



Computer  
Science

# CSC380: Principles of Data Science

## Data Analysis, Collection, and Visualization

**Prof. Jason Pacheco**

TA: Enfa Rose George

TA: Saiful Islam Salim

Material from: Watkins, J. "Intro. to the Science of Statistics"

# Outline

- Data Visualization
- Data Summarization
- Data Collection and Sampling

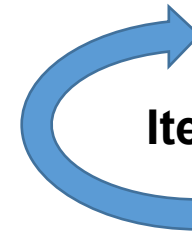
# Outline

- **Data Visualization**
- Data Summarization
- Data Collection and Sampling

# Data Analysis, Exploration, and Visualization

```
141 137 134 134 132 130 129 129 131 135 130 128 129 126 128 128 130
138 136 134 134 135 133 131 129 132 139 133 128 130 128 127 129 131
135 135 134 133 133 132 130 128 132 136 134 130 131 131 132 132 133
133 134 133 132 131 130 130 131 131 129 134 134 130 134 137 134 134
134 134 134 134 133 132 134 138 136 127 135 137 132 136 140 135 139
137 135 136 138 137 135 137 143 142 132 136 138 135 137 138 138 142
139 135 135 138 138 134 135 141 143 133 133 134 135 135 133 138 140
136 137 137 138 141 143 142 144 140 143 142 137 137 139 137 135 136
137 138 136 136 138 140 141 143 140 144 143 139 139 140 138 137 139
137 139 137 136 136 136 137 140 143 146 143 140 141 142 142 143 143
137 140 141 139 138 136 135 137 143 144 142 139 142 144 145 147 146
140 144 144 143 141 137 135 137 139 139 139 143 145 146 147 147
145 148 147 145 143 140 139 141 136 138 140 142 147 147 146 147 149
146 148 147 144 143 141 140 143 137 139 142 145 146 145 145 148 147
145 147 146 143 142 140 140 143 138 140 143 143 143 141 143 148 142
145 145 144 144 143 141 141 142 142 145 146 145 144 141 143 150 144
144 143 142 143 143 142 142 144 143 144 143 144 148 144 142 147 145
146 145 144 143 143 143 144 146 144 144 141 146 157 154 144 143 148
149 148 145 144 143 143 144 145 144 146 142 149 167 169 155 146 151
150 149 147 145 142 142 143 143 145 147 143 147 166 175 164 151 152
150 150 149 147 145 145 145 145 147 148 143 142 154 165 160 148 150
152 152 152 150 149 150 150 149 151 151 150 147 146 152 153 147 151
152 153 153 152 151 151 150 152 152 156 155 148 149 155 153 152
152 152 152 152 152 151 151 151 152 152 152 153 152 151 151 152 154
153 153 153 153 153 153 153 153 154 154 153 153 152 152 150 152 154
153 153 153 153 154 154 154 154 154 154 153 153 153 153 152 153 155
153 153 152 153 154 154 154 154 153 154 154 153 153 153 153 154 157
153 152 152 152 154 155 155 155 153 155 155 154 152 152 152 154 159
```

Encoding

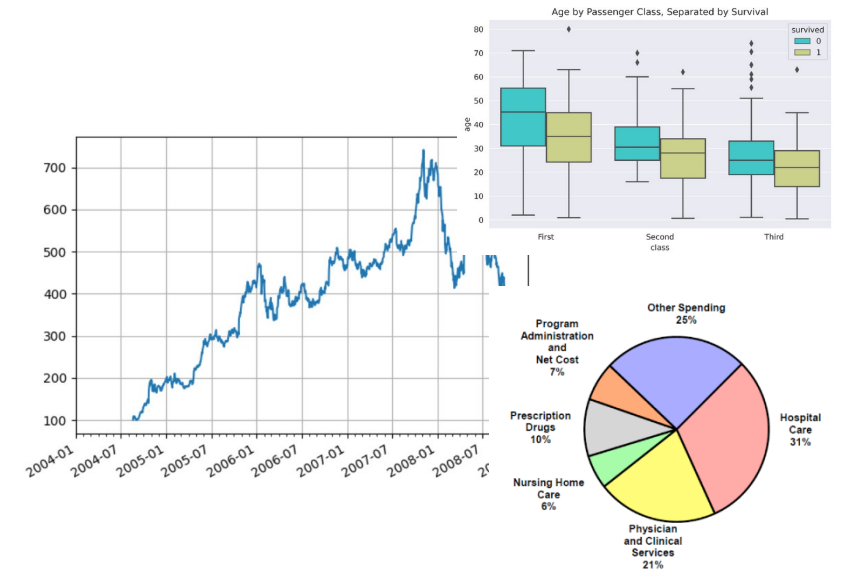
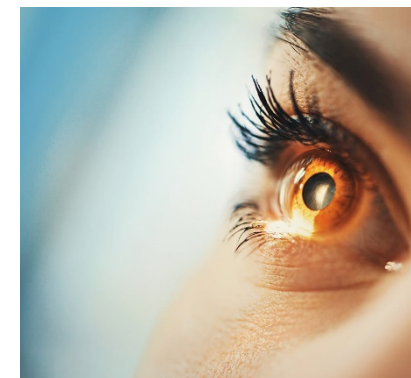


Iterate

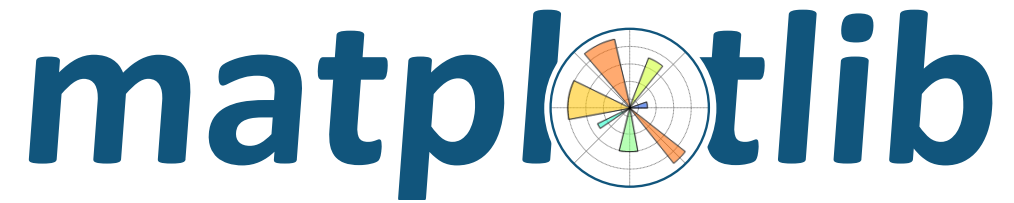
Visual Perception



Understanding



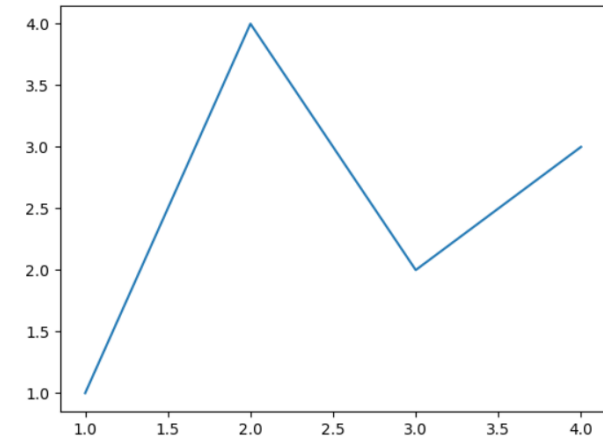
# Data visualization in Python...



```
import matplotlib.pyplot as plt
import numpy as np
```

## Create a simple figure with an axis object,

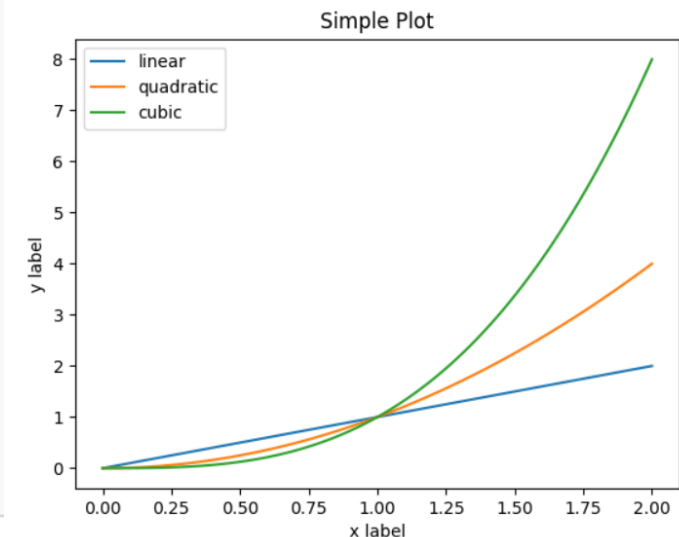
```
fig, ax = plt.subplots() # Create a figure containing a single axes.
ax.plot([1, 2, 3, 4], [1, 4, 2, 3]) # Plot some data on the axes.
```

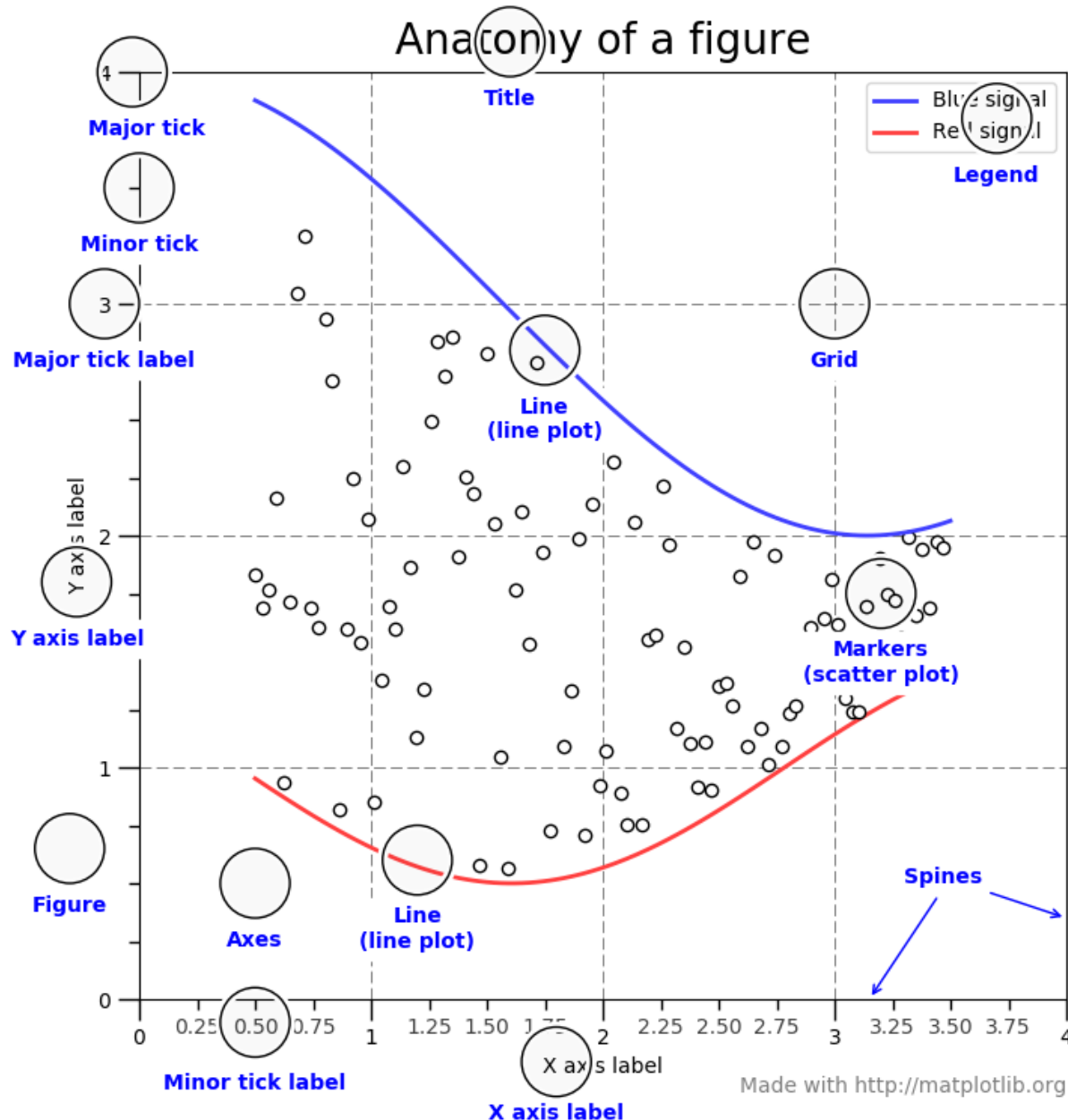


## A more complicated plot...

```
x = np.linspace(0, 2, 100)

# Note that even in the OO-style, we use `.pyplot.figure` to create the figure.
fig, ax = plt.subplots() # Create a figure and an axes.
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.
```





May need to **show** the plot with,  
`plt.show()`  
Typically, a **blocking** event.

Documentation + tutorials:

<https://matplotlib.org/>

- Identifying Consumers
- Recommending Products
- Analyzing Reviews

## E-commerce



- Predicting Potential Problems
- Monitoring Systems
- Automating Manufacturing Units
- Maintenance Scheduling
- Anomaly Detection

## Manufacturing



- Fraud Detection
- Credit Risk Modeling
- Customer Lifetime Value

## Banking



## Healthcare

- Medical Image Analysis
- Drug Discovery
- Bioinformatics
- Virtual Assistants



## Transport

- Self Driving Cars
- Enhanced Driving Experience
- Car Monitoring System
- Enhancing the safety of passengers



## Finance

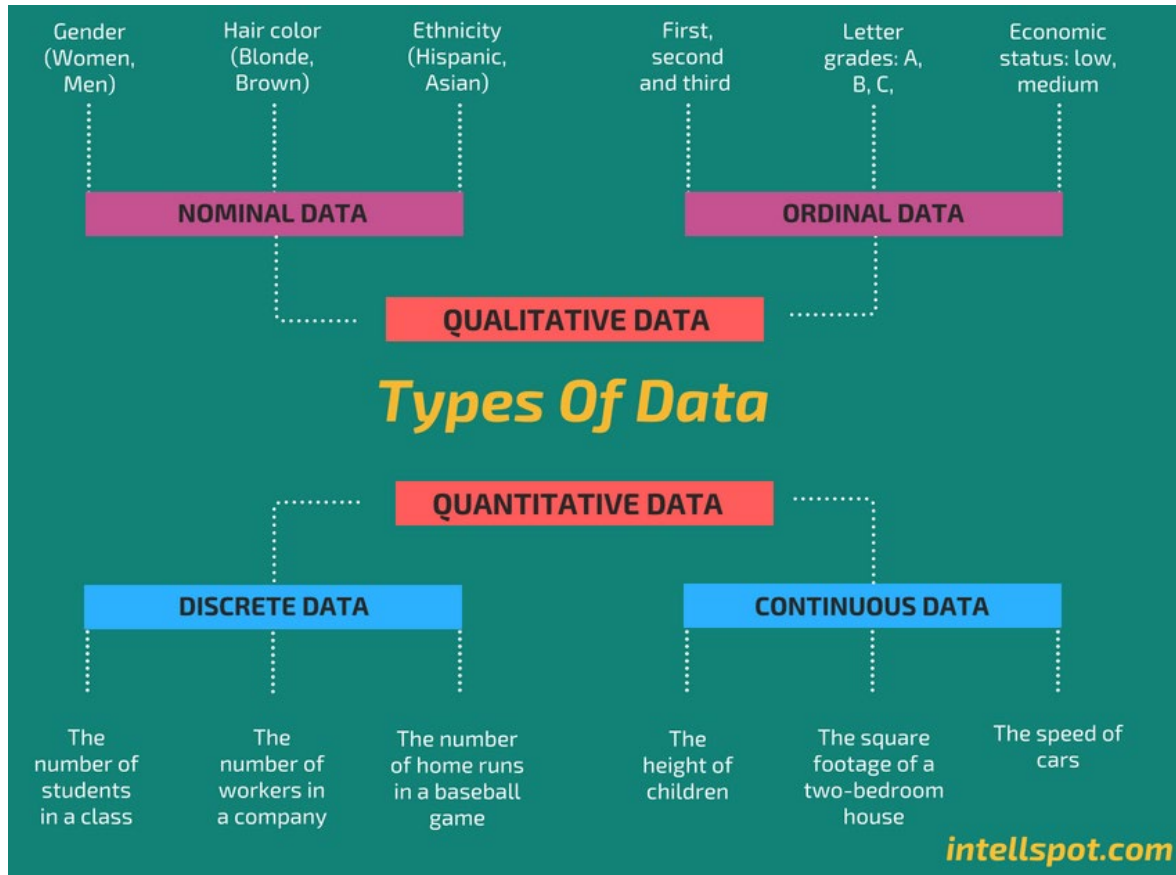
- Customer Segmentation
- Strategic Decision Making
- Algorithmic Trading
- Risk Analytics

# Data Science Applications



# Types of Data

*Data come in many forms, each requiring different approaches & models*



**Qualitative or categorical** : partition data into classes (flexible but imprecise)


**Quantitative** : can perform mathematical operations (e.g. addition, subtraction, ordering)

*We often refer to different types of data as **variables***



# Categorical Variables

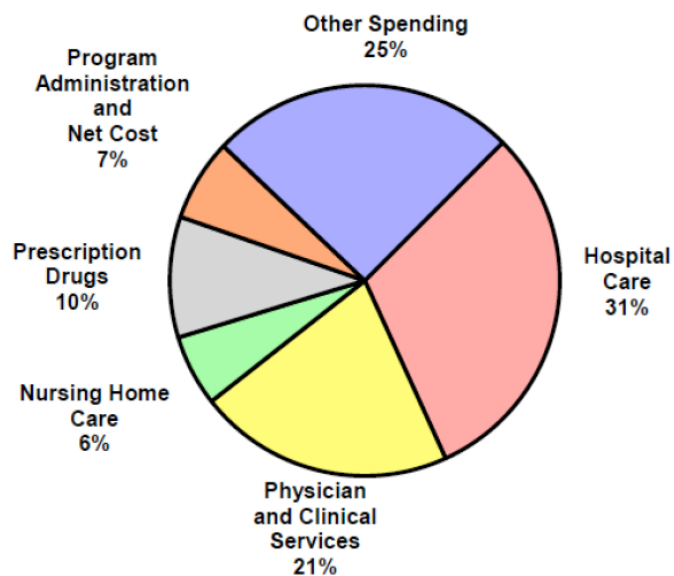
## Examples

- Roll of a die: 1,2,3,4,5 or 6  Numerical data can be categorical or quantitative depending on context
- Blood Type: A, B, AB, or O
- Political Party: Democrat, Republican, etc.
- Type of Rock: Igneous, Sedimentary, or Metamorphic
- Word Identity: NP, VP, N, V, Adj, Adv, etc.

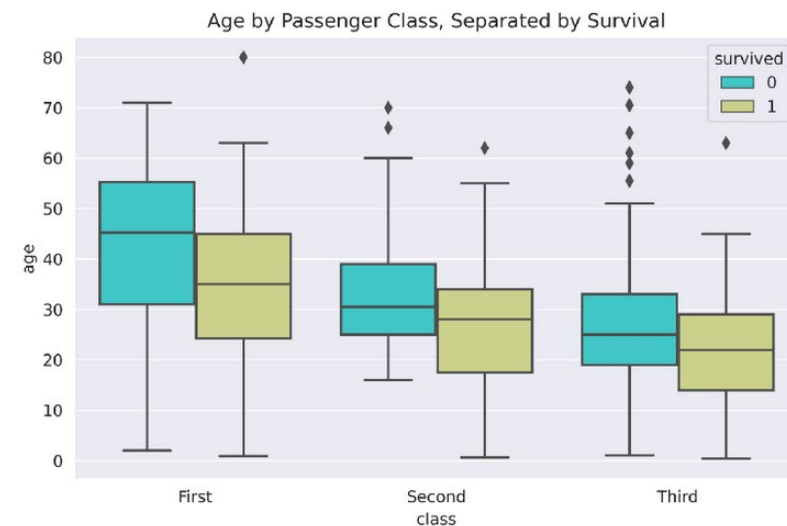
Quantitative data can be converted to categorical by defining ranges:

- Small [0, 10cm), Medium [10, 100cm), Large [100cm, 1m), XL [1m, -)
- Low [ less than -100dB), Moderate [-100dB, -50dB), Loud [over -50dB)

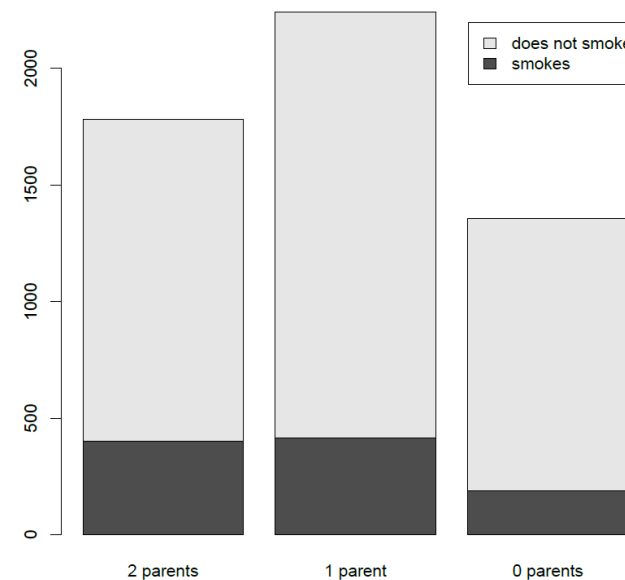
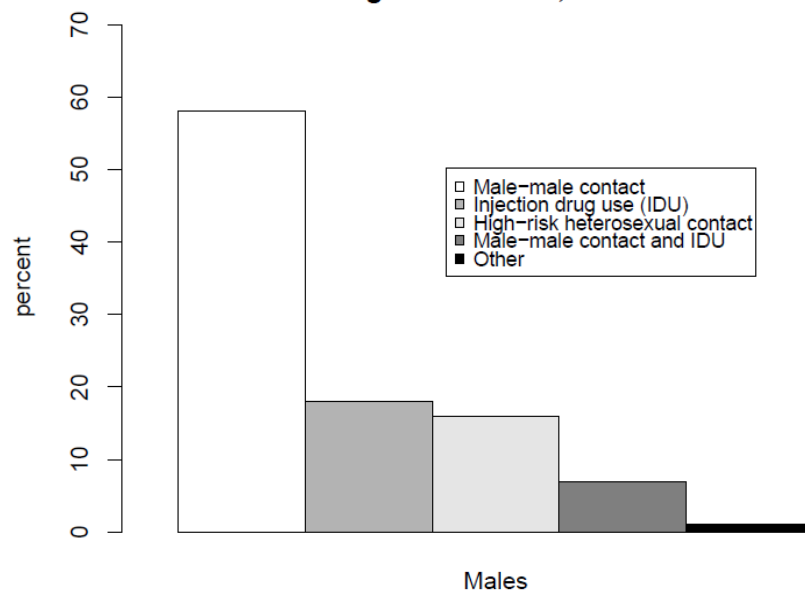
# Visualizing Categorical Variables



	student smokes	student does not smoke	total
2 parents smoke	400	1380	1780
1 parent smokes	416	1823	2239
0 parents smoke	188	1168	1356
total	1004	4371	5375



