INMAS

Data Visualization

- *Exploratory Data Analysis*
- *Graphing in Python*

James Balamuta
INMAS Statistical Methods Workshop Fall 2021

Lecture Objectives

- **Apply** quantitative and visual exploratory techniques on data
- **Deduce** and **interpret patterns** revealed during exploratory data analysis (EDA)
- **Create** common statistical graphs using Seaborn.

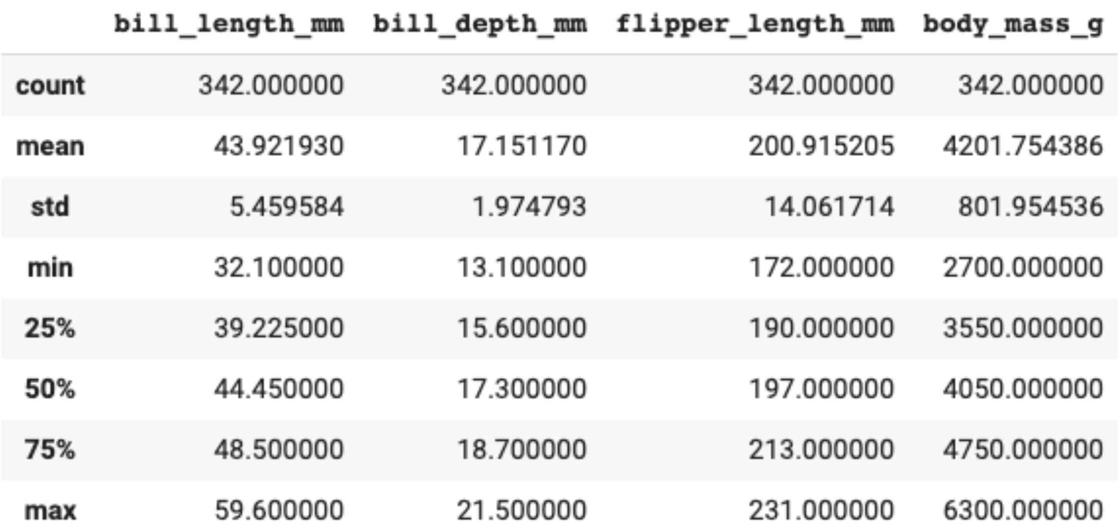
Exploratory Data Analysis

Definition:
Exploratory Data Analysis (EDA) is a philosophy for the beginning of an analysis that describes a variety of techniques that are *quantitative* and *visual* in nature to look for patterns in data.

Source

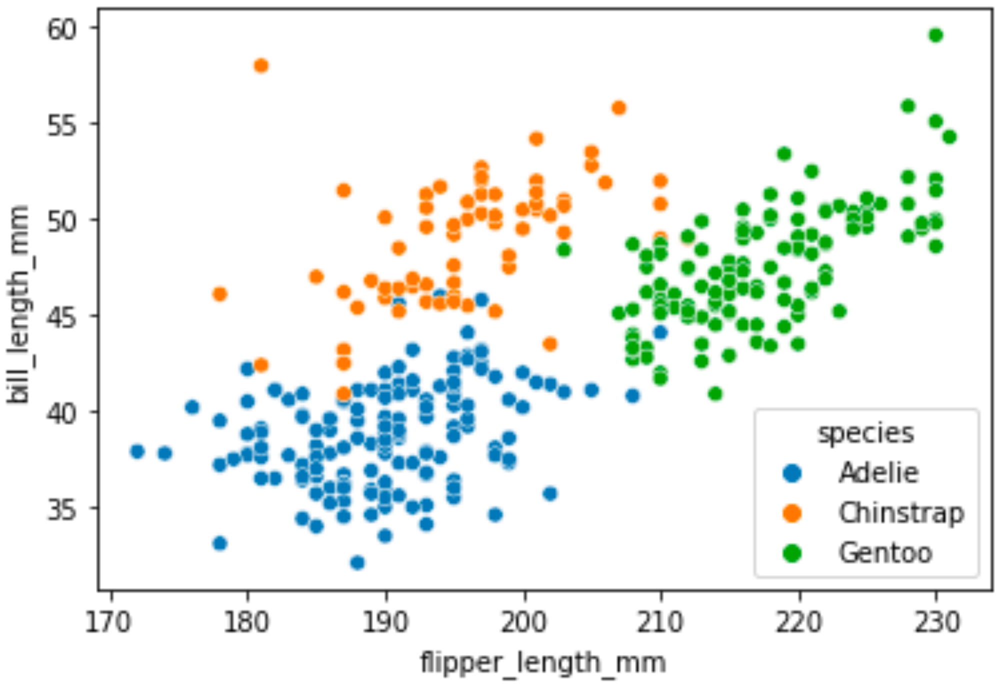
Types of EDA

Quantitative vs. Visual



	bill_length_mm	bill_depth_mm	flipper_length_mm	body_mass_g
count	342.000000	342.000000	342.000000	342.000000
mean	43.921930	17.151170	200.915205	4201.754386
std	5.459584	1.974793	14.061714	801.954536
min	32.100000	13.100000	172.000000	2700.000000
25%	39.225000	15.600000	190.000000	3550.000000
50%	44.450000	17.300000	197.000000	4050.000000
75%	48.500000	18.700000	213.000000	4750.000000
max	59.600000	21.500000	231.000000	6300.000000

```
# Obtain summary information
penguins.describe()
```



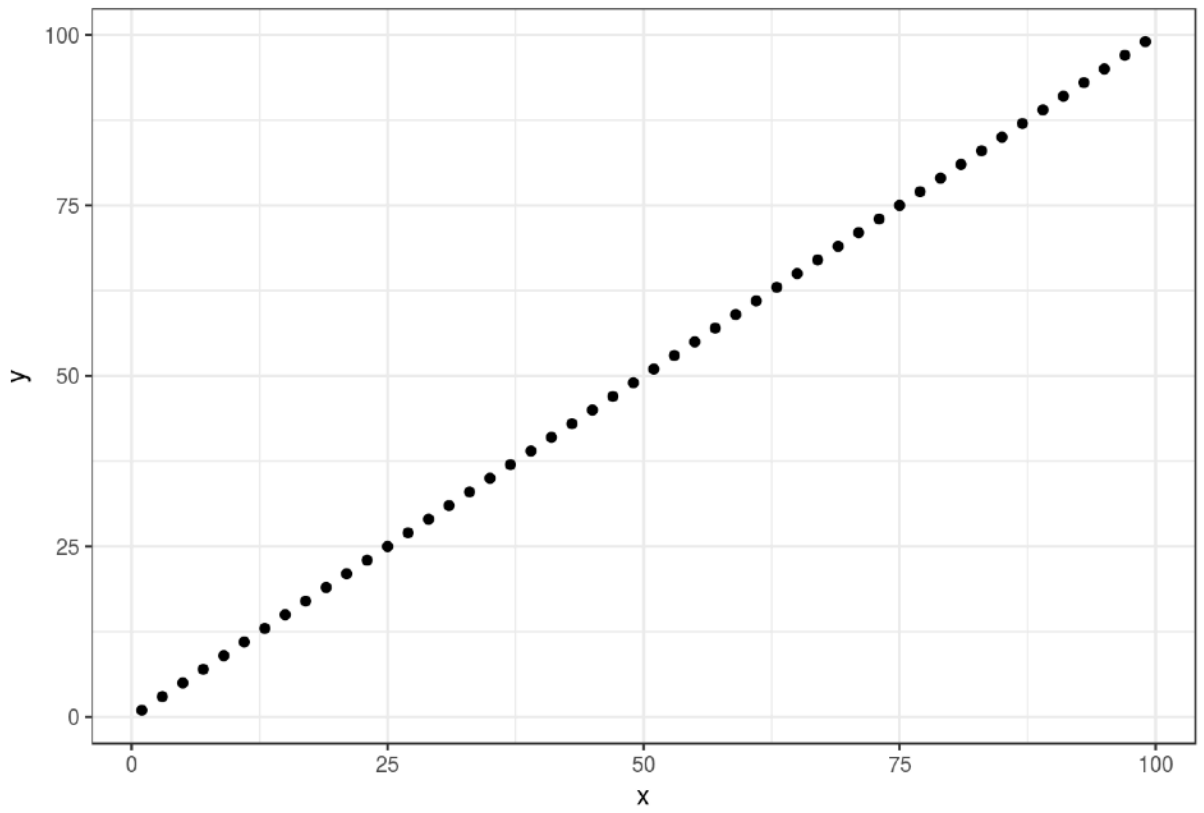
```
# Visualize information with
# a scatterplot
sns.scatterplot(
  data=penguins,
   x="flipper_length_mm",
   y="bill_length_mm",
   hue="species");
```

