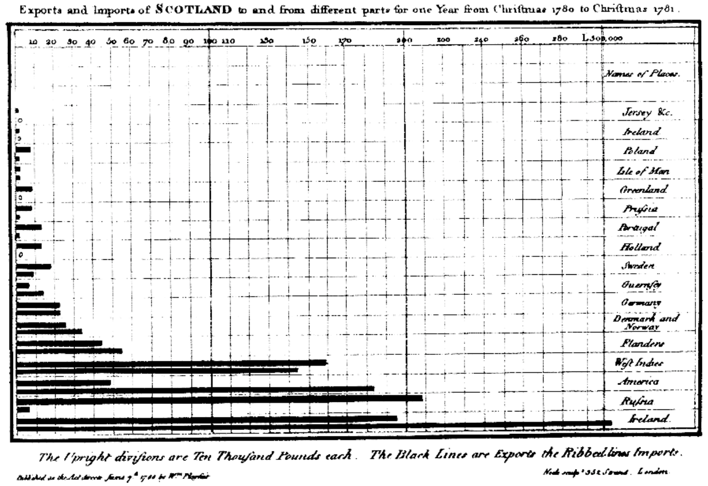
The Galileo Project, Rice University

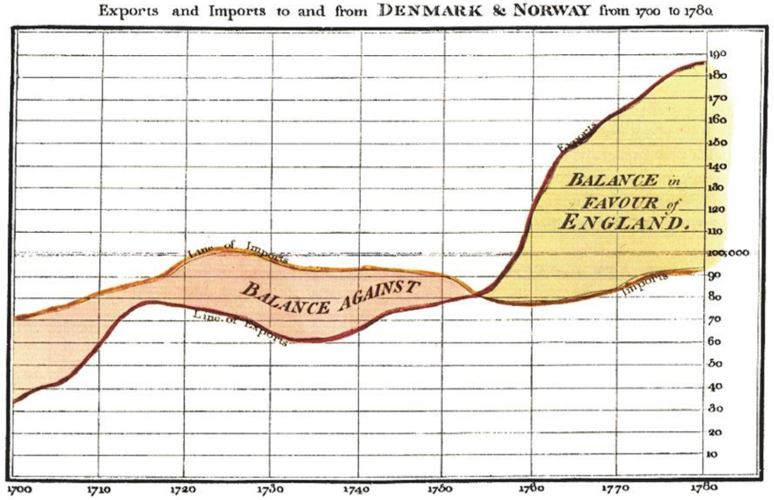
Record



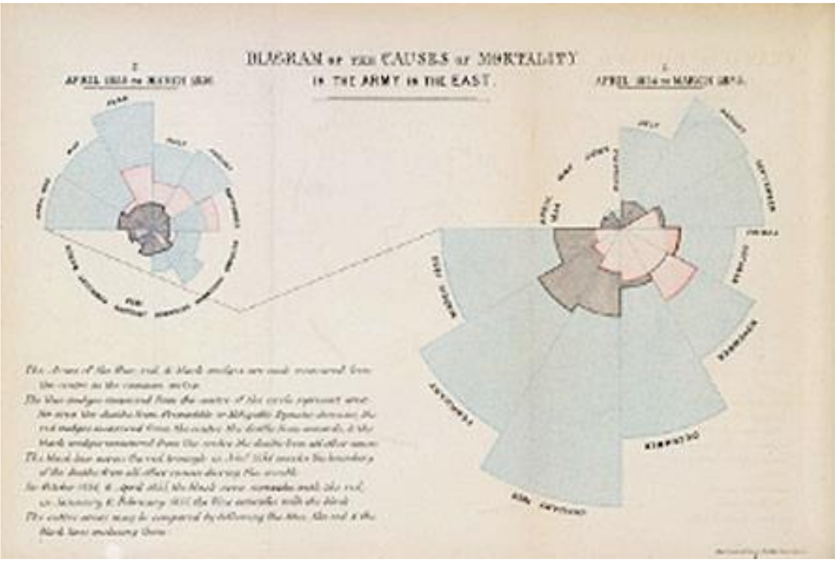
E. J. Muybridge, 1878

Abstract