Proportion of AIDS Cases by Sex and Transmission Category Diagnosed – USA, 2005



# Pie Chart

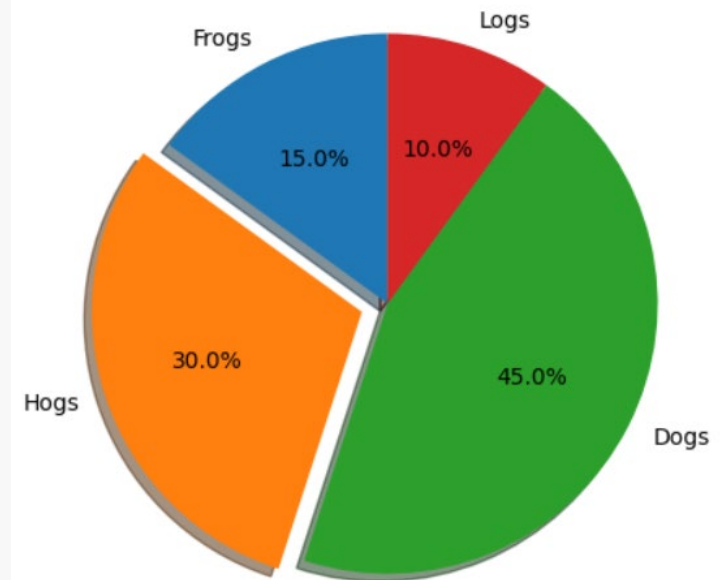
*Circular chart divided into sectors, illustrating relative magnitudes in frequencies or percent. In a pie chart, the area is proportional to the quantity it represents.*

```
import matplotlib.pyplot as plt

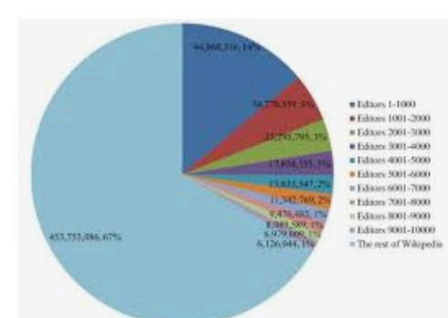
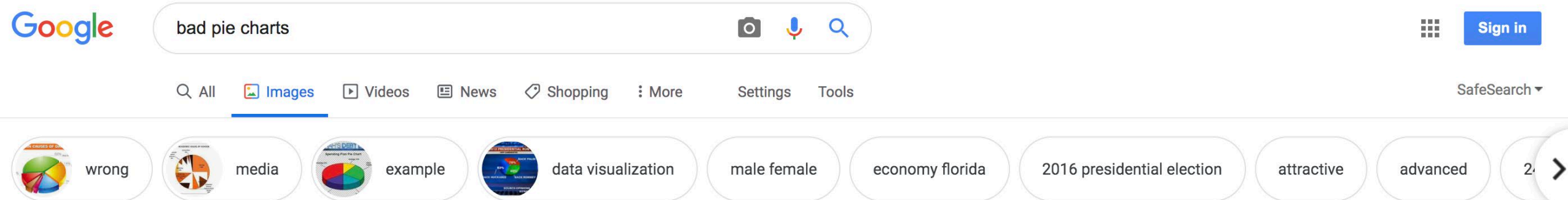
# Pie chart, where the slices will be ordered and plotted counter-clockwise:
labels = 'Frogs', 'Hogs', 'Dogs', 'Logs'
sizes = [15, 30, 45, 10]
explode = (0, 0.1, 0, 0) # only "explode" the 2nd slice (i.e. 'Hogs')

fig1, ax1 = plt.subplots()
ax1.pie(sizes, explode=explode, labels=labels, autopct='%1.1f%%',
        shadow=True, startangle=90)
ax1.axis('equal') # Equal aspect ratio ensures that pie is drawn as a circle.

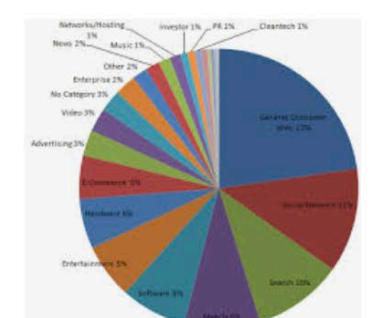
plt.show()
```



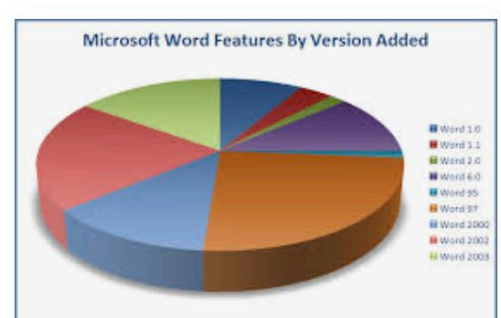
# Maybe the biggest problem with pie charts is that they have been so often done poorly...



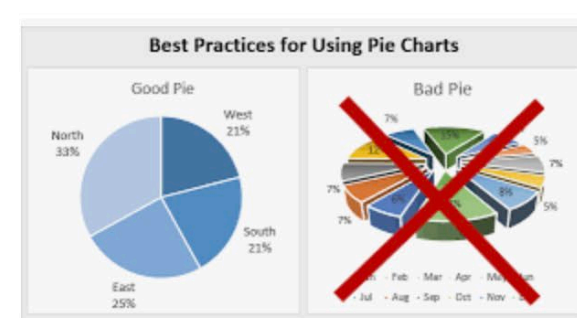
Yet another bad pie chart : datsugly reddit.com



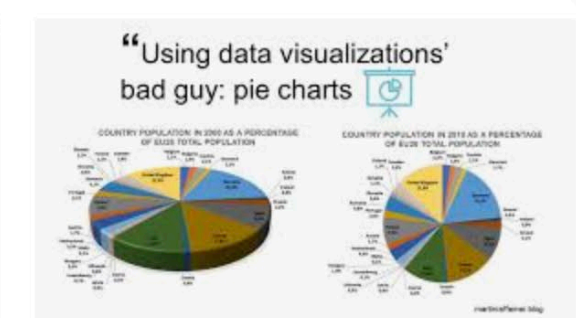
death to pie charts — storyelli... storytellingwithdata.com



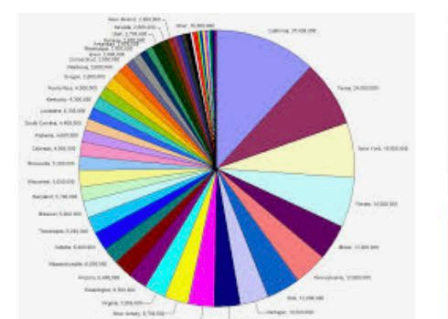
Pie charts: the bad, the worst and the ... visuanalyze.wordpress.com



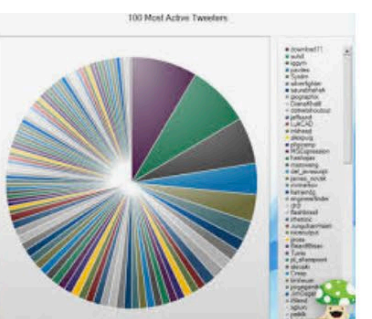
When to use Pie Charts in Dashboards ... excelcampus.com



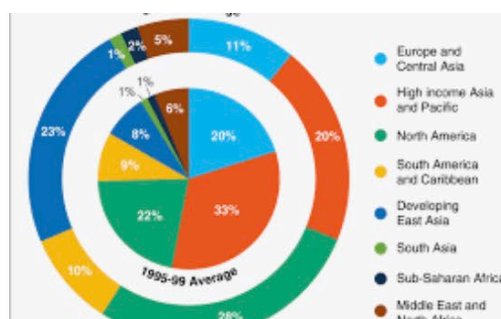
Using data visualizations' bad guy: pie ... martinraffiner.blog



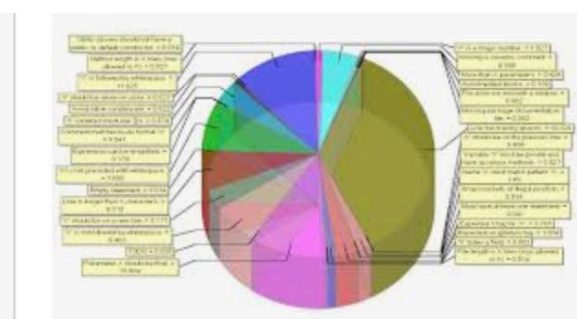
Understanding Pie Charts eagereyes.org



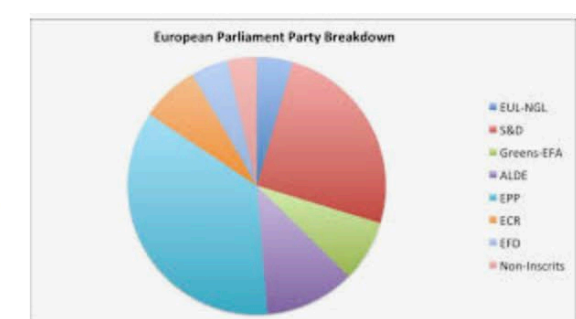
Pie charts: the bad, the worst an... visuanalyze.wordpress.com



Remake: Pie-in-a-Donut Chart - Policy Viz policyviz.com



Pin on Chartjunk Data Visualization pinterest.com



Pie Charts Are The Worst - Business Insider businessinsider.com

# Bar Chart

*We perceive differences in height / length better than area...*

```
plt.bar()
```

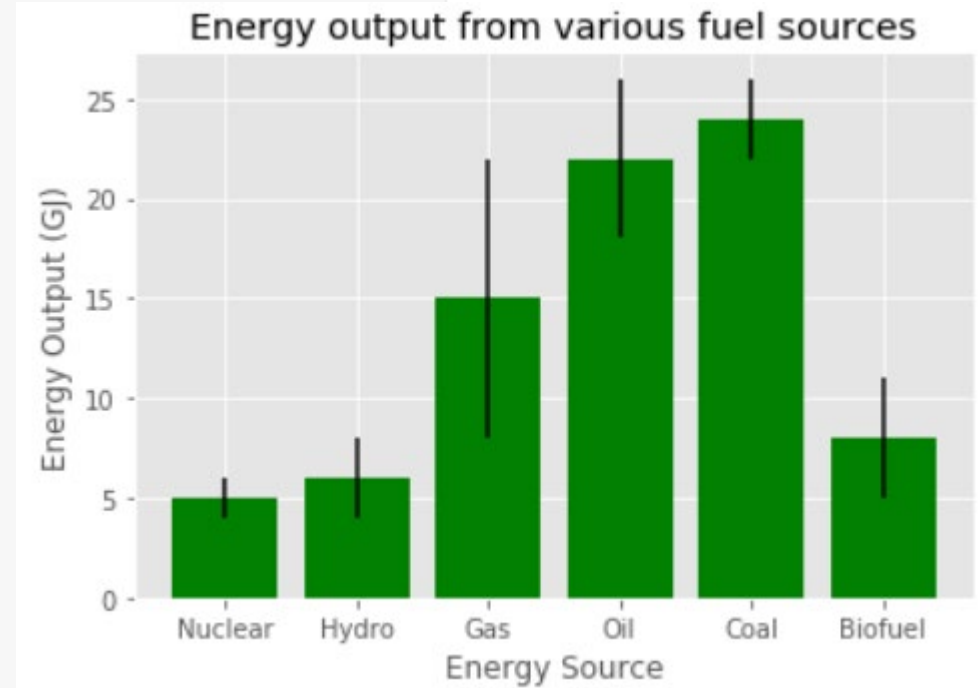
```
x = ['Nuclear', 'Hydro', 'Gas', 'Oil', 'Coal', 'Biofuel']
energy = [5, 6, 15, 22, 24, 8]
variance = [1, 2, 7, 4, 2, 3]

x_pos = [i for i, _ in enumerate(x)]

plt.bar(x_pos, energy, color='green', yerr=variance)
plt.xlabel("Energy Source")
plt.ylabel("Energy Output (GJ)")
plt.title("Energy output from various fuel sources")

plt.xticks(x_pos, x)

plt.show()
```



# Bar Chart

*Don't make readers tilt their heads, consider rotating for readability...*

```
plt.barh()
```

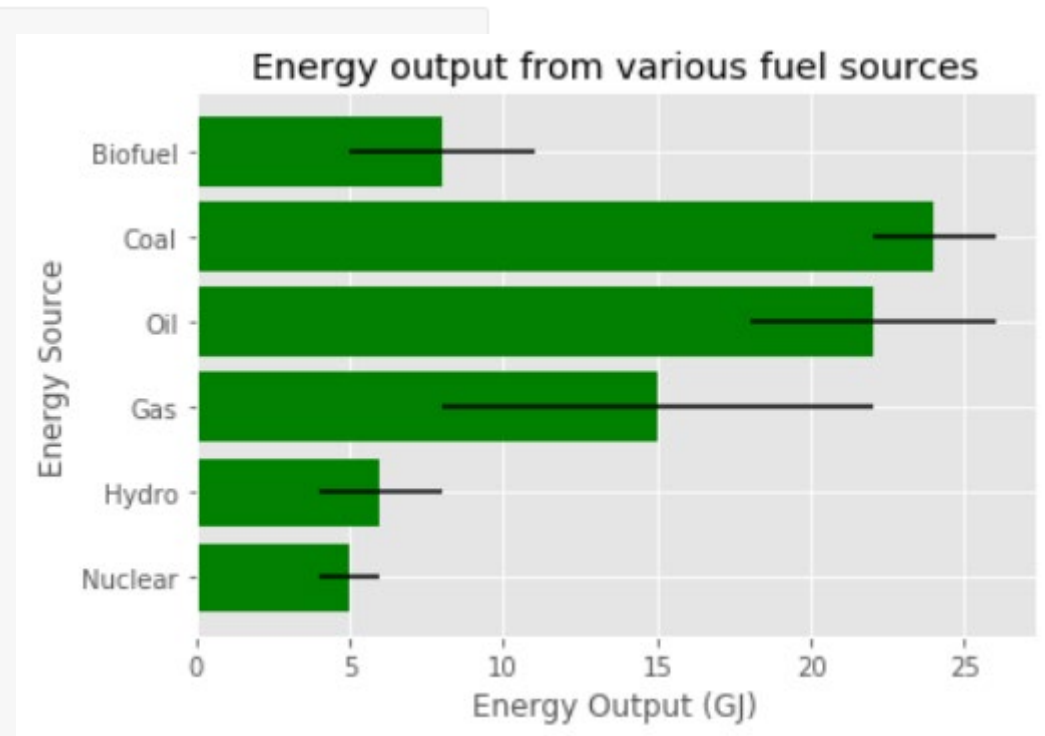
```
x = ['Nuclear', 'Hydro', 'Gas', 'Oil', 'Coal', 'Biofuel']
energy = [5, 6, 15, 22, 24, 8]
variance = [1, 2, 7, 4, 2, 3]

x_pos = [i for i, _ in enumerate(x)]

plt.barh(x_pos, energy, color='green', xerr=variance)
plt.ylabel("Energy Source")
plt.xlabel("Energy Output (GJ)")
plt.title("Energy output from various fuel sources")

plt.yticks(x_pos, x)

plt.show()
```



# Bar Chart

## *Multiple groups of bars...*

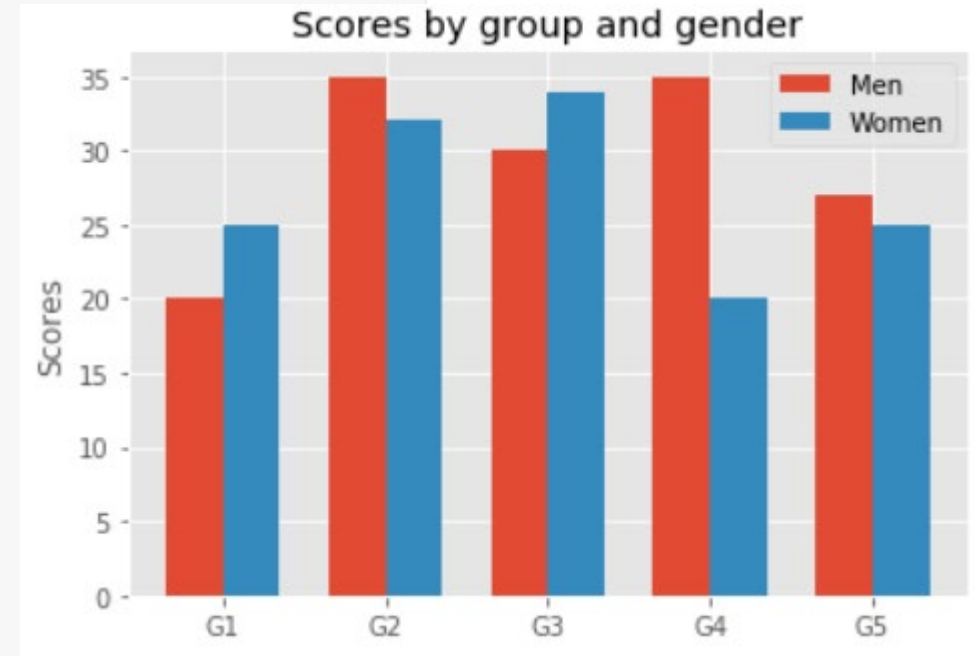
```
import numpy as np

N = 5
men_means = (20, 35, 30, 35, 27)
women_means = (25, 32, 34, 20, 25)

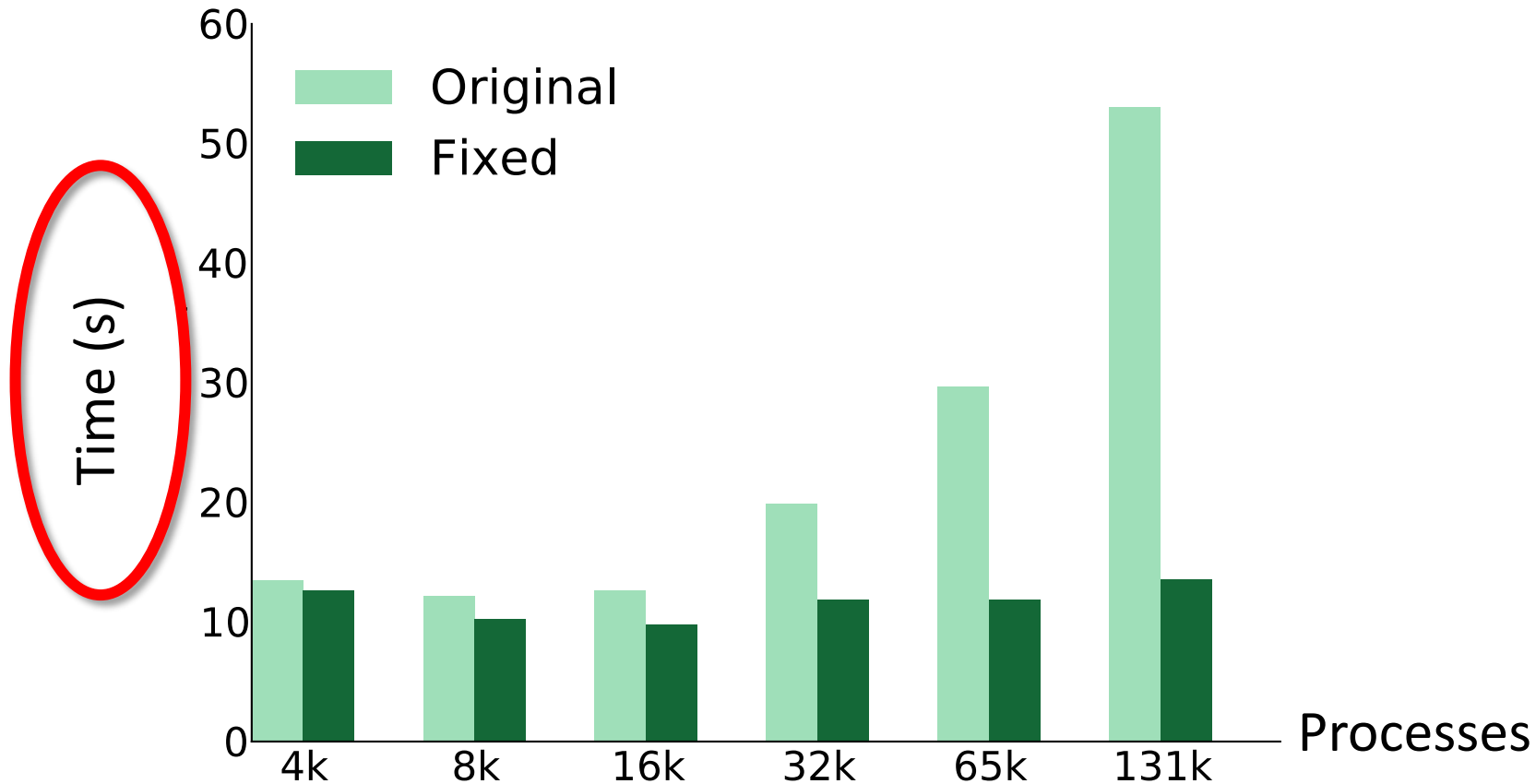
ind = np.arange(N)
width = 0.35
plt.bar(ind, men_means, width, label='Men')
plt.bar(ind + width, women_means, width,
        label='Women')

plt.ylabel('Scores')
plt.title('Scores by group and gender')

plt.xticks(ind + width / 2, ('G1', 'G2', 'G3', 'G4', 'G5'))
plt.legend(loc='best')
plt.show()
```

