

Applied Data Science with Python



Data Visualization



Learning Objectives

By the end of this lesson, you will be able to:

- 👁 Create data visualizations using Pivot Tables
- 👁 Use the data visualization libraries in Python
- 👁 Construct a graph using Matplotlib, Seaborn, Plotly, and Bokeh



Business Scenario

ABC Corporation is a retail company which is experiencing a decline in sales and intends to use data science to explore its sales data to identify the root causes of the decline. The analysis will include data cleaning, exploratory data analysis, feature engineering and modeling. The organization plans to use visualization techniques to communicate the results effectively.

By gaining insights into the data, the company hopes to optimize its sales strategies, enhance customer experience, and increase revenue. By utilizing data science techniques, the company aims to gain a competitive advantage in the retail industry across multiple regions.





Principles of Data Visualization

Discussion: Data Visualization

Duration: 10 minutes



- What is data visualization?
- What are the different categories of information visualization principles?

Introduction

It refers to the visual representation of data using graphs, charts, or other visual elements to extract insights and understand complex patterns and relationships.

It is a crucial part of data analytics and data science.



It is recommended to follow Edward Tufte's principles of data visualization.

Introduction

Edward Tufte's most famous work on visualization was published in 1983 in the book titled:

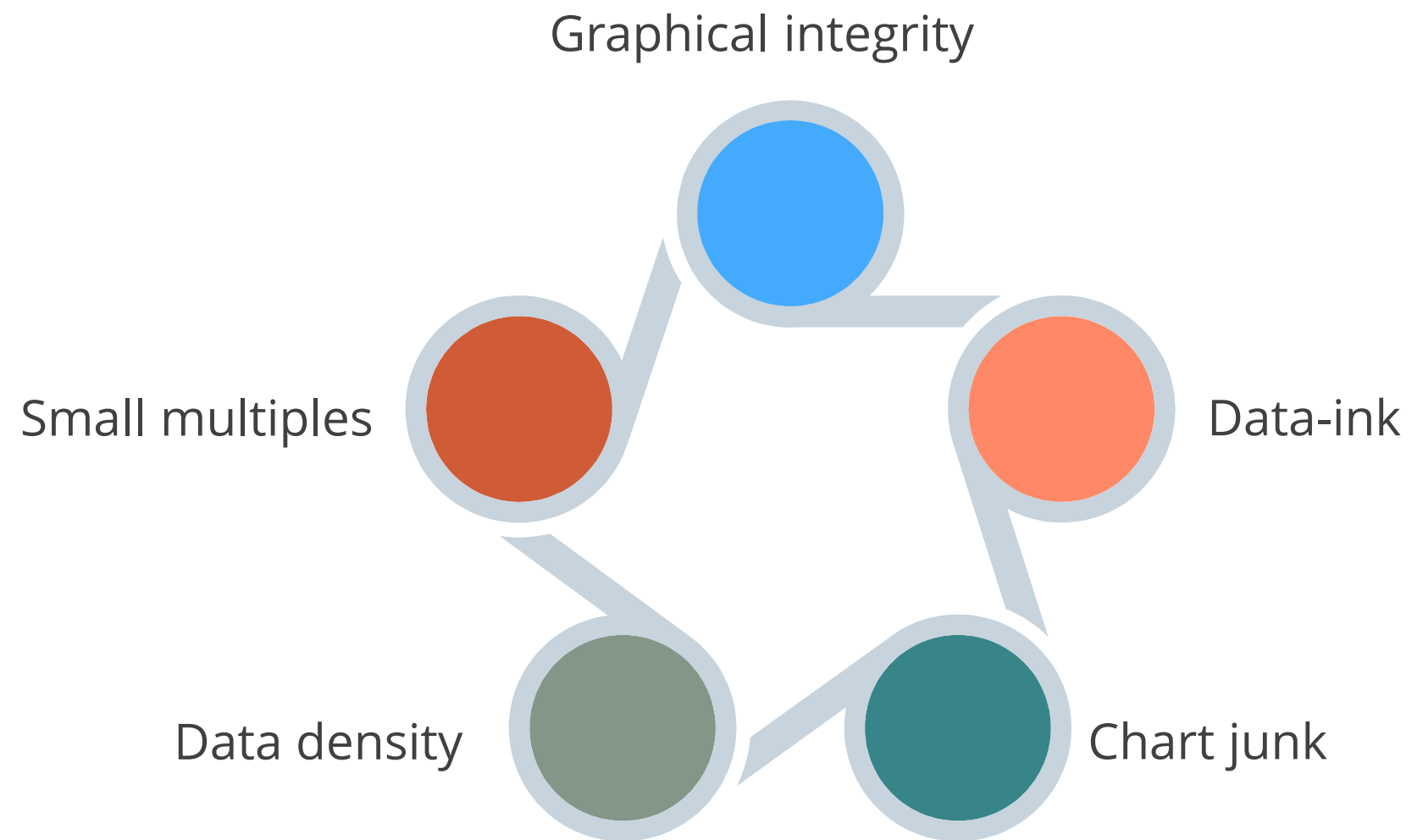


The Visual Display of Quantitative Information

Even now, the work remains relevant.

Information Visualization Principles

They are categorized as follows:

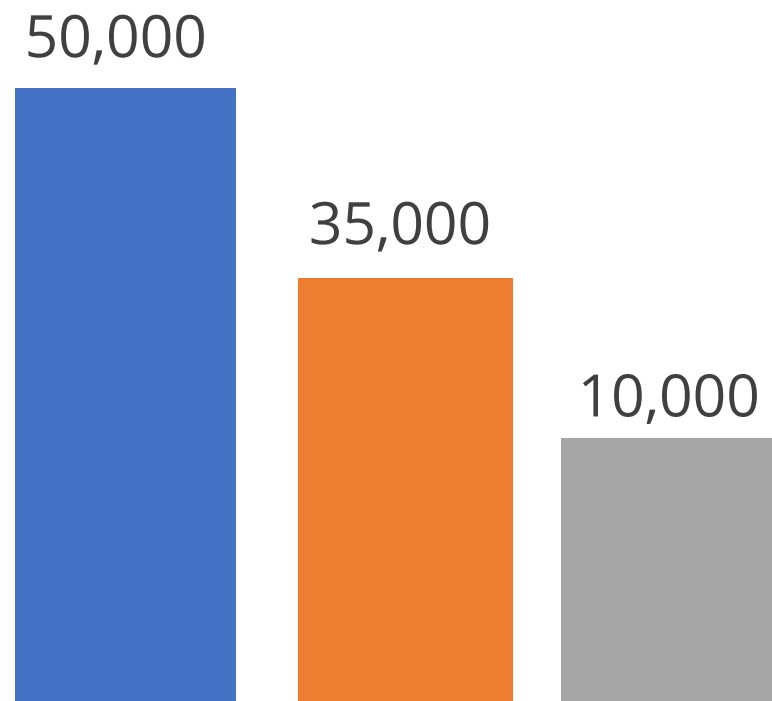


Graphical Integrity

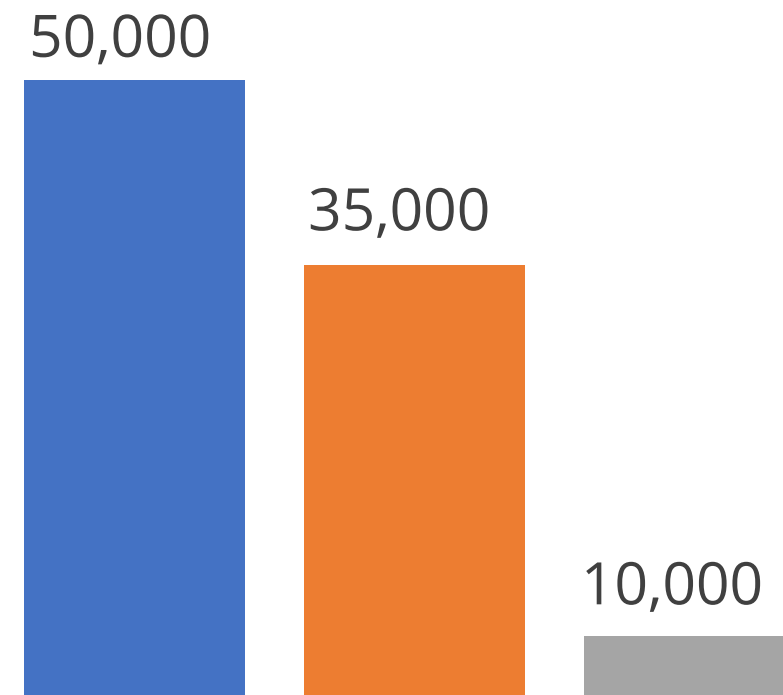
This category has the following principles:

1

Represent numbers that are directly proportional to the numerical quantities represented.