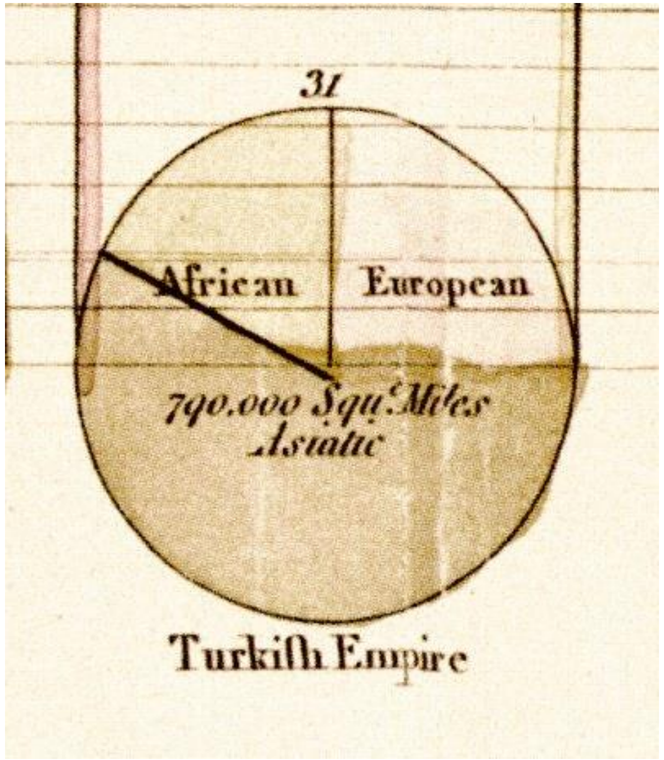


W. Playfair, 1786

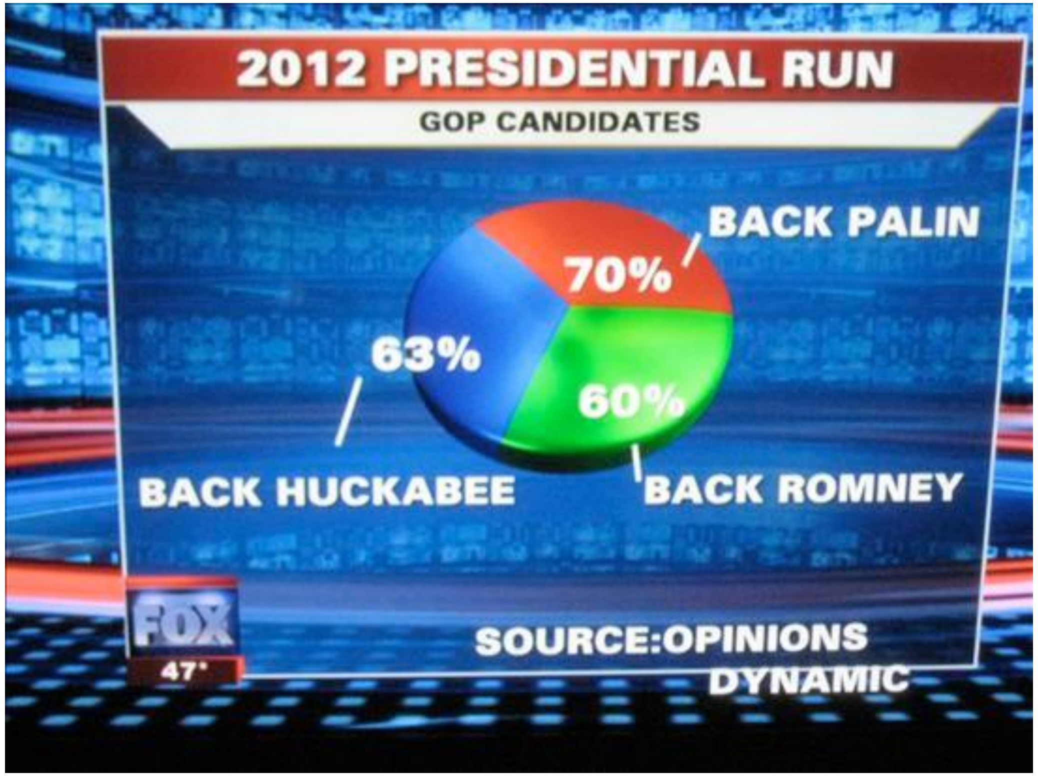


F. Nightingale, 1856

Abstract

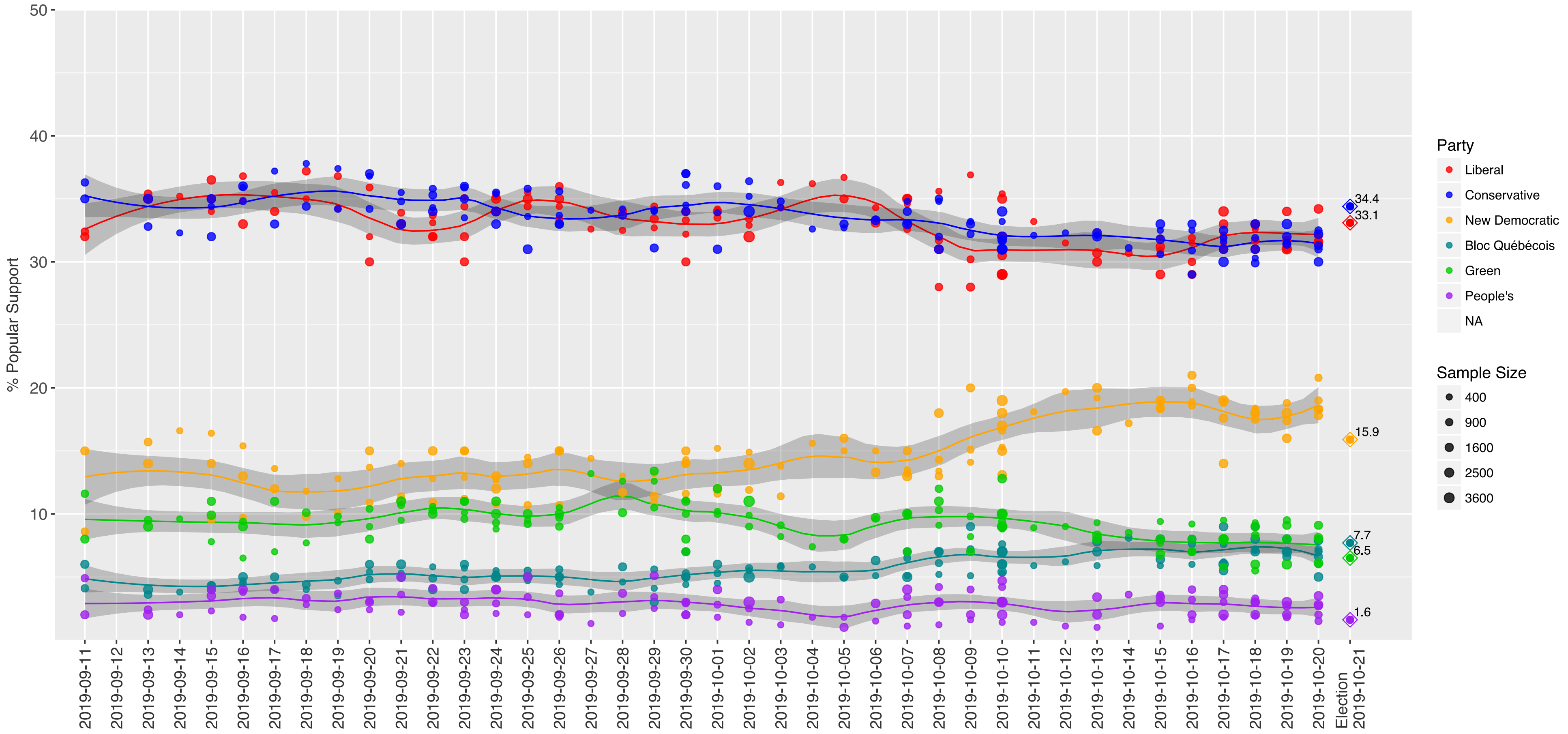


W. Playfair, 1801



Abstract