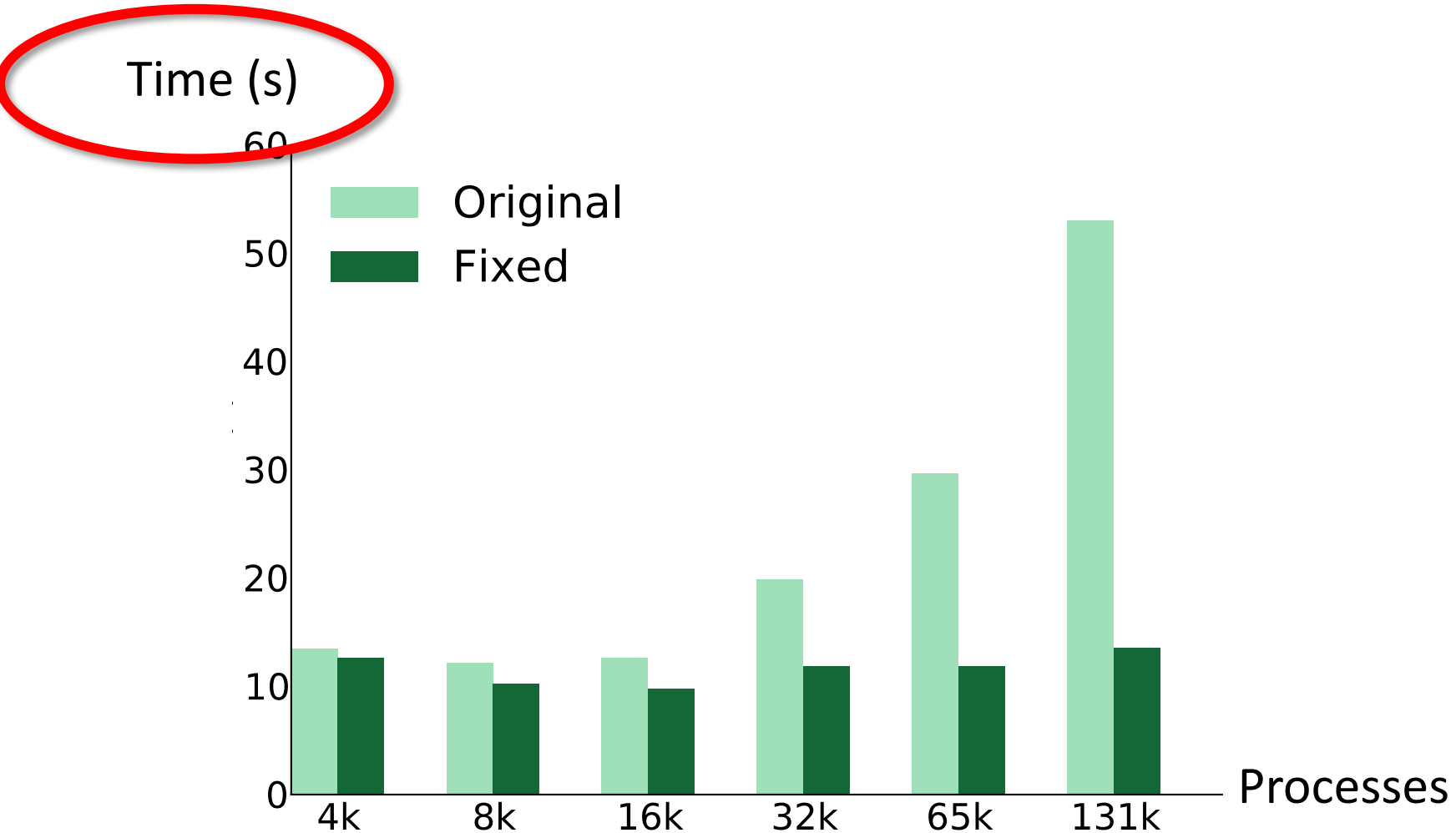


# Labels on the y-axis need not be vertical





# Labels on the y-axis need not be vertical



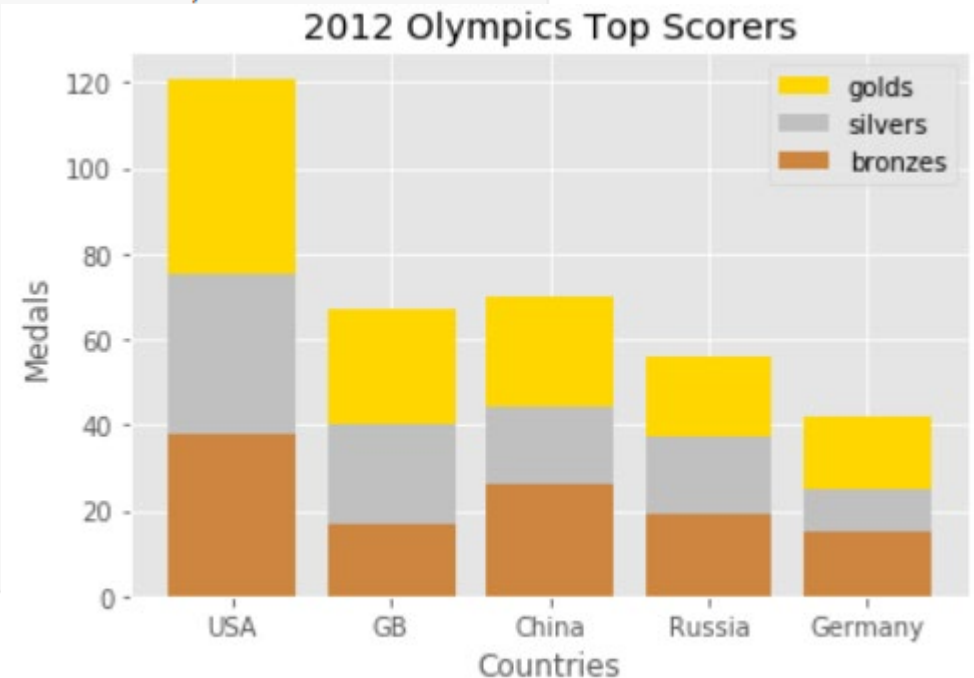
# Stacked Bar Chart

```
countries = ['USA', 'GB', 'China', 'Russia', 'Germany']
bronzes = np.array([38, 17, 26, 19, 15])
silvers = np.array([37, 23, 18, 18, 10])
golds = np.array([46, 27, 26, 19, 17])
ind = [x for x, _ in enumerate(countries)]

plt.bar(ind, golds, width=0.8, label='golds', color='gold', bottom=silvers+bronzes)
plt.bar(ind, silvers, width=0.8, label='silvers', color='silver', bottom=bronzes)
plt.bar(ind, bronzes, width=0.8, label='bronzes', color='#CD853F')

plt.xticks(ind, countries)
plt.ylabel("Medals")
plt.xlabel("Countries")
plt.legend(loc="upper right")
plt.title("2012 Olympics Top Scorers")

plt.show()
```



# Length is still considered more effective than Area

Stacked bar charts are suggested because they encode with length.

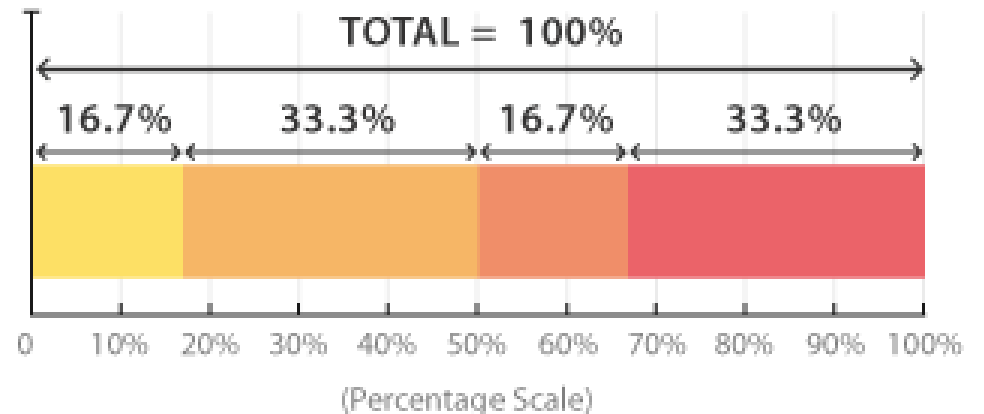
Why might a stacked bar chart not be so good?

- Can imply an order where there is none
- May be hard to compare disparate bars

There is still a lot we don't know about how humans perceive pie charts, area, and angle

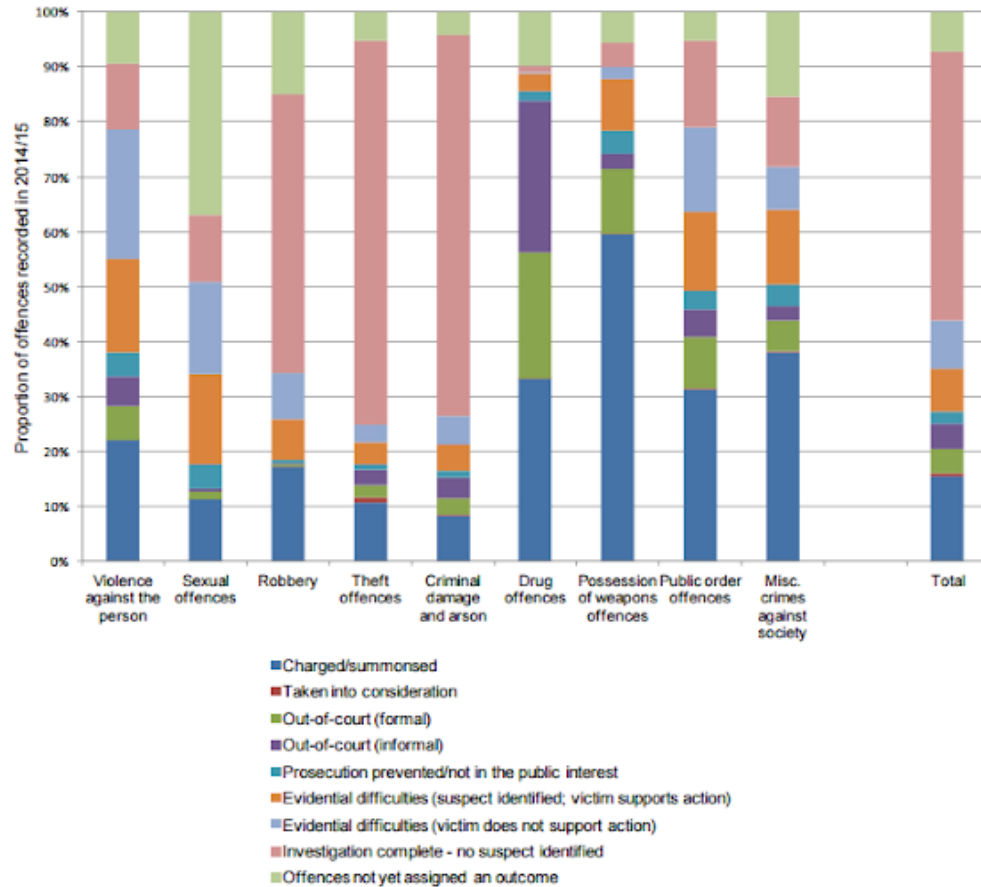


100%

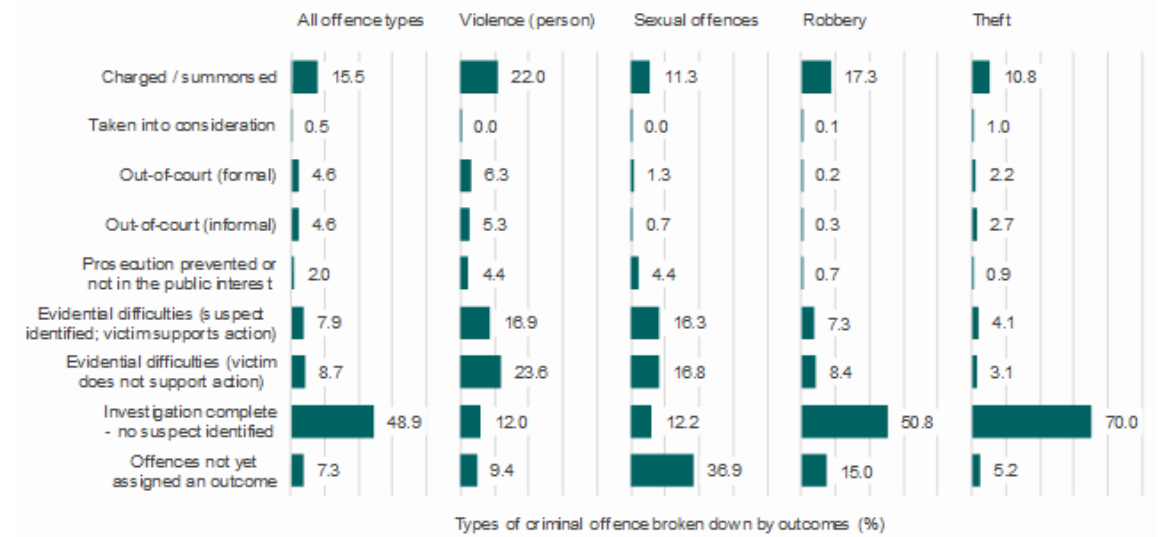


# Stacked Bar Charts vs. Small Multiples

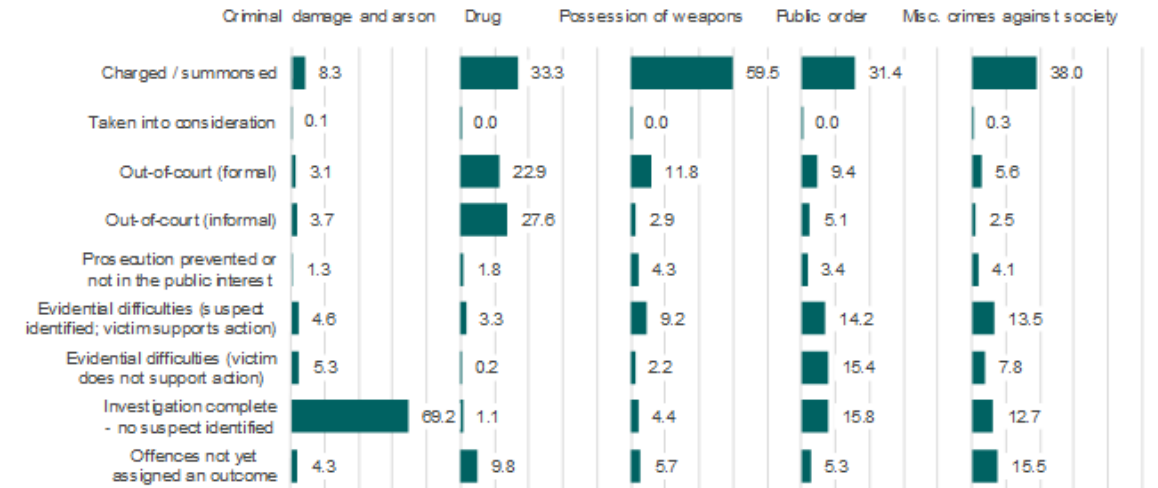
Figure 2.1: Outcomes assigned to offences recorded in 2014/15, by outcome group and offence group



Source: Home Office Data Hub and voluntary spreadsheet return  
 1. Based on 38 forces that supplied data as referenced in Table 2.1.  
 2. The numbers behind this chart are in Table 2.3



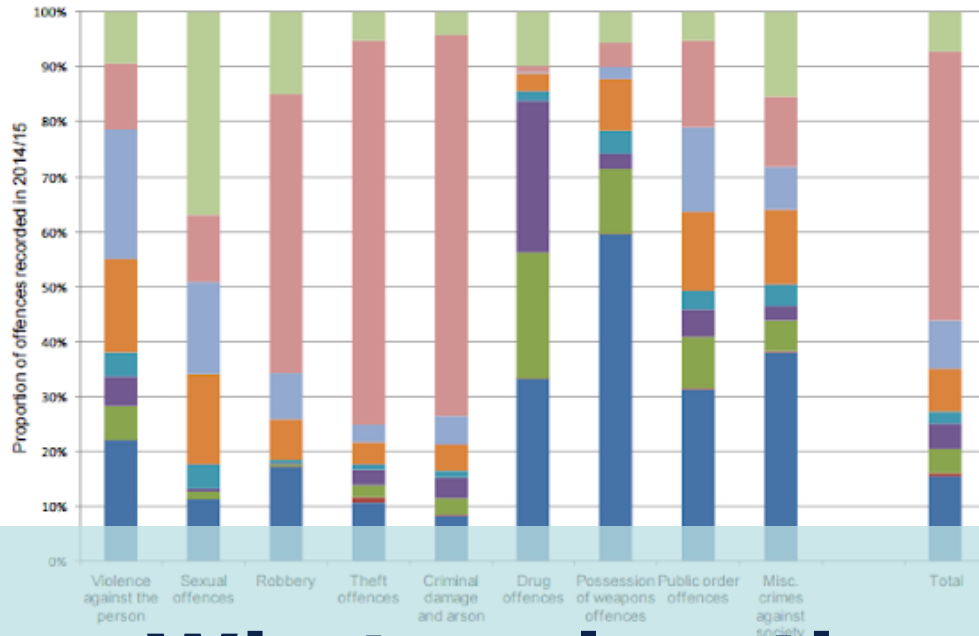
Types of criminal offence broken down by outcomes (%)



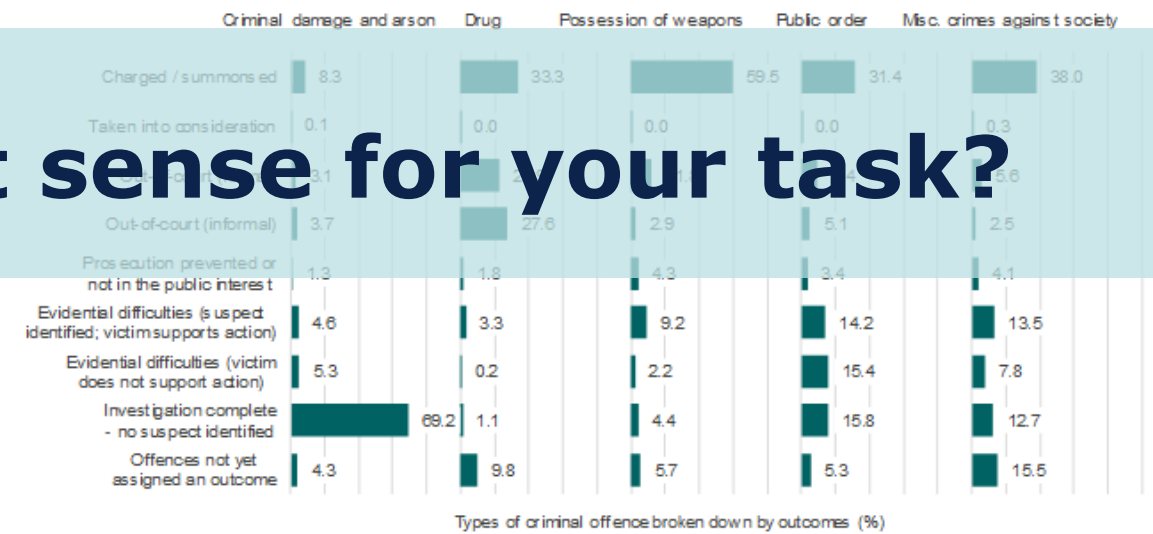
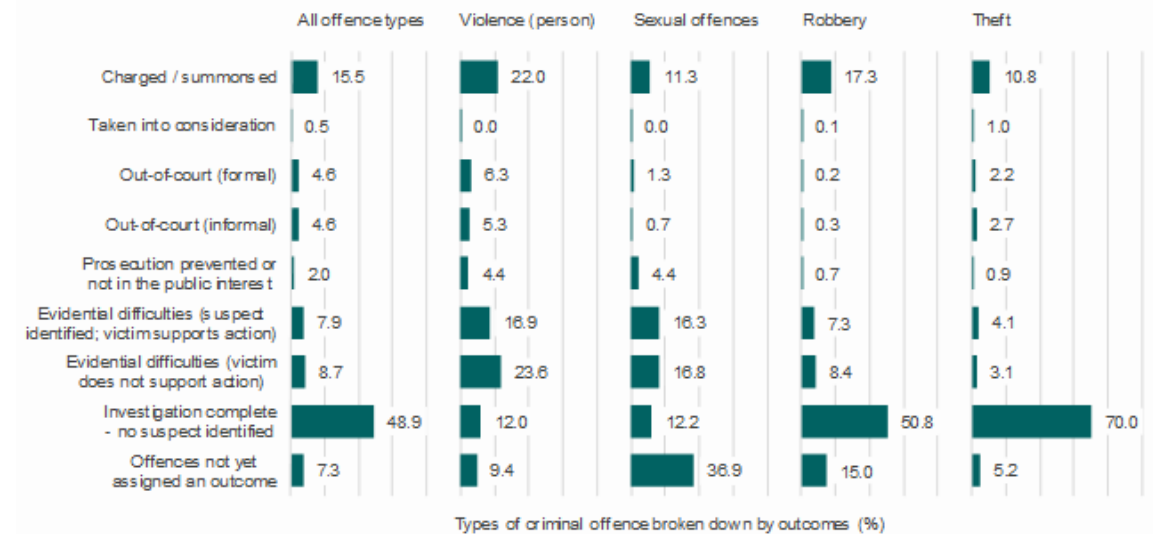
Types of criminal offence broken down by outcomes (%)

# Stacked Bar Charts vs. Small Multiples

Figure 2.1: Outcomes assigned to offences recorded in 2014/15, by outcome group and offence group



## What makes the most sense for your task?



Source: Home Office Data Hub and voluntary spreadsheet return  
 1. Based on 38 forces that supplied data as referenced in Table 2.1.  
 2. The numbers behind this chart are in Table 2.3

# Two-Way Table

Also called contingency table or cross tabulation table...

## Frequency

	student smokes	student does not smoke	total
2 parents smoke	400	1380	1780
1 parent smokes	416	1823	2239
0 parents smoke	188	1168	1356
total	1004	4371	5375

# Two-Way Table

Also called contingency table or cross tabulation table...