Graph that misrepresented the data

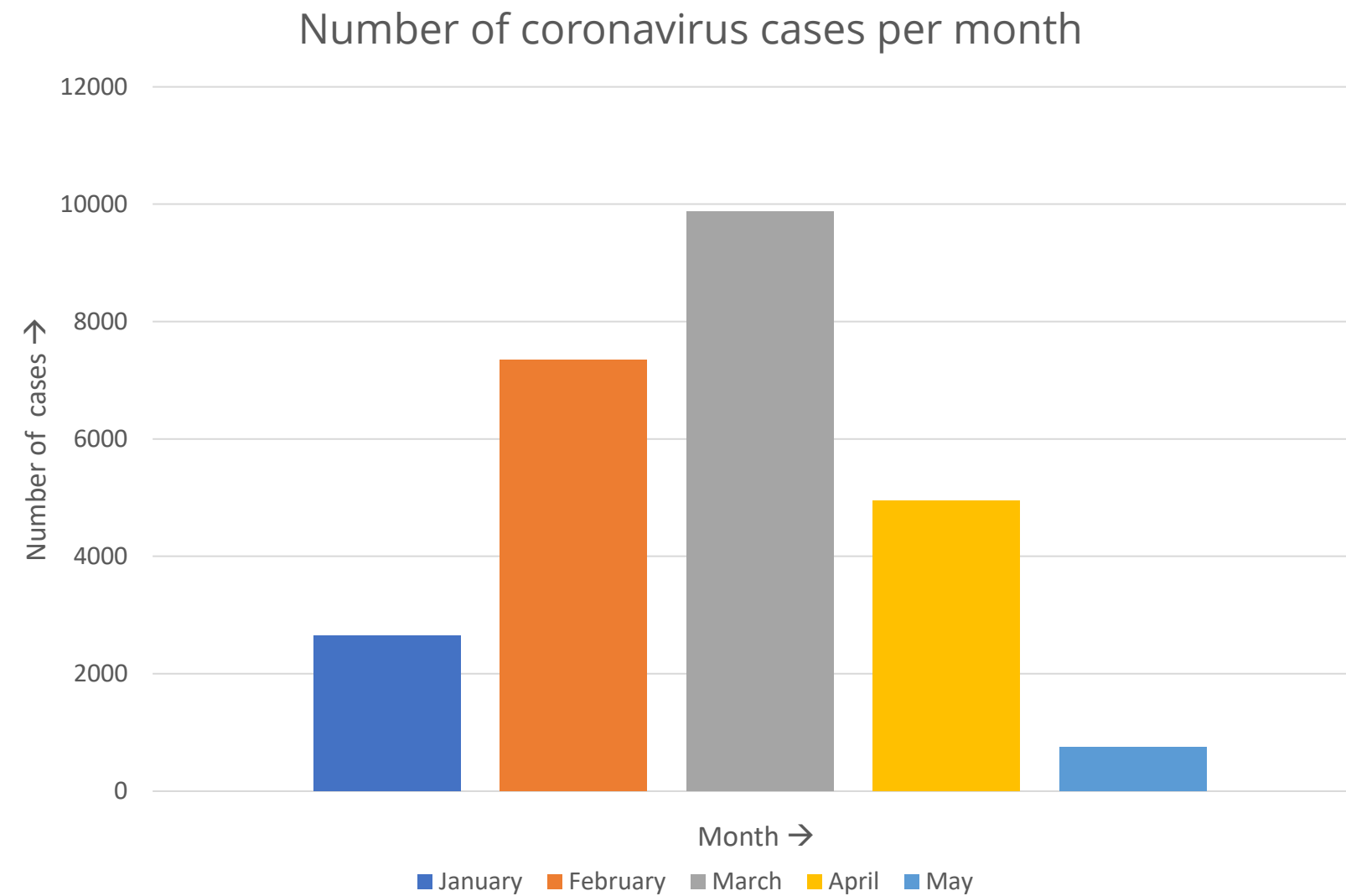


Graph that correctly represented the data

Graphical Integrity

2

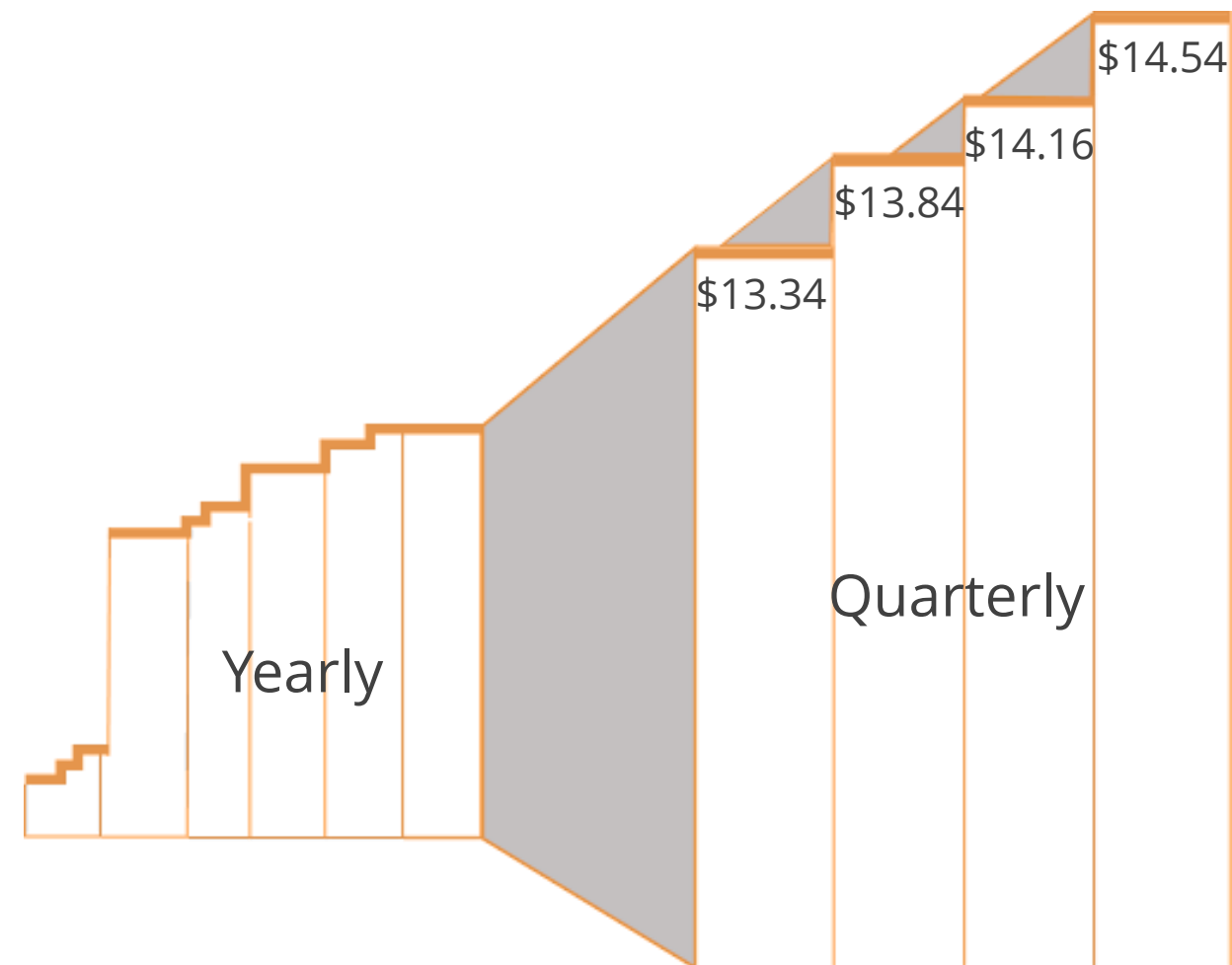
Use clear, detailed, and thorough labeling to defeat graphical distortion and ambiguity



Graphical Integrity

3

Display data variation, not design variation



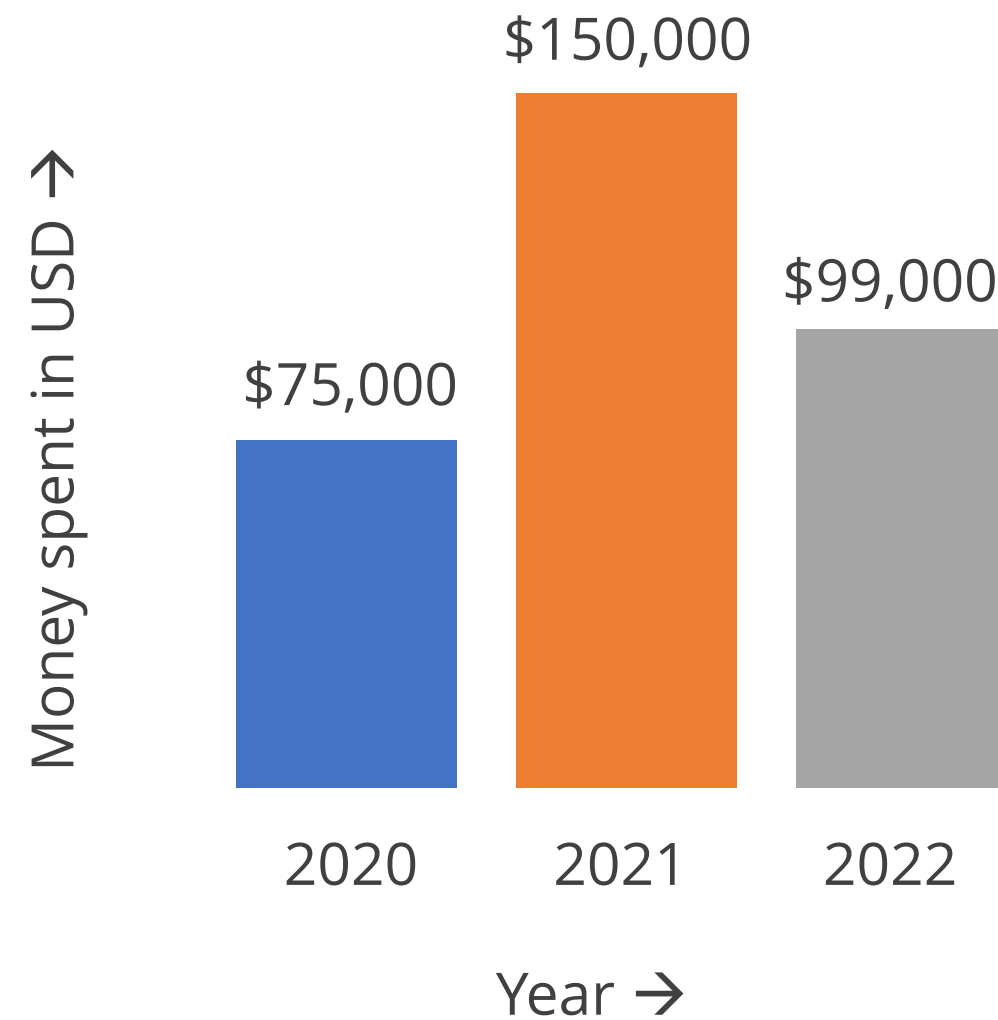
Source:

<https://jcsites.juniata.edu/faculty/rhodes/ida/graphicalIntRedes.html#:~:text=Show%20data%20variation%2C%20not%20design,of%20dimensions%20in%20the%20data.>