Patterns

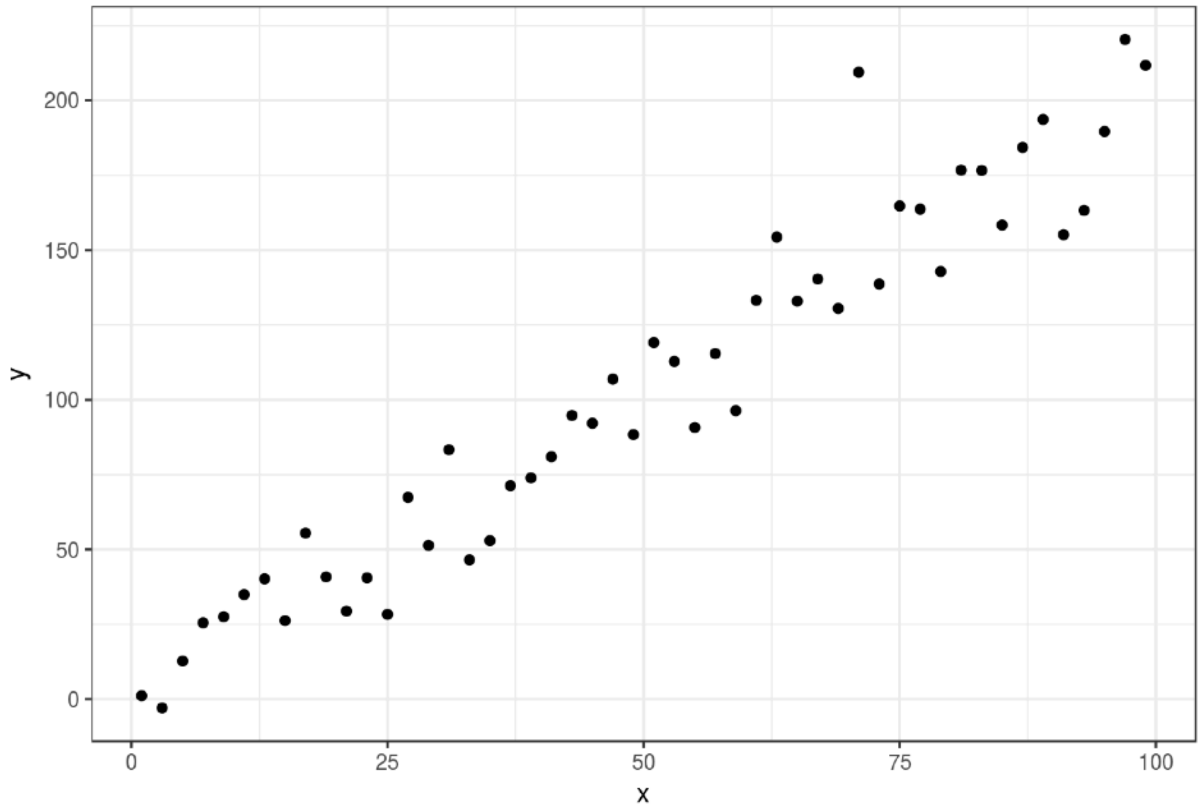
… detecting, analyzing, and communicating …

1. What kind of **relationship exists** is present?
2. What's the **level of strength** of the relationship?
3. Are there any **confounding variables** that might be behind it?
4. How does the **pattern exist in subsets** of the data?

Linear Relationship



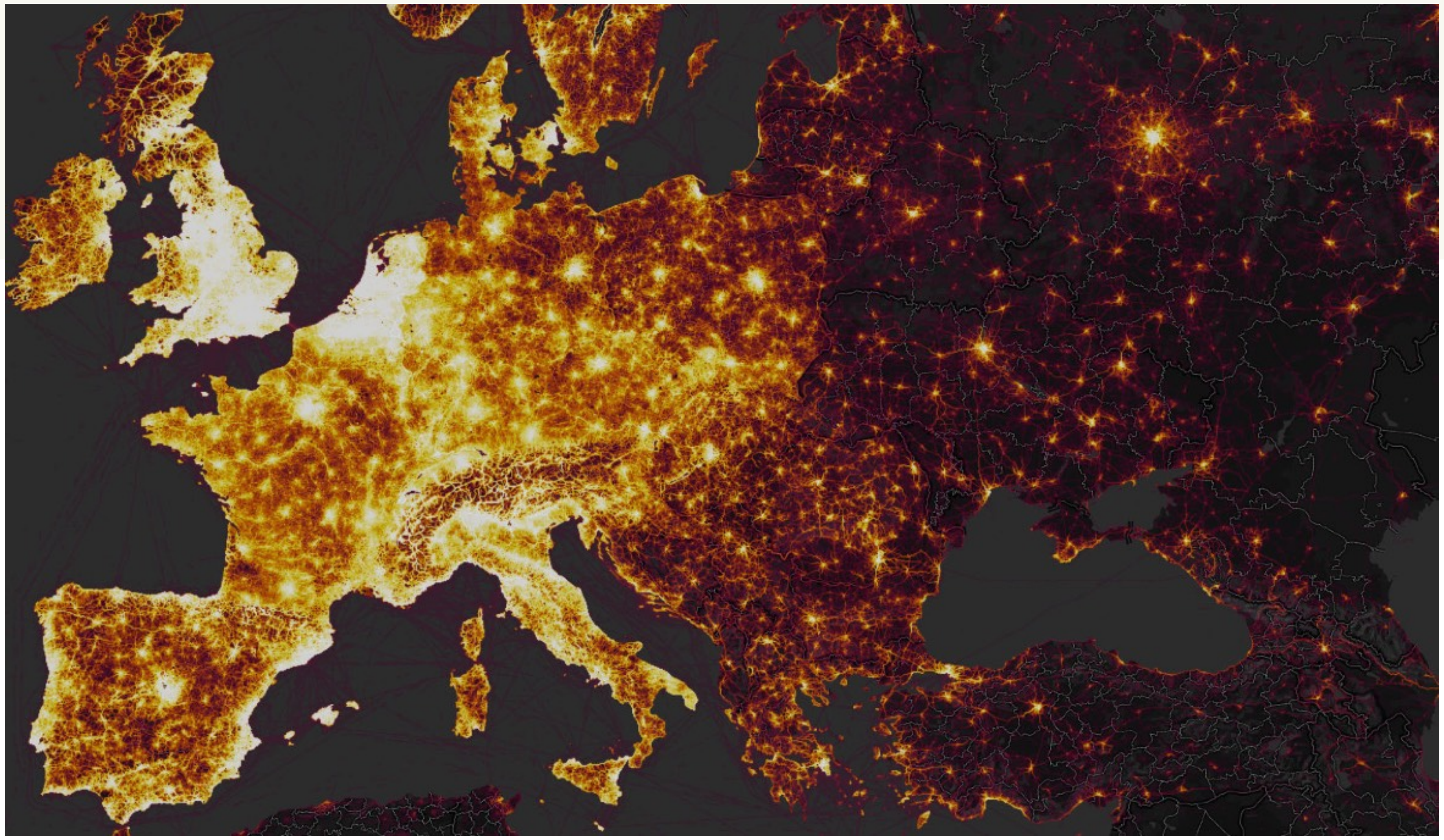
Fuzzed Linear Relationship



Definition:
Variation is the difference between observations in **one variable**.

Source

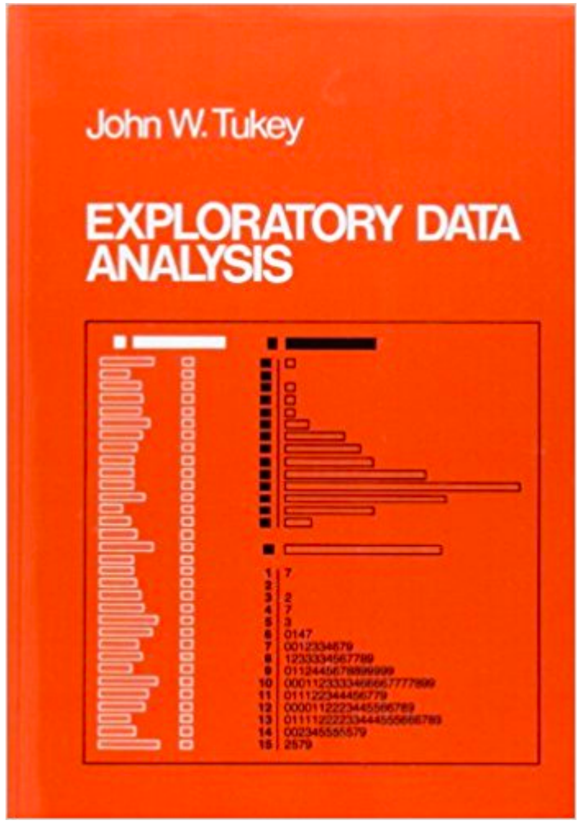
Definition:
Covariation is the difference among observations between **two or more** variables.



Source

“...make **both** calculations **and** graphs. Both sorts of output should be studied; each will contribute to understanding.”

– F. J. Anscombe, 1973

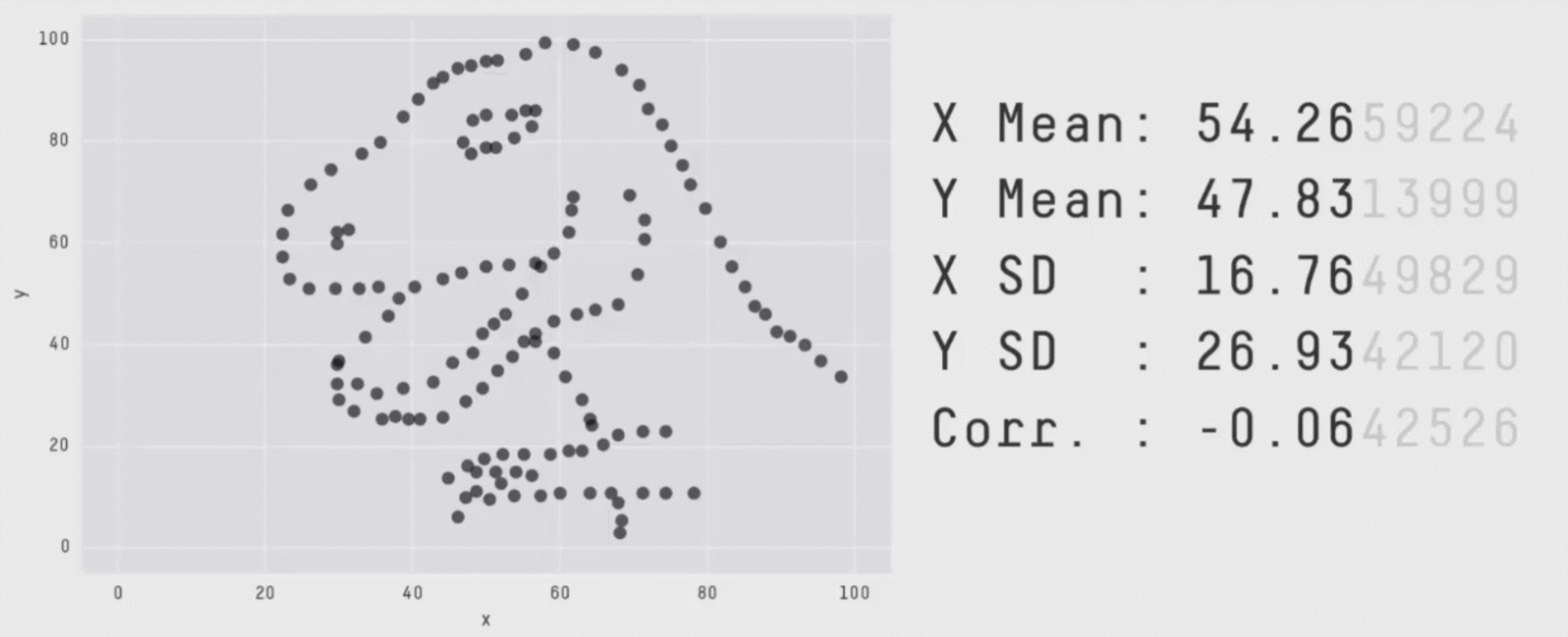


> “The greatest value of a picture is when it forces us to notice what we never expected to see.”