**Relative Frequency**

	student smokes	student does not smoke	total
2 parents smoke	7.4%	25.7%	33.1%
1 parent smokes	7.7%	33.9%	41.7%
0 parents smoke	3.5%	21.8%	25.2%
total	18.7%	81.3%	100%

**Row Variable** (indicated by an upward arrow pointing to the row labels)

**Column Variable** (indicated by a leftward arrow pointing to the column headers)

**Marginal Distribution Of Row Variable** (indicated by a red circle around the 'total' column)

**Marginal Distribution Of Column Variable** (indicated by a red circle around the 'total' row)

**Joint Distribution** (indicated by a red circle around the individual data cells)

# Two-Way Table

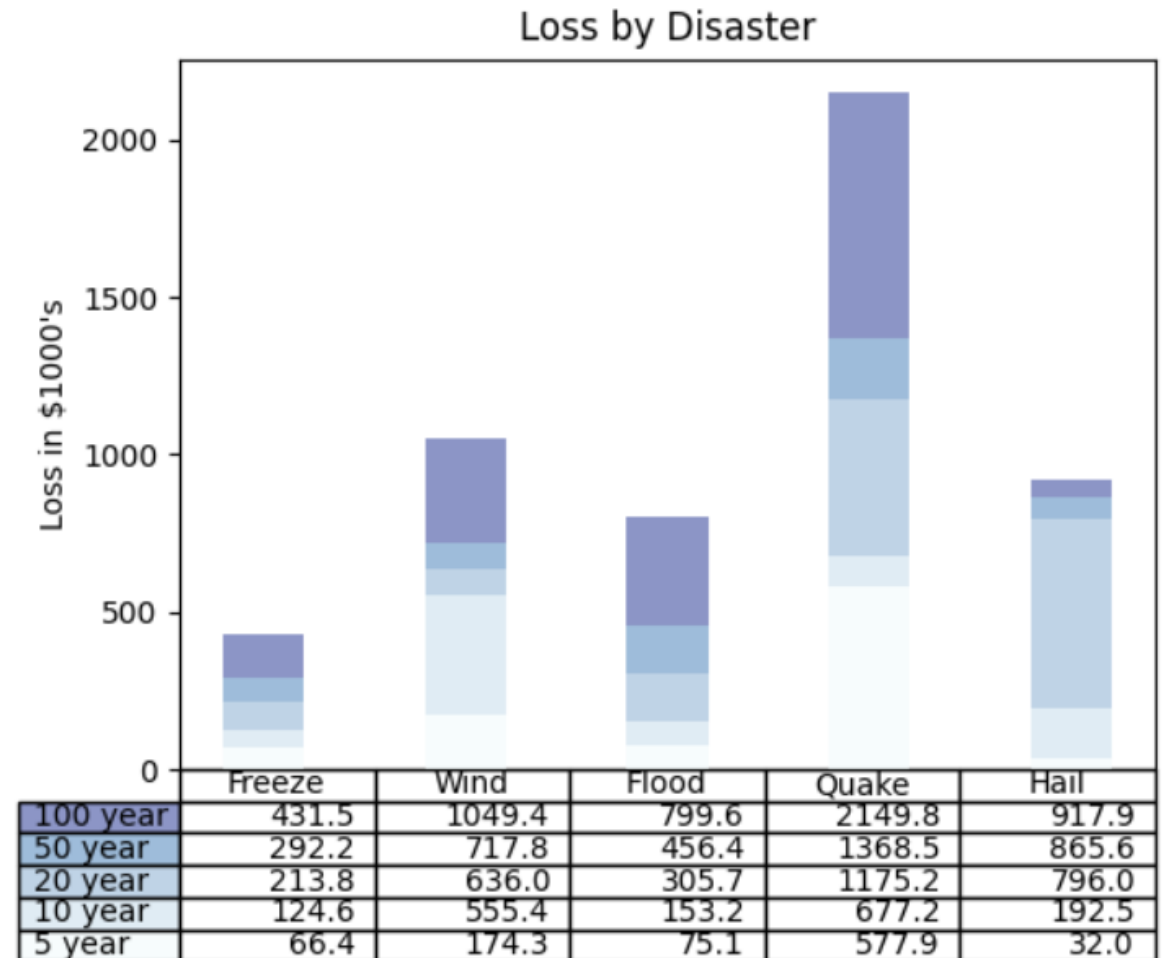
```
data = [[ 66386, 174296, 75131, 577908, 32015],
        [ 58230, 381139, 78045, 99308, 160454],
        [ 89135, 80552, 152558, 497981, 603535],
        [ 78415, 81858, 150656, 193263, 69638],
        [139361, 331509, 343164, 781380, 52269]]

columns = ('Freeze', 'Wind', 'Flood', 'Quake', 'Hail')
rows = ['%d year' % x for x in (100, 50, 20, 10, 5)]
colors = plt.cm.BuPu(np.linspace(0, 0.5, len(rows)))
```

```
the_table = plt.table(cellText=cell_text,
                      rowLabels=rows,
                      rowColours=colors,
                      colLabels=columns,
                      loc='bottom')
```

*Adding stacked bars requires more steps, full code here:*

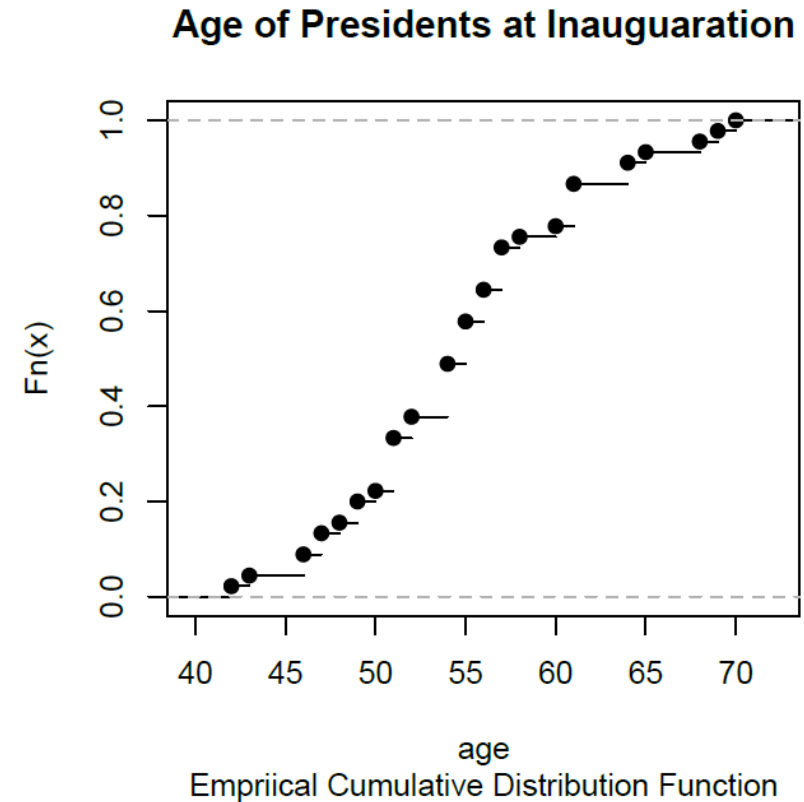
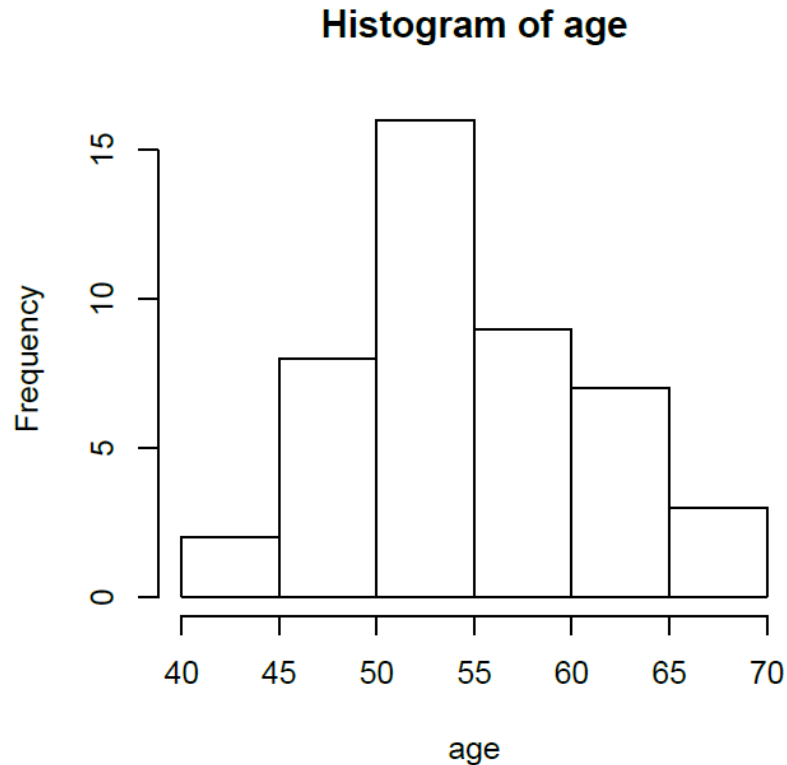
[https://matplotlib.org/stable/gallery/misc/table\\_demo.html](https://matplotlib.org/stable/gallery/misc/table_demo.html)





# Histogram

*Empirical approximation of (quantitative) data generating distribution*



Empirical CDF for each  $x$  gives  $P(X < x)$ ,

$$F_n(x) = \frac{1}{n} \#(\text{observations less than or equal to } x)$$

# Histogram

```
import numpy as np
import matplotlib.pyplot as plt

np.random.seed(19680801)

# example data
mu = 100 # mean of distribution
sigma = 15 # standard deviation of distribution
x = mu + sigma * np.random.randn(437)

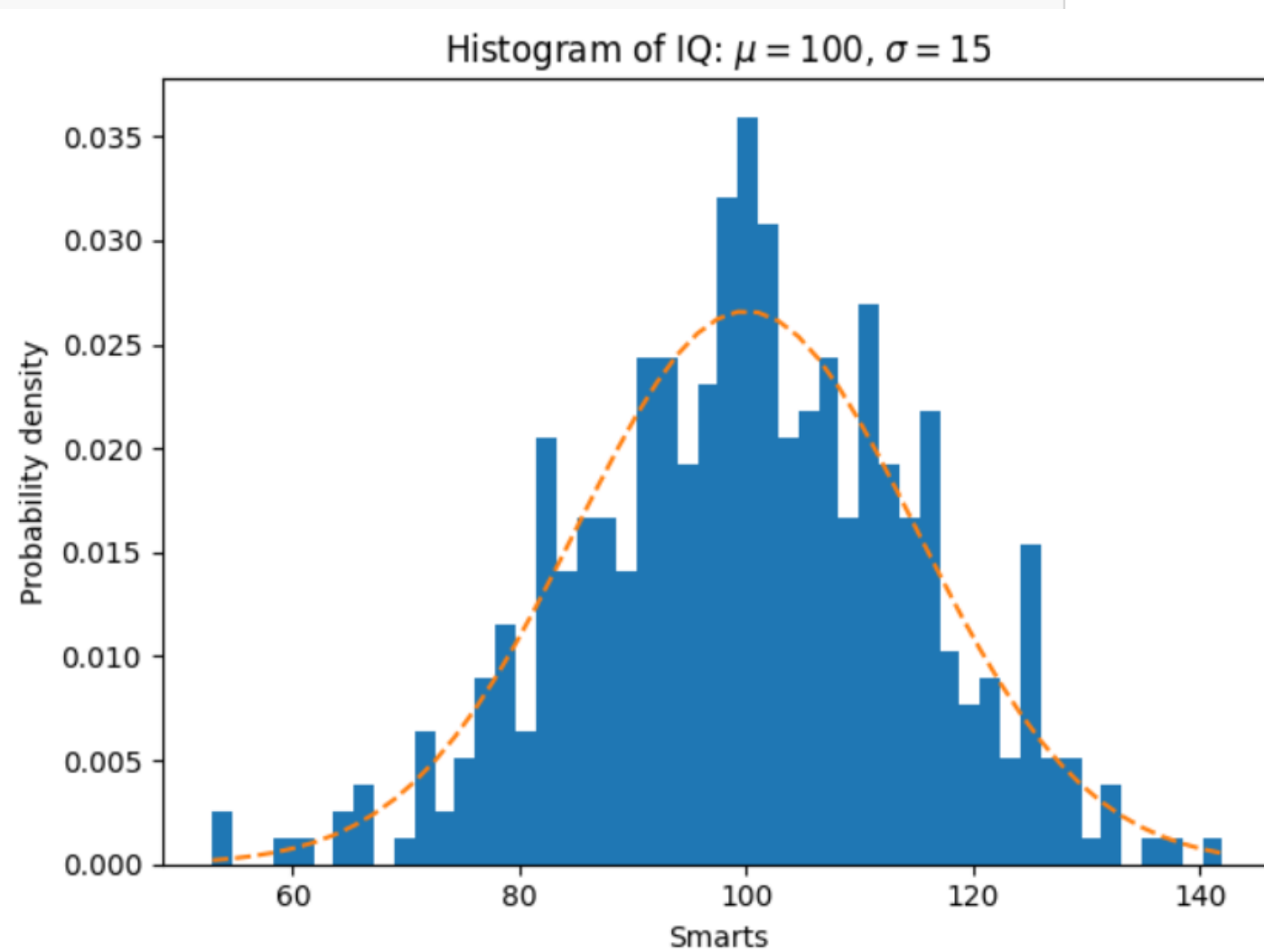
num_bins = 50

fig, ax = plt.subplots()

# the histogram of the data
n, bins, patches = ax.hist(x, num_bins, density=True)

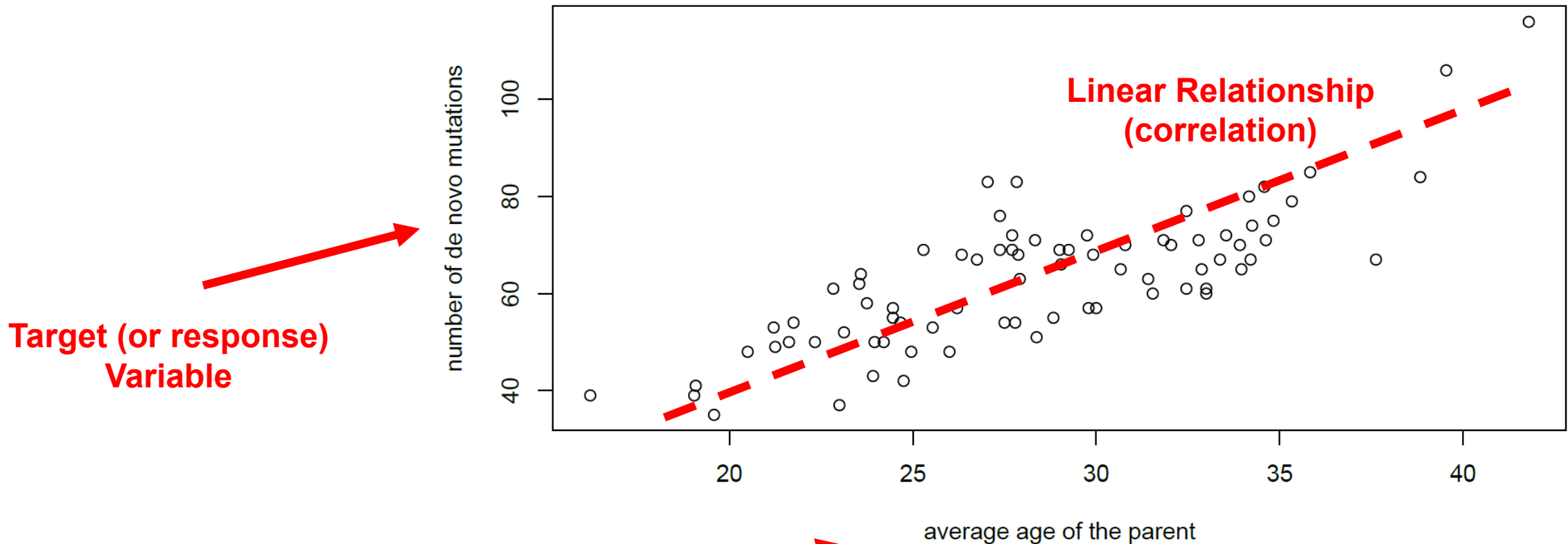
# add a 'best fit' line
y = ((1 / (np.sqrt(2 * np.pi) * sigma)) *
      np.exp(-0.5 * (1 / sigma * (bins - mu)**2)))
ax.plot(bins, y, '--')
ax.set_xlabel('Smarts')
ax.set_ylabel('Probability density')
ax.set_title(r'Histogram of IQ:  $\mu=100$ ,  $\sigma=15$ ')

# Tweak spacing to prevent clipping of ylabel
fig.tight_layout()
plt.show()
```



# Scatterplot

*Compares relationship between two quantitative variables...*



**Target (or response)  
Variable**

**Explanatory Variable**

**Also: predictor, descriptor, input**

Relationship can also be:

- Nonlinear (e.g. “curvy”)
- Clustered or grouped

# Scatterplot

```
import numpy as np
import matplotlib.pyplot as plt

# Fixing random state for reproducibility
np.random.seed(19680801)

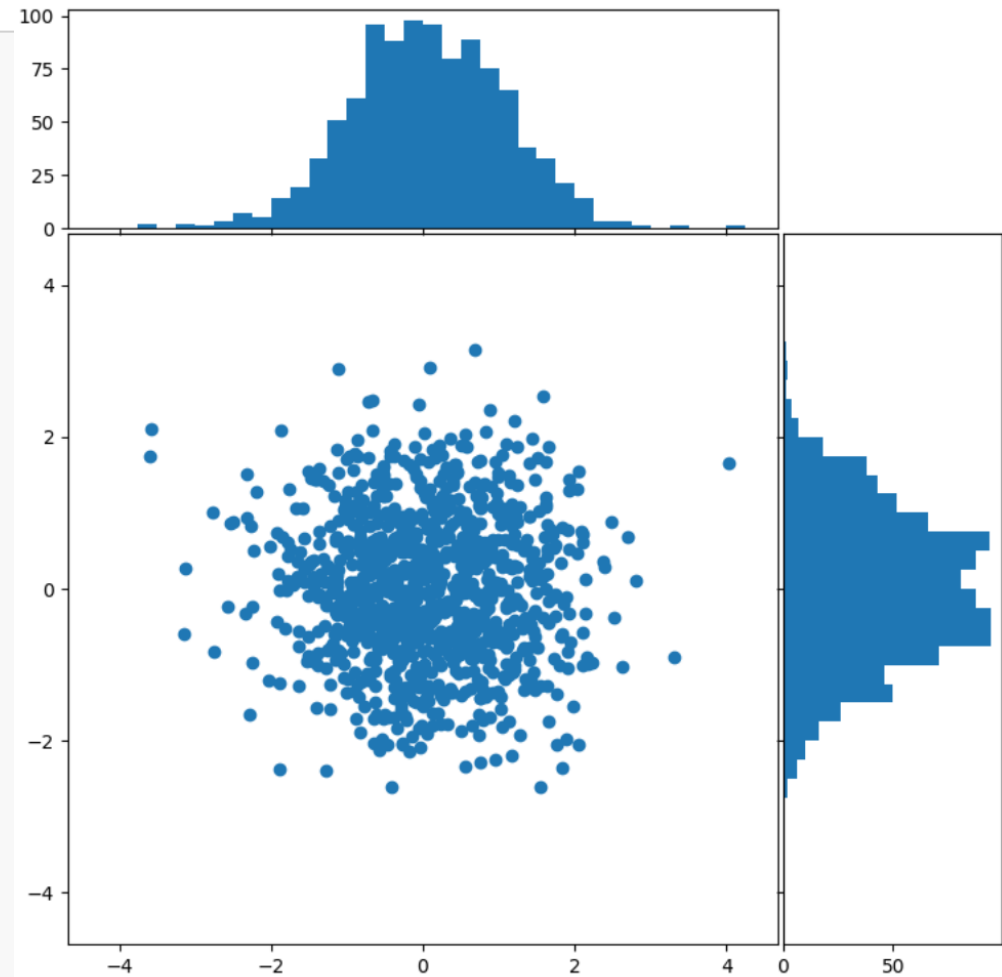
# some random data
x = np.random.randn(1000)
y = np.random.randn(1000)

def scatter_hist(x, y, ax, ax_histx, ax_histy):
    # no labels
    ax_histx.tick_params(axis="x", labelbottom=False)
    ax_histy.tick_params(axis="y", labelleft=False)

    # the scatter plot:
    ax.scatter(x, y)

    # now determine nice limits by hand:
    binwidth = 0.25
    xymax = max(np.max(np.abs(x)), np.max(np.abs(y)))
    lim = (int(xymax/binwidth) + 1) * binwidth

    bins = np.arange(-lim, lim + binwidth, binwidth)
    ax_histx.hist(x, bins=bins)
    ax_histy.hist(y, bins=bins, orientation='horizontal')
```



Full Code:

[https://matplotlib.org/stable/gallery/lines\\_bars\\_and\\_markers/scatter\\_hist.html](https://matplotlib.org/stable/gallery/lines_bars_and_markers/scatter_hist.html)