Graphical Integrity

4

Use deflated standardized units of measurement when displaying time series

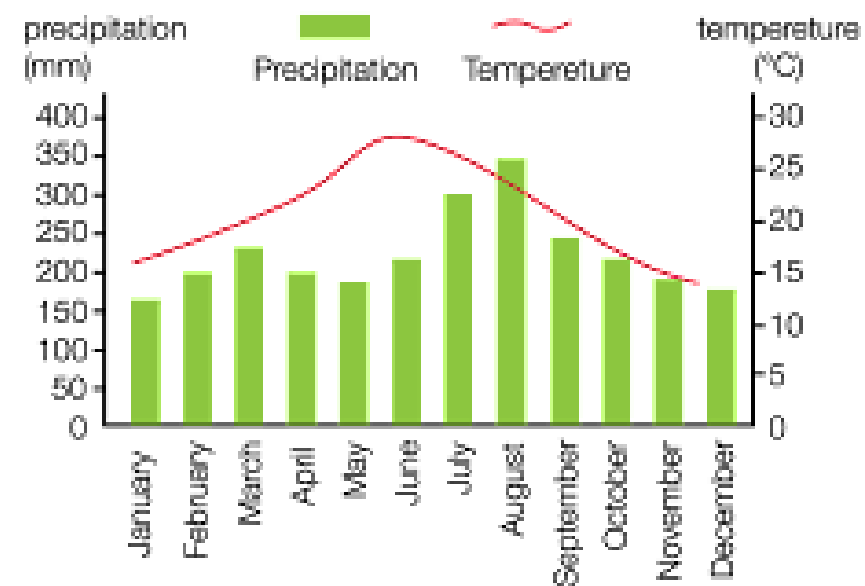


Graphical Integrity

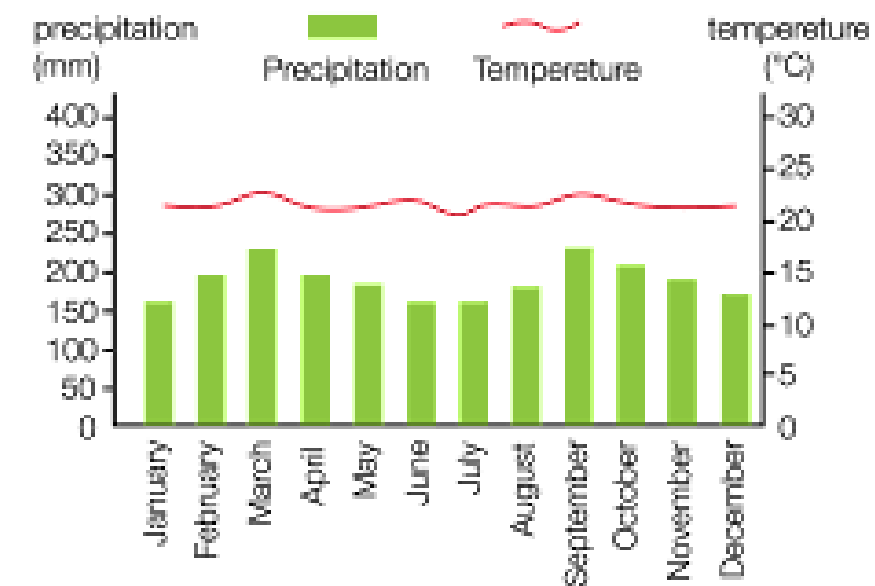
5

Do not let graphics quote data out of context

Monsoon Climate



Equatorial Climate



Climate data is often misquoted by climate change deniers.

Principles of Data-Ink

The following are the principles in this category:

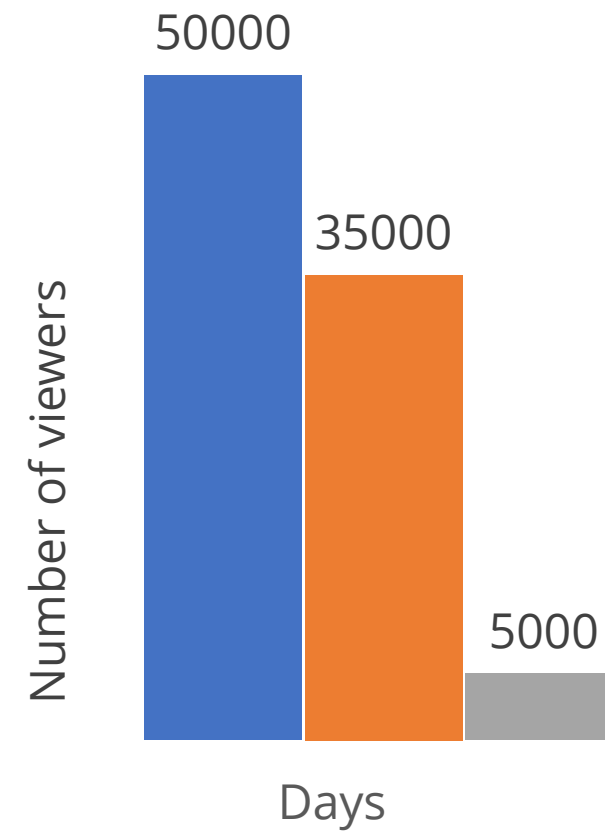
1. Show data above all else:



Principles of Data-Ink

2. Maximize the data-ink ratio:

Number of viewers for a movie X

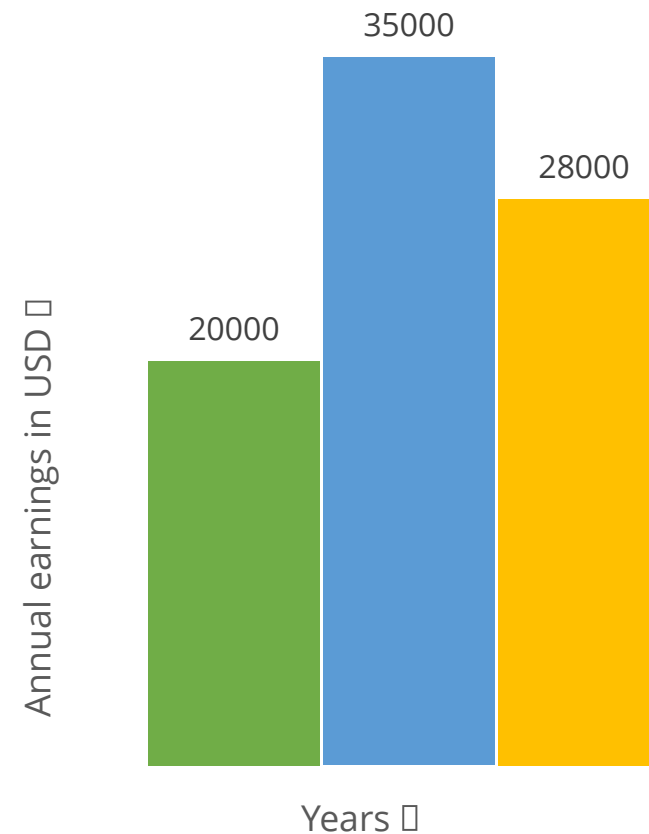


Most of the ink in a visual display should be used to depict the data.