Canadian pre-campaign voting intentions for the federal election 2019



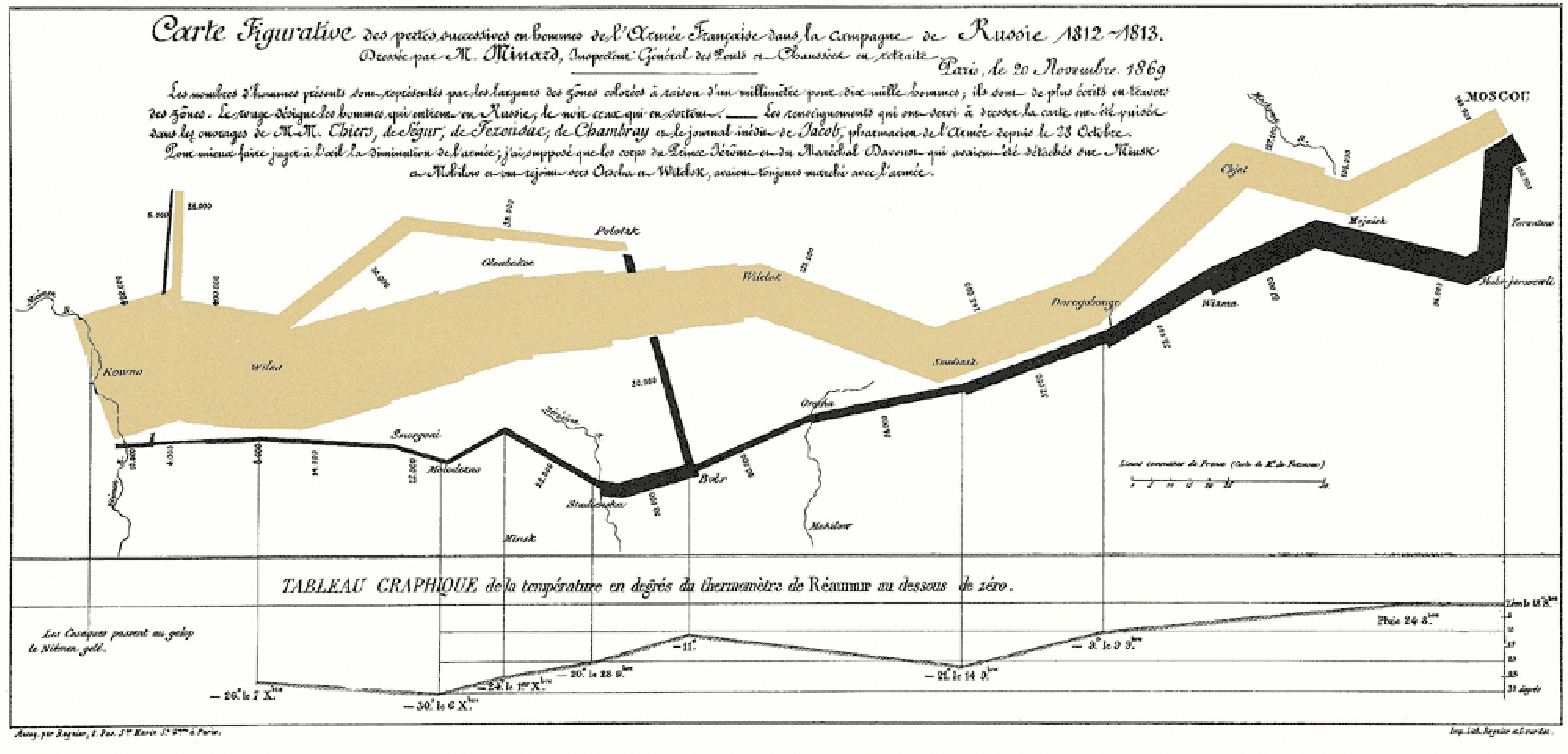
Code available at https://github.com/tylerecouture/wikiplot/blob/master/canadian-federal-polls-pre-19td.R