–John Turkey in Exploratory Data Analysis (1977)

Datasaurus

… the new anscombe's quartet …

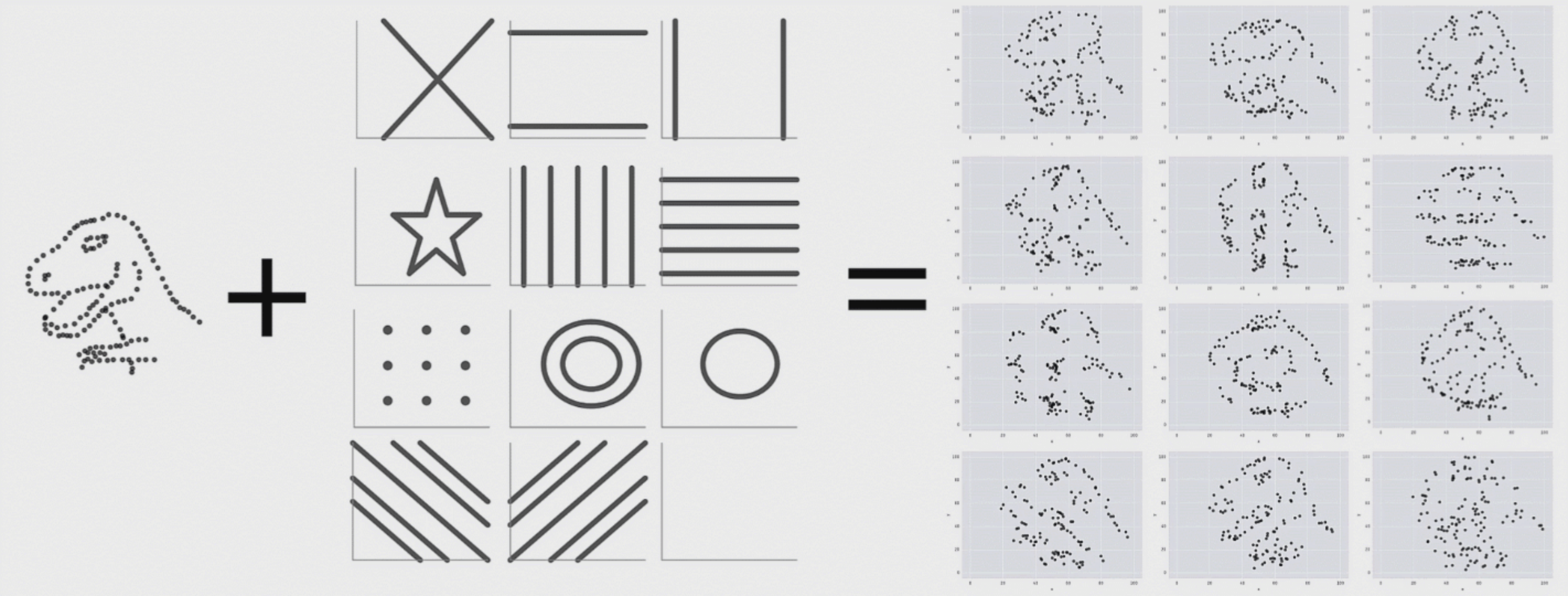


```
X Mean: 54.2659224
Y Mean: 47.8313999
X SD  : 16.7649829
Y SD  : 26.9342120
Corr. : -0.0642526
```

Source

Pattern

... the new anscombe's quartet ...