# Timeseries

```
fig, ax = plt.subplots()
ax.plot('date', 'adj_close', data=data)

# Major ticks every 6 months.
fmt_half_year = mdates.MonthLocator(interval=6)
ax.xaxis.set_major_locator(fmt_half_year)

# Minor ticks every month.
fmt_month = mdates.MonthLocator()
ax.xaxis.set_minor_locator(fmt_month)

# Text in the x axis will be displayed in 'YYYY-mm' format.
ax.xaxis.set_major_formatter(mdates.DateFormatter('%Y-%m'))

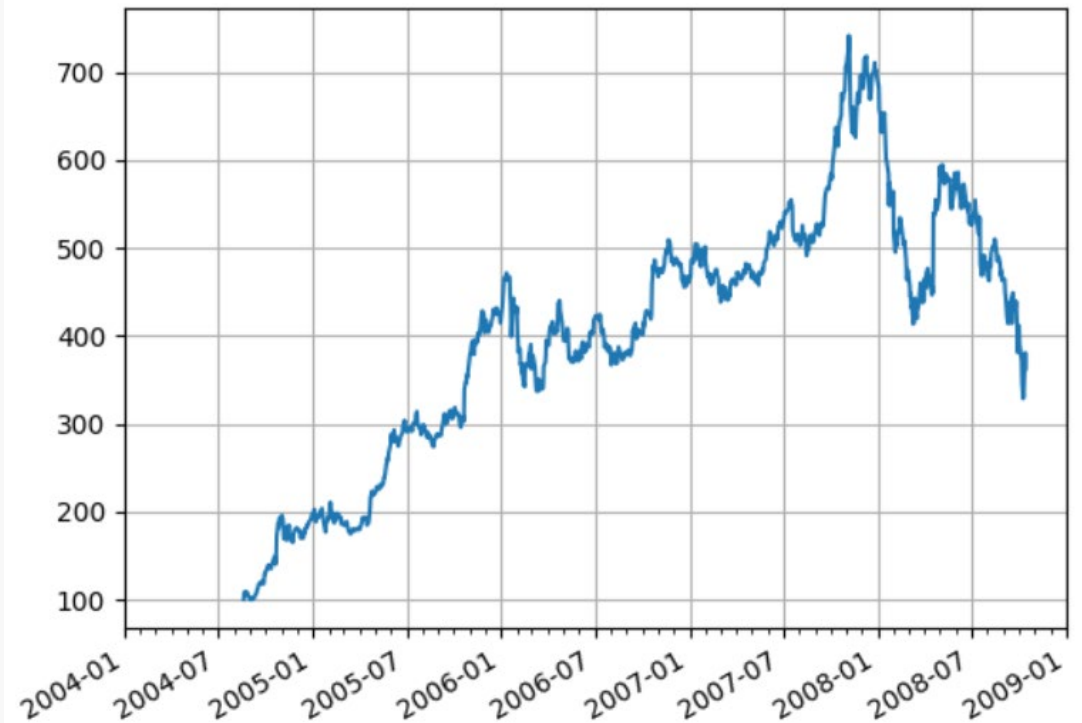
# Round to nearest years.
datemin = np.datetime64(data['date'][0], 'Y')
datemax = np.datetime64(data['date'][-1], 'Y') + np.timedelta64(1, 'Y')
ax.set_xlim(datemin, datemax)

# Format the coords message box, i.e. the numbers displayed as the cursor moves
# across the axes within the interactive GUI.
ax.format_xdata = mdates.DateFormatter('%Y-%m')
ax.format_ydata = lambda x: f'${x:.2f}' # Format the price.
ax.grid(True)

# Rotates and right aligns the x labels, and moves the bottom of the
# axes up to make room for them.
fig.autofmt_xdate()

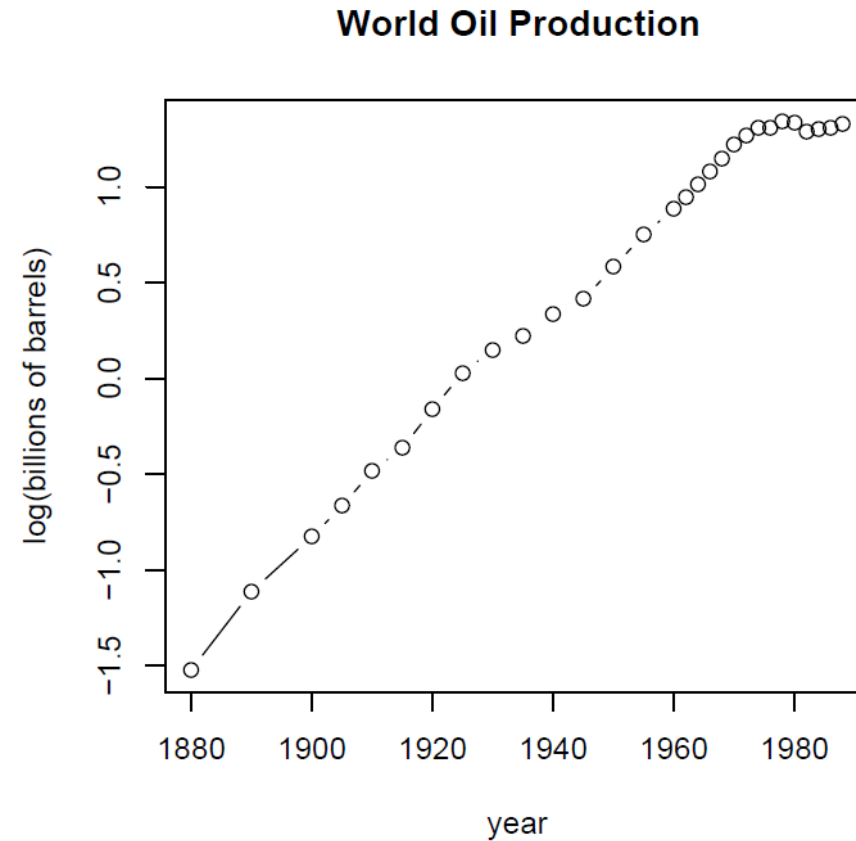
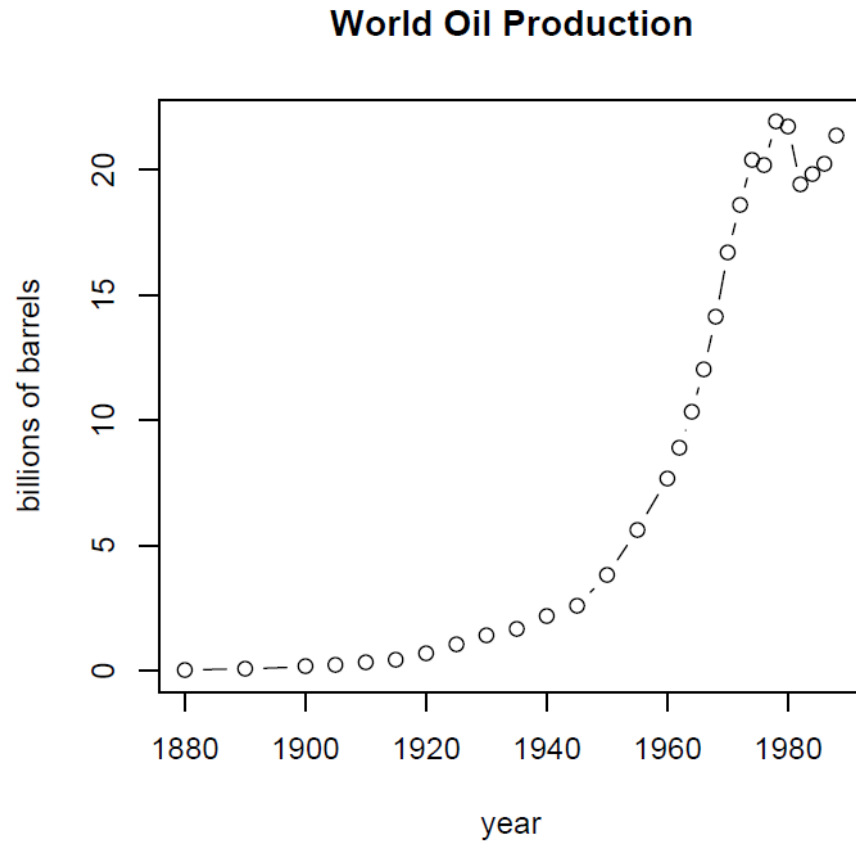
plt.show()
```

*Data follow an explicit ordering*



# Logarithm Scale

*Changing limits and base of y-scale highlights different aspects...*



*...log-scale emphasizes relative changes in smaller quantities*

# Line Plots in Log-Domain

```
# Data for plotting
t = np.arange(0.01, 20.0, 0.01)

# Create figure
fig, ((ax1, ax2), (ax3, ax4)) = plt.subplots(2, 2)

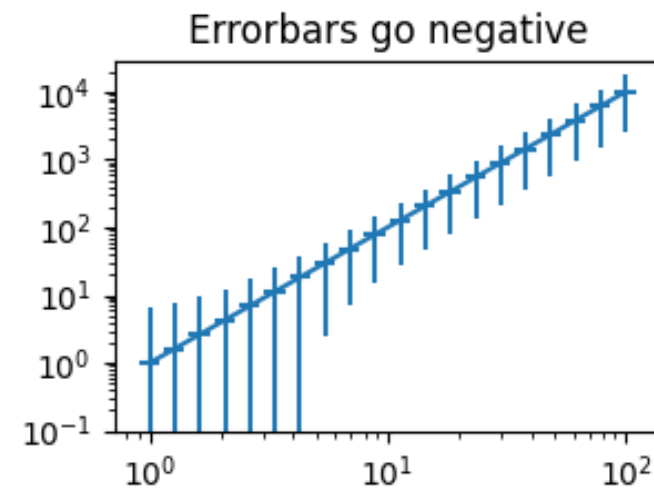
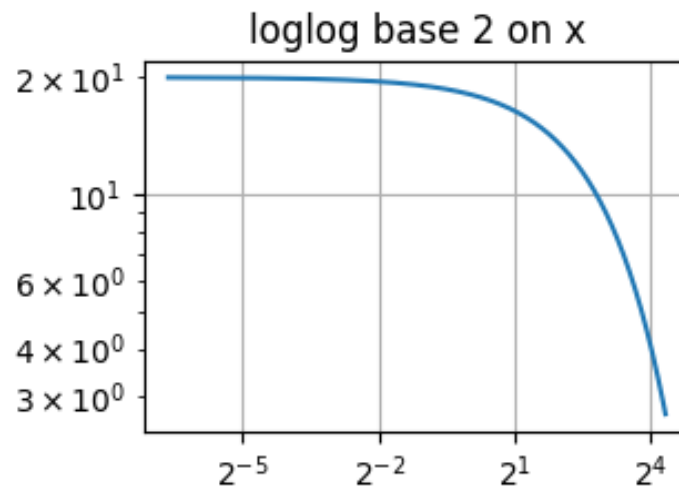
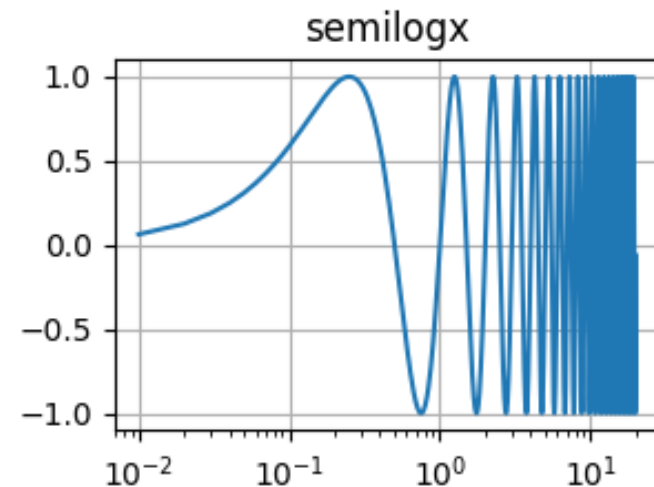
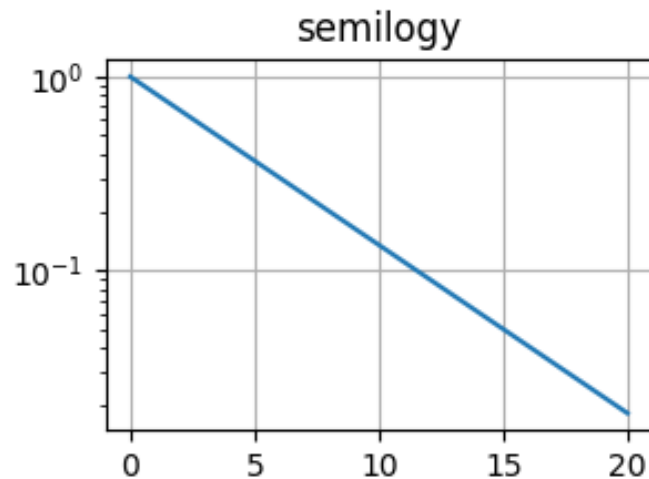
# log y axis
ax1.semilogy(t, np.exp(-t / 5.0))
ax1.set(title='semilogy')
ax1.grid()

# log x axis
ax2.semilogx(t, np.sin(2 * np.pi * t))
ax2.set(title='semilogx')
ax2.grid()

# log x and y axis
ax3.loglog(t, 20 * np.exp(-t / 10.0))
ax3.set_xscale('log', base=2)
ax3.set(title='loglog base 2 on x')
ax3.grid()

# With errorbars: clip non-positive values
# Use new data for plotting
x = 10.0**np.linspace(0.0, 2.0, 20)
y = x**2.0

ax4.set_xscale("log", nonpositive='clip')
ax4.set_yscale("log", nonpositive='clip')
ax4.set(title='Errorbars go negative')
ax4.errorbar(x, y, xerr=0.1 * x, yerr=5.0 + 0.75 * y)
# ylim must be set after errorbar to allow errorbar to autoscale limits
ax4.set_ylim(bottom=0.1)
```



# More Visualization Resources

[datavizcatalogue.com](https://datavizcatalogue.com)



Arc Diagram



Connection Map



Bubble Map



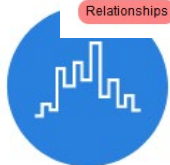
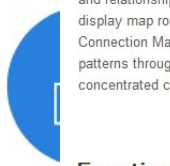
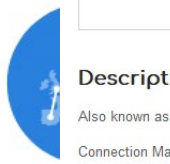
Circle Packing



Error Bars



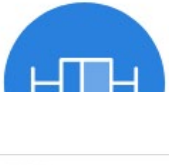
Illustration Diagram



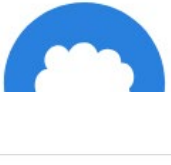
Kagi Chart



Line Graph



Marimekko Chart



Multi-set Bar Chart



Network Diagram

## Description

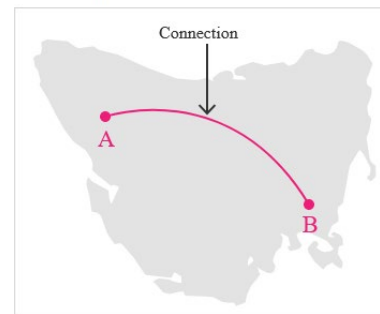
Also known as a *Link Map* or *Ray Map*.

Connection Maps are drawn by connecting points placed on a map by straight or curved lines.

While Connection Maps are great for showing connections and relationships geographically, they can also be used to display map routes through a single chain of links.

Connection Maps can also be useful in revealing spatial patterns through the distribution of connections or by how concentrated connections are on a map.

## Anatomy



## Functions

- Distribution
- Location
- Movement
- Patterns
- Relationships

# matplotlib

[matplotlib.org](https://matplotlib.org)



[scikit-learn.org](https://scikit-learn.org)



# Outline

- Data Visualization
- **Data Summarization**
- Data Collection and Sampling

# Data Summarization

- Raw data are hard to interpret
- Visualizations summarize important aspects of the data
- The *empirical distribution* estimates the distribution on data, but can be hard to interpret
- Summary statistics characterize aspects of the data distribution like:
  - Location / center
  - Scale / spread
  - Skew

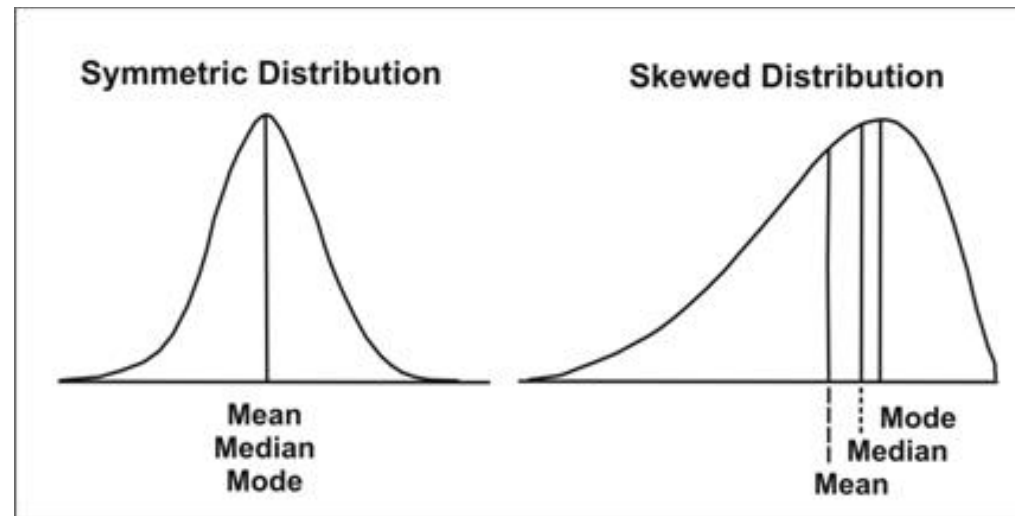
# Measuring Location

Three common measures of the distribution location...

**Mean** Average (expected value) of the data distribution

**Median** Midpoint – 50% of the probability is below and 50% above

**Mode** Value of highest probability (mass or density)



...align with symmetric distributions, but diverge with asymmetry

# Median

For data  $x_1, x_2, \dots, x_N$  sort the data,

$$x_{(1)}, x_{(2)}, \dots, x_{(n)}$$

- Notation  $x_{(i)}$  means the  $i$ -th *lowest* value, e.g.  $x_{(i-1)} \leq x_{(i)} \leq x_{(i+1)}$
- $x_{(1)}, x_{(2)}, \dots, x_{(n)}$  are called *order statistics*

If  $n$  is **odd** then find the middle datapoint,

$$\text{median}(x_1, \dots, x_n) = x_{((n+1)/2)}$$

If  $n$  is **even** then average between both middle datapoints,

$$\text{median}(x_1, \dots, x_n) = \frac{1}{2} (x_{(n/2)} + x_{(n/2+1)})$$

# Median

What is the median of the following data?

1, 2, 3, 4, 5, 6, 8, 9      **4.5**

What is the median of the following data?

1, 2, 3, 4, 5, 6, 8, 100      **4.5**


**Median is *robust* to outliers**

# Sample Mean

Empirical estimate of the true mean of the data distribution,

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i$$

Alternatively, if the value  $x$  occurs  $n(x)$  times in the data then,

$$\bar{x} = \frac{1}{N} \sum_x x n(x) = \sum_x x p(x) \quad \text{where} \quad p(x) = \frac{n(x)}{N}$$


**Empirical  
Distribution**

## Recall

- Law of Large Numbers says  $\bar{x}$  goes to mean  $E[X]$
- Central Limit Theorem says  $\bar{x}$  has Normal distribution

# Sample Mean

**Example 2.1.** For the data set  $\{1, 2, 2, 2, 3, 3, 4, 4, 4, 5\}$ , we have  $n = 10$  and the sum

$$\begin{aligned} 1 + 2 + 2 + 2 + 3 + 3 + 4 + 4 + 4 + 5 &= 1n(1) + 2n(2) + 3n(3) + 4n(4) + 5n(5) \\ &= 1(1) + 2(3) + 3(2) + 4(3) + 5(1) = 30 \end{aligned}$$

Thus,  $\bar{x} = 30/10 = 3$ .

# Sample Mean

**Example 2.2.** *For the data on the length in microns of wild type Bacillus subtilis data, we have*

length $x$	frequency $n(x)$	proportion $p(x)$	product $xp(x)$
1.5	18	0.090	0.135
2.0	71	0.355	0.710
2.5	48	0.240	0.600
3.0	37	0.185	0.555
3.5	16	0.080	0.280
4.0	6	0.030	0.120
4.5	4	0.020	0.090
sum	200	1	2.490

*So the sample mean  $\bar{x} = 2.49$ .*



# Sample Mean

For any real-valued function  $h(x)$  we can compute the mean as,

$$\overline{h(x)} = \frac{1}{N} \sum_{i=1}^N h(x_i)$$

Note  $\overline{h(x)} \neq h(\bar{x})$  in general.

**Example** Compute the average of the square of values,

$$\{ 1, 2, 3, 4, 5, 5, 6 \}$$

$$\overline{x^2} = \frac{1}{7} (1 + 2^2 + 3^2 + 4^2 + 2(5^2) + 6^2) \approx 16.57$$

# Weighted Mean

In some cases we may weight data differently,

$$\sum_{i=1}^N w_i x_i \quad \text{where} \quad \sum_{i=1}^N w_i = 1 \quad 0 \leq w_i \text{ for } i = 1, \dots, N$$

For example, grades in this class:

$$\text{Grade} = 0.2 \cdot x_{\text{midterm}} + 0.2 \cdot x_{\text{final}} + 0.6 \cdot x_{\text{homework}}$$

## Grading Breakdown

- Homework: 60%
- Midterm: 20%
- Final: 20%

# Measuring Spread

We have seen estimates of spread via the sample variance,

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \bar{x})^2$$

**Biased**

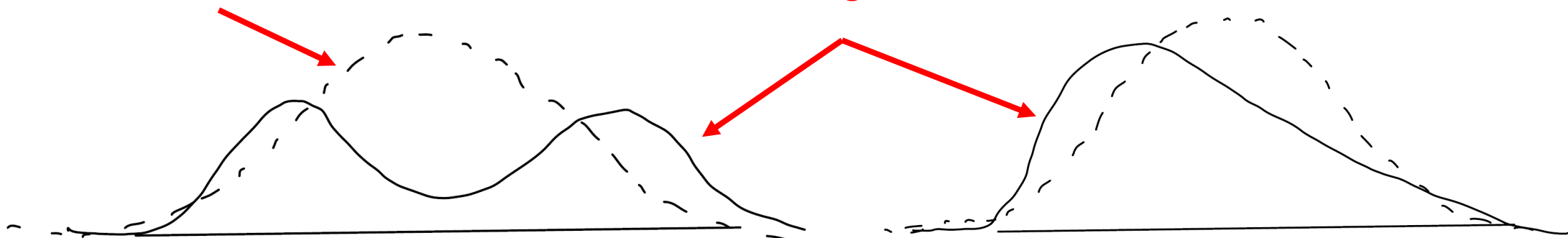
$$s^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2$$

**Unbiased**

Misleading estimate of spread for multimodal / skew distributions

**Normal with same variance**

**Target**



# Measuring Spread

**Quartile** divide data into 4 equally-sized bins,

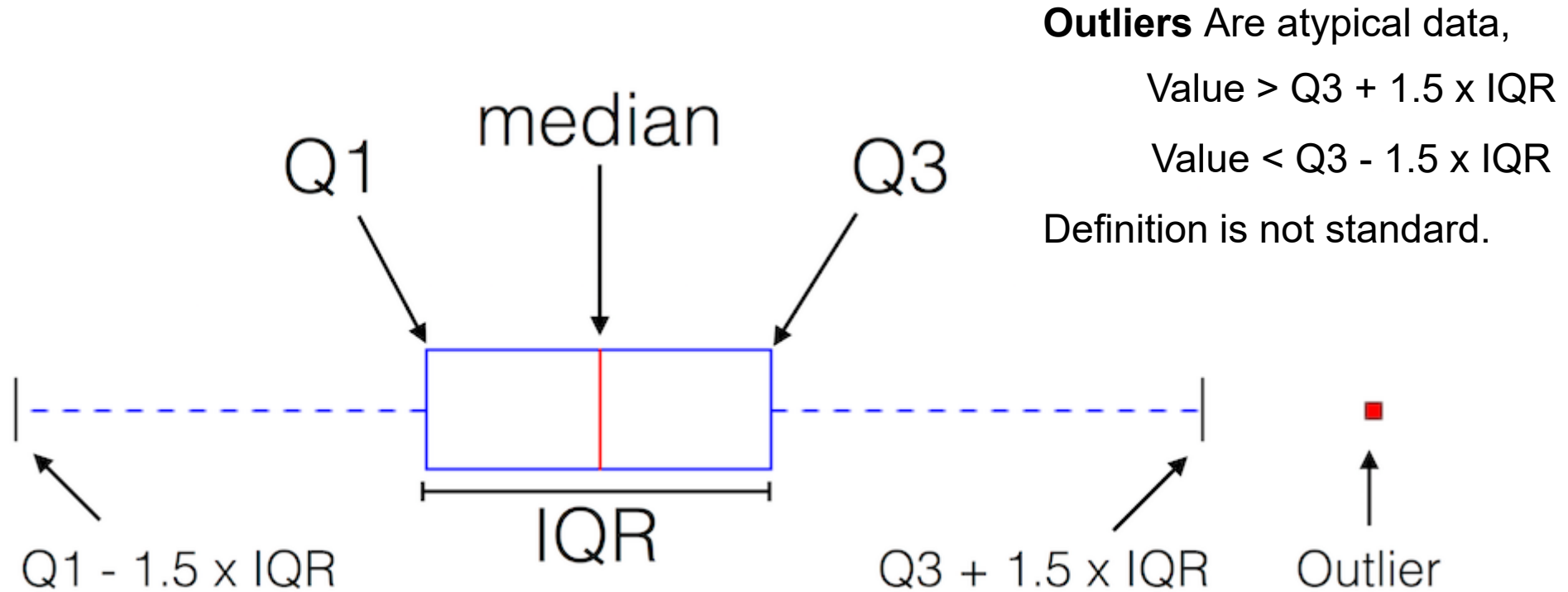
- **1<sup>st</sup> Quartile** : Lowest 25% of data
- **2<sup>nd</sup> Quartile** : Median (lowest 50% of data)
- **3<sup>rd</sup> Quartile** : 75% of data is below 3<sup>rd</sup> quartile
- **4<sup>th</sup> Quartile** : All the data... not useful

Compute using `np.quantile()` :

```
x = np.random.rand(10) * 100
q = np.quantile(x, (0.25, 0.5, 0.75))
np.set_printoptions(precision=1)
print( "X: " , x )
print( "Q: " , q )
```

```
X:  [90.7 73.9 31.7  2.8 56.3 95.7 15.6 75.8  4.1 19.5]
Q:  [16.6 44.  75.3]
```