Principles of Data-Ink

3. Remove non-data-ink:

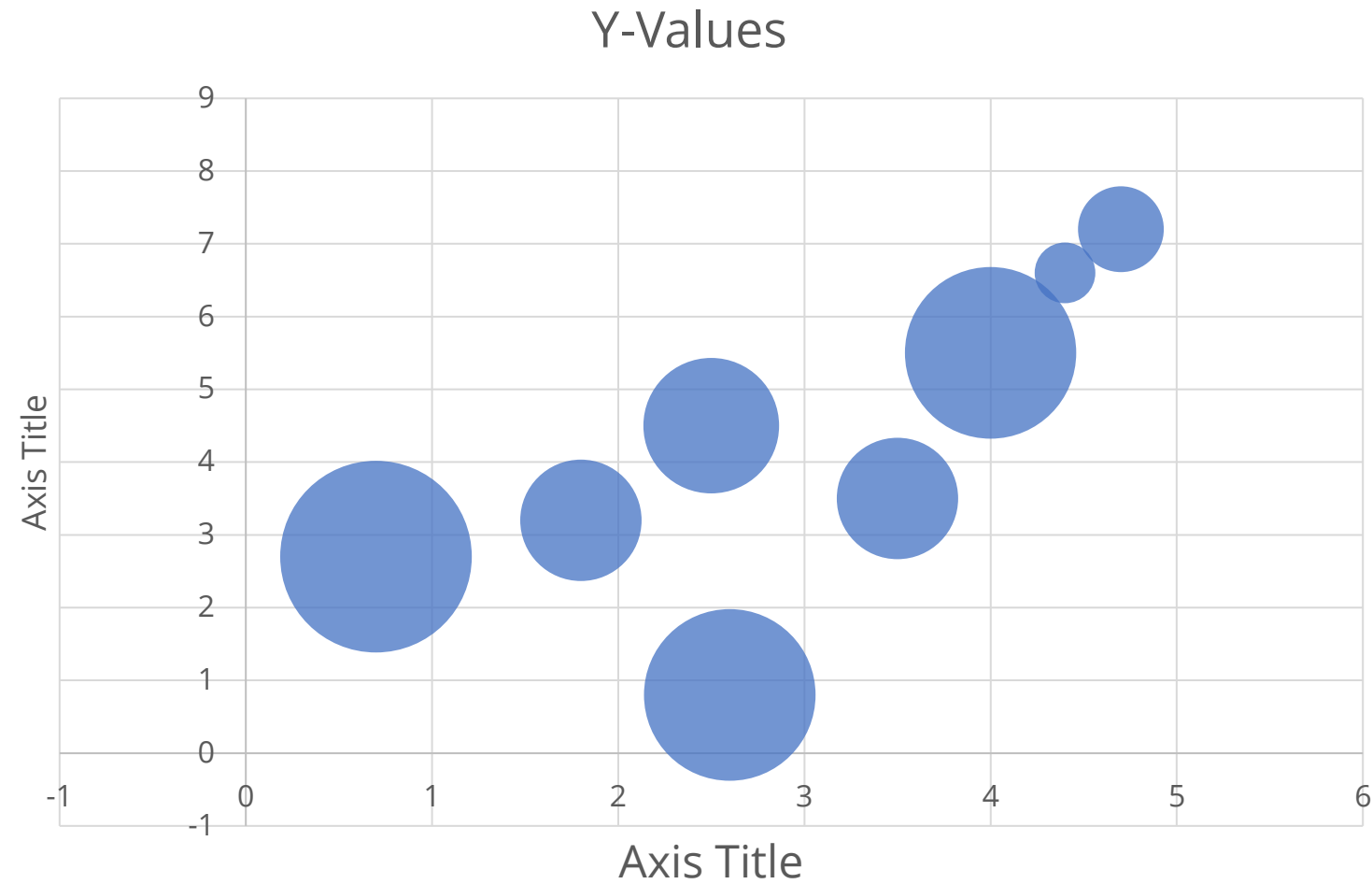
News paper earnings in 3 consecutive years



Minimize or remove the elements like borders, gridlines, and excessive labels to reduce visual clutter and focus attention on the actual data

Principles of Data-Ink

Remove redundant data-ink:



Remove unnecessary or redundant data when presenting the data to avoid overburdening the viewer

Principles of Data-Ink

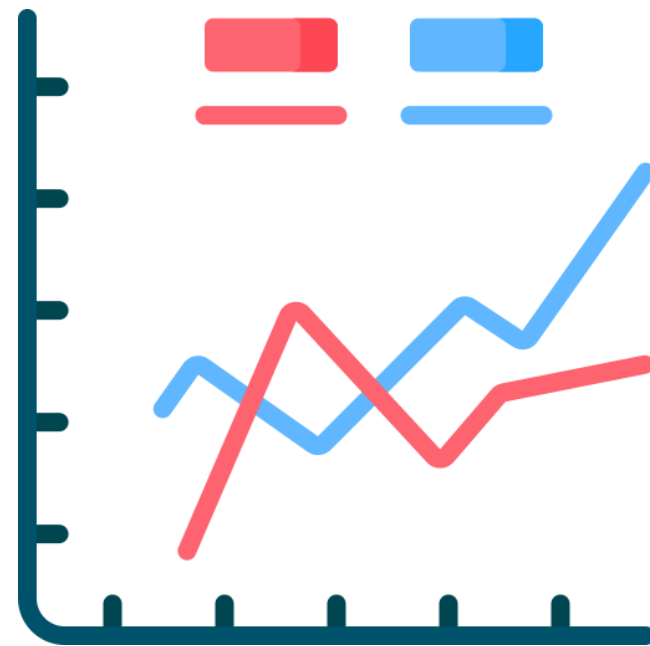
5. Revise and edit:



Go through the graph and the data to check for accuracy and understandability

Chart Junk

This pertains to excessive and unnecessary use of graphical effects.

