

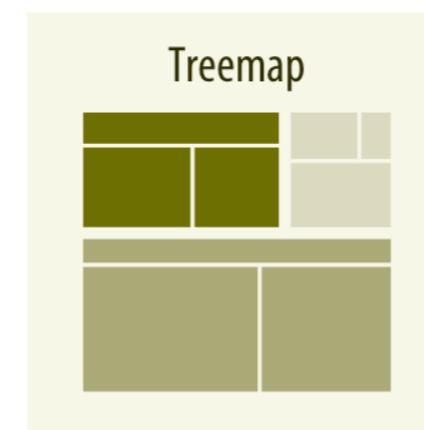
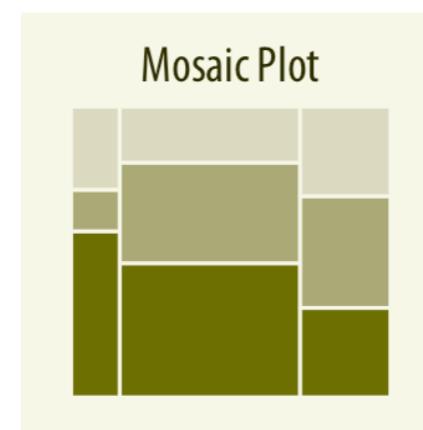
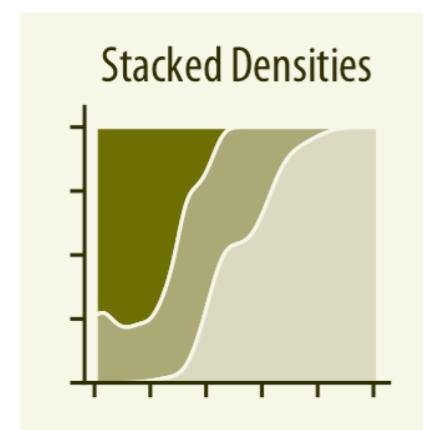
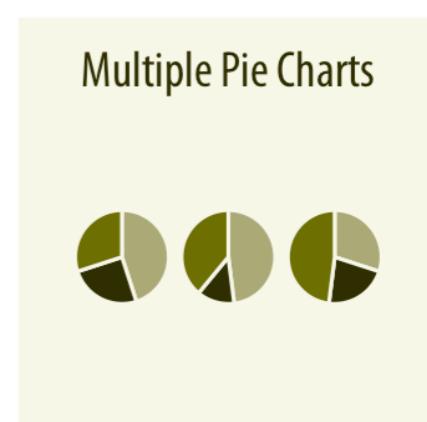
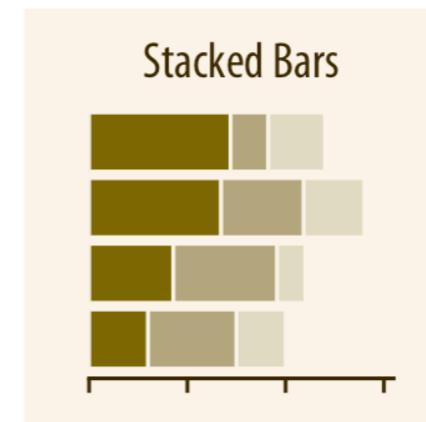
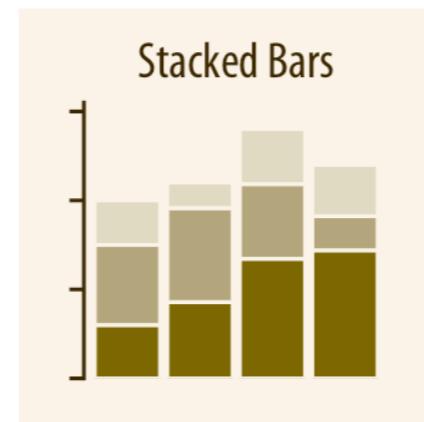
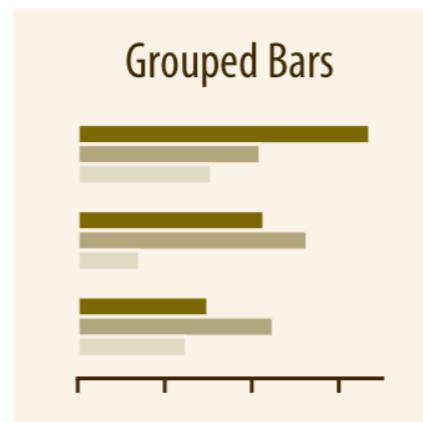
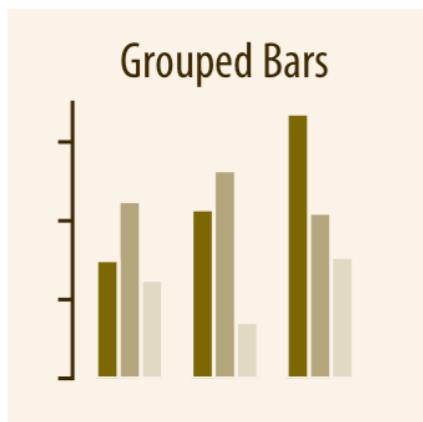
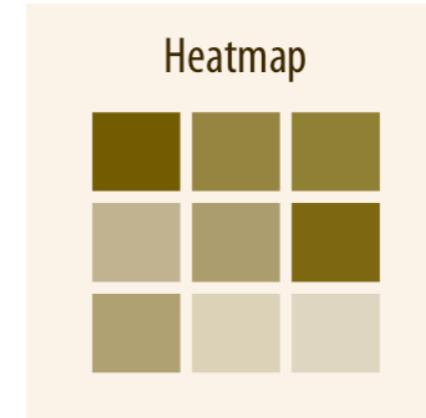
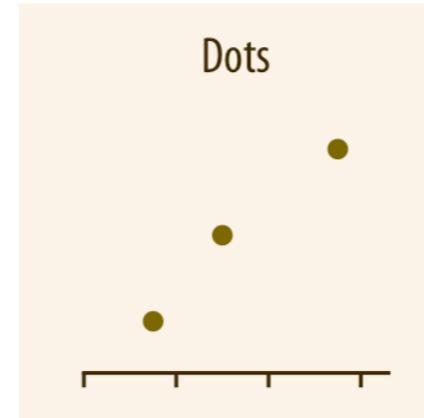
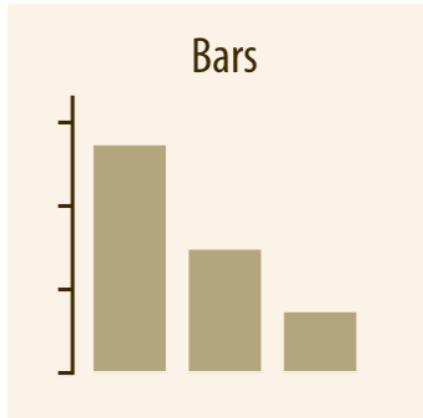
Introduction to Data Visualization