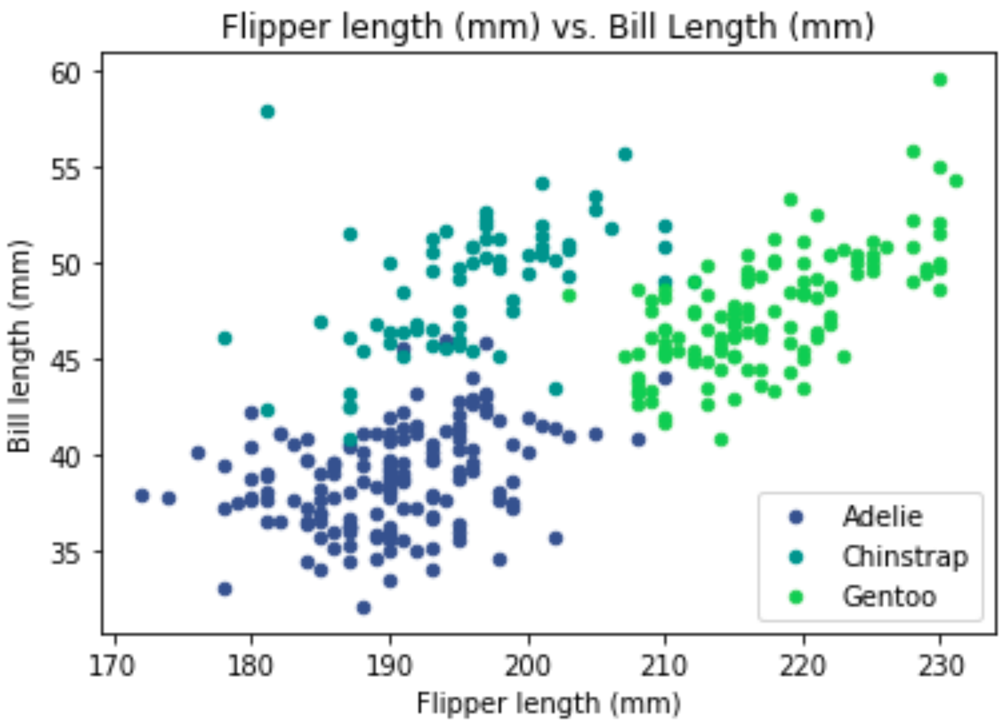


Source

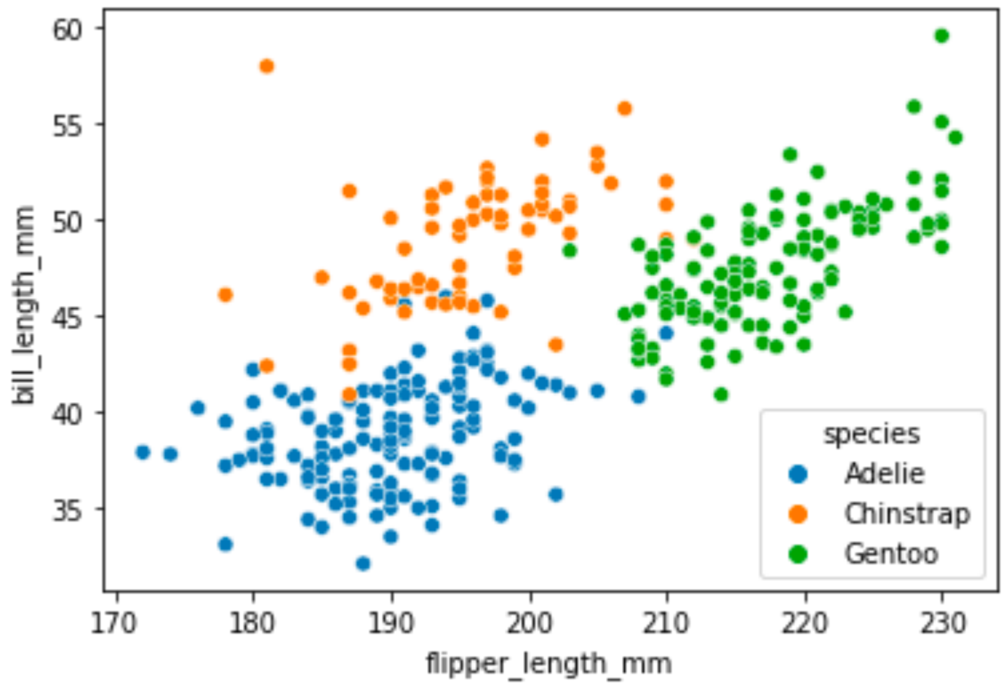
Graphing in Python

Python Graphing Systems

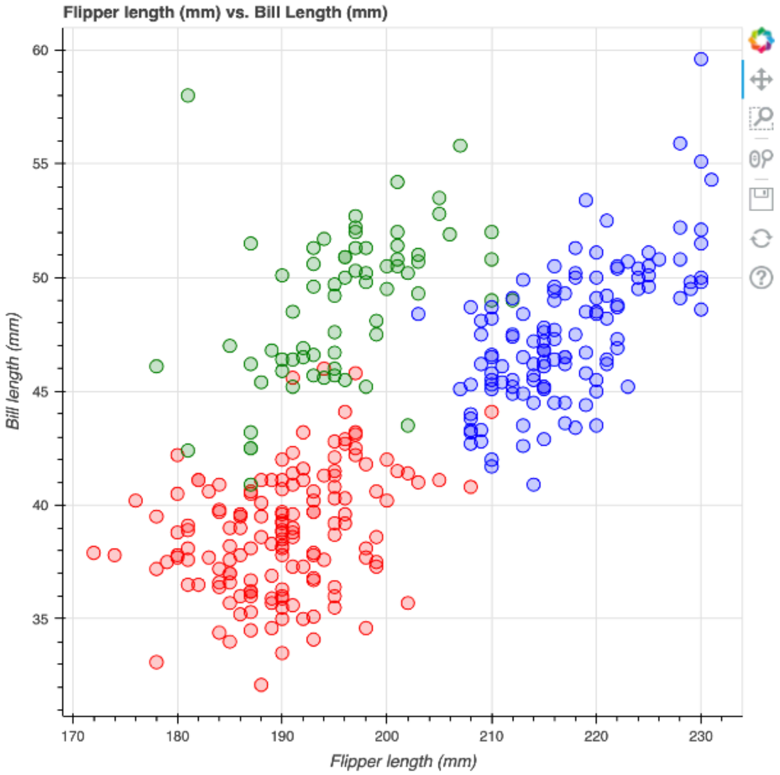
Comparison of Graphs on similar data



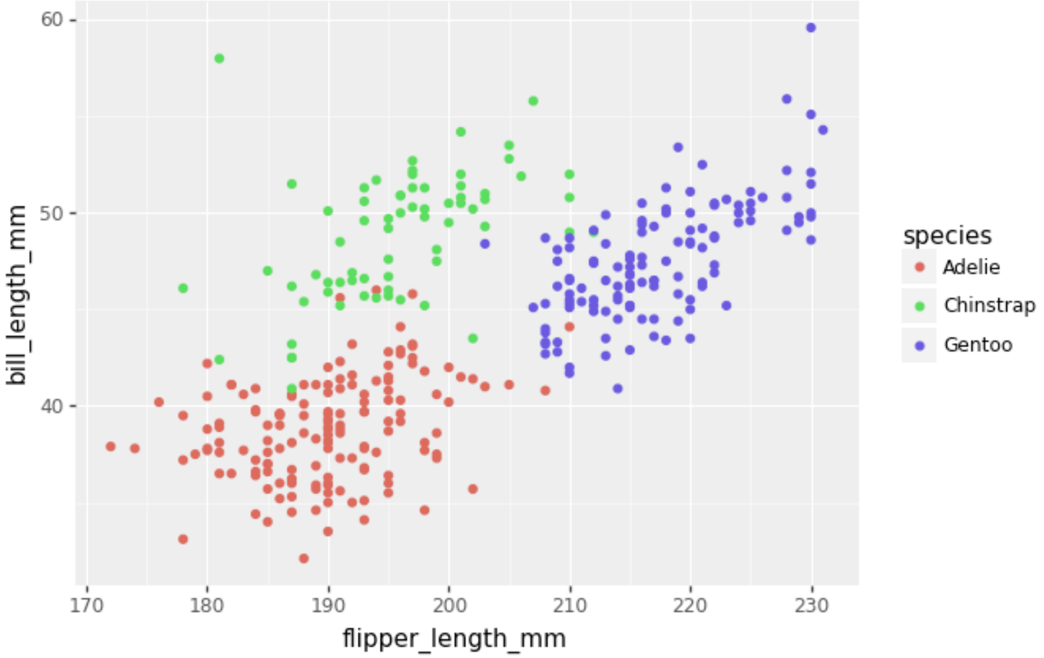
matplotlib



seaborn



Bokeh



plotnine

Goldilocks Scenario

... what graph system to choose ???