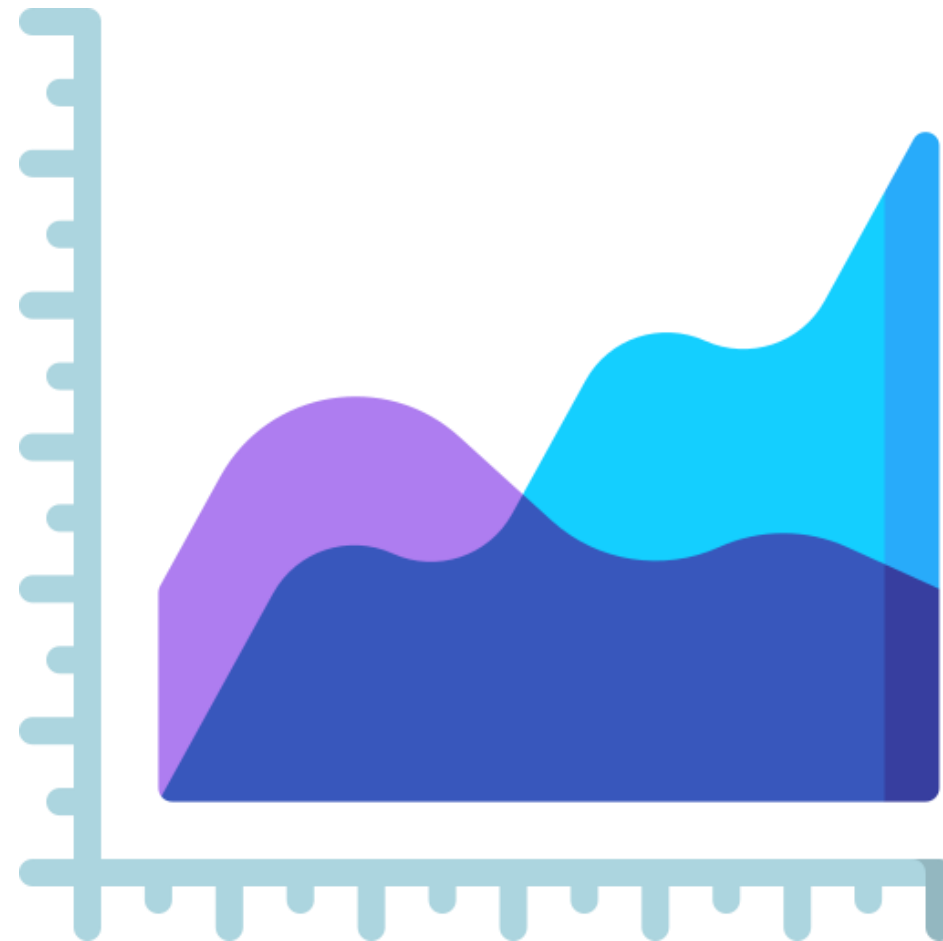


Avoid moiré vibration, heavy grids, and self-promoting graphs as they showcase design skills rather than data.

Tufte dedicates an entire chapter to chart junk in his book.

Data Density

Tufte suggests using high-density graphs rather than low density graphs.



Graphs can be shrunk without compromising on legibility or information.

Small Multiples

Small multiples are a great tool to visualize large quantities of data with a high number of dimensions.



Small multiples pack in a series of the same small graph frequented in one visual.



Data Visualization Using Pivot Tables

Visualizing Data Using Pivot Tables

A pivot table is a tabular representation that combines and summarizes the individual elements of a larger dataset into specific categories or groups.

```
pd.pivot_table(df, index = ['city'])
```

	bathrooms	bedrooms	condition	floors	price	sqft_above
city						
Algona	1.900000	3.200000	3.200000	1.400000	2.072880e+05	1608.600000
Auburn	2.092330	3.420455	3.409091	1.448864	2.993404e+05	1862.250000
Beaux Arts Village	1.750000	3.000000	4.000000	1.000000	7.450000e+05	1140.000000
Bellevue	2.453671	3.804196	3.590909	1.398601	8.471807e+05	2182.604895
Black Diamond	1.750000	3.222222	3.222222	1.388889	3.396056e+05	1807.777778
Bothell	2.431818	3.606061	3.151515	1.500000	4.814419e+05	1957.878788
Burien	1.746622	3.337838	3.554054	1.209459	3.489472e+05	1468.445946
Carnation	2.170455	3.090909	3.090909	1.522727	5.087520e+05	2205.181818
Clyde Hill	2.613636	4.181818	3.545455	1.181818	1.321945e+06	2522.727273
Covington	1.970930	3.325581	3.651163	1.348837	2.962304e+05	1648.837209
Des Moines	1.862069	3.241379	3.482759	1.250000	3.049925e+05	1509.517241
Duvall	2.267857	3.380952	3.071429	1.642857	4.039941e+05	2010.833333
Enumclaw	1.758929	3.178571	3.928571	1.303571	3.076146e+05	1823.928571

In Python, a pivot table can be created using the method **pandas.pivot_table()**.

Discussion: Data Visualization

Duration: 10 minutes



- What is data visualization?

Answer: Data visualization refers to the use of graphical elements such as graphs, charts, or other visuals to extract insights, understand complex patterns, and identify relationships within data.

- What are the different categories of information visualization principles?

Answer: The categories of information visualization principles include graphical integrity, small multiples, data intensity, data-ink ratio, and chart junk.

Assisted Practices



Let's understand the topics below using Jupyter Notebooks.

- 12.4_Visualizing Data Using Pivot Tables

Note: Please download the pdf files for each topic mentioned above from the Reference Material section.



Data Visualization Libraries in Python: Matplotlib

Discussion: Data Visualization Libraries in Python

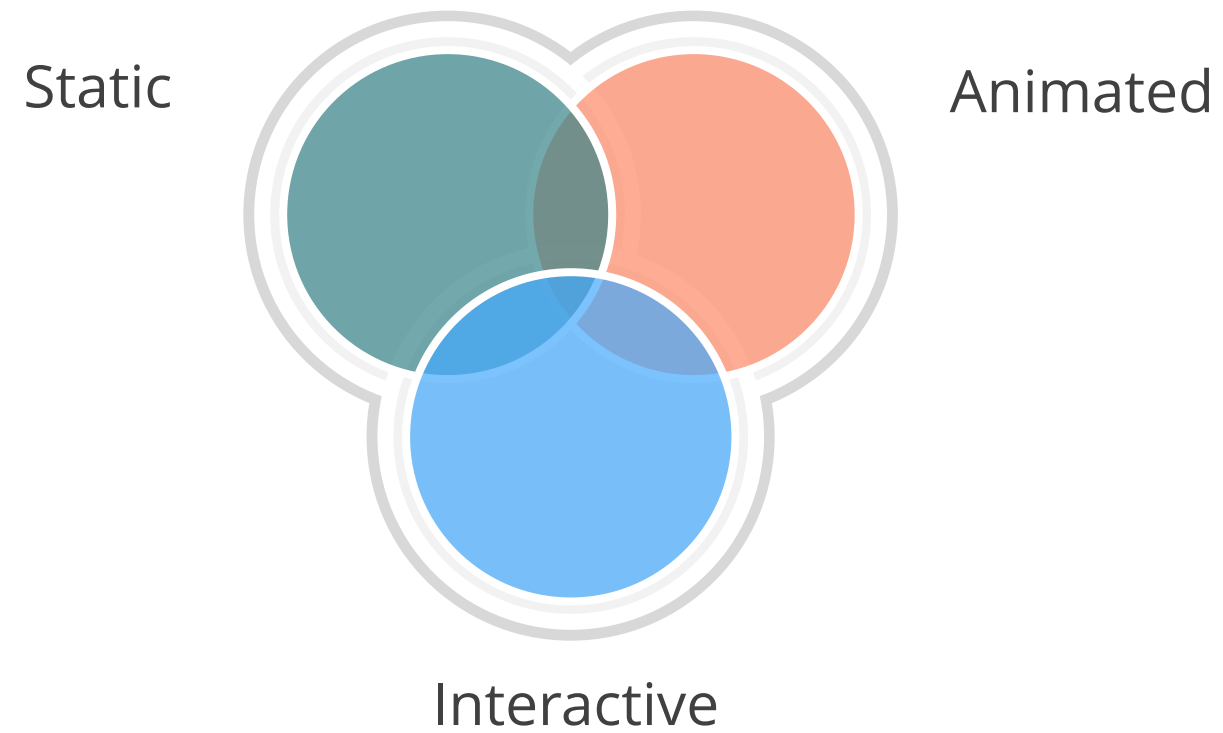
Duration: 10 minutes



- What is the difference between `plot()` and `show()` in Matplotlib?
- What are the advantages of Seaborn?