QB Bootcamp, Day 3
Friday, 29 August 2019
2:00pm - 2:30pm

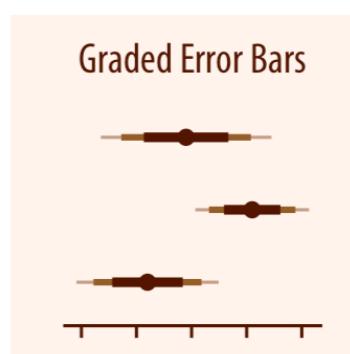
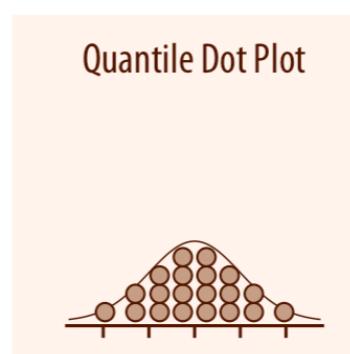
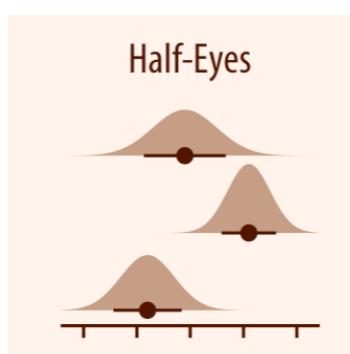
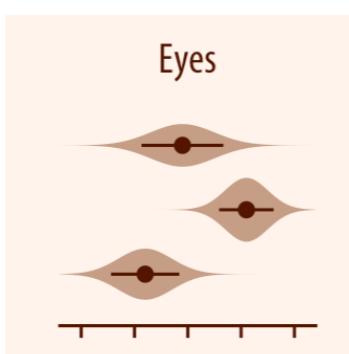
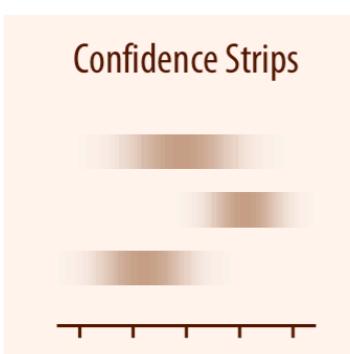
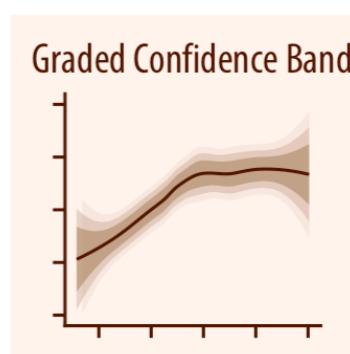
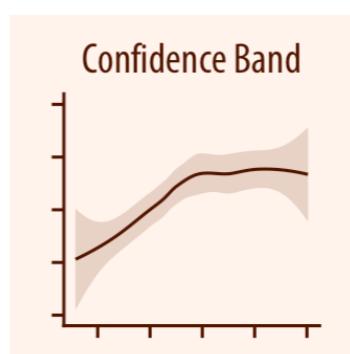
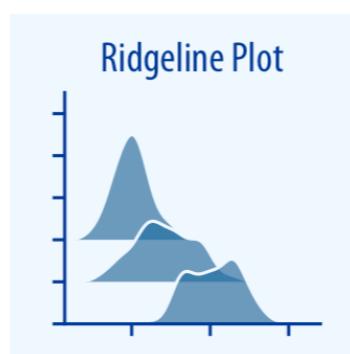
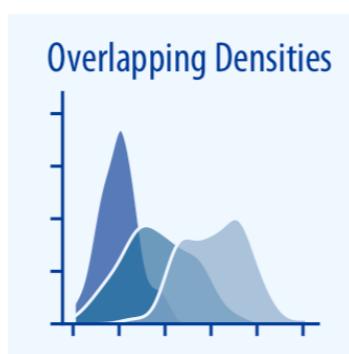
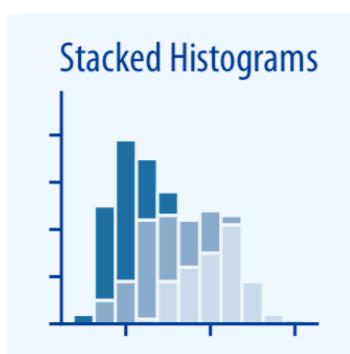
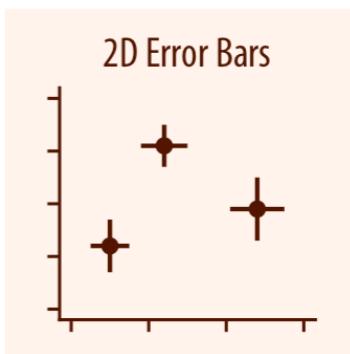
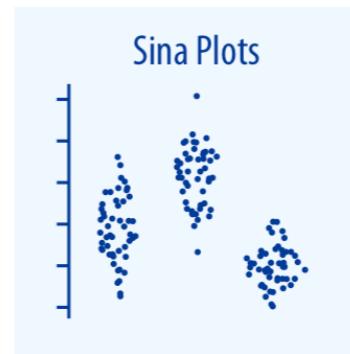
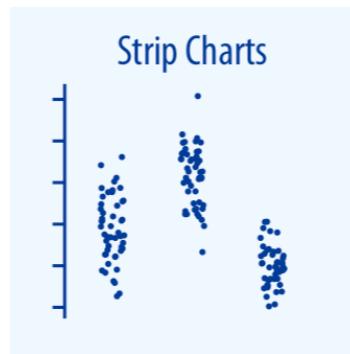
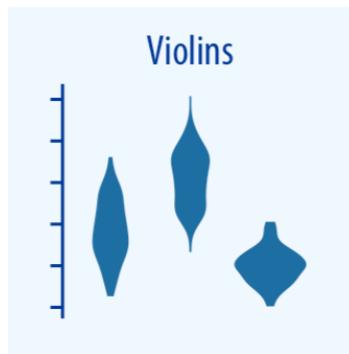
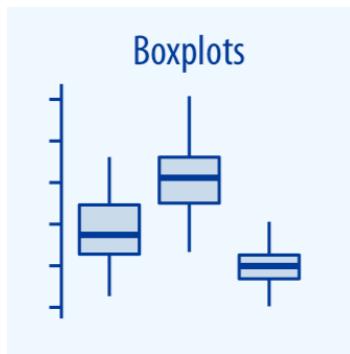
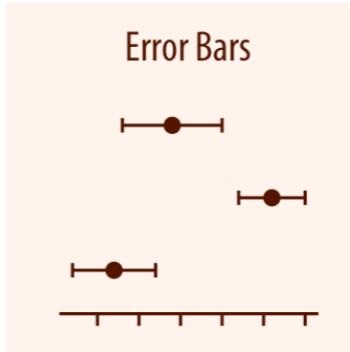
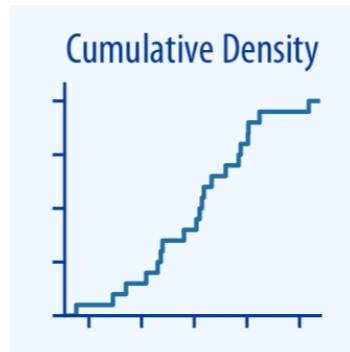
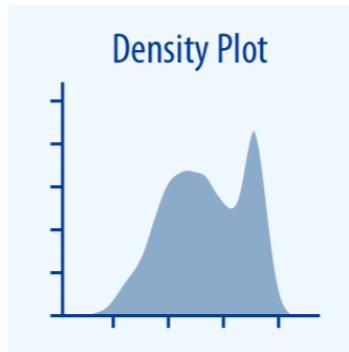
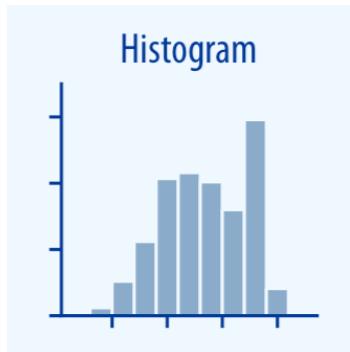
There are many ways to visualize data