Discover

John Snow, 1854

E. Tufte, Visual Explanations, 1997

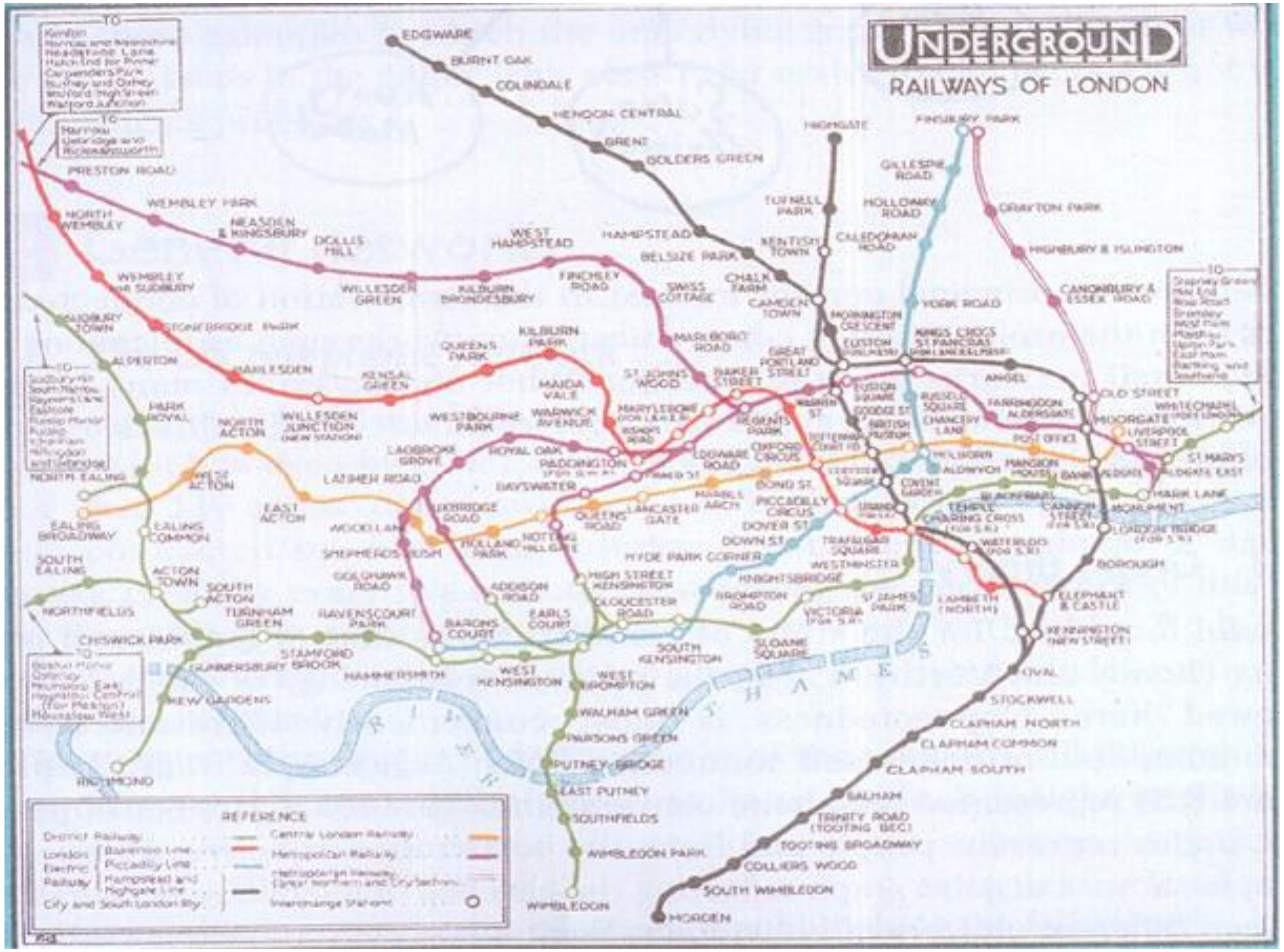
Discover



C.J. Minard, 1869

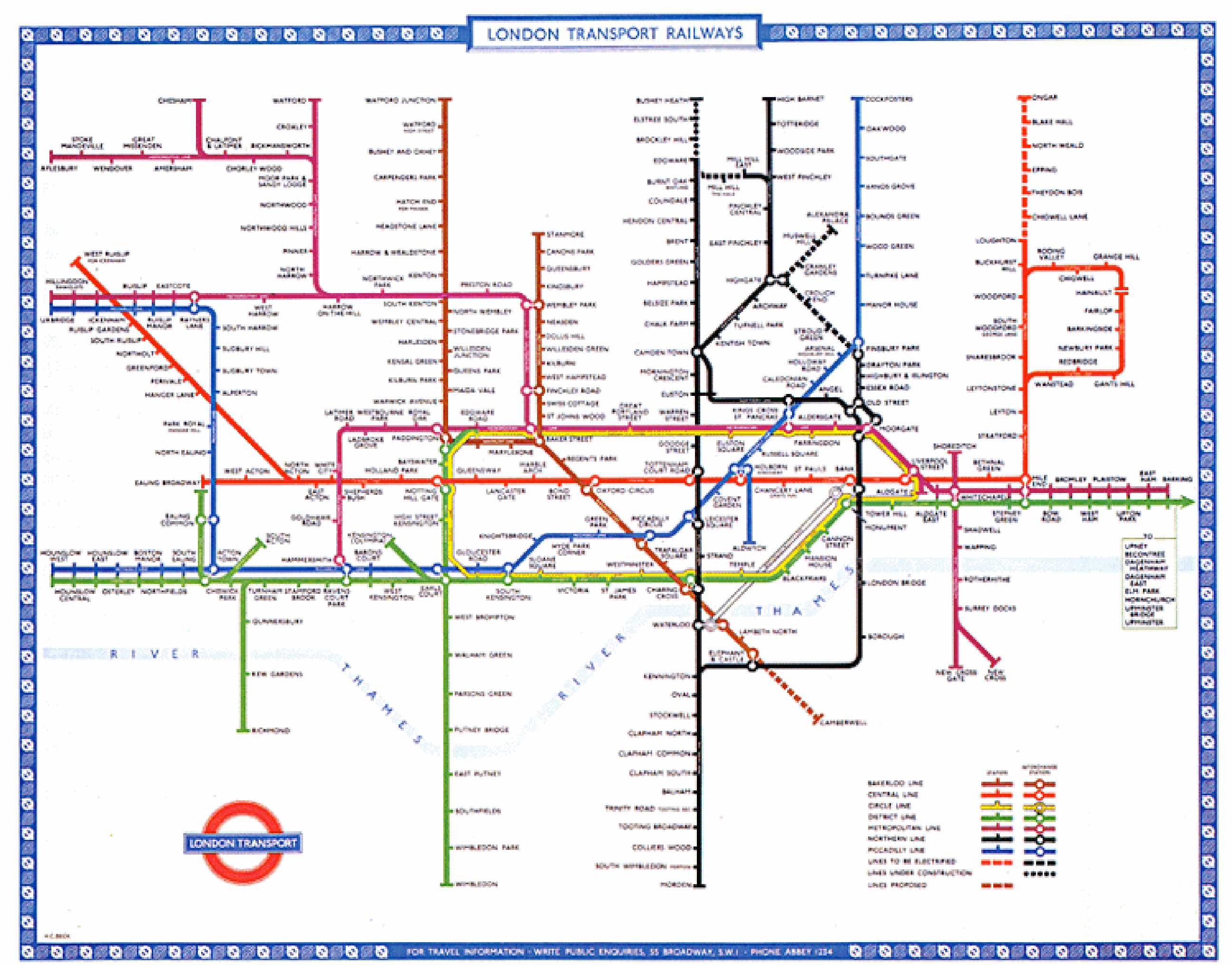
E. Tufte, Writings, Artworks, News

Clarify



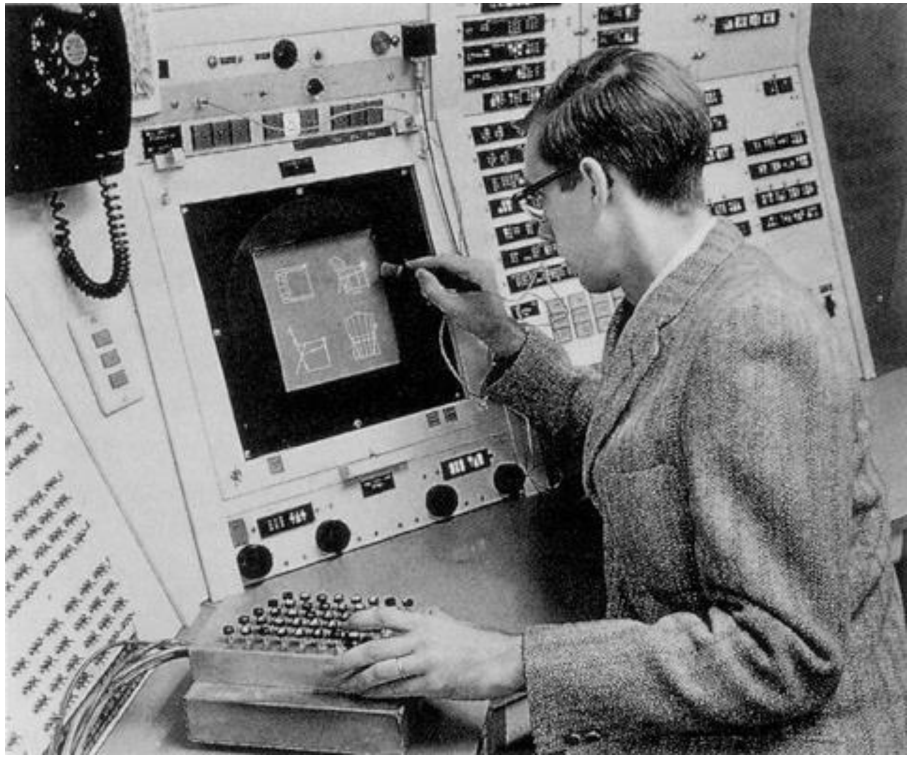
London Subway Map, 1927

Clarify



Harry Beck, 1933

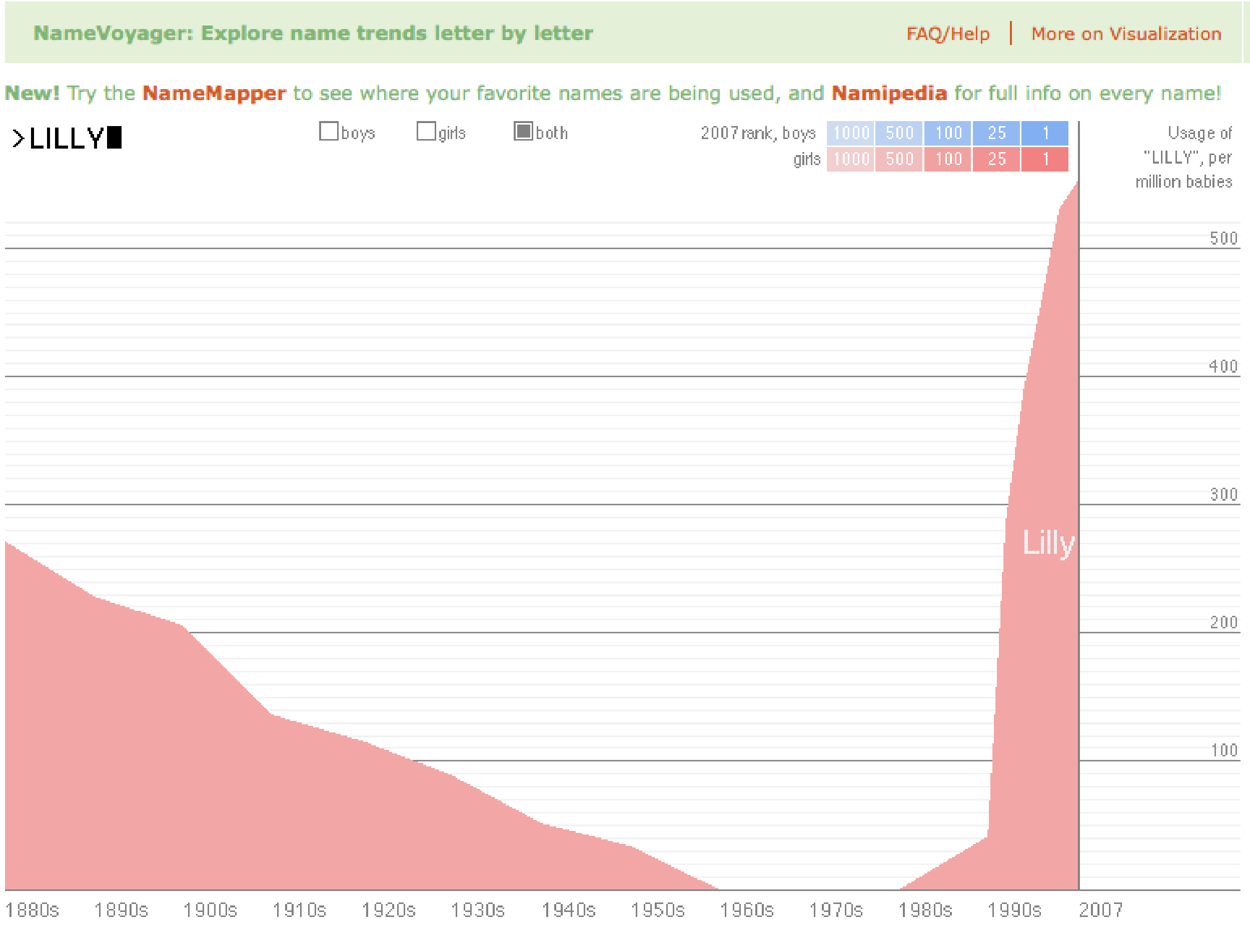
Interact



Ivan Sutherland, Sketchpad, 1963

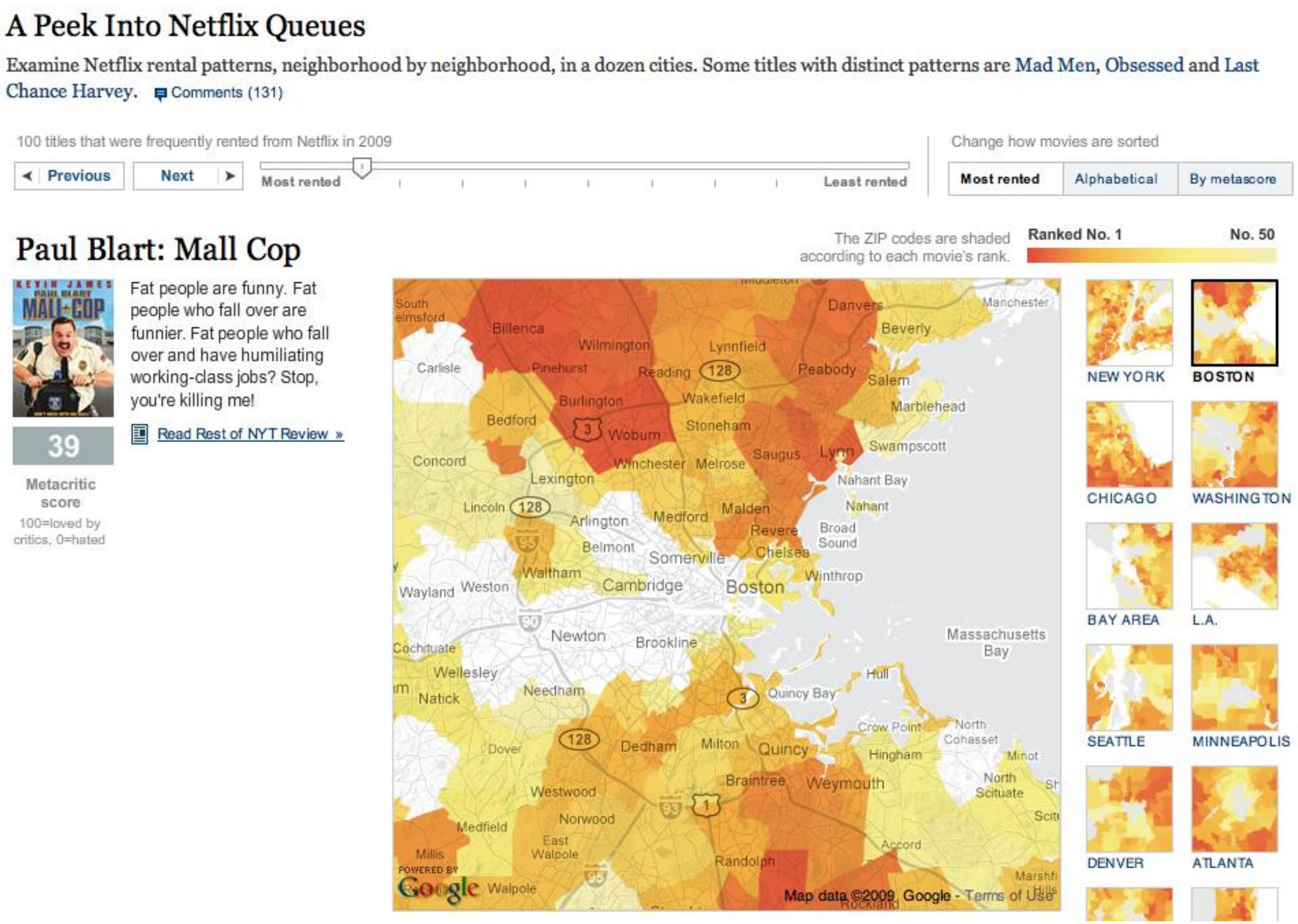
Doug Engelbart, 1968