Matplotlib

Matplotlib is one of the earliest libraries for developing the following types of visualizations:

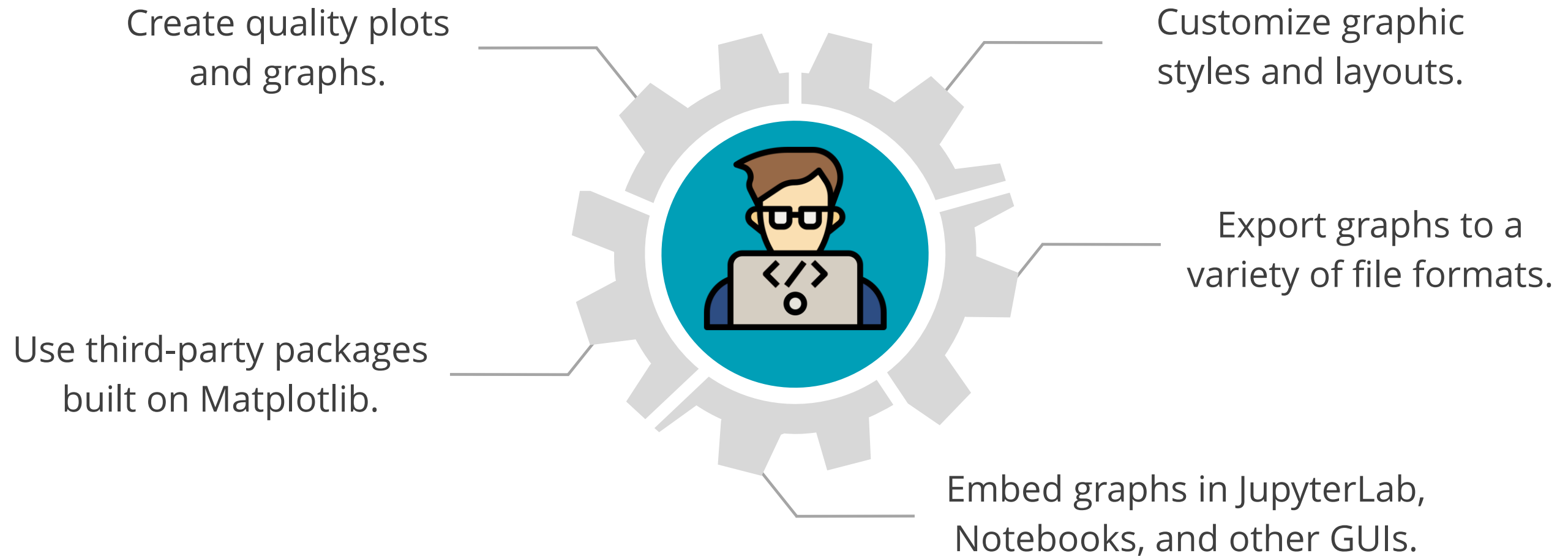


Matplotlib makes difficult things feasible.

Matplotlib

Created by John D Hunter, Matplotlib is an open-source library that can be used freely by developers.

The libraries help to:



Using Matplotlib

Most developers use a submodule called Pyplot (Python plot) for quick and immediate graphing needs.

It is imported with an alias plt, as shown below:

```
import matplotlib.pyplot as plt.
```

The Pyplot library is full of features, methods, and attributes that help develop elegant and rich visuals from underlying data.

Features of Matplotlib

Some of the important features are:

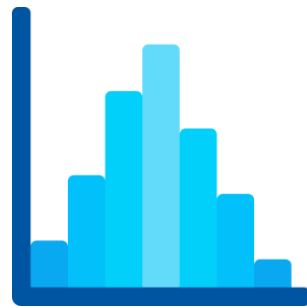
- The plot() method is used for plotting two-dimensional graphs.
- The show() method is used for displaying graphs.
- Markers of different varieties can be opted for in the parameters of the plot() method.
- The attribute linestyle can be used for parameters of the plot() method.
- The xlabel() and ylabel() can be used for labelling the two dimensions.
- The grid() method can be used to add grids to the graph.



Graph Types

Graph Types

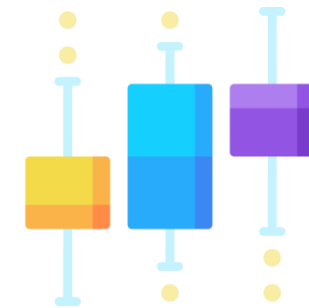
Matplotlib can plot a variety of graphs, which helps in drawing:



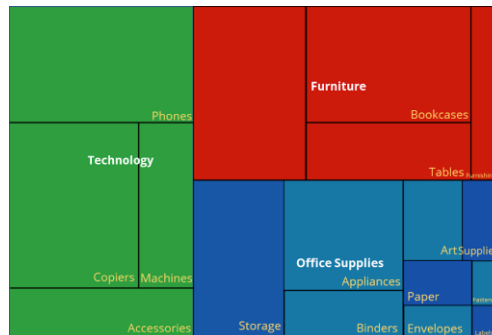
Histogram



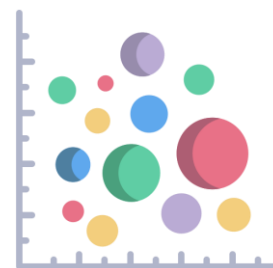
Pie chart



Sub plot



Treemap



Scatter plot



Bar chart

Graph Types

Pyplot provides various methods for plotting graphs:



scatter()

It is used to build a scatter plot which receives X and Y data, usually as a NumPy.



bar()

It is used to build a bar chart. Its input includes two NumPy arrays and the x-axis carries non-numerical data.

Graph Types

Pyplot provides various methods for plotting graphs:



hist()

It helps in plotting Histograms, which, in turn, help to plot frequencies.



pie()

It helps in plotting pie charts and the input is a NumPy array. The input must add up to 100% for the chart to be correct.

Graph Types

Pyplot provides various methods for plotting graphs:



subplot()

It helps in plotting more than one sub plot in a graphical area.

Have all the subplots ready before calling the plot() method.

Call the show() method to view subplot.



Data Visualization Libraries in Python: Seaborn

Seaborn

Seaborn, along with Matplotlib, is a high-level library for preparing statistical graphics.



seaborn

It makes visualization a central part of exploring and understanding complex datasets.

It is integrated with Pandas data structures.

Seaborn

It is usually imported into the app with an alias sns.