visualizing amounts or proportions



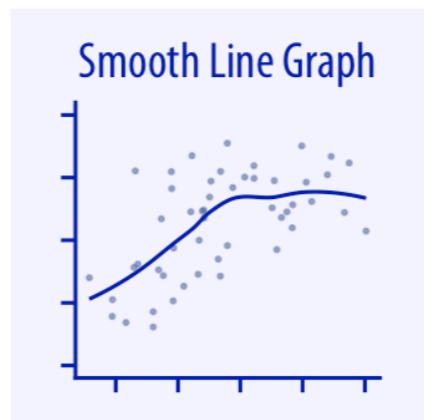
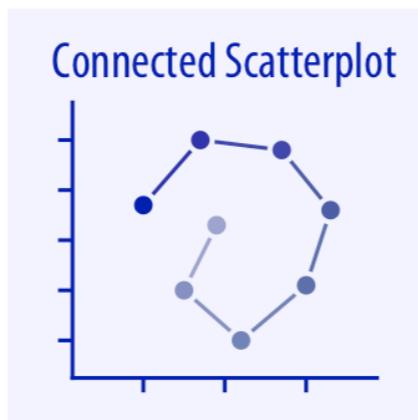
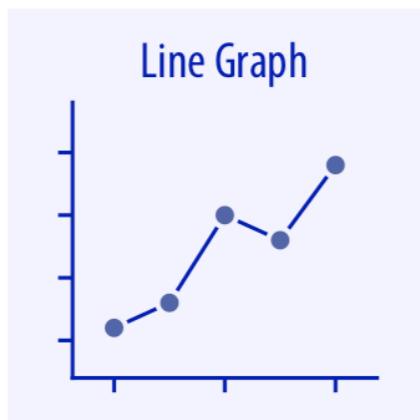
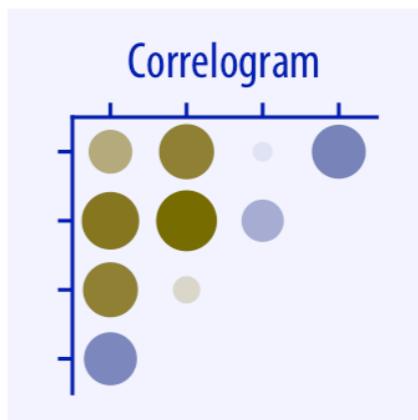
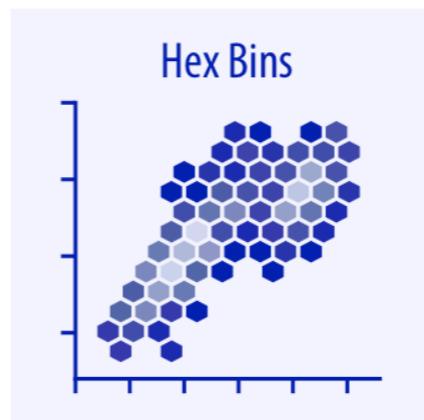
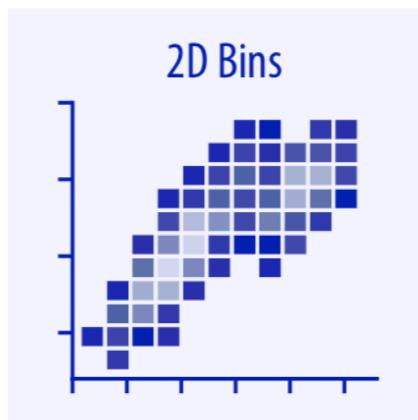
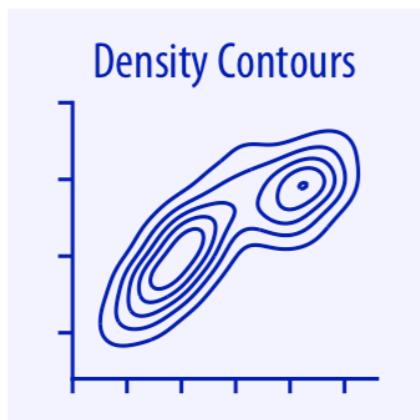
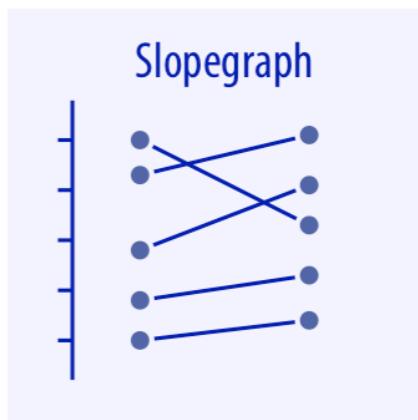
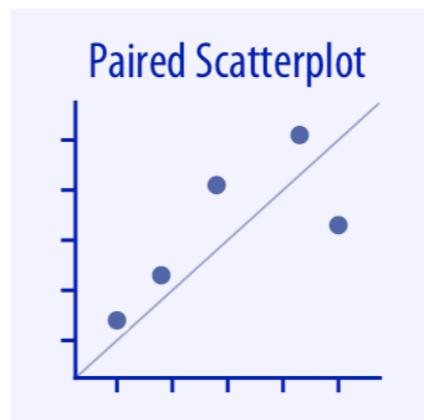
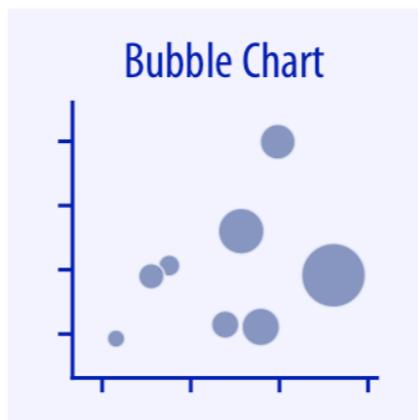
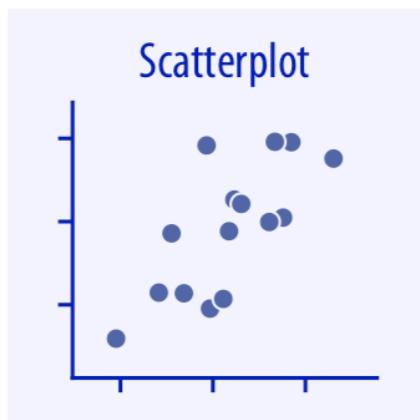
There are many ways to visualize data

visualizing distributions or uncertainties



There are many ways to visualize data

visualizing pairwise relationships



Data visualization resources

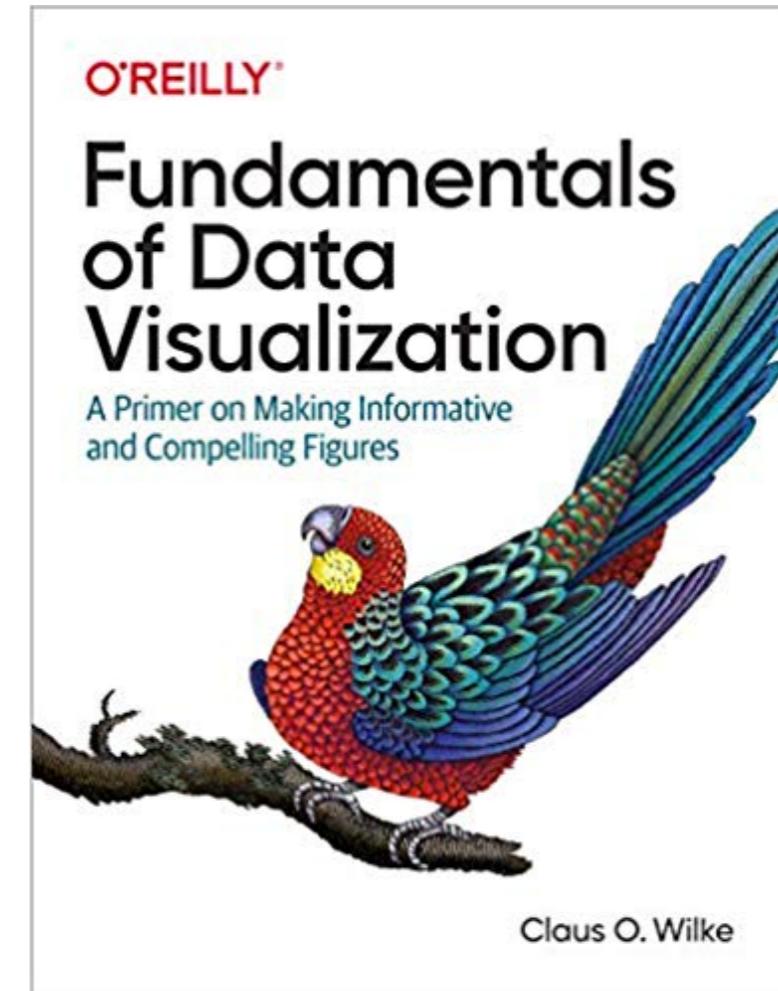
Claus Wilke gives excellent guidance on data visualization do's and don'ts

Molly Hammell will discuss data visualization later on in SEE.

I will briefly discuss 2 issues:

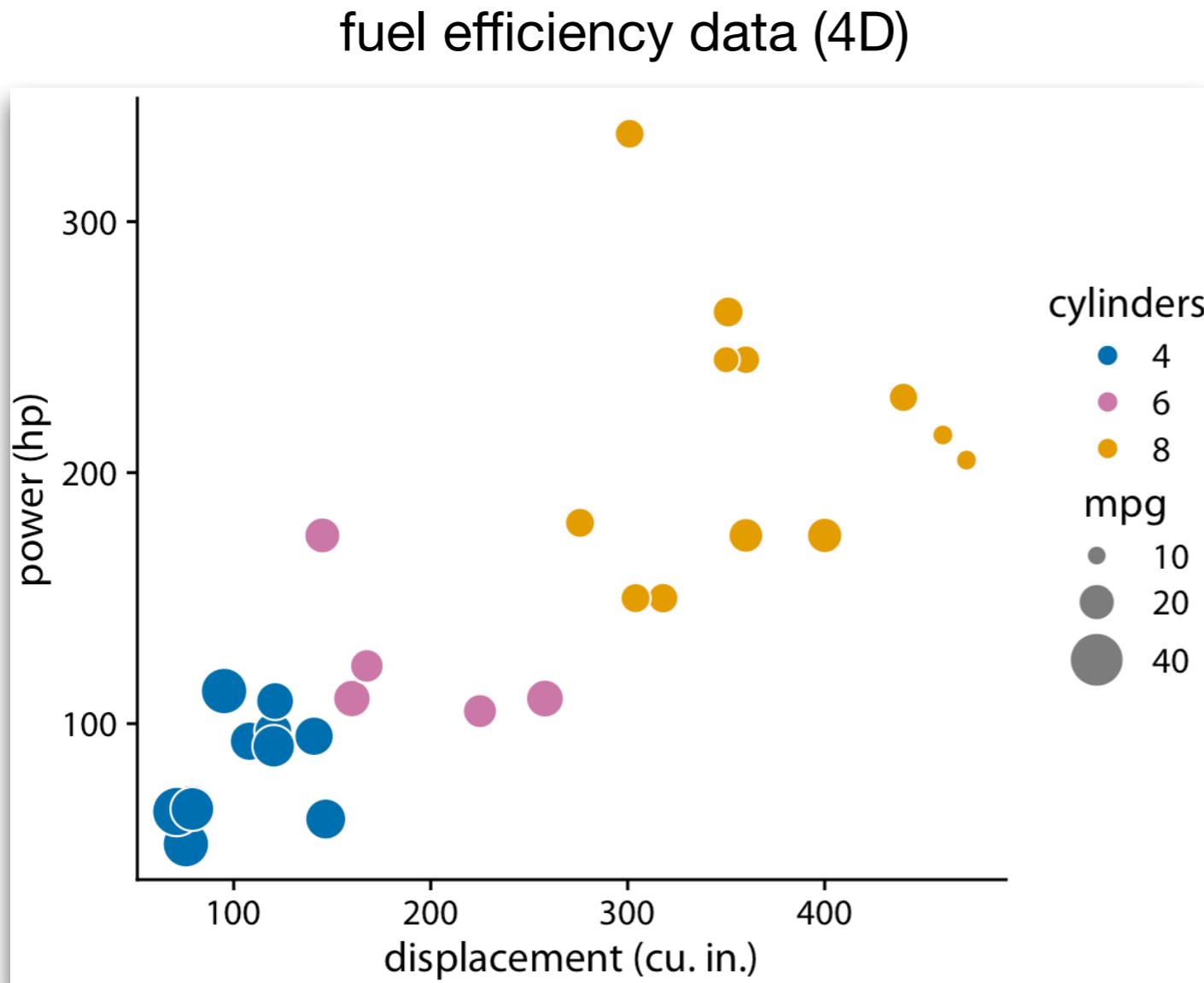
- Aesthetics
- 3D plots

We will then dive into Matplotlib and Seaborn



**Fundamentals of
Data Visualization**
Wilke, 2019

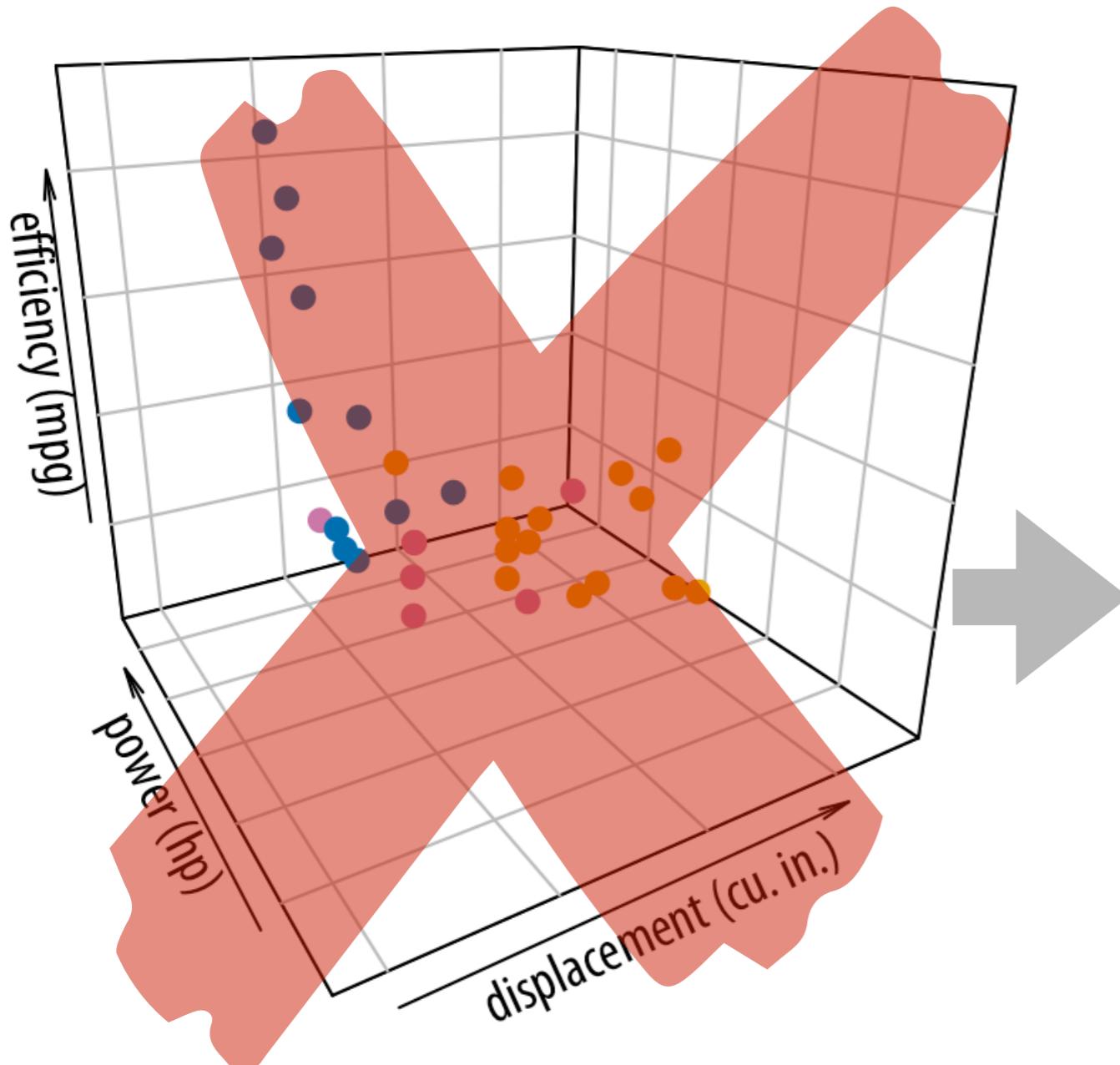
All data visualization uses “aesthetics” to encode quantitative information



aesthetic	quantity
position	x: displacement y: power
size	mpg
color	# cylinders

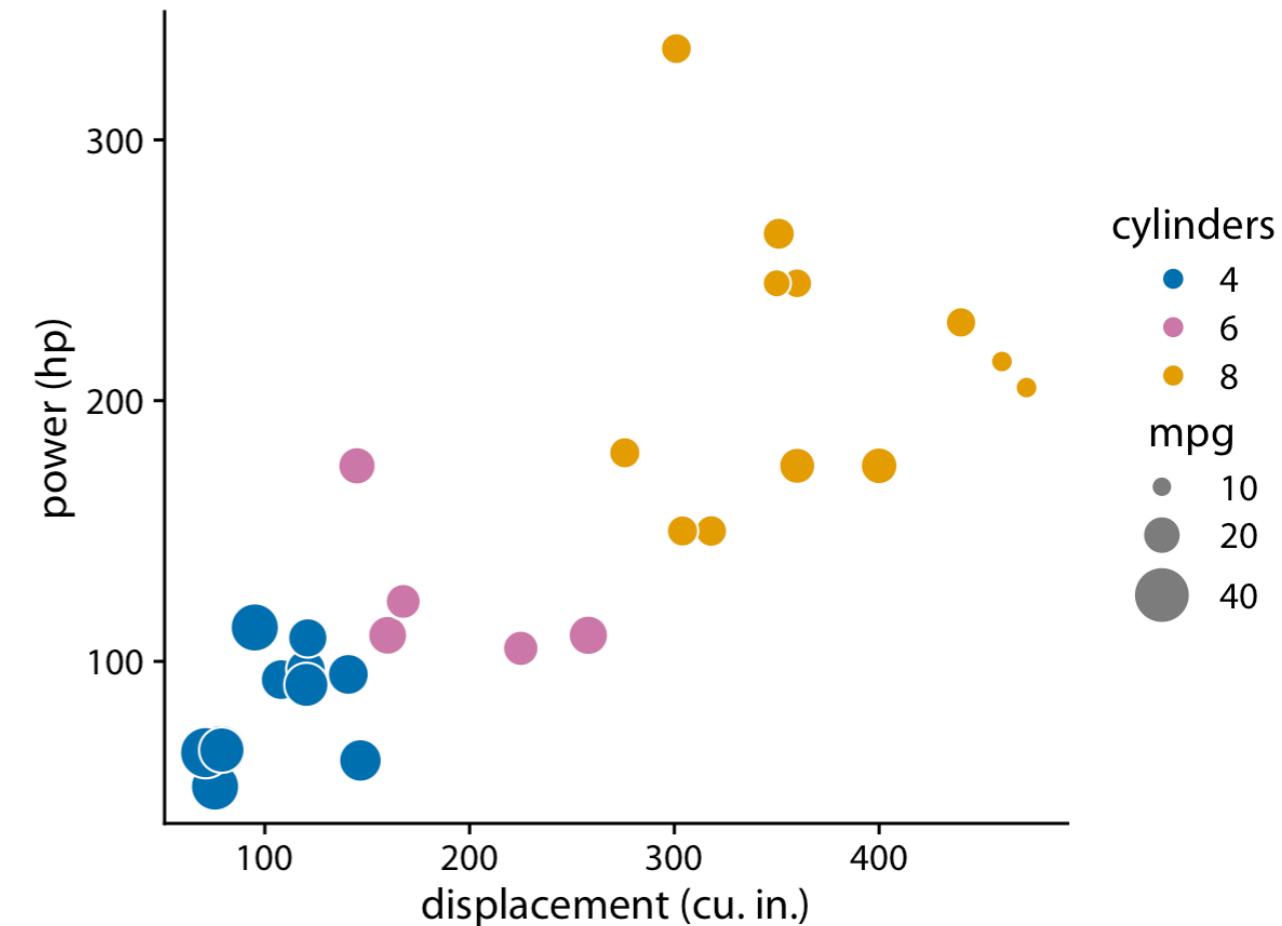
Please don't plot data in 3D

In a 3D figure, it's virtually impossible to figure out the coordinates of each point by eye.