# Box Plot



**Interquartile-Range (IQR)** Measures interval containing 50% of data

$$IQR = Q3 - Q1$$

Region of *typical* data

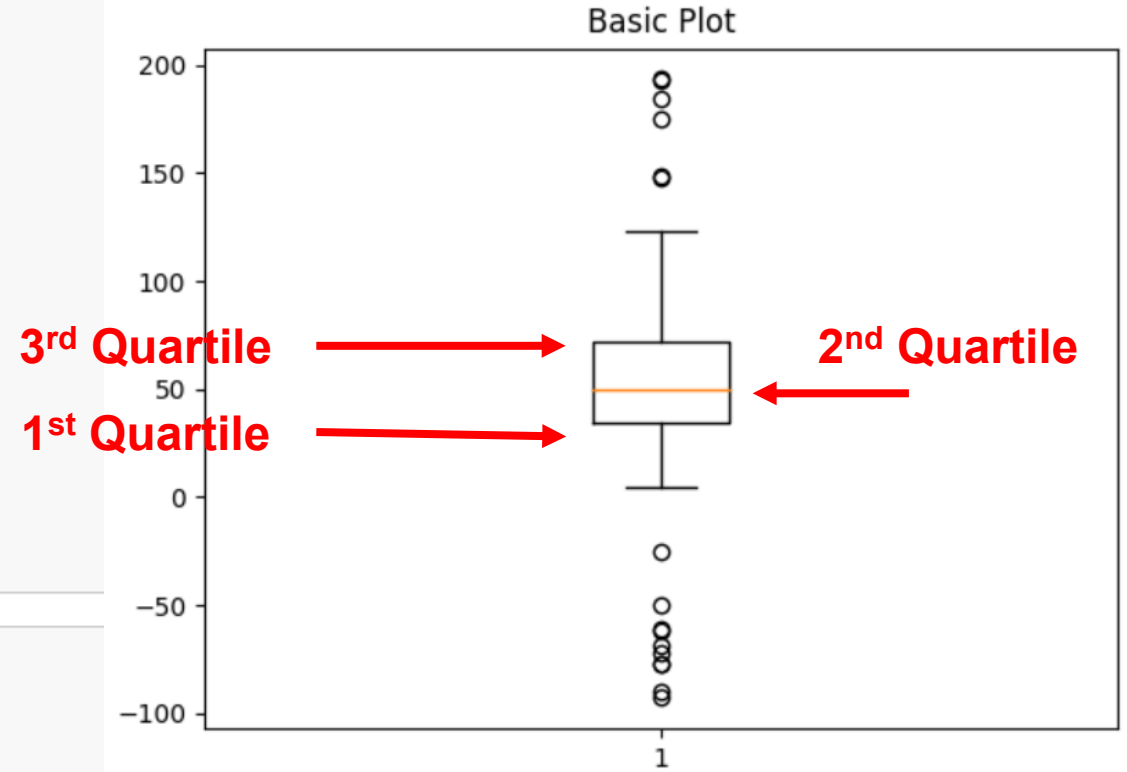
# Box Plot

```
import numpy as np
import matplotlib.pyplot as plt

# Fixing random state for reproducibility
np.random.seed(19680801)

# fake up some data
spread = np.random.rand(50) * 100
center = np.ones(25) * 50
flier_high = np.random.rand(10) * 100 + 100
flier_low = np.random.rand(10) * -100
data = np.concatenate((spread, center, flier_high, flier_low))
```

```
fig1, ax1 = plt.subplots()
ax1.set_title('Basic Plot')
ax1.boxplot(data)
```

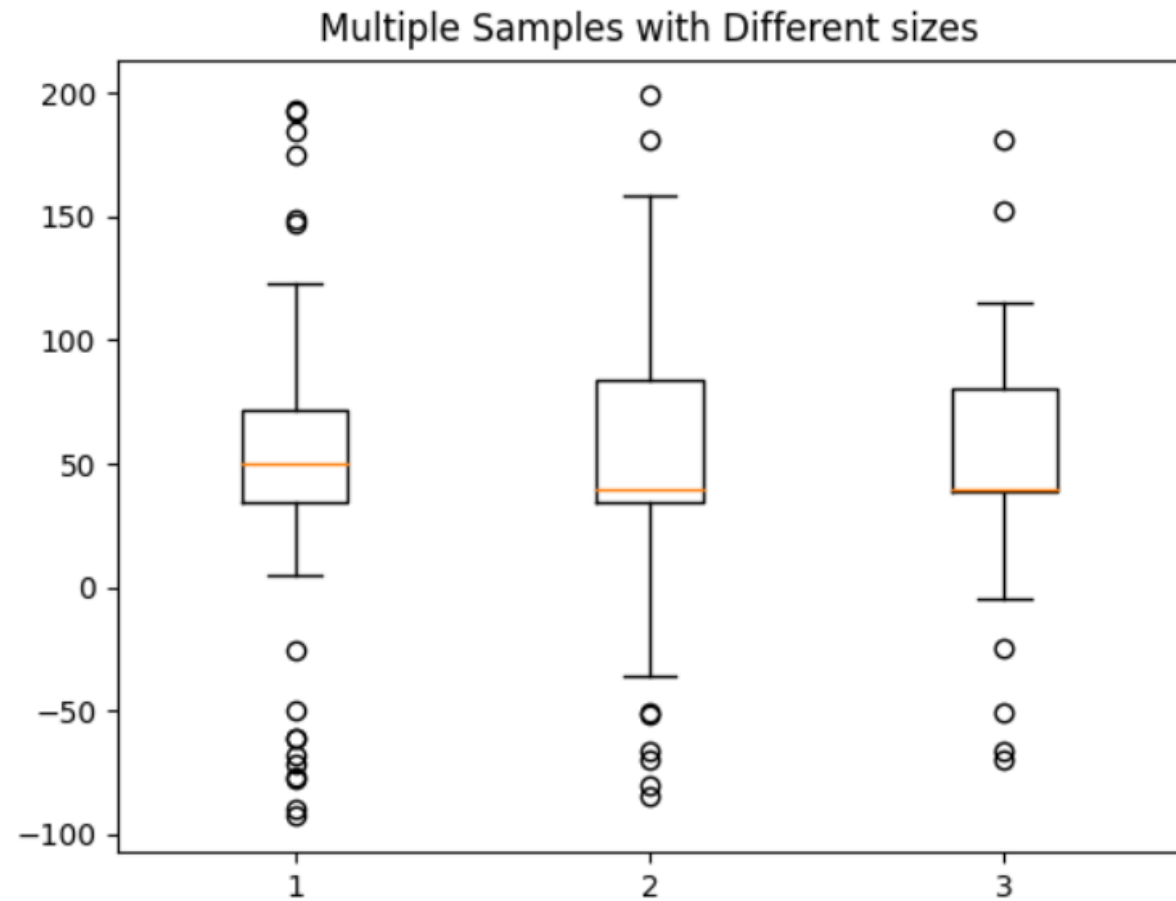


# Box Plot

```
spread = np.random.rand(50) * 100
center = np.ones(25) * 40
flier_high = np.random.rand(10) * 100 + 100
flier_low = np.random.rand(10) * -100
d2 = np.concatenate((spread, center, flier_high, flier_low))
```

```
data = [data, d2, d2[::2]]
fig7, ax7 = plt.subplots()
ax7.set_title('Multiple Samples with Different sizes')
ax7.boxplot(data)

plt.show()
```



# Measuring Spread

For nonnegative numbers we can look at the **coefficient of variation**,

$$CV = \frac{s}{\bar{x}}$$

where  $s^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2$  and  $\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i$

- It is a *pure number* – it has no units
- It represents spread relative to the mean

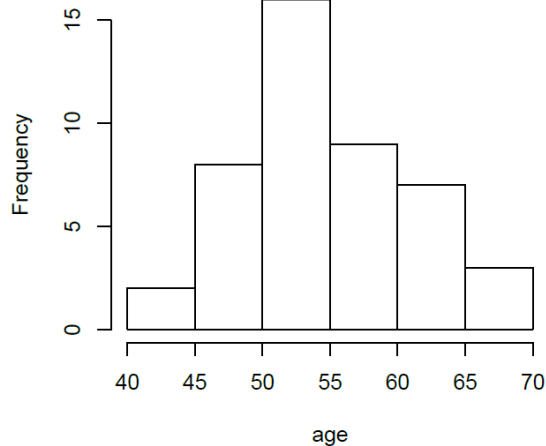
**Question** Why would we want to compute this?



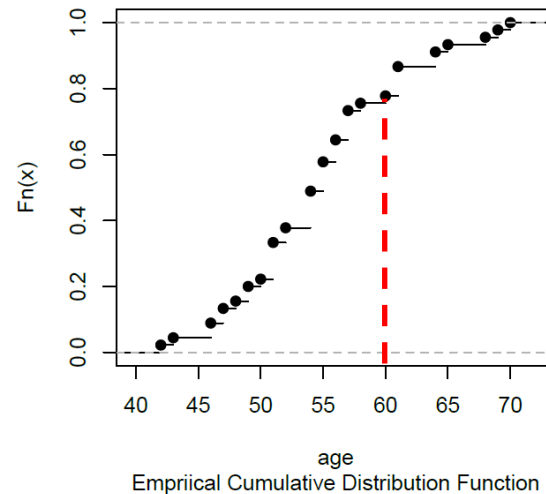
# Quantile / Percentile

**Question** Is 60yrs old for a US president? Why or why not?

Histogram of age



Age of Presidents at Inauguration



Empirical CDF for each  $x$  gives  $P(X < x)$ ,

$$F_n(x) = \frac{1}{n} \#(\text{observations less than or equal to } x)$$

Compute probability of being  $< 60$ ,

$$F_n(60) \approx 0.8$$

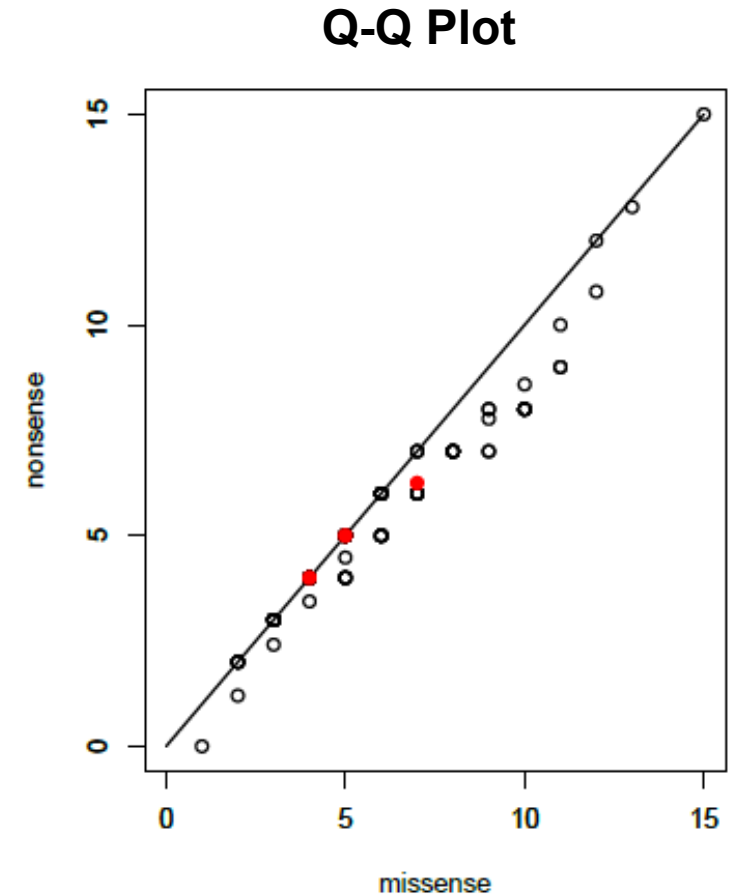
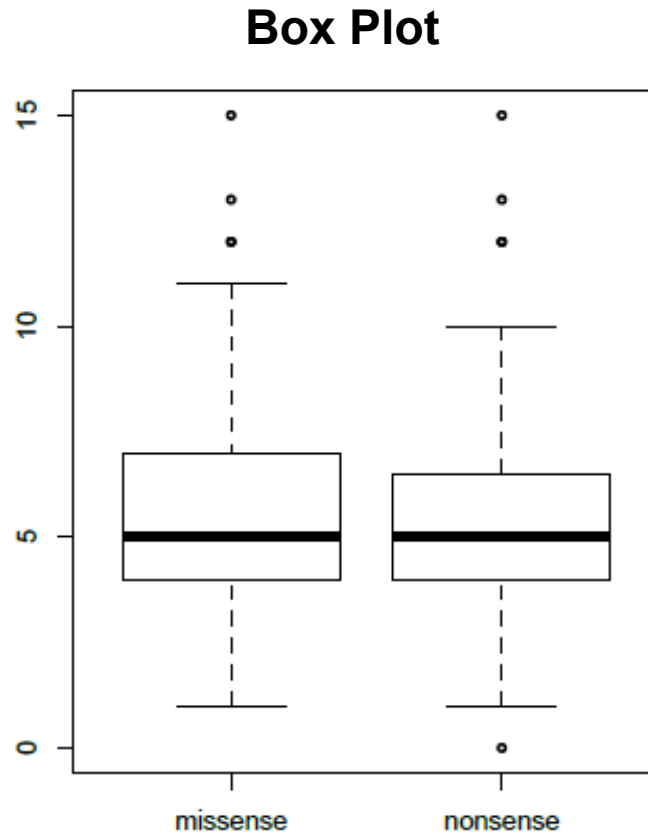
0.8 Quantile or 80<sup>th</sup> Percentile  $\rightarrow$  About 80% of presidents younger than 60

# Quantile-Quantile Plot

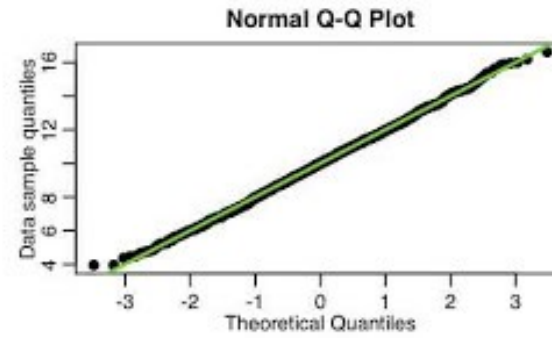
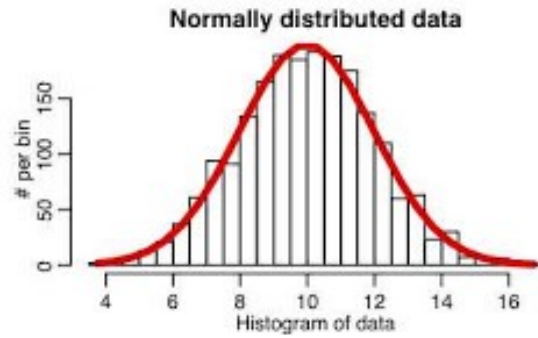
Plot quantiles of two variables against each other...

Variables with similar distributions will fall along a 45-degree (slope 1) line

In Q-Q plot correlation = distribution similarity

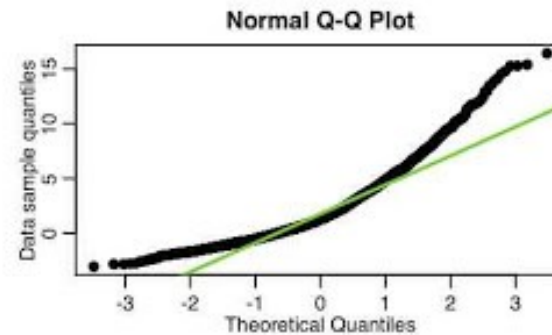
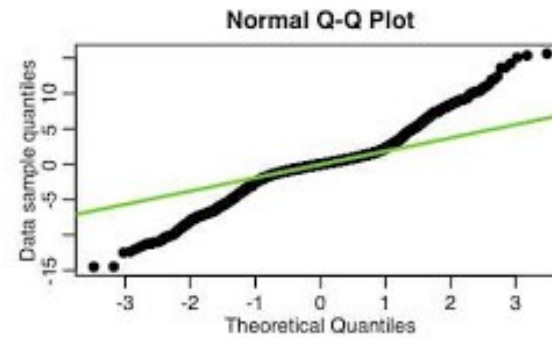


# Interpreting Q-Q Plots

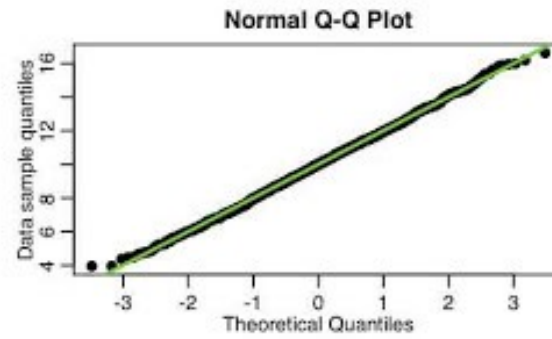
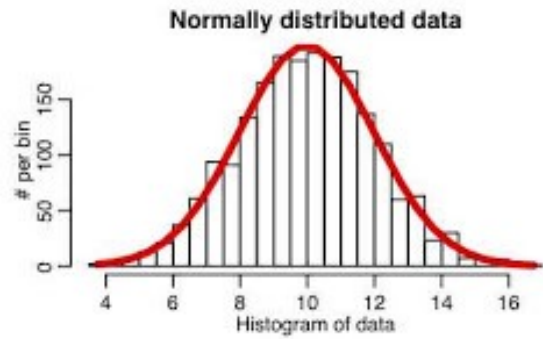


Plot against theoretical quantiles to check model fit

**Good Fit**

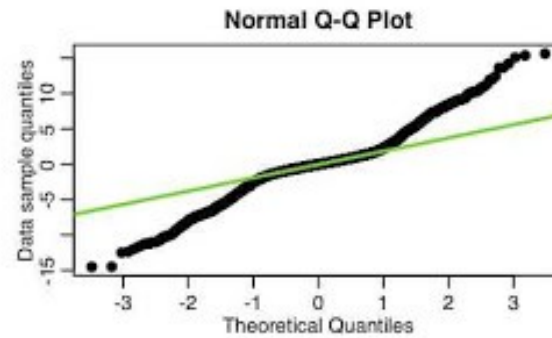
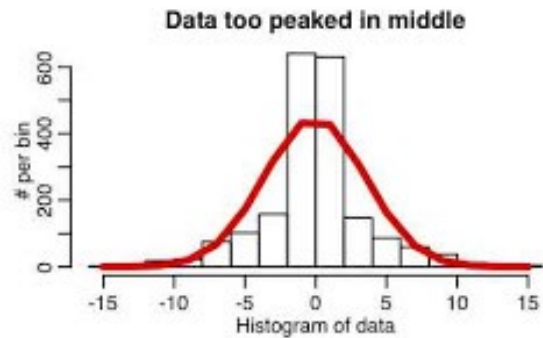


# Interpreting Q-Q Plots

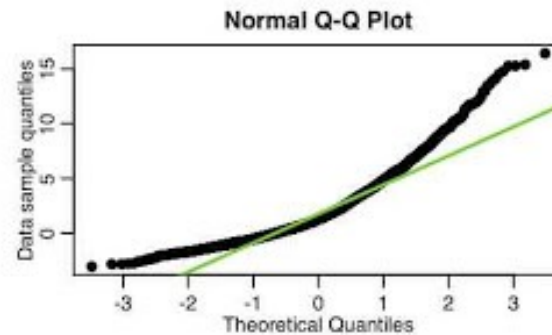


Plot against theoretical quantiles to check model fit

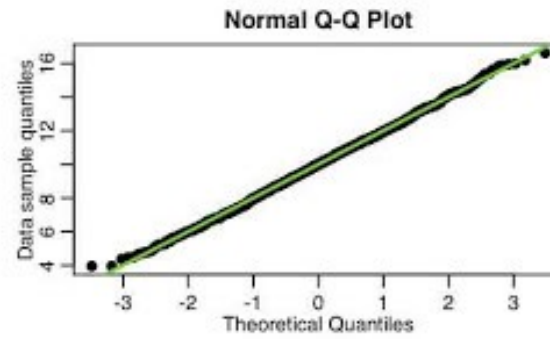
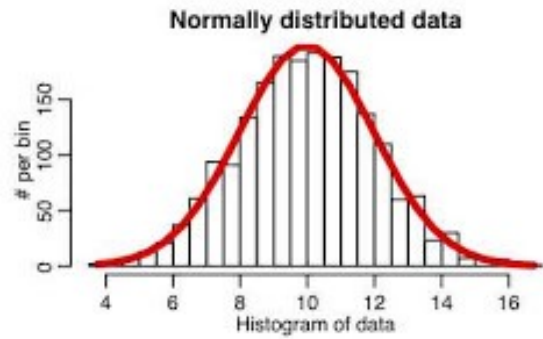
**Good Fit**



**Fat Tails**

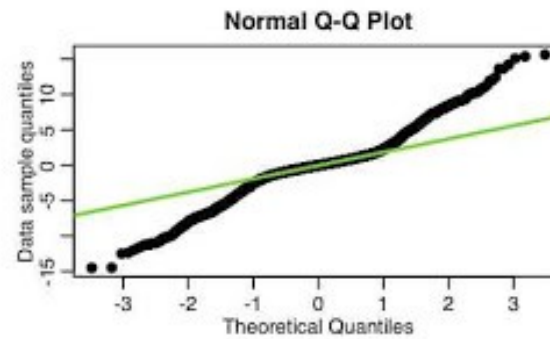
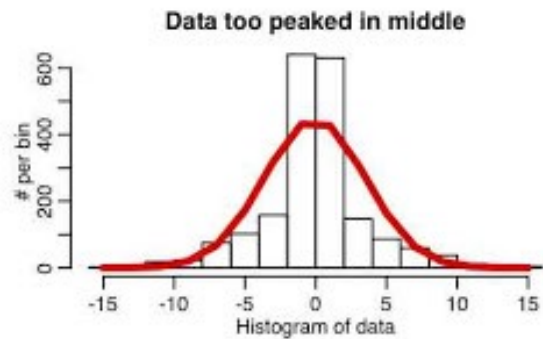


# Interpreting Q-Q Plots

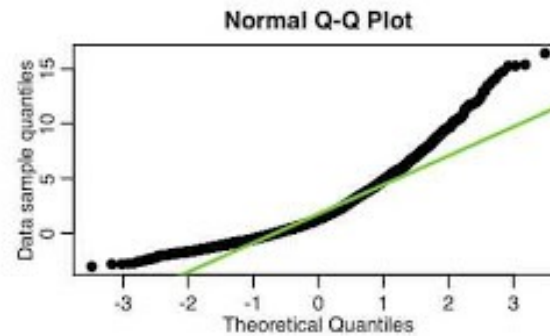
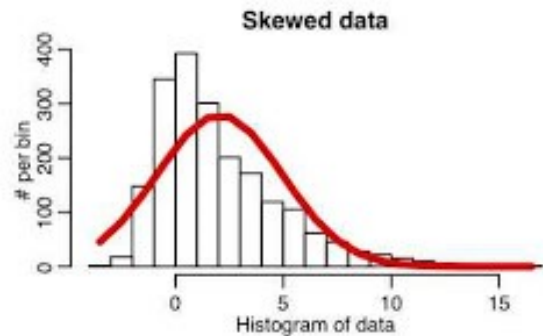


Plot against theoretical quantiles to check model fit

**Good Fit**



**Fat Tails**



**Positive Skew**

# SciPy

*Python-based ecosystem for math, science  
and engineering.*



As usual, install with Anaconda:

```
> conda install scipy
```

Or with PyPI:

```
> pip install scipy
```

SciPy includes some libraries that you are already familiar with:



# SciPy Statistics

*SciPy is a large library, so we import it in bits and pieces...*



```
>>> from scipy import stats
```

In some cases, you will import only the functions that you need:

```
>>> from scipy.stats import norm
```

```
>>> norm.mean(), norm.std(), norm.var()
(0.0, 1.0, 1.0)
>>> norm.stats(moments="mv")
(array(0.0), array(1.0))
```

# SciPy Statistics

To compute summary stats (e.g. **mode**):



```
>>> a = np.array([[6, 8, 3, 0],
...               [3, 2, 1, 7],
...               [8, 1, 8, 4],
...               [5, 3, 0, 5],
...               [4, 7, 5, 9]])
>>> from scipy import stats
>>> stats.mode(a)
ModeResult(mode=array([[3, 1, 0, 0]]), count=array([[1, 1, 1, 1]]))
```

Compute the mode of the whole array set `axis=None`:

```
>>> stats.mode(a, axis=None)
ModeResult(mode=array([3]), count=array([3]))
```





## *Other useful summary statistics:*

**moment**(a[, moment, axis, nan\_policy])

Calculate the nth moment about the mean for a sample.