Interact



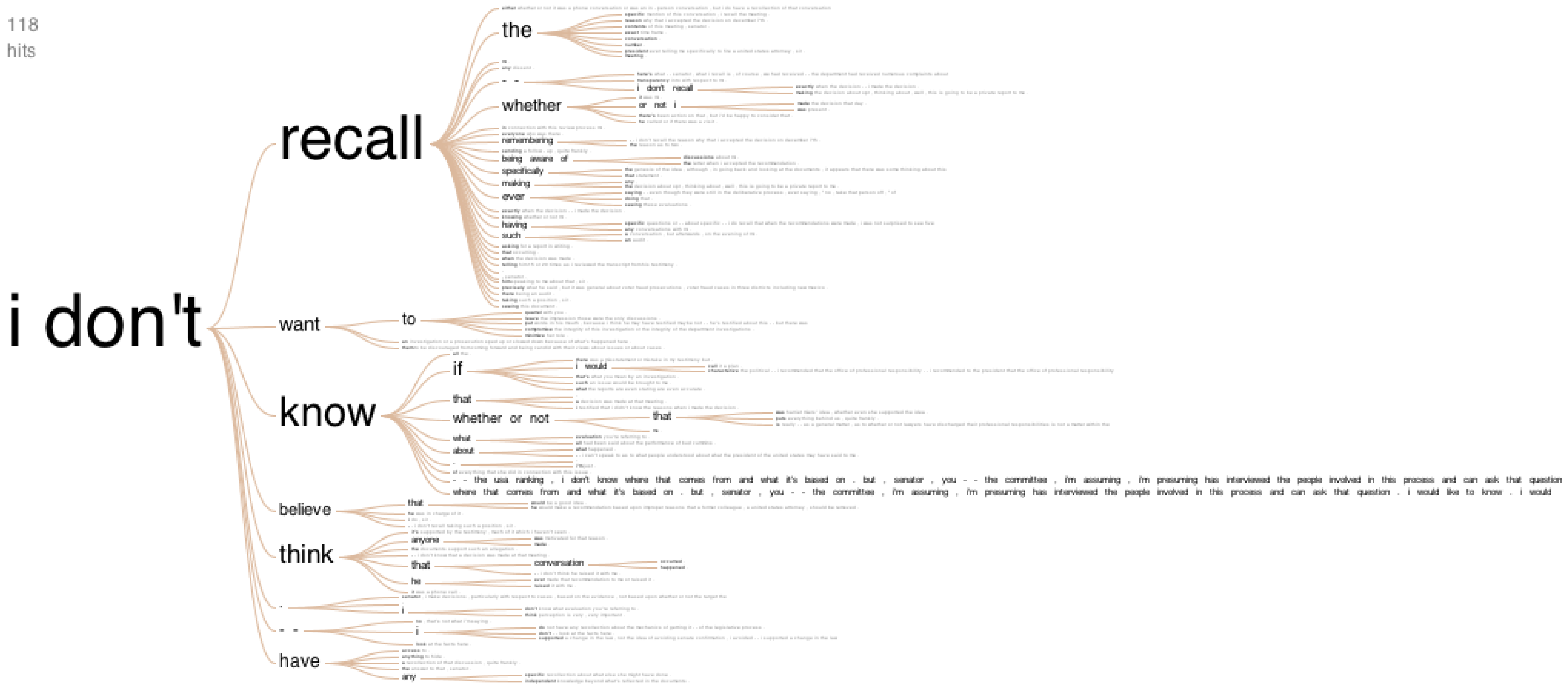
M. Wattenberg, 2005

Interact



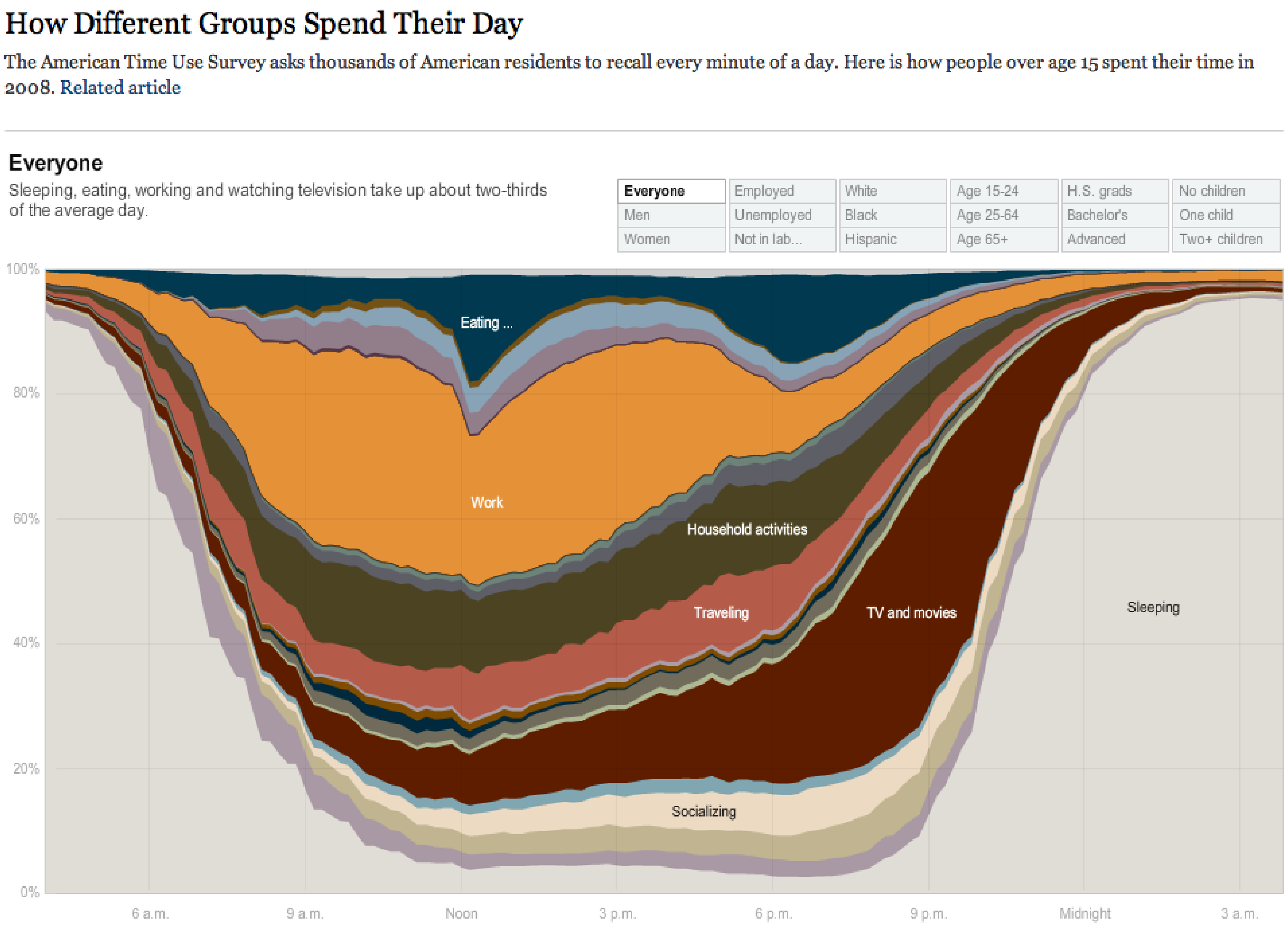
NY Times

Communicate



"Many Eyes", M. Wattenberg 2007

Communicate



NY Times

Inspire / Tell a Story

Hans Rosling, TED 2006

Visualization

- To convey information through visual representations

  Map
  Record
  Abstract
  Discover
  Clarify
  Interact
  Communicate
  Inspire

Goals

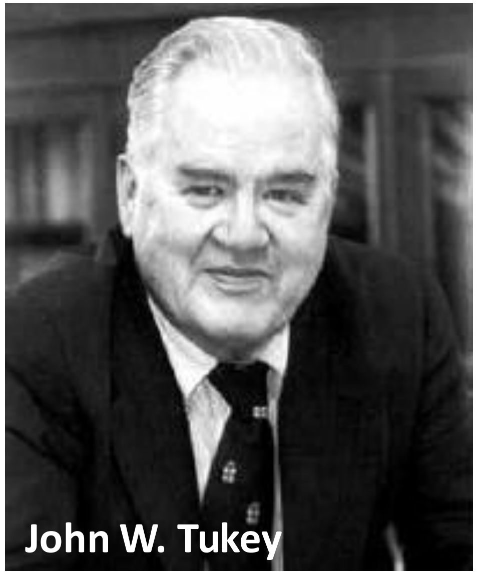
- Insight and analysis