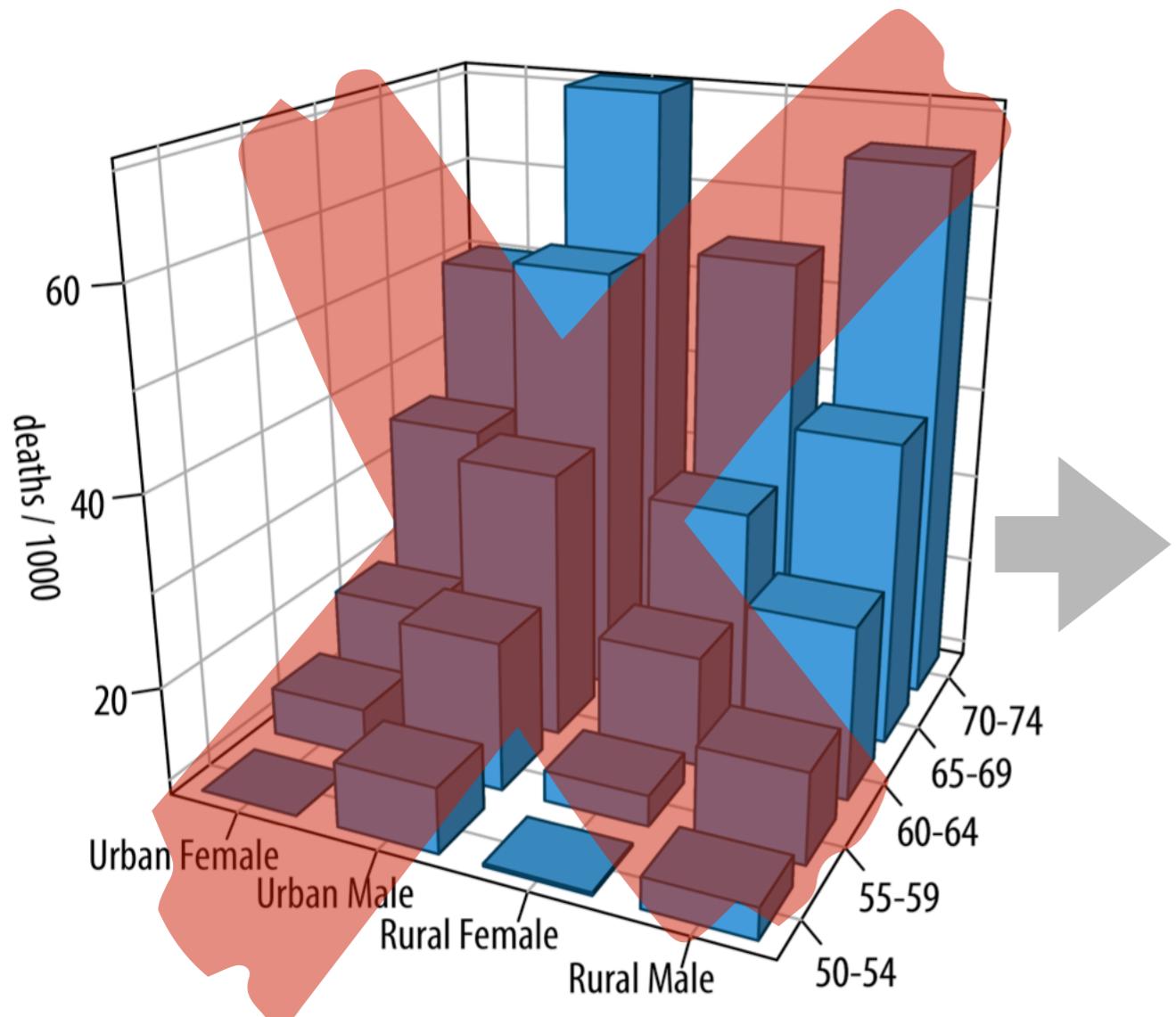


cylinders • 4 • 6 • 8

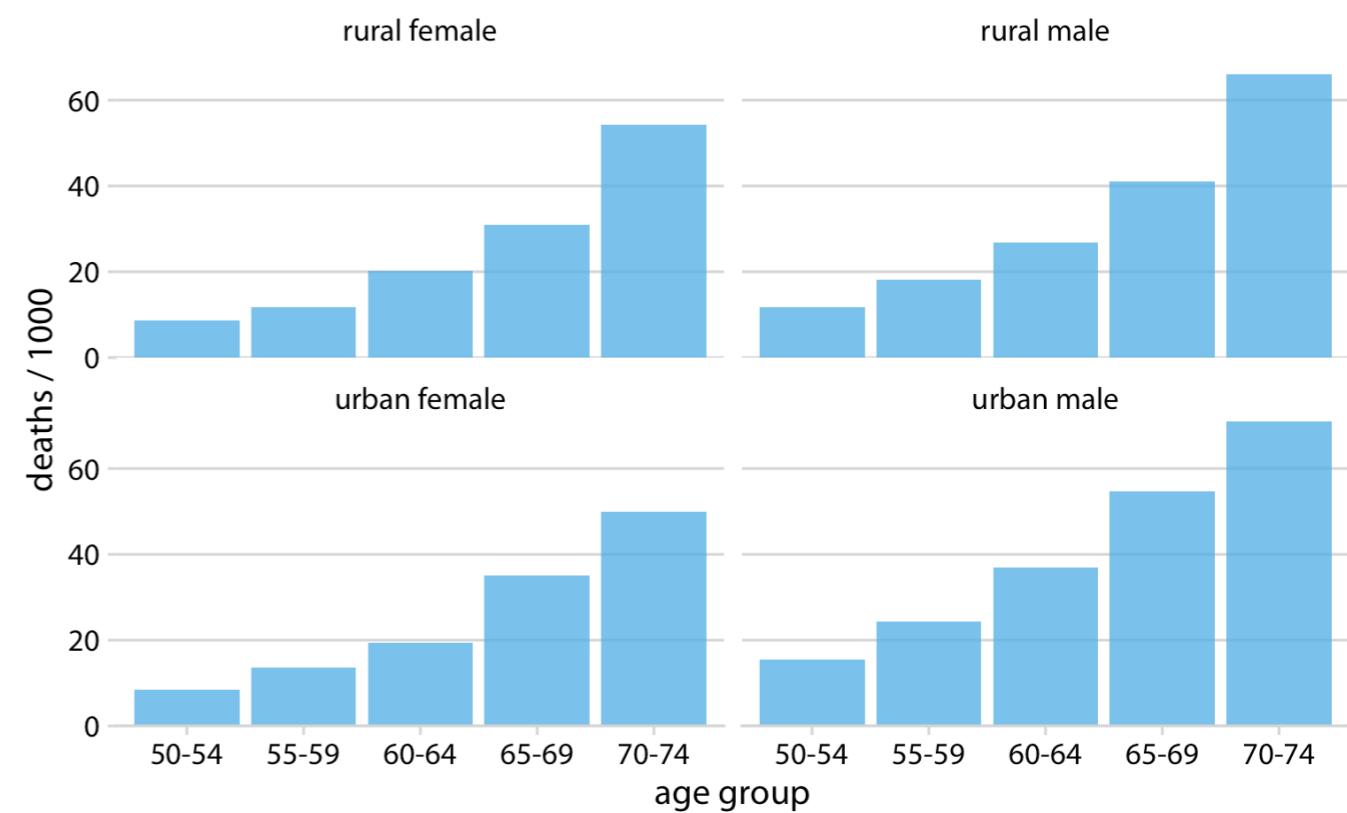
Plot in 2D using multiple aesthetics instead