**trim\_mean**(a, proportiontocut[, axis])

Return mean of array after trimming distribution from both tails.

**iqr**(x[, axis, rng, scale, nan\_policy, ...])

Compute the interquartile range of the data along the specified axis.

**bootstrap**(data, statistic, \*[, vectorized, ...])

Compute a two-sided bootstrap confidence interval of a statistic.

**variation**(a[, axis, nan\_policy, ddof])

Compute the coefficient of variation.

...

# Anscomb's Quartet : The Data

Four distinct datasets of paired variables X and Y...

I		II		III		IV	
x	y	x	y	x	y	x	y
10.0	8.04	10.0	9.14	10.0	7.46	8.0	6.58
8.0	6.95	8.0	8.14	8.0	6.77	8.0	5.76
13.0	7.58	13.0	8.74	13.0	12.74	8.0	7.71
9.0	8.81	9.0	8.77	9.0	7.11	8.0	8.84
11.0	8.33	11.0	9.26	11.0	7.81	8.0	8.47
14.0	9.96	14.0	8.10	14.0	8.84	8.0	7.04
6.0	7.24	6.0	6.13	6.0	6.08	8.0	5.25
4.0	4.26	4.0	3.10	4.0	5.39	19.0	12.50
12.0	10.84	12.0	9.13	12.0	8.15	8.0	5.56
7.0	4.82	7.0	7.26	7.0	6.42	8.0	7.91
5.0	5.68	5.0	4.74	5.0	5.73	8.0	6.89

# Anscomb's Quartet : Summary Statistics

```
# Import the csv file
df = pd.read_csv("anscombe.csv")

# Convert pandas dataframe into pandas series
list1 = df['x1']
list2 = df['y1']

# Calculating mean for x1
print('%.1f' % statistics.mean(list1))

# Calculating standard deviation for x1
print('%.2f' % statistics.stdev(list1))

# Calculating mean for y1
print('%.1f' % statistics.mean(list2))

# Calculating standard deviation for y1
print('%.2f' % statistics.stdev(list2))

# Calculating pearson correlation
corr, _ = pearsonr(list1, list2)
print('%.3f' % corr)

# Similarly calculate for the other 3 samples

# This code is contributed by Amiya Rout
```

Start by computing summary statistics, e.g. Dataset 1:

**Mean X1: 9.0**

**STDEV X1: 3.32**

**Mean Y1: 7.5**

**STDEV Y1: 2.03**

**Correlation: 0.816**

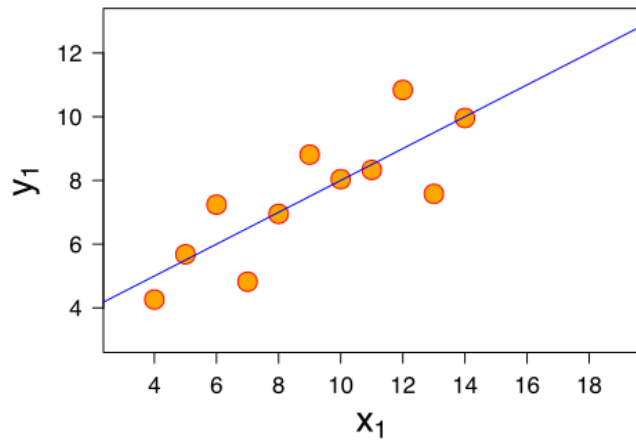
**Actually, all datasets have the same statistics...**

**Question** What can we conclude about these data? Are they the same?

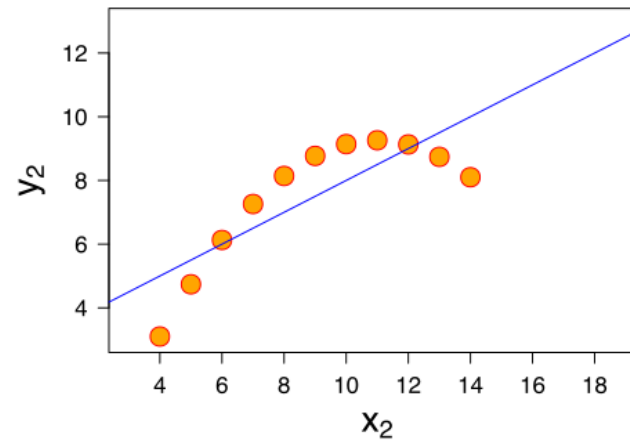
[ Source: <https://www.geeksforgeeks.org/anscombes-quartet/> ]

# Anscomb's Quartet : Visualization

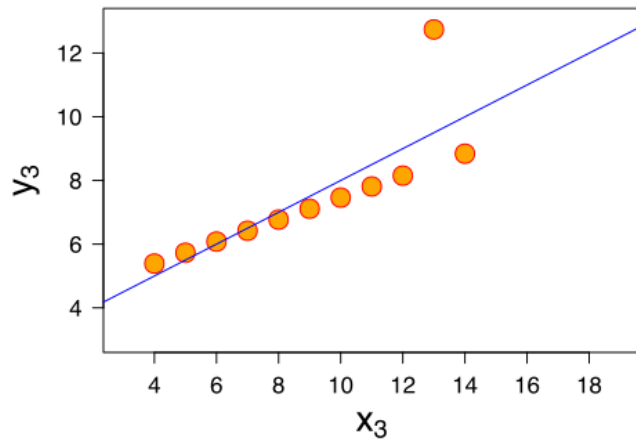
**Dataset 1**



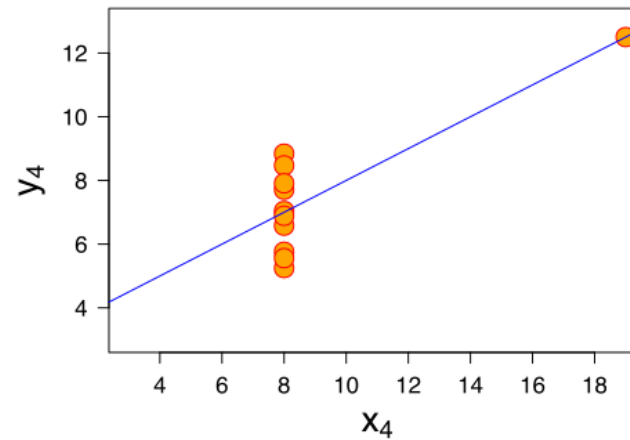
**Dataset 2**



**Dataset 3**



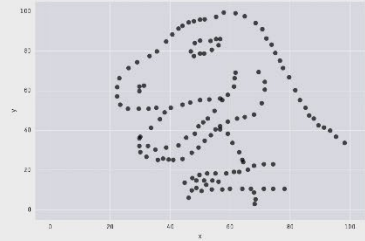
**Dataset 4**



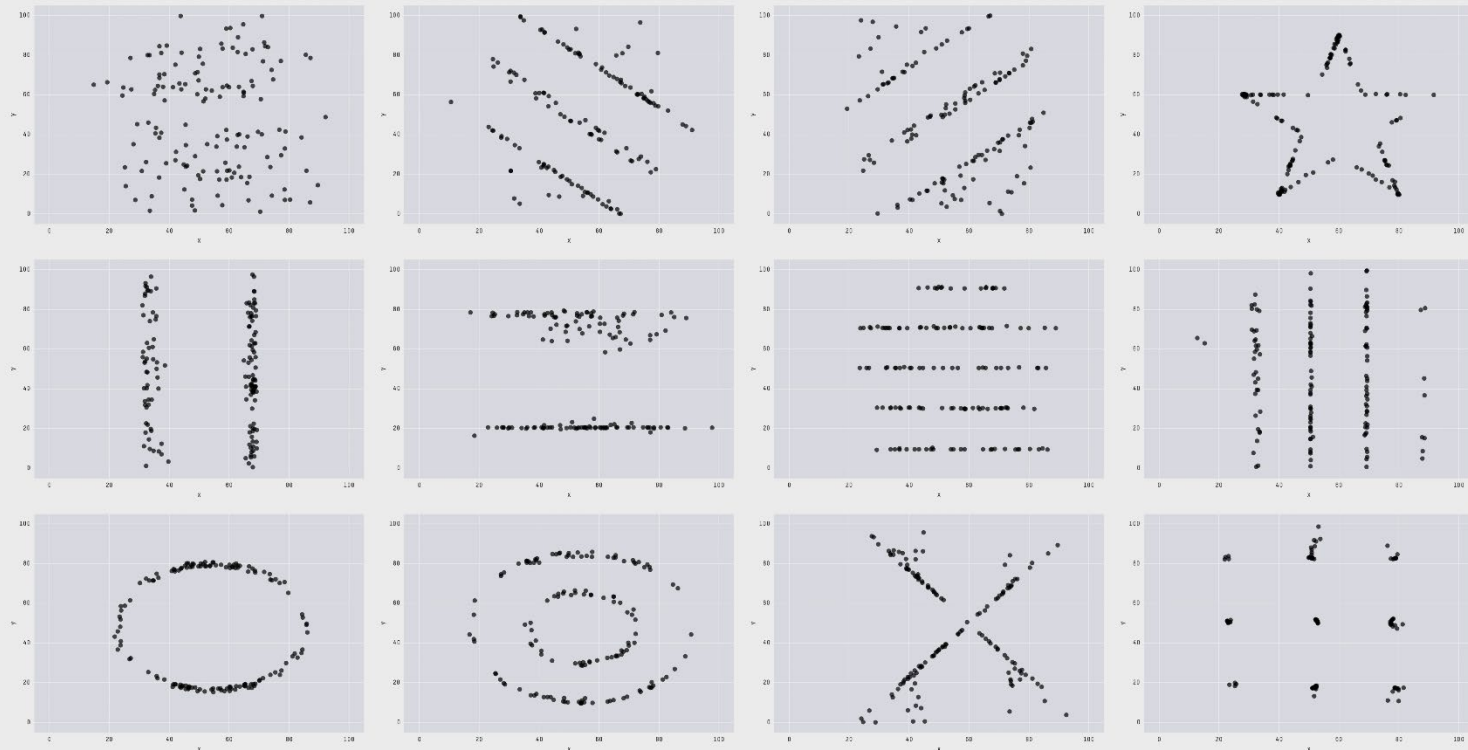
Visualizing data clearly indicates that these are *very different* datasets...

...this highlights the **importance of visualizing data**

# Datasaurus



X Mean: 54.26  
Y Mean: 47.83  
X SD : 16.76  
Y SD : 26.93  
Corr. : -0.06



13 datasets that all have the same summary statistics, but look very different in simple visualizations

Can be very difficult to see differences in high dimensions, however

# Administrative Items

- HW5 Due Tuesday (10/12) @ 11:59pm
- Midterm Out Thursday (10/14)
  - I will announce when it's due next week
- Midterm review next Thursday

# Outline

- Data Visualization
- Data Summarization
- **Data Collection and Sampling**

Much of the content in this section from [Scribbr.com](https://www.scribbr.com) and Shona McCombes

# Motivation

Not understanding how data are collected is one of the top reasons behind bad data science...

How Bad Data Is  
Undermining Big Data  
Analytics

Forbes

**How to be a bad data scientist!**



Pascal Potvin Feb 27, 2018

**8 telltale signs of  
a bad data scientist**

InfoWorld

**If Your Data Is Bad, Your  
Machine Learning Tools  
Are Useless**

by Thomas C. Redman

...we will not do data collection or experimental design, but students should be familiar with the basics



# Statistical Analysis

1. Plan research design
2. Collect data from a sample
3. Visualize and summarize the data (plots and summary stats)
4. Make inferences from data (i.e. estimate stuff, test hypotheses, ...)
5. Interpret results

# Statistical Analysis

1. Plan research design
2. Collect data from a sample
3. Visualize and summarize the data (plots and summary stats)
4. Make inferences from data (i.e. estimate stuff, test hypotheses, ...)
5. Interpret results

**Have touched on these already...**

# Statistical Analysis

1. Plan research design

**Will focus on these**

2. Collect data from a sample

3. Visualize and summarize the data (plots and summary stats)

4. Make inferences from data (i.e. estimate stuff, test hypotheses, ...)

5. Interpret results

# Research Design

**Observational** Collect data by observing a population. If there are treatments, they are not under control of the researcher.

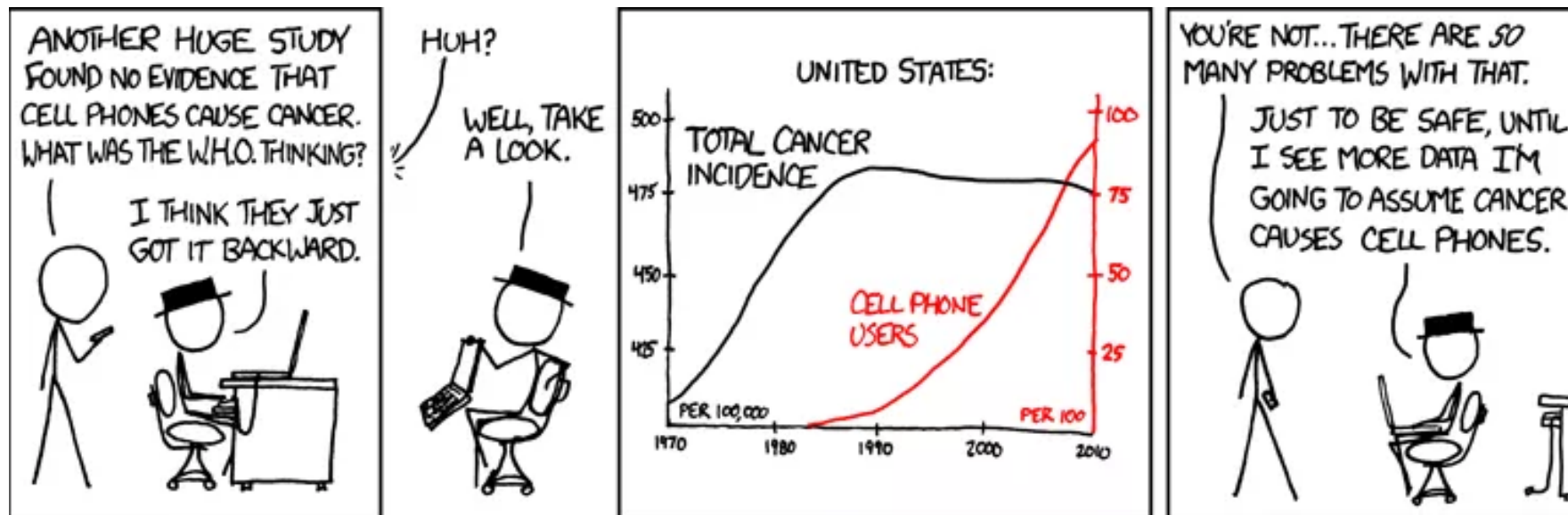
**Natural Experiment** Observe naturally-occurring phenomena. Approximates a controlled study, despite the researcher not having control of any groups. (e.g. different US state policies of COVID protocols)

**Case Studies and Surveys** Analysis based on previously-collected data. E.g. Analysis of US census data, or US current population survey (CPS)

**Randomized Control** Researcher controls treatment among groups. Used to assess *causal* relationships. Stronger than correlational study but difficult to conduct.

# Causation vs. Correlation

Studies generally try to show *either* correlation (association) or causation, but they are not the same...



[ Image: XKCD.com ]

*What is an alternative likely explanation for this chart?*

# Confounding Variables

A variable that influences the *response* but is unaccounted for in data collection

**Example** You are studying whether birth order affects Down's Syndrome in the child. You collect a simple random sample of children, their birth order, and instances of Down's syndrome.

**Explanation** Maternal age (confounder) was not recorded.  
Scenarios that are not distinguishable from data:

1. Higher maternal age is directly associated with Down's
2. Higher maternal age is directly associated, regardless of birth order
3. Maternal age directly assoc. with birth order (mother is older with later children)
4. Maternal age has no effect

# Controlling for Confounders

Two primary ways to control for confounders...

**Stratified Sampling** Divide population into smaller groups. Previous example can divide population of children by maternal age at birth and sample from each strata using a simple random sample.

**Probabilistic Model** Use the techniques learned earlier in this course to model effects among each of the variables.

We will learn more about probabilistic modeling in the Machine Learning portion of this course...

# Example

Do children of parents that smoke tend to smoke at higher rates?

Does having parents that smoke *cause* a child to smoke at higher rates?

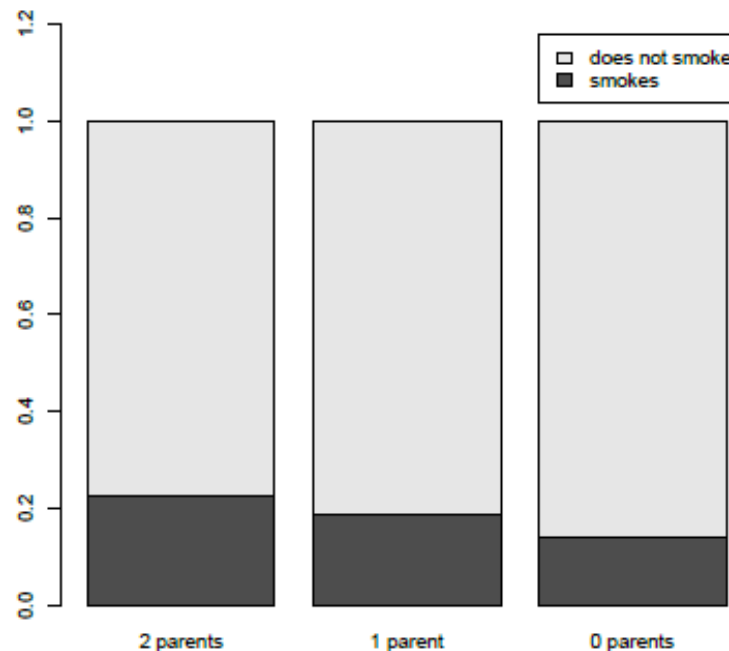
0 parents smoke	
student smokes 0.1386	student does not smoke 0.8614

1 parent smoke	
student smokes 0.1858	student does not smoke 0.8142

2 parents smoke	
student smokes 0.2247	student does not smoke 0.7753



This is an **observational study** – they tend not to distinguish causation from correlation...

What are alternative explanations (e.g. confounders)?



# Randomized Controlled Experiments

## Approach

1. **Control** for effects of confounders by comparing several *treatments*
2. **Randomize** the assignment of subjects to treatments to eliminate bias due to systematic differences in categories
3. **Replicate** experiment on many subjects, to reduce chance of variation in the results

# Example: Pfizer COVID Phase 3 Vaccine Trials

- 1. Placebo Control** Subjects are randomly selected to receive either the vaccine or an injection of saline solution
- 2. Randomize** Stratified sampling with age strata: 12-15yrs, 16-55yrs, 55+yrs with ~40% in the latter strata
- 3. Replicate** Experiment is repeated at multiple sites in several countries

Full statistical procedures are published and publicly available:

[https://cdn.pfizer.com/pfizercom/2020-11/C4591001\\_Clinical\\_Protocol\\_Nov2020.pdf](https://cdn.pfizer.com/pfizercom/2020-11/C4591001_Clinical_Protocol_Nov2020.pdf)

# Example: Pfizer COVID Phase 3 Vaccine Trials

The landmark phase 3 clinical trial enrolled **46,331** participants at **153** clinical trial sites around the world.

## Trial Geography



Our trial sites are located in **Argentina, Brazil, Germany, Turkey, South Africa** and the **United States**.

## Participant Diversity

Approximately **42%** of overall and **30%** of U.S. participants have diverse backgrounds.

Participants	Overall Study	U.S. Only
Asian	5%	6%
Black	10%	10%
Hispanic/Latinx	26%	13%
Native American	1.0%	1.3%

**49.1%** of participants are male and **50.9%** are female

## Participant Age



Ages 12-15 2,260

Ages 16-17 754

Ages 18-55 25,427

Ages 56+ 17,879

# Example: Honeybees

Designing controlled experiments can be hard—especially if your subjects are ornery...

People have been trying to control breeding to create more productive honeybees for over 100 years. **Gregor Mendel** (of inheritance fame) tried and failed...

In 1956 an apiary in southeast Brazil tried to control breeding by hybridizing African and European honeybees. The “Africanized” bees escaped. We now have them all over the US, including Arizona, and they are mean!