matplotlib

Highly
customizable

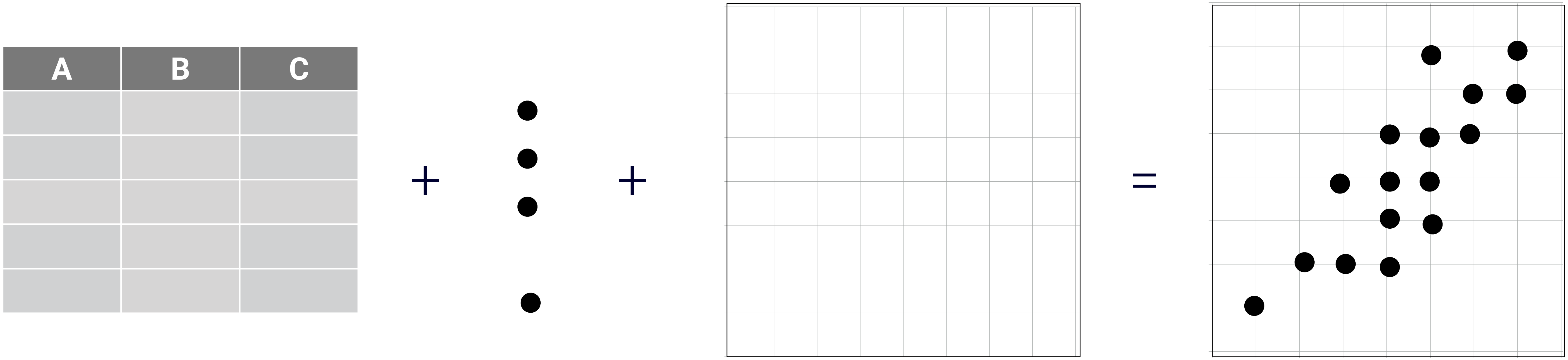
seaborn
bokeh

plotnine

Images Source

Grammar of a Graph

Underlying Structure of Graphs

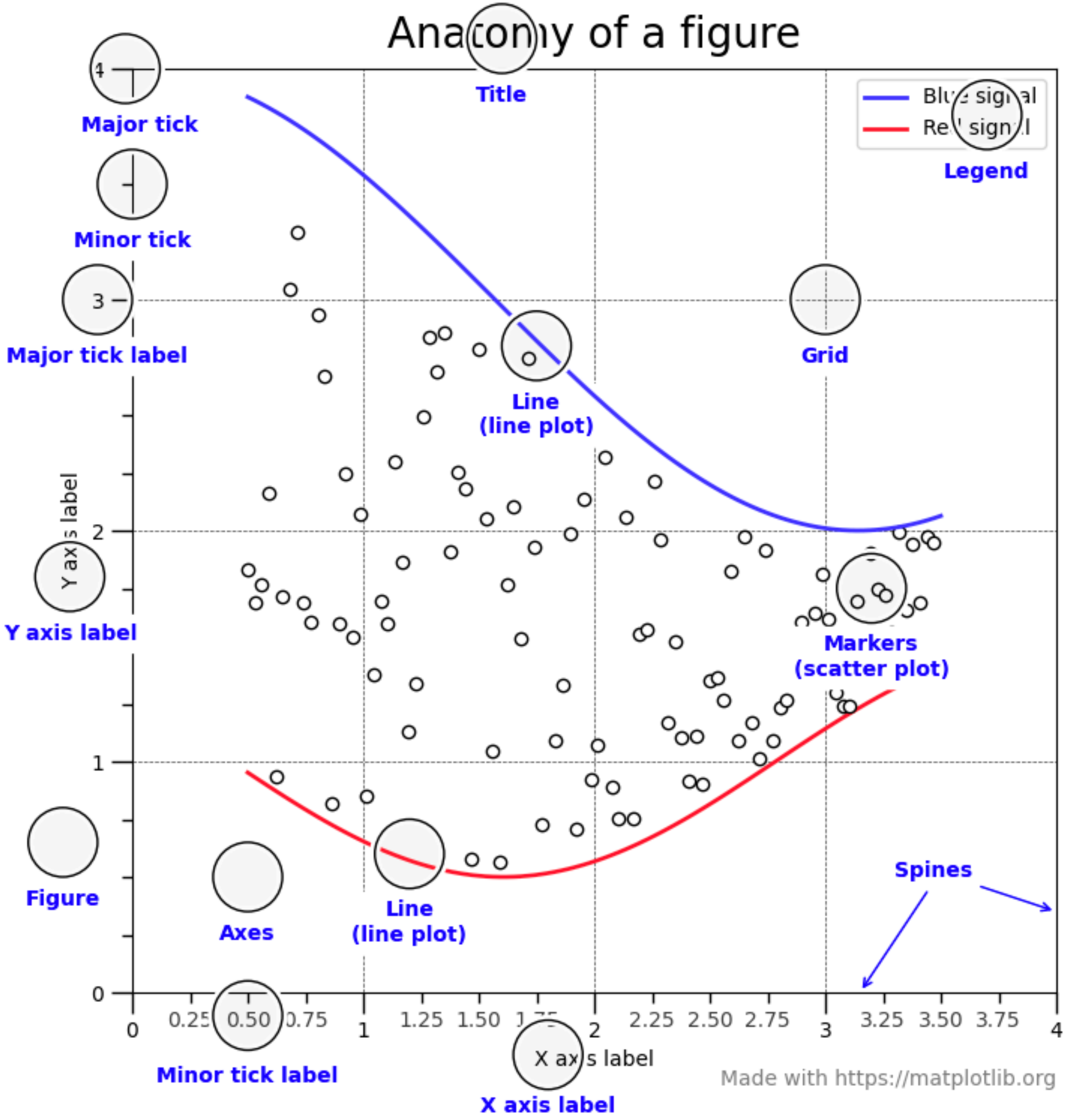


Data

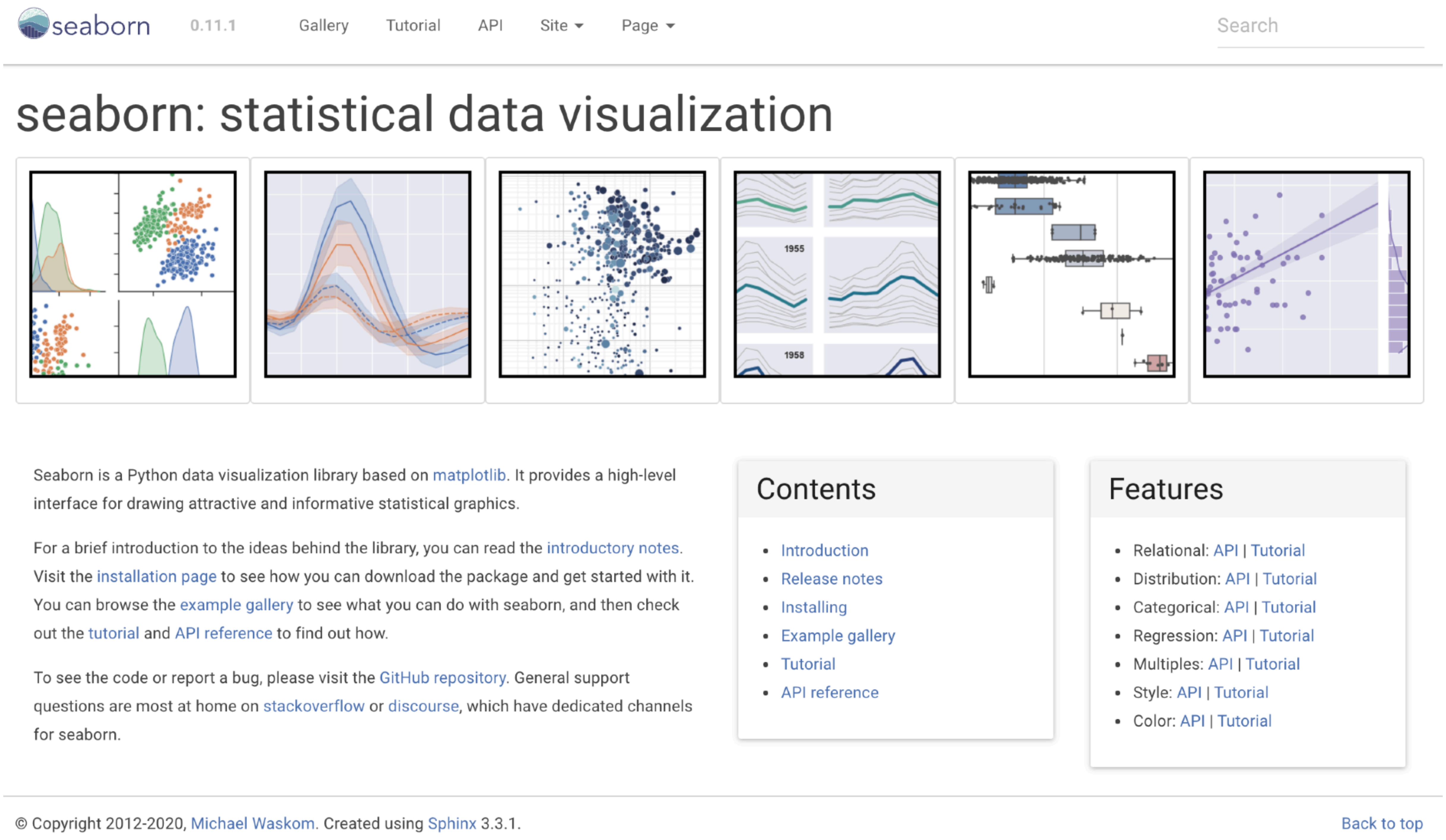
Geometric Object

Coordinate System

Plot


Anatomy of a figure
Title
Major tick
Minor tick
Major tick label
Y axis label
Figure
Axes
Line
(line plot)
Line
(line plot)
Grid
Legend
Blue signal
Red signal
Markers
(scatter plot)
Spines
Minor tick label
X axis label
Y axis label
X axis label
0
1
2
3
4
0
0.25
0.50
0.75
1
1.25
1.50
1.75
2
2.25
2.50
2.75
3
3.25
3.50
3.75
4
Made with https://matplotlib.org

Source



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 and get started with it. 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 questions are most at home on stackoverflow or discourse, which have dedicated channels for seaborn.

Contents

- Introduction
- Release notes
- Installing
- Example gallery
- Tutorial
- API reference

Features

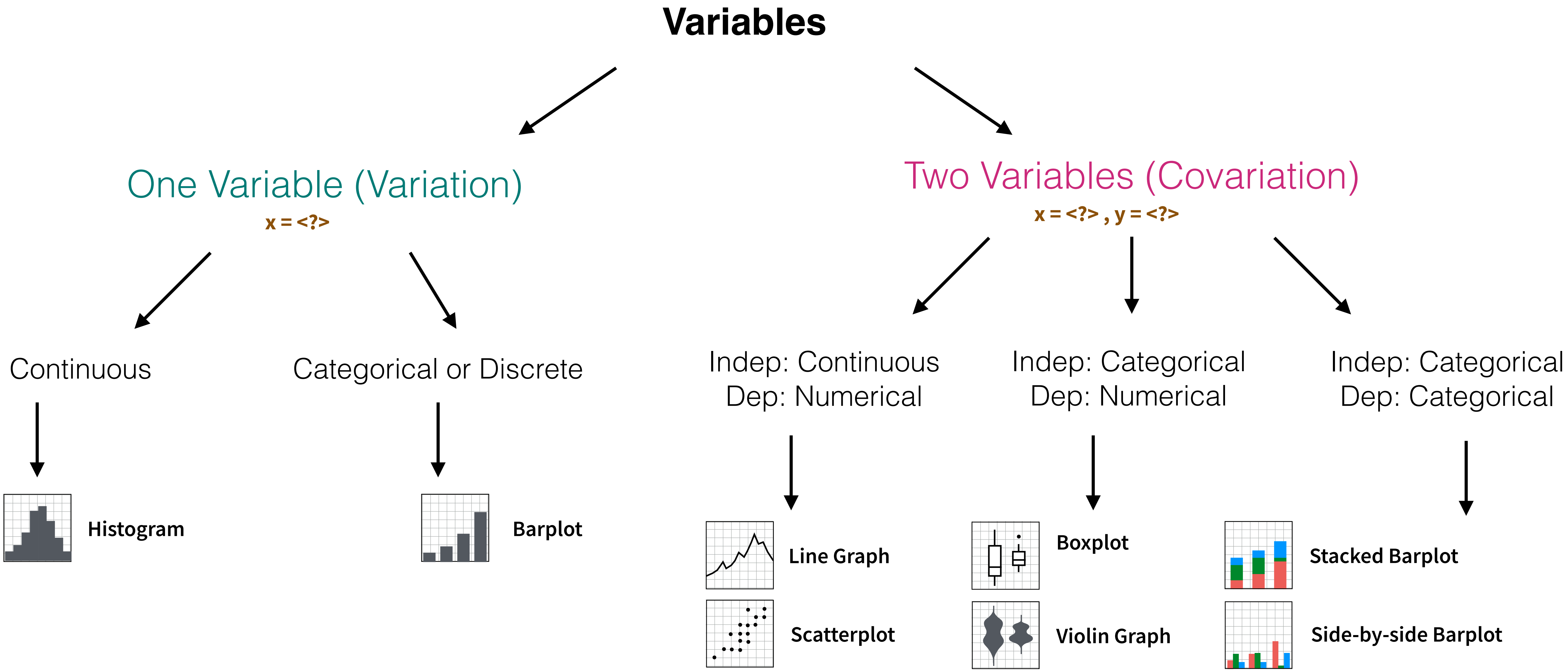
- Relational: API | Tutorial
- Distribution: API | Tutorial
- Categorical: API | Tutorial
- Regression: API | Tutorial
- Multiples: API | Tutorial
- Style: API | Tutorial
- Color: API | Tutorial

Back to top

https://seaborn.pydata.org/

Graphing Outline

... when to use a graph ...



Variables

One Variable (Variation)

x = <?>

Continuous

Histogram

Categorical or Discrete

Barplot

Two Variables (Covariation)