Please don't plot data in 3D

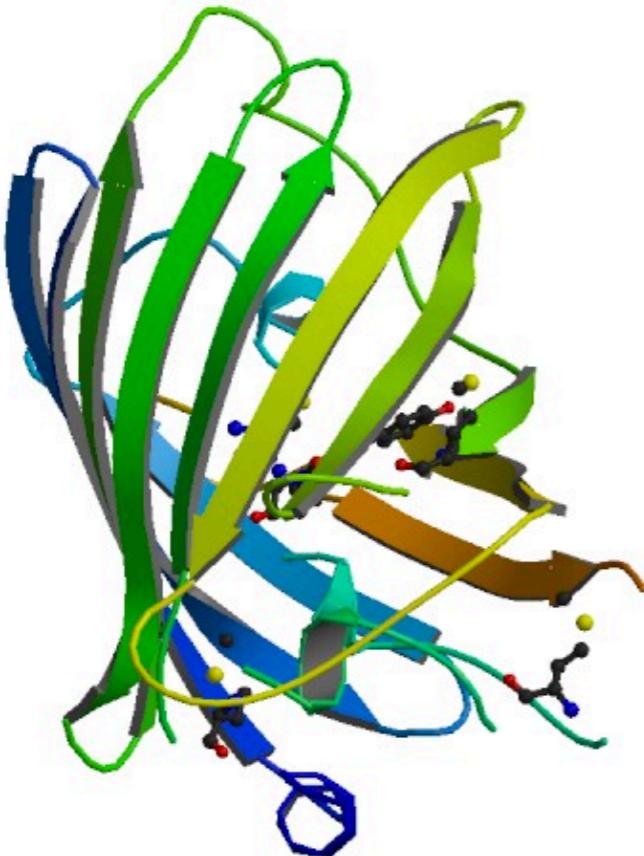


Alternatively,
use “small multiples” of 2D plots

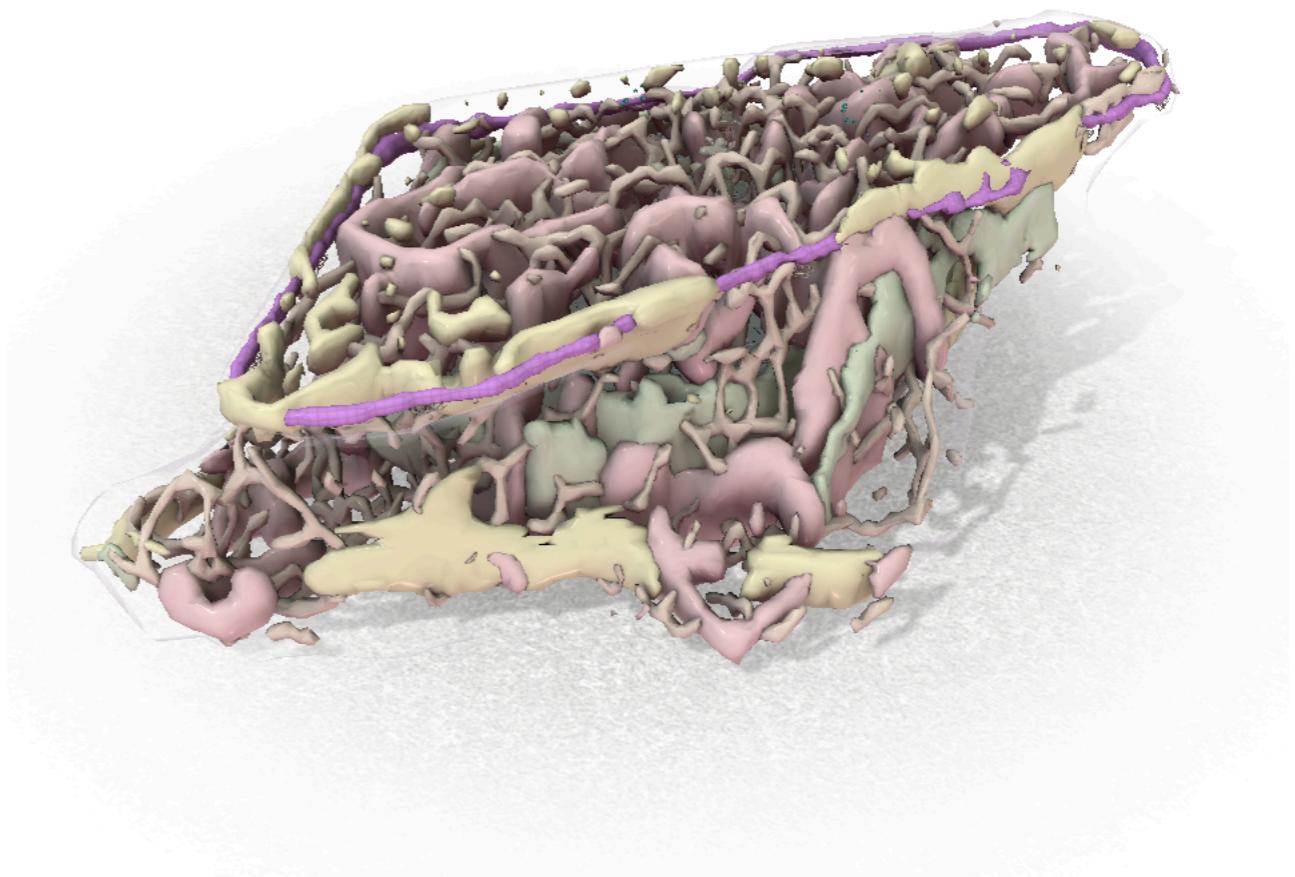


Only use 3D if you're illustrating some inherently 3D object

Structure of GFP
(PDB:1EMA)



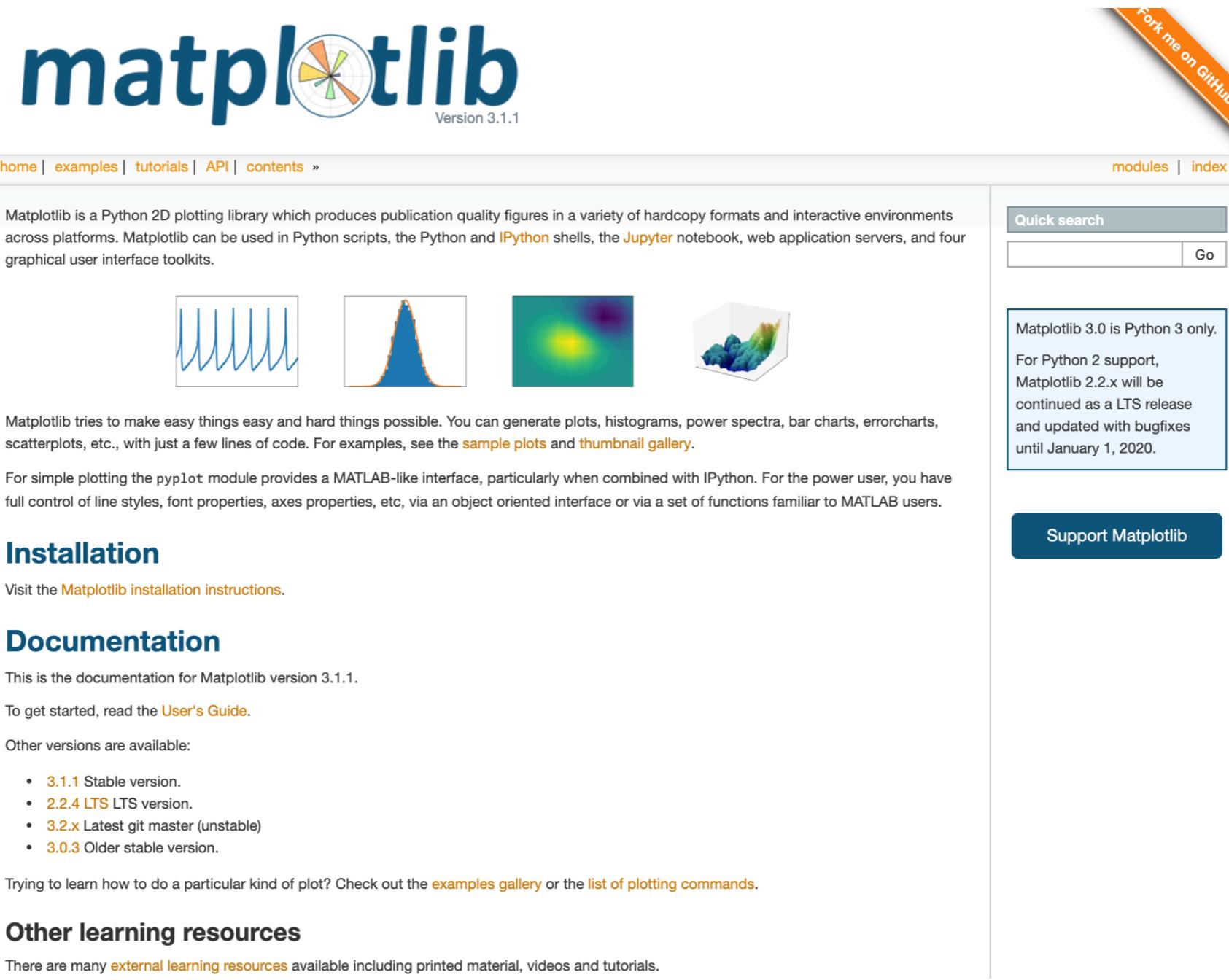
cellular structures
(Allen Cell Explorer)



<https://www.rcsb.org/structure/1ema>
<https://www.allencell.org/visual-guide-to-human-cells.html>

Matplotlib is the foundation for most graphics in Python

Matplotlib provides basic graphing infrastructure.
It is functional, but the capabilities are rather basic and
it takes effort to make professional-looking plots.