  - Extract the information content
  - Make things and relationships visible
  - Analyze the data by means of the visual representation
- Communication
  - Allow the non–expert to understand
  - Guide the expert into the right direction
- Exploration
  - Interactive control
  - Use visual representation to understand the phenomena

- “The purpose of computing is insight not numbers” (Hamming 1962)

Exploratory Data Analysis (EDA)

EDA is the process of doing Descriptive Statistics

- Aim to understand the data
- Data summarization, visualization, etc.



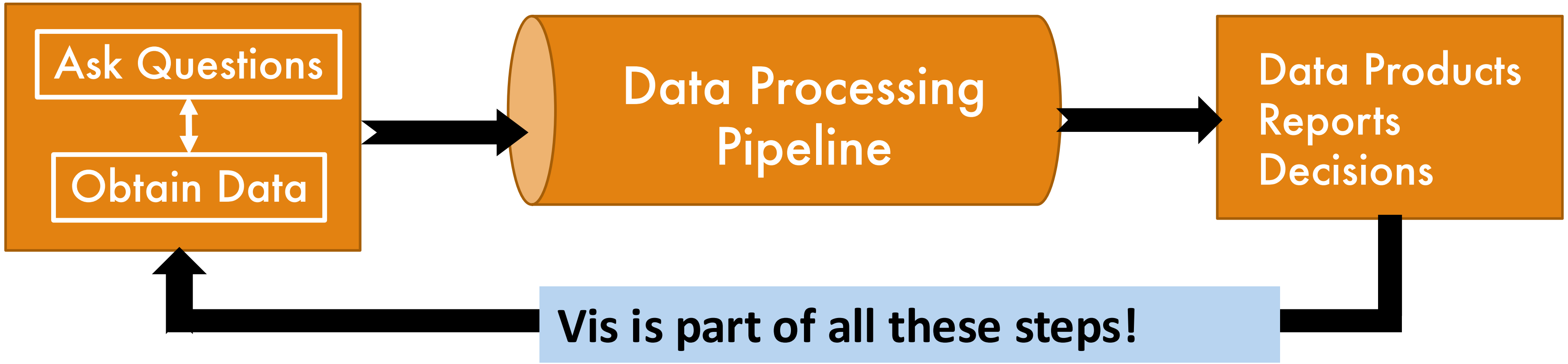
- Professor at Princeton University
- Founding chairman of the Princeton statistics department in 1965
- Worked on EDA at Bell Labs since 60's
- Wrote a book entitled "Exploratory Data Analysis" in 1977

EDA is like detective work

John Tukey:

"Exploratory data analysis is an attitude, a state of flexibility, a willingness to look for those things that we believe are not there, as well as those that we believe to be there."

Recap: Data 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

Sources

Books

- Tamara Munzner: [“Visualization Analysis and Design”](), 2014
- Lau, Gonzalez, Nolan: “Principles and Techniques of Data Science”

Slides

- Jiannan Wang’s CMPT 733 slides, Spring 2017
- Torsten Möller’s Visualization course, Spring 2018
- UC Berkley Data 100 (Lau, Nolan, Dudoit, Perez)