x = <?> , y = <?>

Indep: Continuous
Dep: Numerical

Line Graph

Scatterplot

Indep: Categorical
Dep: Numerical

Boxplot

Violin Graph

Indep: Categorical
Dep: Categorical

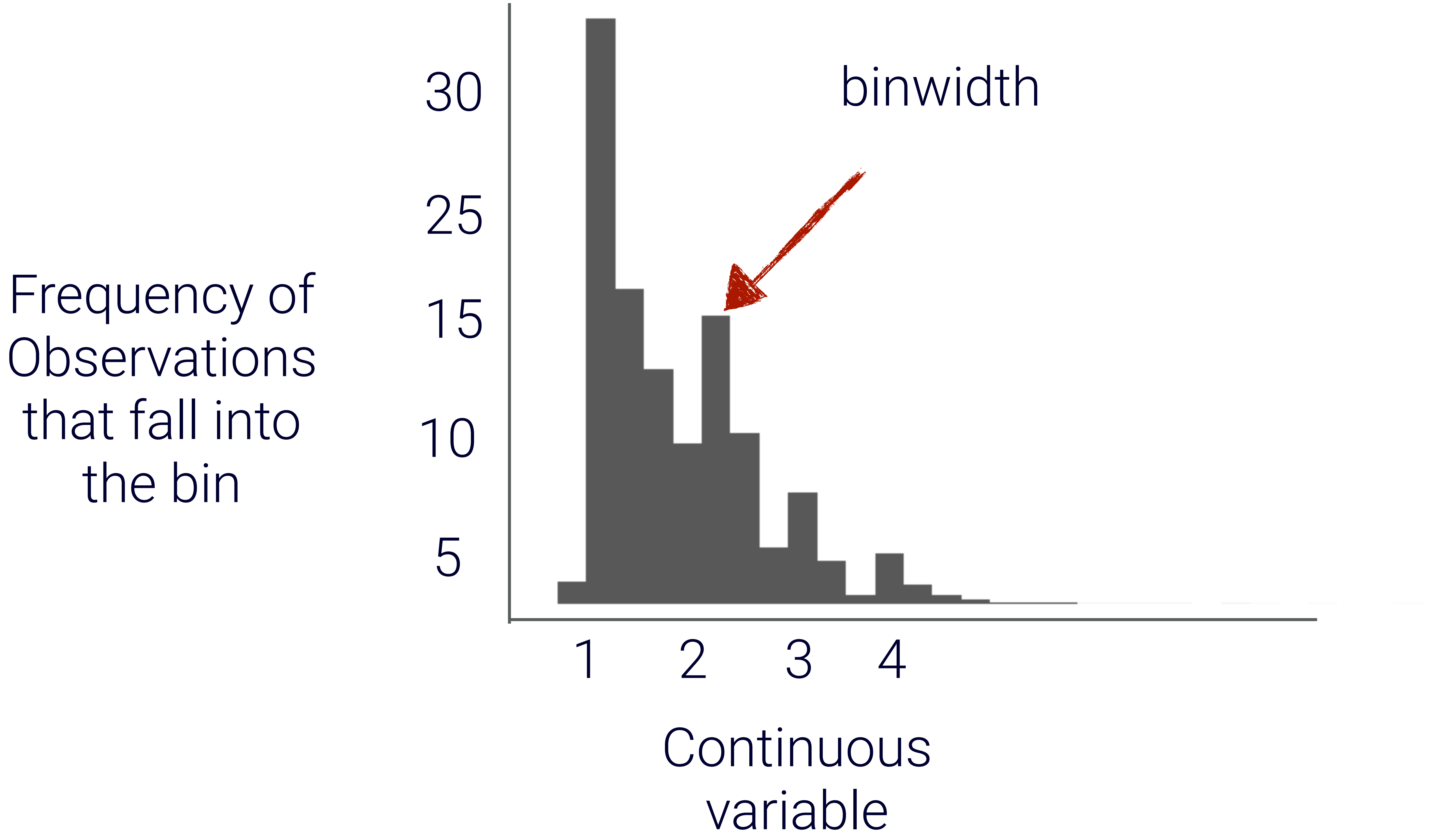
Stacked Barplot

Side-by-side Barplot

Independent
Dependent

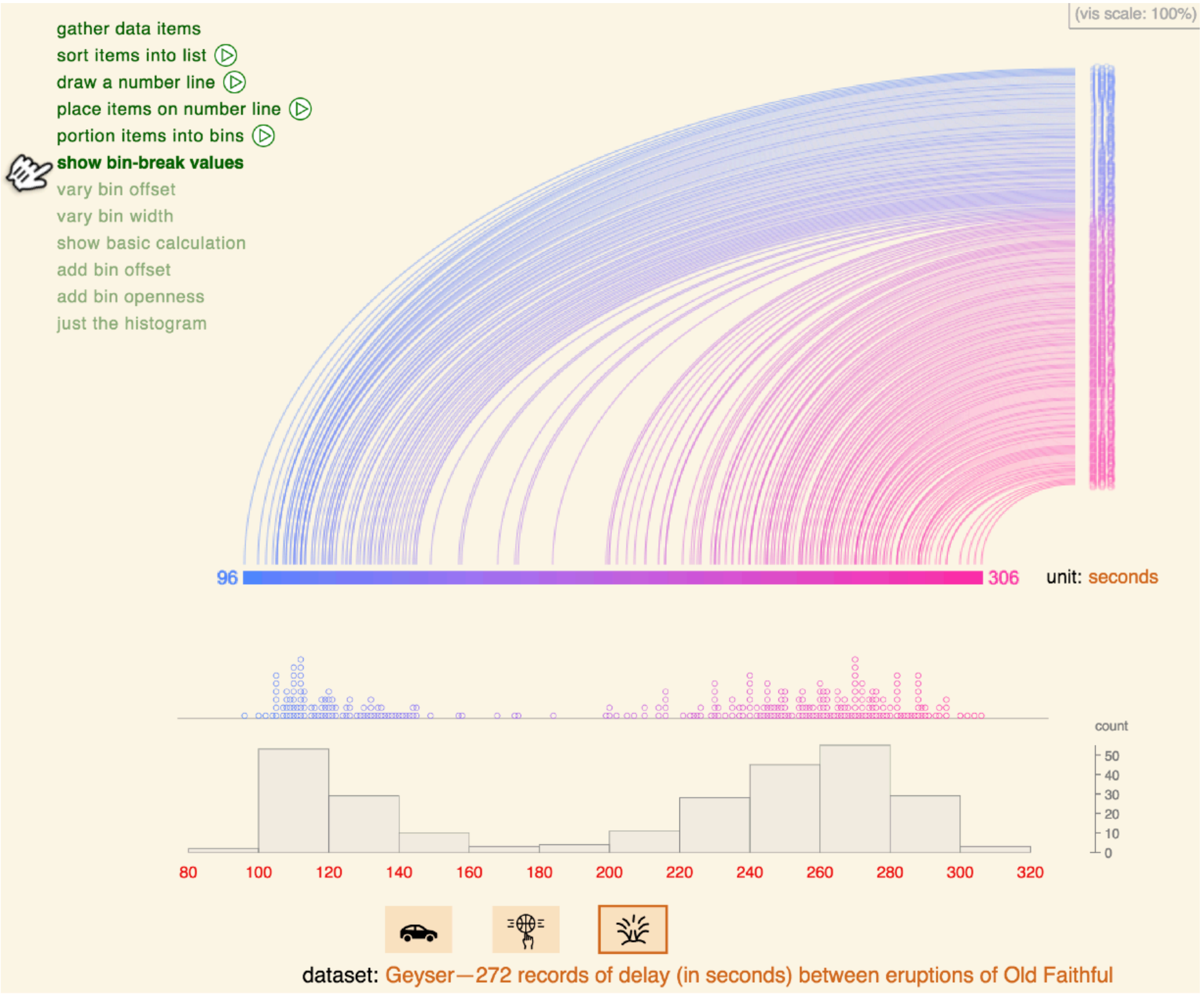
Histogram

Continuous data for one variable



Histograms

imagined "bins" for continuous data

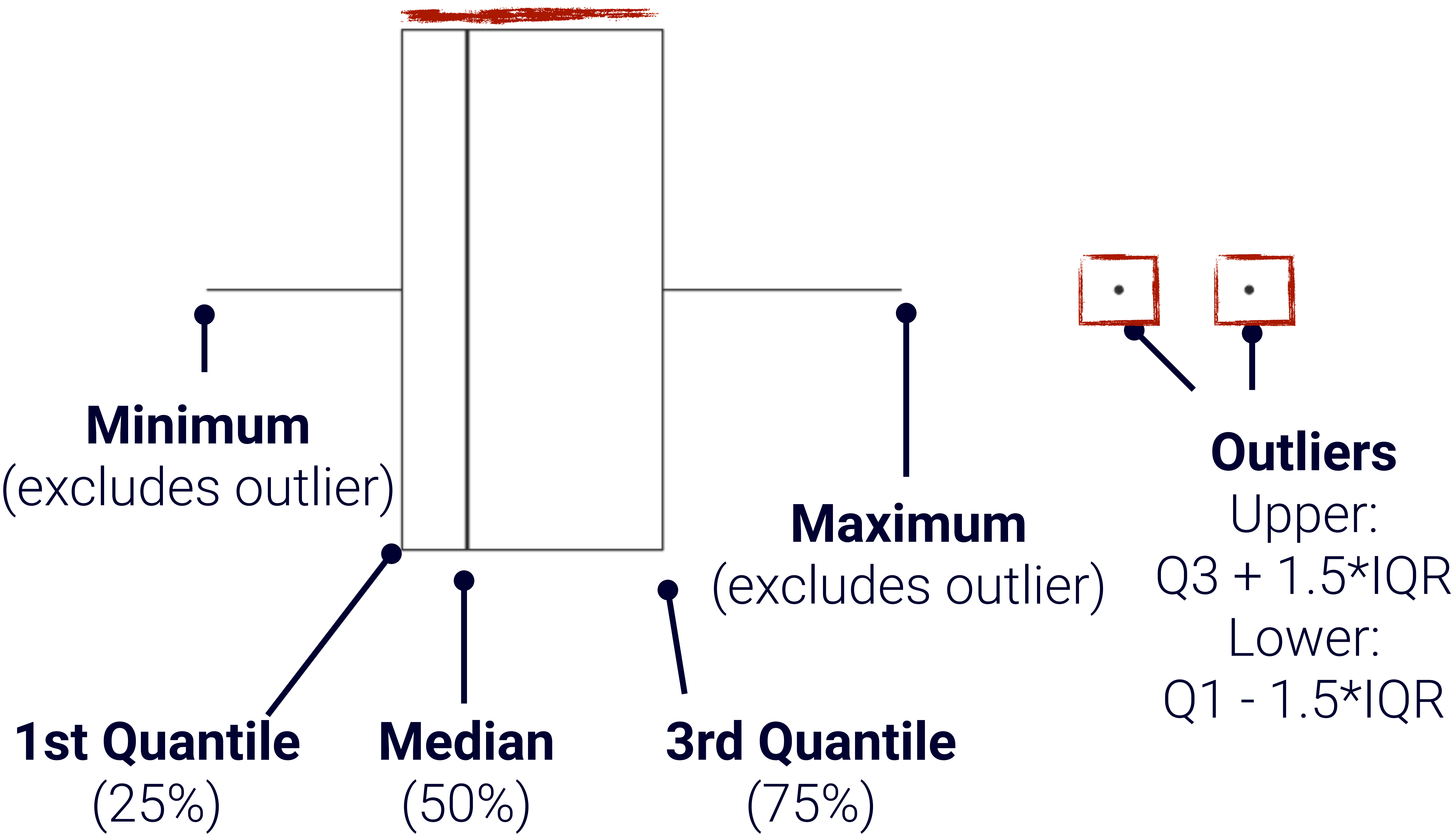


dataset: Geyser—272 records of delay (in seconds) between eruptions of Old Faithful

https://tinlizzie.org/histograms/

Boxplot

Categorical paired with **numerical** to observe *covariation*



Interquartile range (**IQR** = Q3 - Q1)

Minimum
(excludes outlier)

Maximum
(excludes outlier)