The screenshot shows the official Matplotlib website at matplotlib.org. The header features the "matplotlib" logo with "Version 3.1.1" below it. A "Fork me on GitHub" button is in the top right. The main content area includes navigation links (home, examples, tutorials, API, contents, modules, index), a "Quick search" bar, and a note about Python 3 support. It also displays four sample plots: a line plot, a histogram, a heatmap, and a 3D surface plot. Below these are sections for Installation, Documentation, and Other learning resources, each with its own sub-content and links.

Installation
Visit the [Matplotlib installation instructions](#).

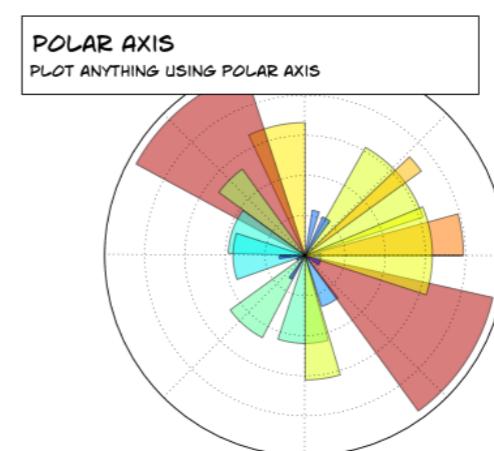
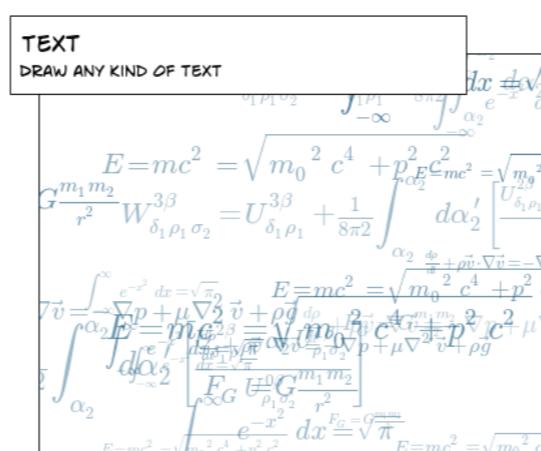
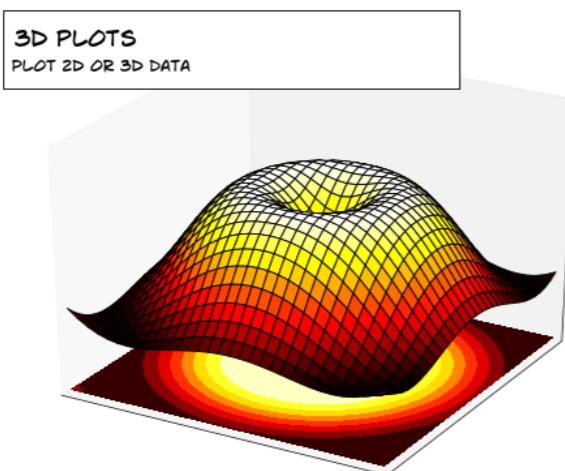
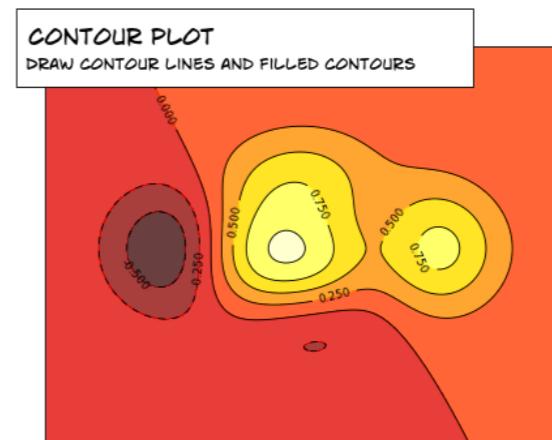
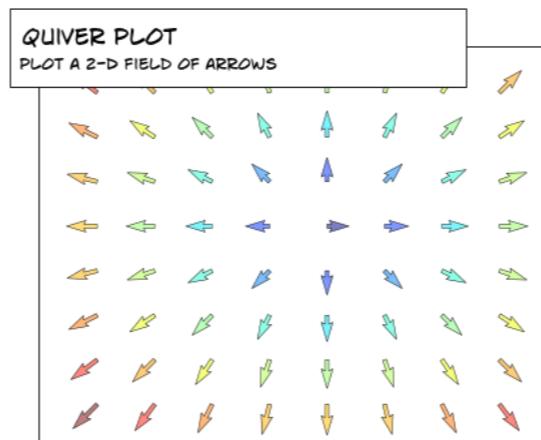
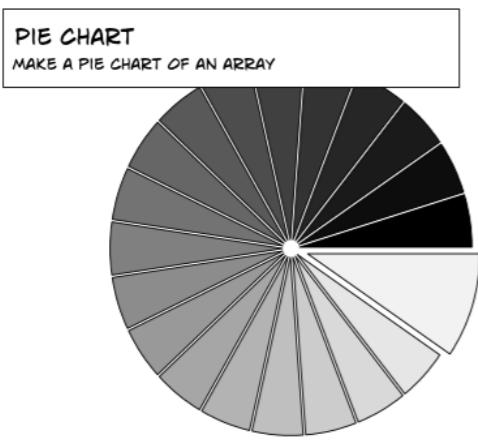
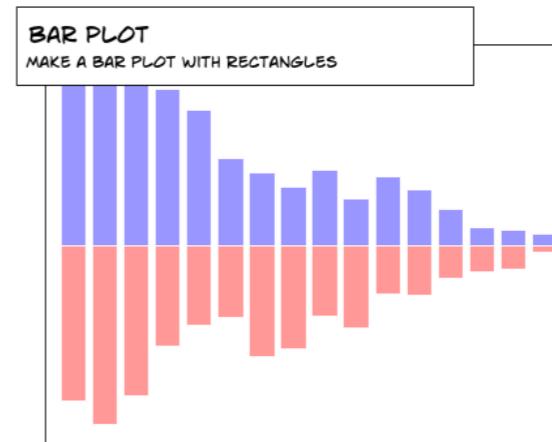
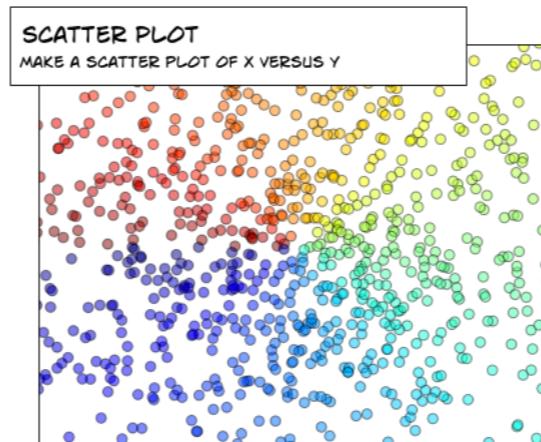
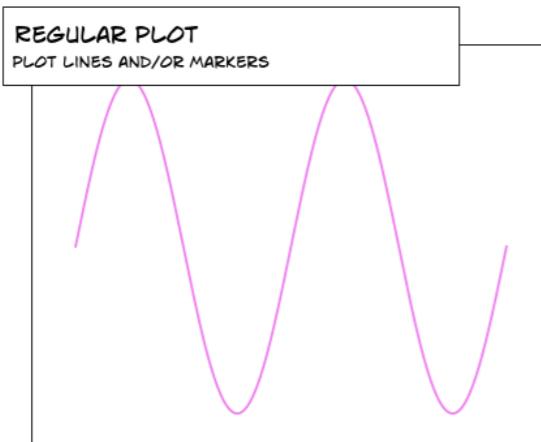
Documentation
This is the documentation for Matplotlib version 3.1.1.
To get started, read the [User's Guide](#).
Other versions are available:

- [3.1.1 Stable version](#).
- [2.2.4 LTS LTS version](#).
- [3.2.x Latest git master \(unstable\)](#)
- [3.0.3 Older stable version](#).

Trying to learn how to do a particular kind of plot? Check out the [examples gallery](#) or the [list of plotting commands](#).

Other learning resources
There are many [external learning resources](#) available including printed material, videos and tutorials.