# Example: Polio Vaccine

In 1954 the National Foundation of Infantile Paralysis tested Jonas Salk's Polio vaccine in a controlled trial with the following cohorts:

- Vaccinate all 2<sup>nd</sup> grade children with parental consent
- Use grades 1 and 3 as control (unvaccinated)

*Do you see anything wrong with this design?*

To address study flaws the US Public Health Service (PHS) conducted a new randomized control study:

- Flip coin for each child (randomized control)
- Kids in control get salt water injection
- Diagnosticians not told what group each child is in (double blind)



Rates per 100,000

	PHS		NFIP	
	Size	Rate	Size	Rate
Treatment	200,000	28	225,000	25
Control	200,000	71	725,000	54
No consent	350,000	46	125,000	44

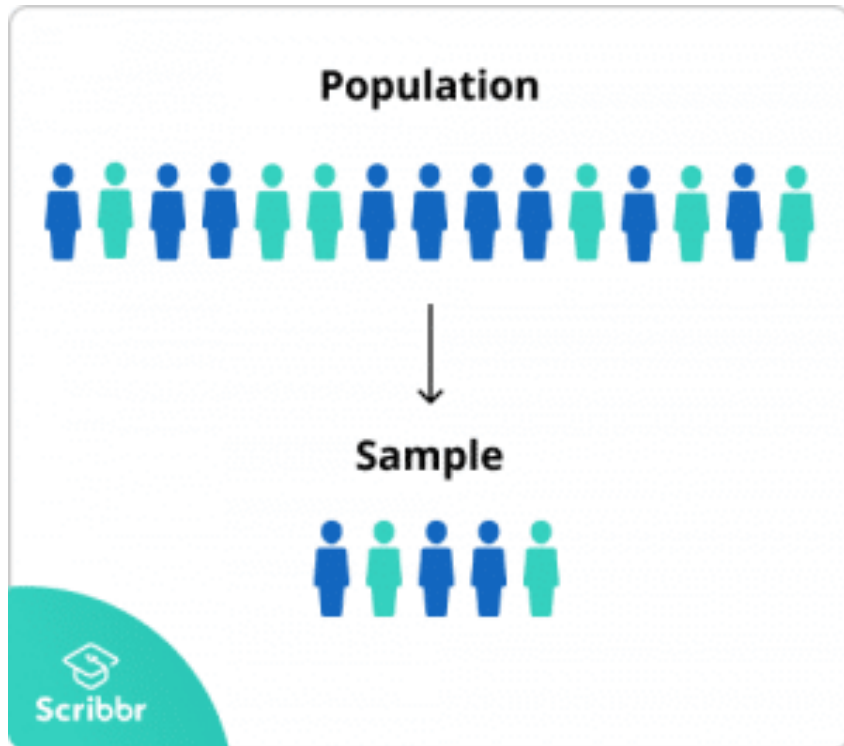
Source: Watkins, J.

# Questions for Data Collection

- What can I measure?
- What *shall* I measure?
- How shall I measure it?
- How frequently shall I measure it?
- What obstacles prevent reliable measurement?

# Population vs. Sample

Generally infeasible to collect data from entire *population*



**Population** Entire group that we want to draw conclusions about.

Can be defined in terms of location, age, income, etc.

**Sample** Specific group that we collect data from.

# Examples of Population vs. Sample

<b>Population</b>	<b>Sample</b>
Advertisements for IT jobs in the Netherlands	The top 50 search results for advertisements for IT jobs in the Netherlands on May 1, 2020
Songs from the Eurovision Song Contest	Winning songs from the Eurovision Song Contest that were performed in English
Undergraduate students in the Netherlands	300 undergraduate students from three Dutch universities who volunteer for your psychology research study
All countries of the world	Countries with published data available on birth rates and GDP since 2000



# Reasons for Sampling

**Necessity** It is usually impractical or impossible to collect data from an entire population due to size or inaccessibility.

**Practicality** It is just easier and more efficient to collect data from a *sample* of the population

**Cost-effectiveness** There are fewer participant, laboratory, equipment, and researcher costs involved.

**Manageability** Storing data and running statistical analyses is easier on smaller datasets.

# Population Parameter vs. Sample Statistic

**Population parameter** A measure that describes *the whole population*.

**Sample statistic** A measure that describes the sample and reflects the population parameter.

***Example** We are studying student political attitudes and ask students to rate themselves on a scale: 1, very liberal, to 7, very conservative. The **population parameter** of interest is the average political leaning. The sample mean of 3.2 is our **statistic**.*

# Sampling Error

**Definition** The *sampling error* is the difference between the population parameter and the sample statistic.

- Sampling errors are normal, but we want them to be low
- Samples are random, so sample statistics are estimates and thus subject to random noise
- **Sample bias** occurs when the sample is not representative of the population (for various reasons)

# Sampling Methods

Sampling must be conducted properly, to avoid sample bias

Two primary types of sampling...

**Probability Sampling** Random selection allowing strong statistical inferences about the population

**Non-Probability Sampling** Based on convenience or other criteria to easily collect data (but no random sampling)

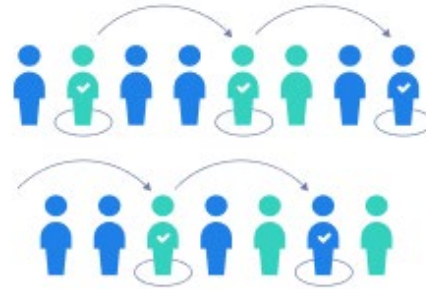
# Probability Sampling



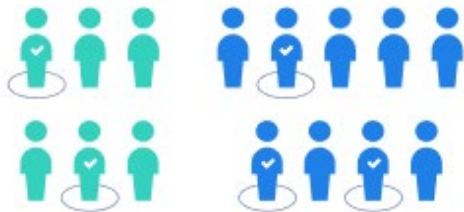
Simple random sample



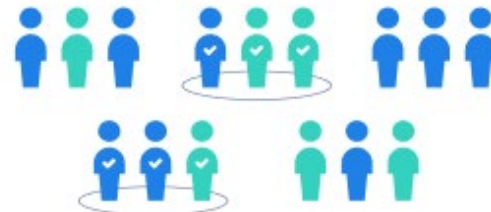
Systematic sample



Stratified sample



Cluster sample



## Simple Random Sample (SRS)

Each member of the population has the *same chance* of being selected (i.e. uniform over the population)

### Example : American Community Survey (ACS)

Each year the US Census Bureau use simple random sampling to select individuals in the US. They follow those individuals for 1 year to draw conclusions about the US population as a whole.

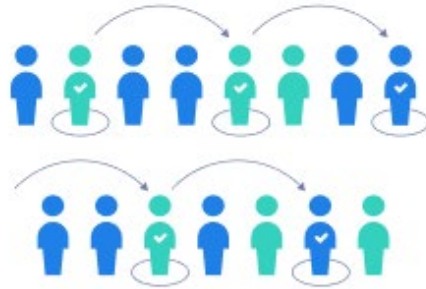
# Probability Sampling



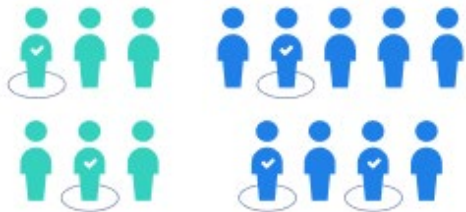
Simple random sample



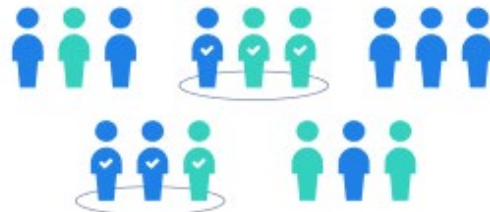
Systematic sample



Stratified sample



Cluster sample



## Simple Random Sample (SRS)

Each member of the population has the *same chance* of being selected (i.e. uniform over the population)

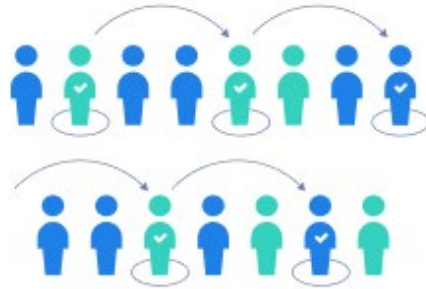
- Most straightforward probability sampling method
- Impractical unless you have a complete list of every member of population

# Probability Sampling

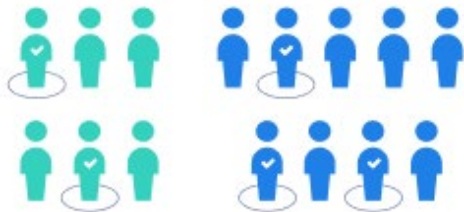
Simple random sample



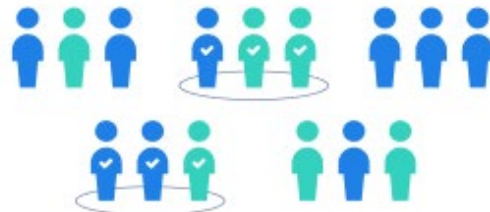
Systematic sample



Stratified sample



Cluster sample



## Systematic Sample

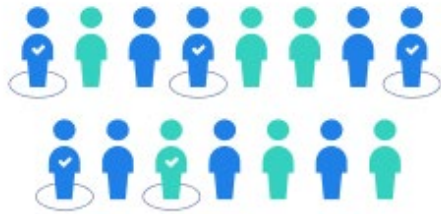
Select members of population at a regular interval, determined in advance

**Example** You own a grocery store and want to study customer satisfaction. You ask *every 20<sup>th</sup> customer* at checkout about their level of satisfaction.

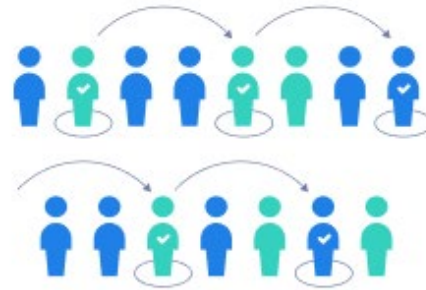
**Note** We cannot itemize the whole population in this example, so SRS is not possible.

# Probability Sampling

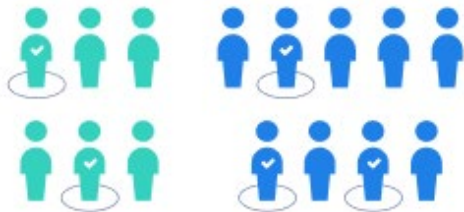
Simple random sample



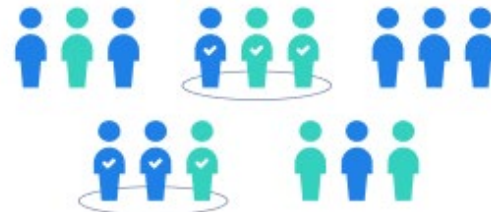
Systematic sample



Stratified sample



Cluster sample



## Systematic Sample

Select members of population at a regular interval, determined in advance

- Imitates SRS but is easier in practice
- Can even do systematic sampling when you can't access the entire population in advance
- **Do not** use when population is ordered



# Simple vs. Systematic Random Sample

Consider a school with 1,000 students and suppose we want to select 100 for data collection...

**Simple Random Sample** Place all names in a bucket and draw 100 randomly. Each student has a 10% chance of being selected.

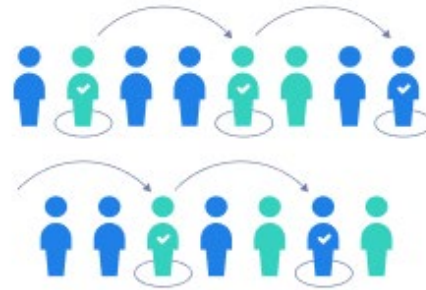
**Systematic Random Sample** Any systematic pattern may produce a random sample. One possibility: assume students have ID numbers from 1 to 1,000 and we choose a random starting point (e.g. 533). We pick every 10<sup>th</sup> name thereafter, in a circular fashion (i.e. wraparound at 1,000). Note: students {3, 13, 23, ..., 993} have nonzero probability of being selected, whereas students outside this set have zero probability.

# Probability Sampling

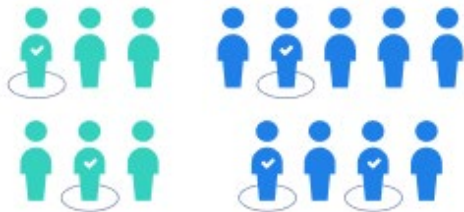
Simple random sample



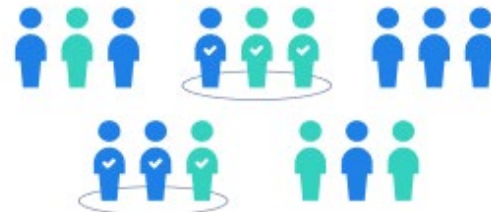
Systematic sample



Stratified sample



Cluster sample



## Stratified Sample

Divide population into *homogeneous* subpopulations (strata). Probability sample the strata.

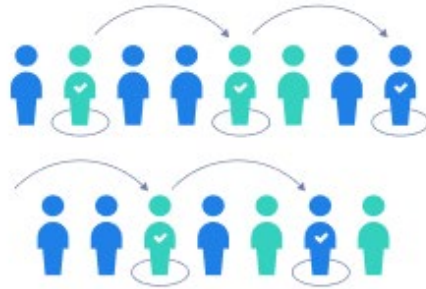
**Example** We wish to solicit opinions of UA CS freshman, but they are about 14% women. SRS will fail to capture adequate proportion of women. We divide into men / women and perform SRS within each group.

# Probability Sampling

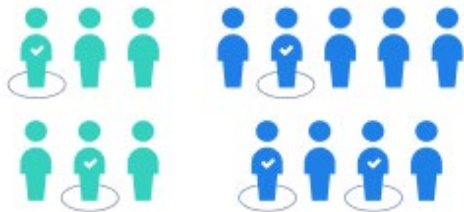
Simple random sample



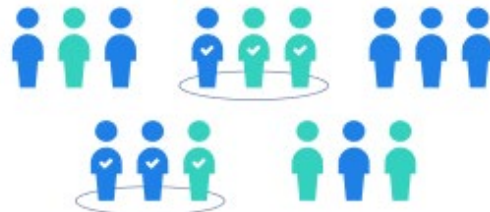
Systematic sample



Stratified sample



Cluster sample



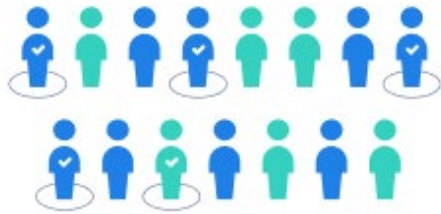
## Stratified Sample

Divide population into *homogeneous* subpopulations (strata). Probability sample the strata.

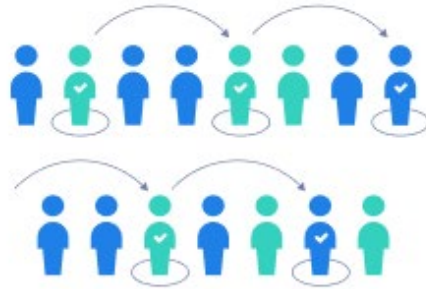
- Use when population is diverse and want to accurately capture characteristic of each group
- Ensures similar variance across subgroups
- Lowers overall variance in the population

# Probability Sampling

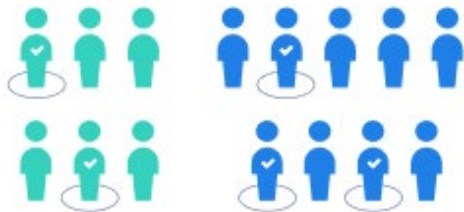
Simple random sample



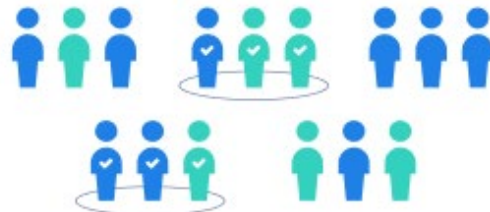
Systematic sample



Stratified sample



Cluster sample



## Cluster Sample

Divide population into subgroups (clusters). Randomly select entire clusters.

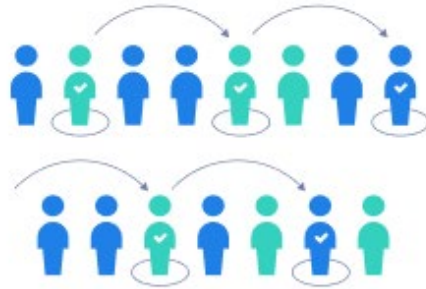
**Example** We wish to study the average reading level of *all 7<sup>th</sup> graders in the city* (population). Create a list of all schools (clusters) then randomly select a subset of schools and test every student.

# Probability Sampling

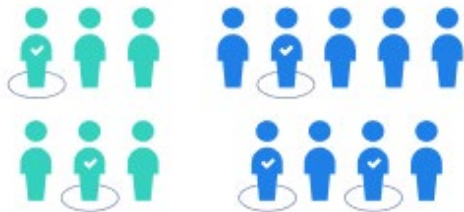
Simple random sample



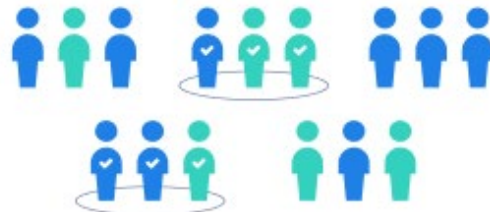
Systematic sample



Stratified sample



Cluster sample

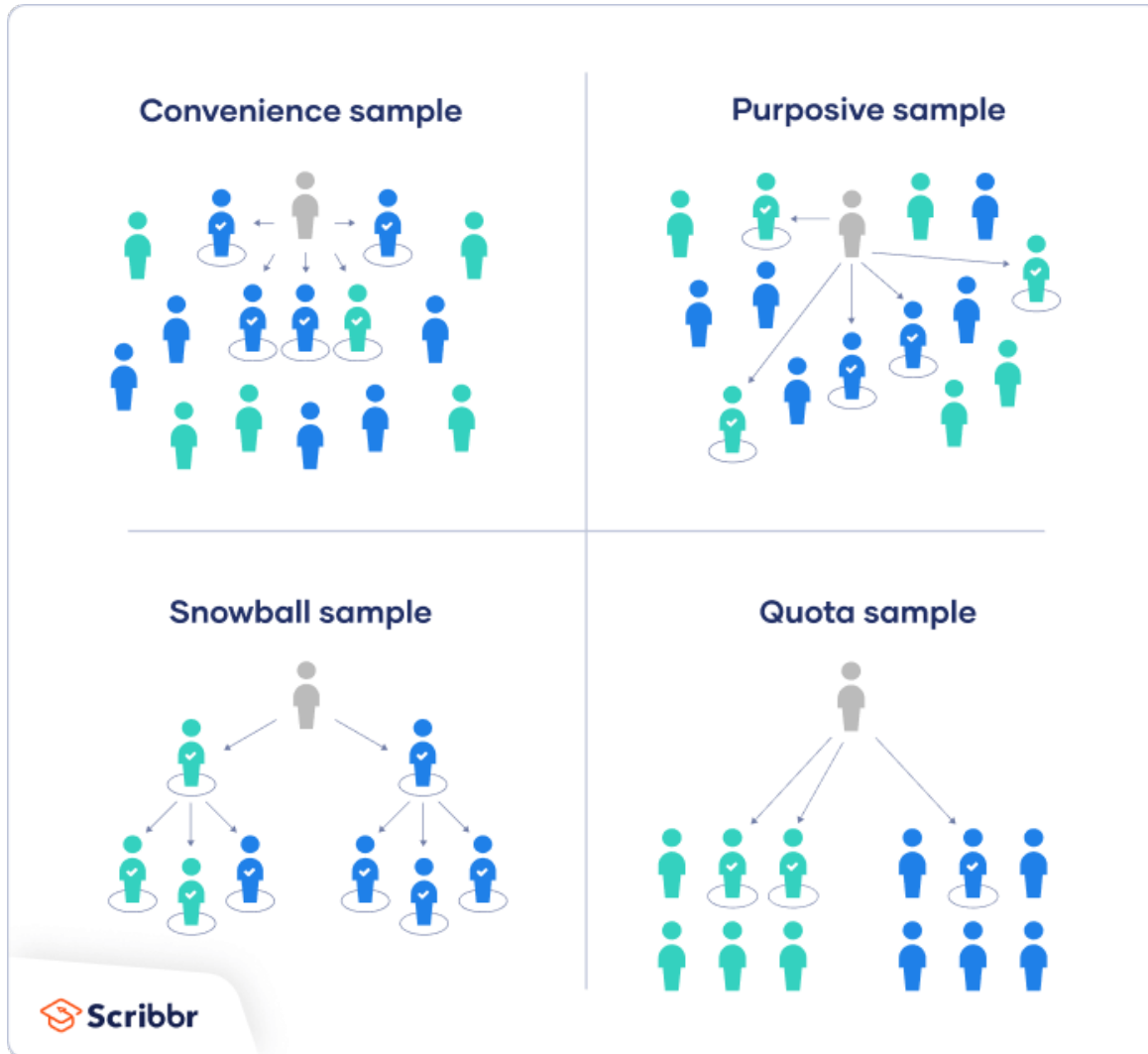


## Cluster Sample

Divide population into subgroups (clusters). Randomly select entire clusters.

- This is *single-stage* cluster sampling
- *Multi-stage* avoids sampling every member of a group
- Related to stratified sampling, but groups are not homogeneous

# Non-Probability Sampling



Easier to access data, but higher risk of *sample bias* compared to probability sampling

Usually used to perform *qualitative research* (e.g. gathering student opinions, experiences, etc.)

We will not focus on these, but you should be aware if your data are from non-probability methods

# Sampling Bias



Occurs if data are collected in a way that some members of the population have lower/higher probability of being sampled than others

Sometimes is unavoidable (e.g. not all members are equally accessible) but it must then be corrected for

**Example** We conduct a poll by randomly calling numbers in a phone book. People that have less time are less likely to response. Called *non-response bias*.

# Common Types of Sampling Bias

**Self-selection** Possible whenever members (typically people) under study have control over whether to participate. E.g. online or phone-in poll—user can choose whether to initiate participation.

**Exclusion** Results from excluding certain groups from the sample. E.g. excluding groups that move in or out of a study area during follow-up.

**Survivorship** Only *surviving* subjects are selected. Here “surviving” is a loose definition, non-survivors may simple fall out of view. E.g. using record of current companies as indicator of economic climate.

**Survivorship** Only *surviving* subjects are selected. Here “surviving” is a loose definition, non-survivors may simple fall out of view. E.g. using record of current companies as indicator of economic climate.



# Example of Bias in a Simple Random Sample

SRS is least prone to bias, but not always...

You want to study procrastination and social anxiety levels in undergraduate students at your university using a simple random sample. You assign a number to every student in the research participant database from 1 to 1500 and use a random number generator to select 120 numbers.

***What is the cause of bias in this simple random sample?***

# Example of Bias in a Simple Random Sample

SRS is least prone to bias, but not always...

You want to study procrastination and social anxiety levels in undergraduate students at your university using a simple random sample. You assign a number to every student in the research participant database from 1 to 1500 and use a random number generator to select 120 numbers.

Although you used a random sample, not every member of your target population –undergraduate students at your university – had a chance of being selected. Your sample misses anyone who did not sign up to be contacted about participating in research. This may bias your sample towards people who have less social anxiety and are more willing to participate in research.