Outliers
Upper:
Q3 + 1.5*IQR
Lower:
Q1 - 1.5*IQR

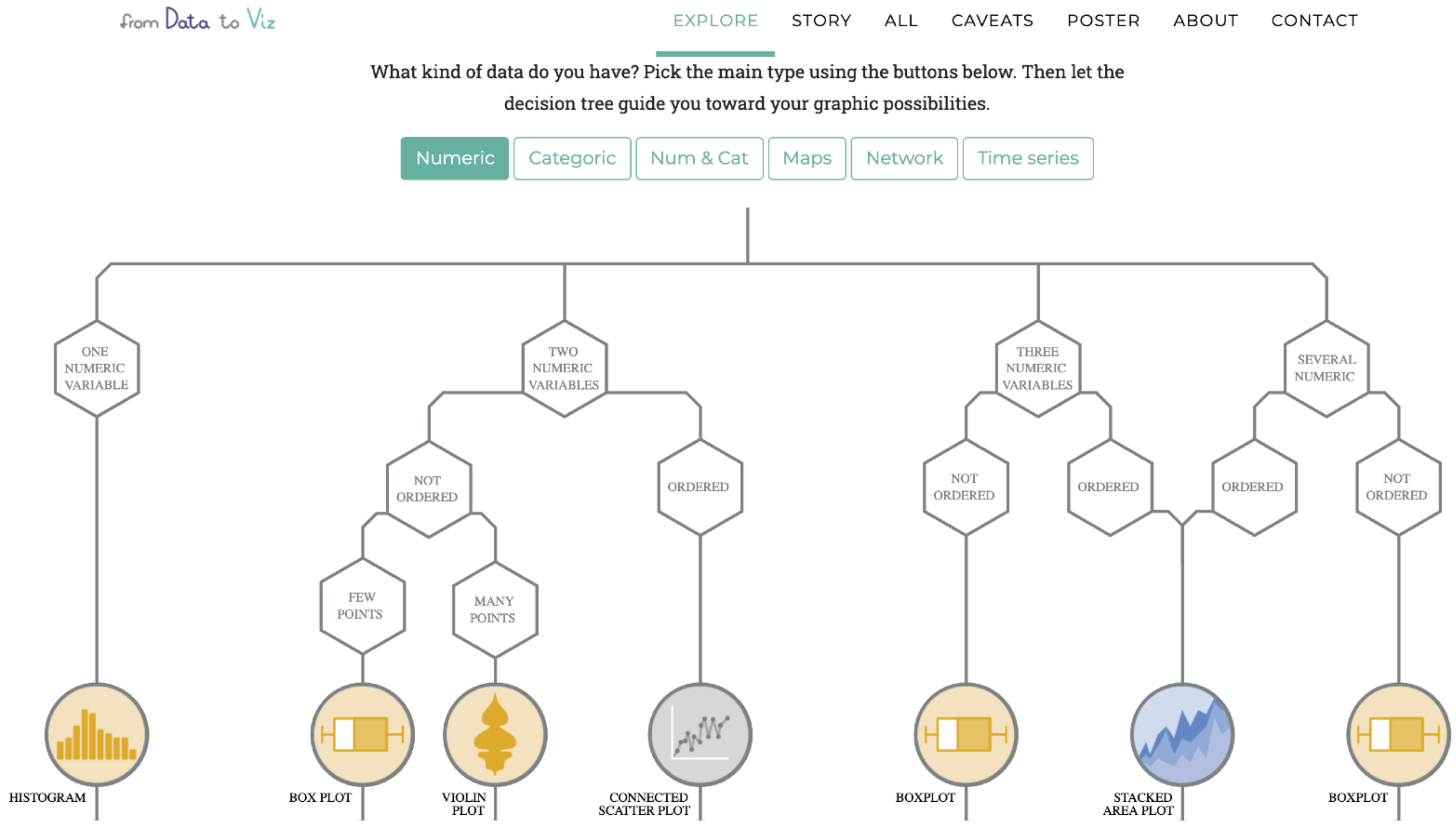
1st Quantile
(25%)

Median
(50%)

3rd Quantile
(75%)

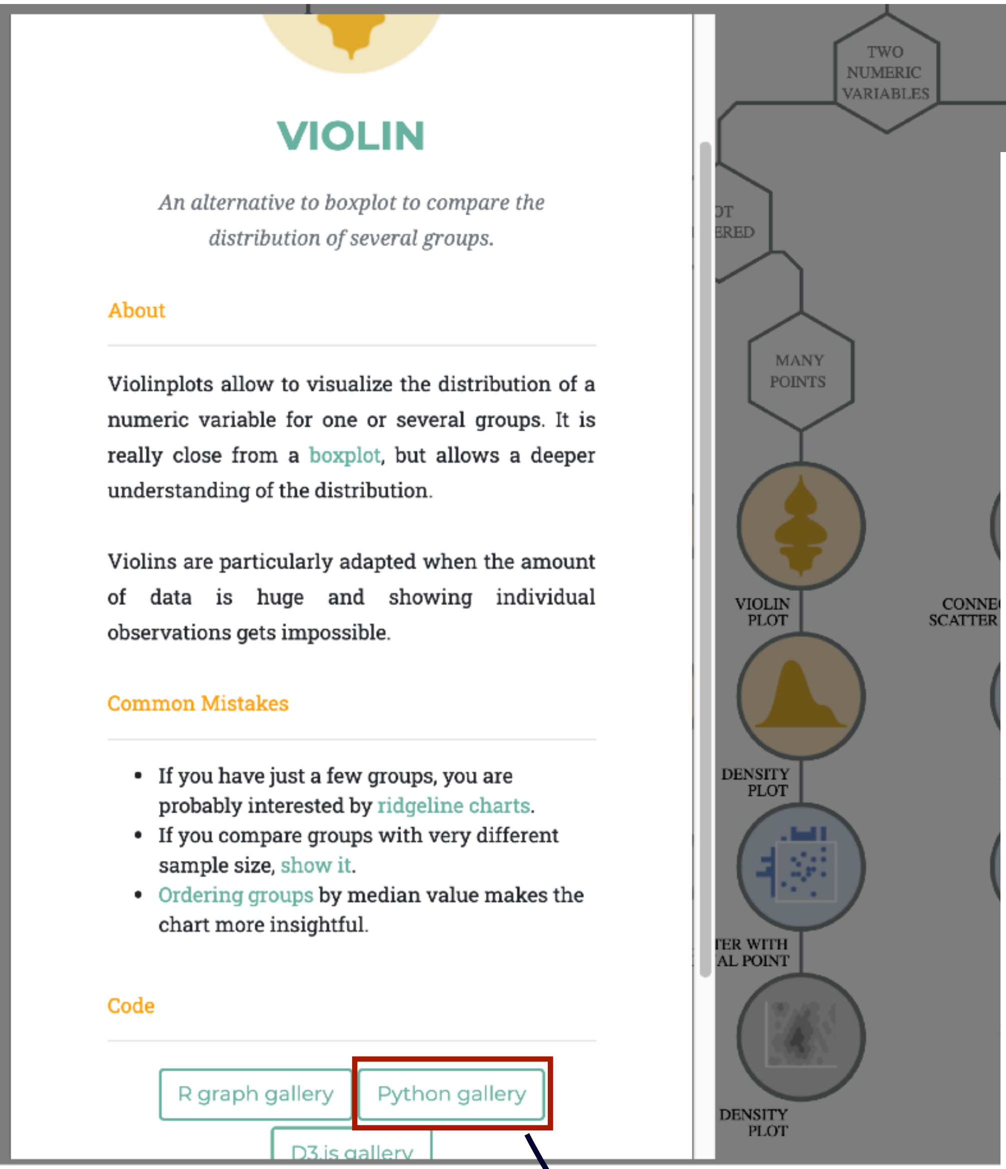
data-to-viz.com

Flowchart for picking the best visualization given data



What kind of data do you have? Pick the main type using the buttons below. Then let the decision tree guide you toward your graphic possibilities.

Code Examples



VIOLIN

An alternative to boxplot to compare the distribution of several groups.

About

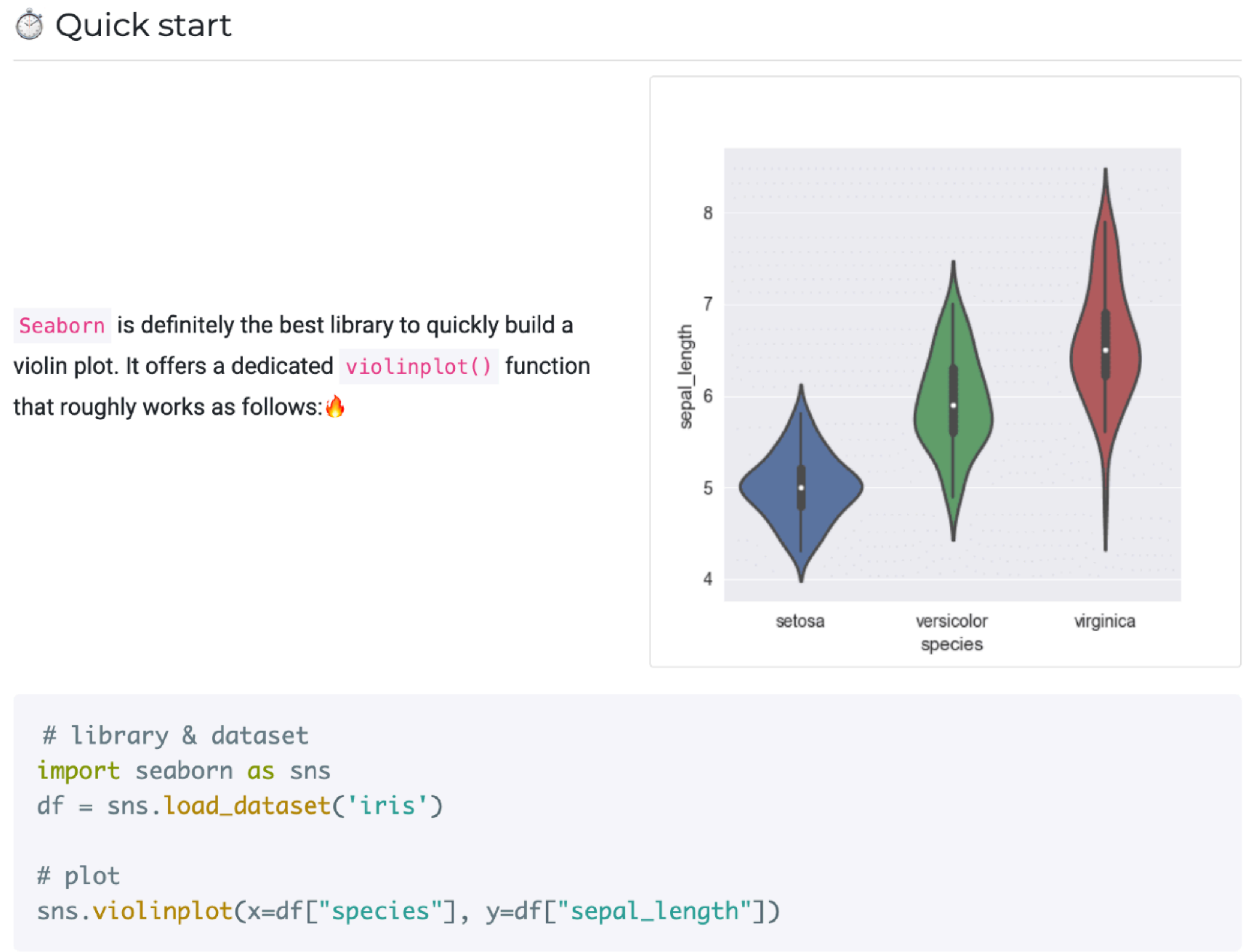
Violinplots allow to visualize the distribution of a numeric variable for one or several groups. It is really close from a boxplot, but allows a deeper understanding of the distribution.