There are many, many types of plots that one can make in Matplotlib



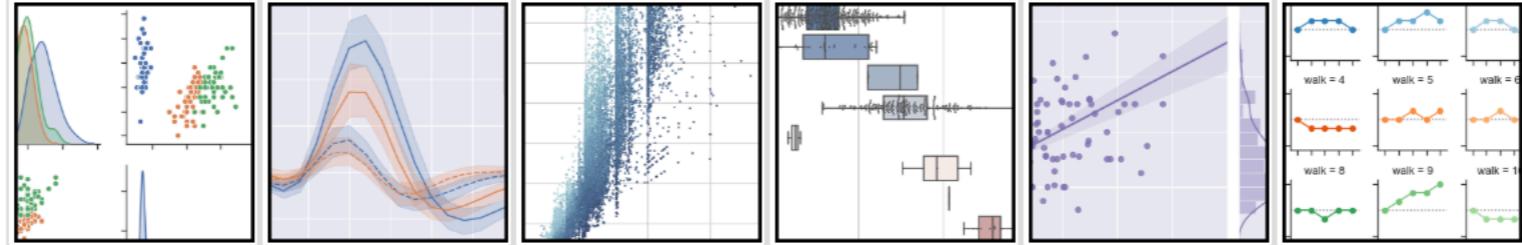
Seaborn facilitates rapid data visualization

Seaborn is a *wrapper* for Matplotlib.

It provides methods for rapidly generating a wide range of decent-looking plots from data stored in Pandas dataframes.

seaborn 0.9.0 [Gallery](#) [Tutorial](#) [API](#) [Site](#) ▾ [Page](#) ▾ [Search](#)

seaborn: statistical data visualization



Seaborn is a Python data visualization library based on [matplotlib](#). It provides a high-level interface for drawing attractive and informative statistical graphics.

For a brief introduction to the ideas behind the library, you can read the [introductory notes](#). Visit the [installation page](#) to see how you can download the package. You can browse the [example gallery](#) to see what you can do with seaborn, and then check out the [tutorial](#) and [API reference](#) to find out how.

To see the code or report a bug, please visit the [github repository](#). General support issues are most at home on [stackoverflow](#), where there is a seaborn tag.

Contents

- [Introduction](#)
- [Release notes](#)
- [Installing](#)
- [Example gallery](#)
- [Tutorial](#)
- [API reference](#)

Features

- Relational: [API](#) | [Tutorial](#)
- Categorical: [API](#) | [Tutorial](#)
- Distributions: [API](#) | [Tutorial](#)
- Regressions: [API](#) | [Tutorial](#)
- Multiples: [API](#) | [Tutorial](#)
- Style: [API](#) | [Tutorial](#)
- Color: [API](#) | [Tutorial](#)

The Python Graph Gallery provides detailed plotting instructions for beginners



THE PYTHON
GRAPH GALLERY

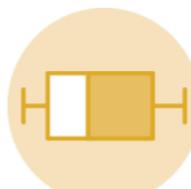
DISTRIBUTION



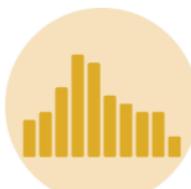
VIOLIN



DENSITY



BOXPLOT



HISTOGRAM

CORRELATION



Scatterplot



Connected
Scatter plot



Bubble plot



Heatmap



2D density plot

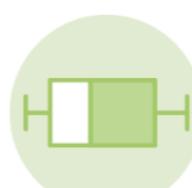


Correlogram

RANKING



Barplot



Boxplot



parallel plot



Lollipop plot



Wordcloud



Spider

PART OF A WHOLE



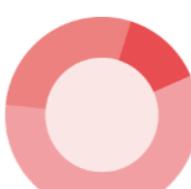
Stacked barplot



Tree plot



Venn diagram



Doughnut plot

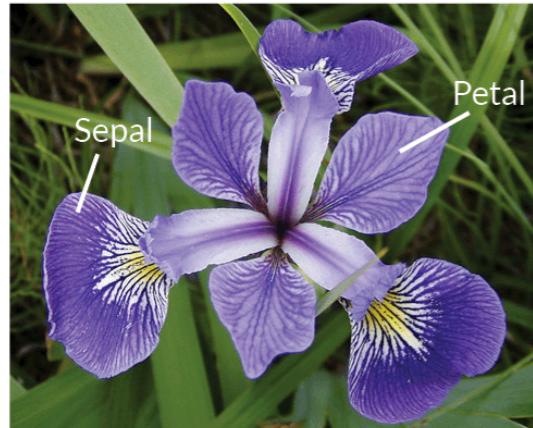


Pie plot



Tree diagram

Fisher's “Iris” dataset is a famous example dataset in statistics and dataviz



Iris Versicolor



Iris Setosa



Iris Virginica

R. A. Fisher (1936).
"The use of multiple measurements
in taxonomic problems".
Annals of Eugenics. 7 (2): 179–188.
(data collected by Edgar Anderson)

150 rows
(50 per species)

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
			:		

Matplotlib figures contain one or more Axes objects

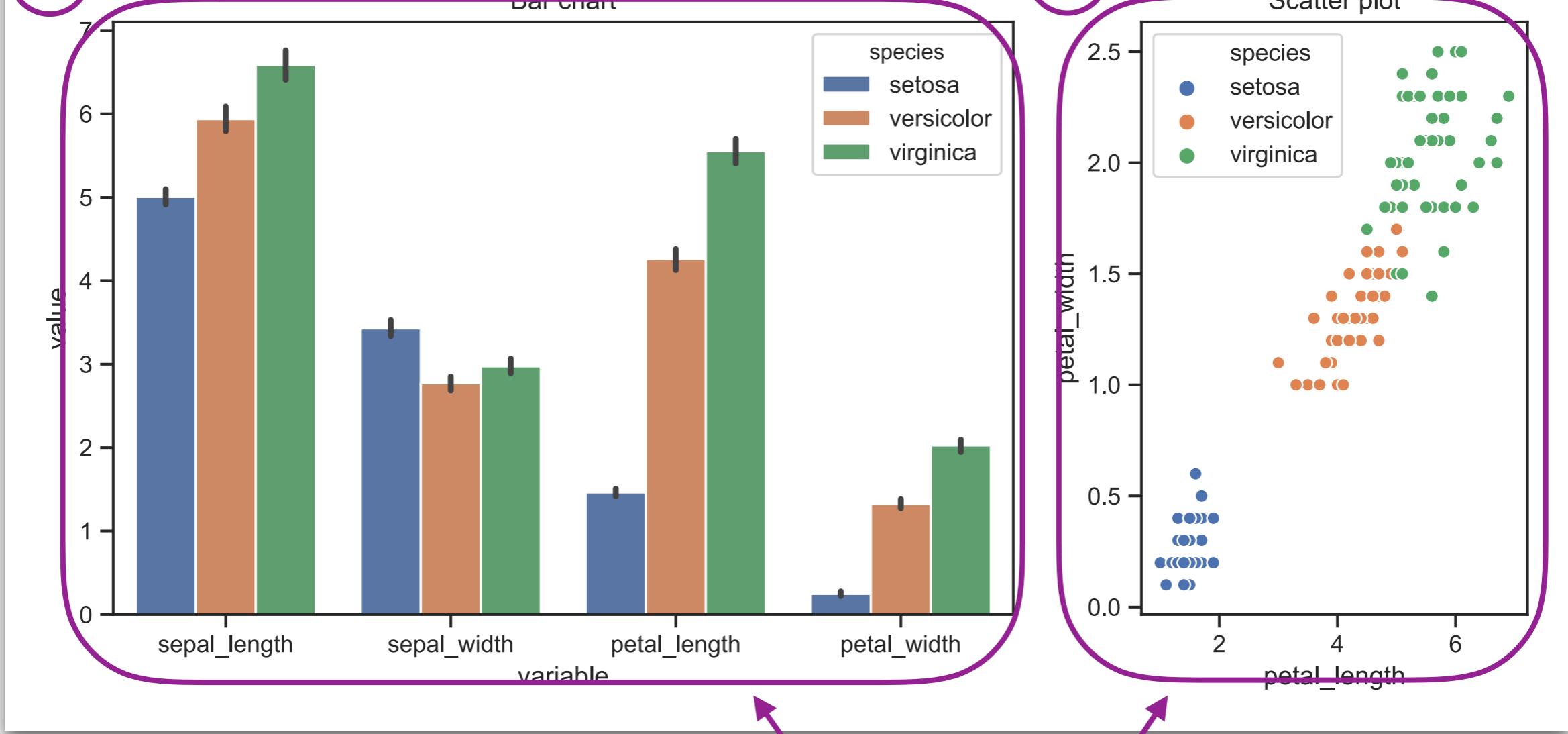
text

(A)

Figure object

text

(B)



Axes objects

Each Axes object contains many individual elements that can be adjusted