Violins are particularly adapted when the amount of data is huge and showing individual observations gets impossible.

Common Mistakes

- If you have just a few groups, you are probably interested by ridgeline charts.
- If you compare groups with very different sample size, show it.
- Ordering groups by median value makes the chart more insightful.

Code



⏱ Quick start

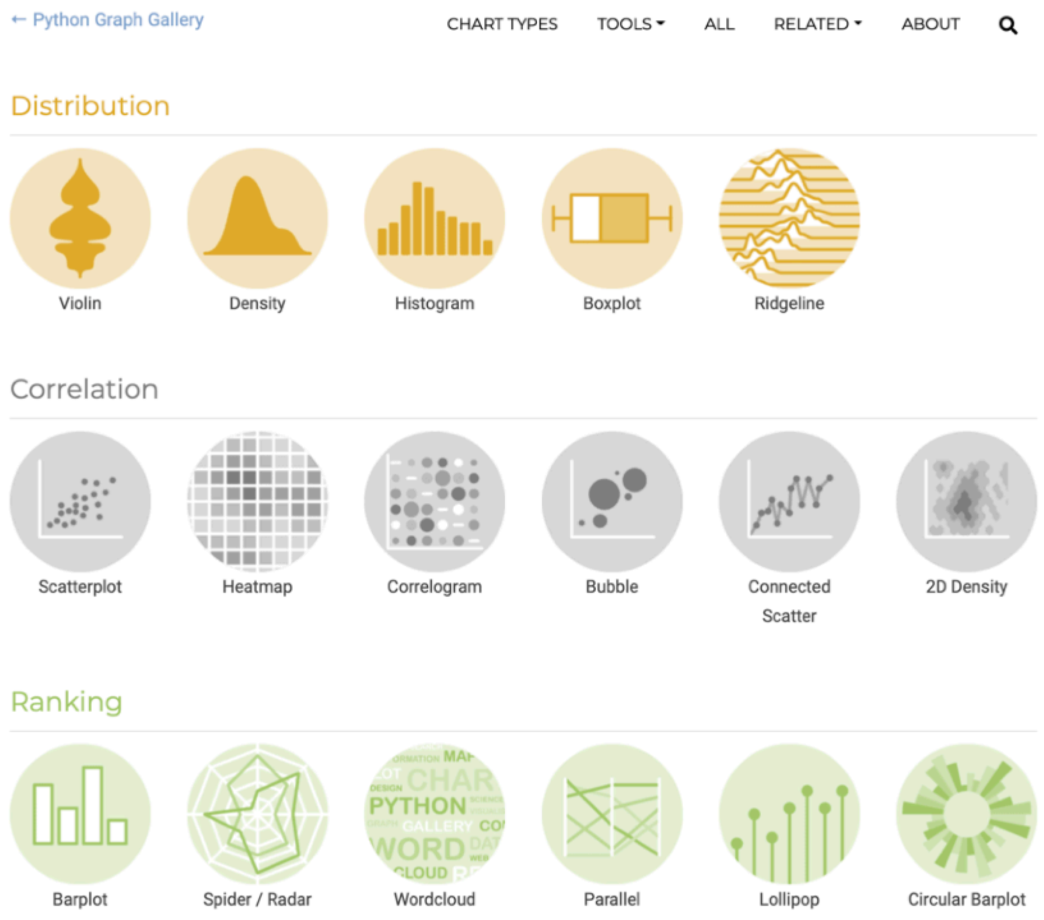
Seaborn is definitely the best library to quickly build a violin plot. It offers a dedicated violinplot() function that roughly works as follows:🔥

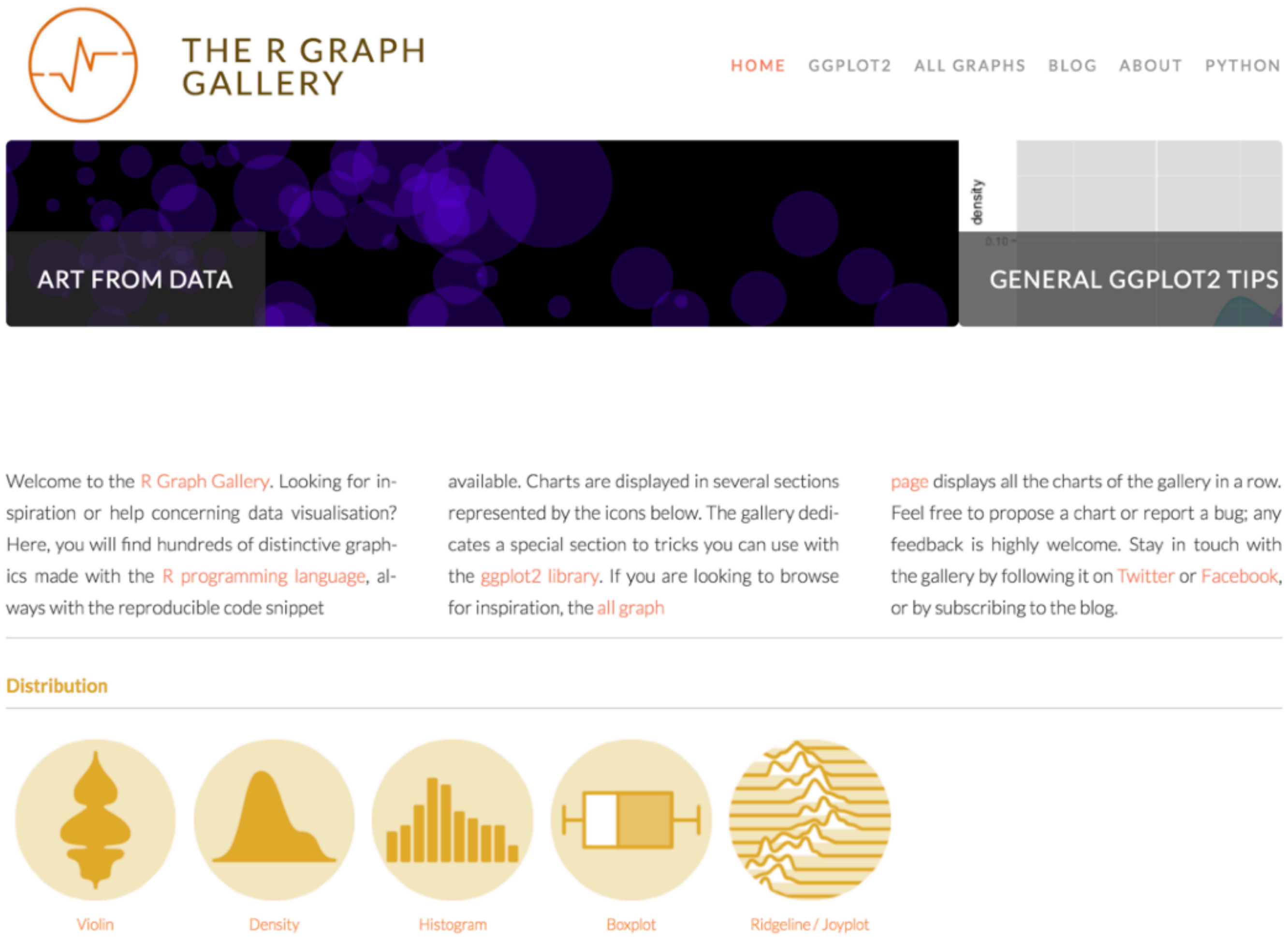
```
# library & dataset
import seaborn as sns
df = sns.load_dataset('iris')

# plot
sns.violinplot(x=df["species"], y=df["sepal_length"])
```

Sample Implementations

Code to Generate Various Graphs in Python and R





Welcome to the R Graph Gallery. Looking for inspiration or help concerning data visualisation? Here, you will find hundreds of distinctive graphics made with the R programming language, always with the reproducible code snippet available. Charts are displayed in several sections represented by the icons below. The gallery dedicates a special section to tricks you can use with the ggplot2 library. If you are looking to browse for inspiration, the all graph page displays all the charts of the gallery in a row. Feel free to propose a chart or report a bug; any feedback is highly welcome. Stay in touch with the gallery by following it on Twitter or Facebook, or by subscribing to the blog.

Source Source

Summary

- Emphasized the benefits of graphing data.
- Discussed different approaches to graphing data with one variable vs. two variables.