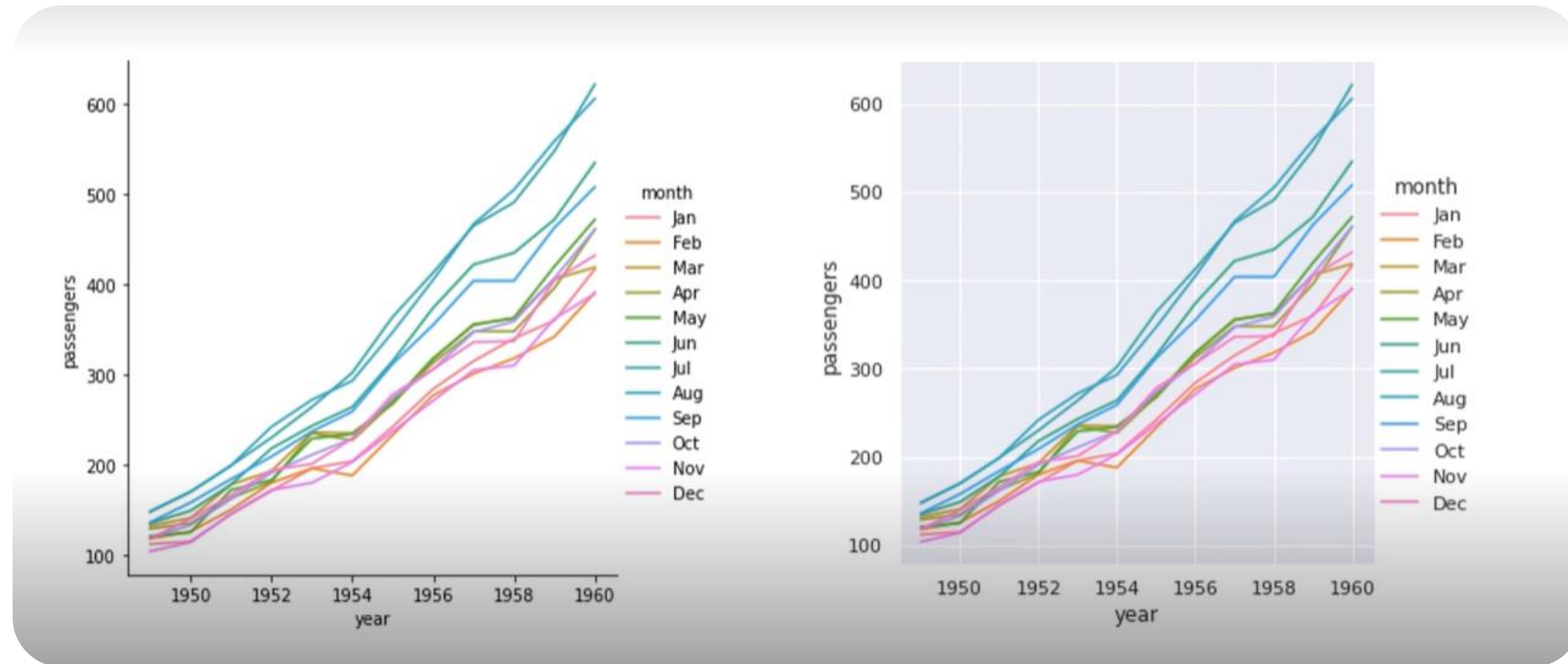
The Seaborn library helps a developer to understand and learn Seaborn statistical plotting techniques.

Seaborn supports several different dataset formats and almost all functions.

Seaborn uses Pivot to provide features of data manipulation that convert long-form and wide-form datasets to fine tune visualization.

Seaborn's Visualization Features

Seaborn uses Matplotlib for plotting a variety of graphs.

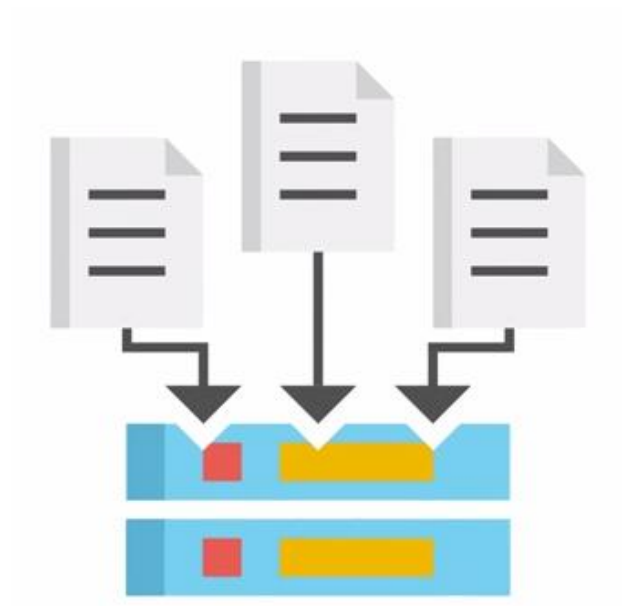


Sample plots with and without a theme

A default theme can be set using `set_theme()` method on the imported `sns`.

Seaborn's Visualization Features

Use the `load_dataset()` method to load data.

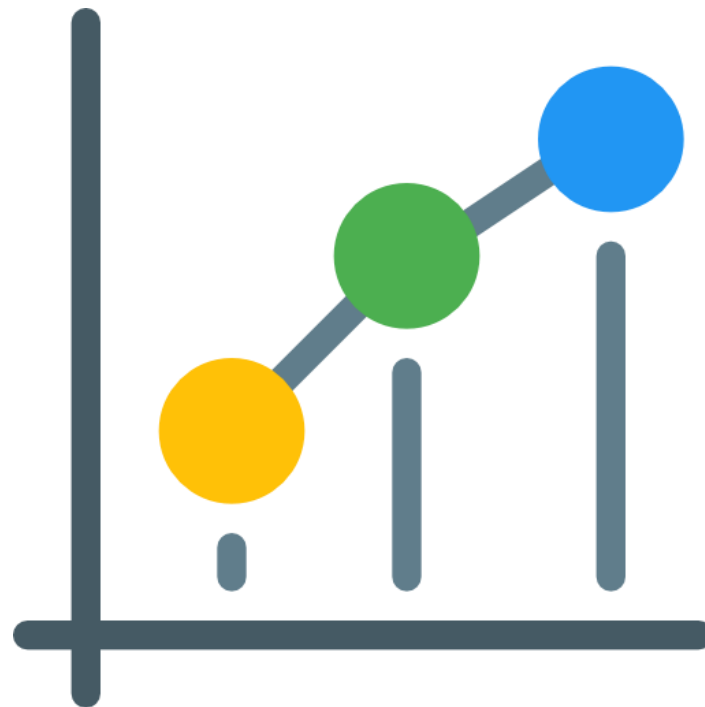


- It is a function to import default data from Seaborn that's useful to practice visualization.
- The data can then be queried and inspected before plotting and visualizing it.

The data can be inspected with functions such as `head()` and `tail()` to view the top and bottom parts of the dataset.

Seaborn's Visualization Features

The replot() method can be used to plot graphs by providing necessary parameters like data, X, Y, hue, style, size, and so on.



The replot() method is used to plot the relation between two variables of a dataset on a graph (x-y axis) with semantic mappings of the subset.

Seaborn's Visualization Features

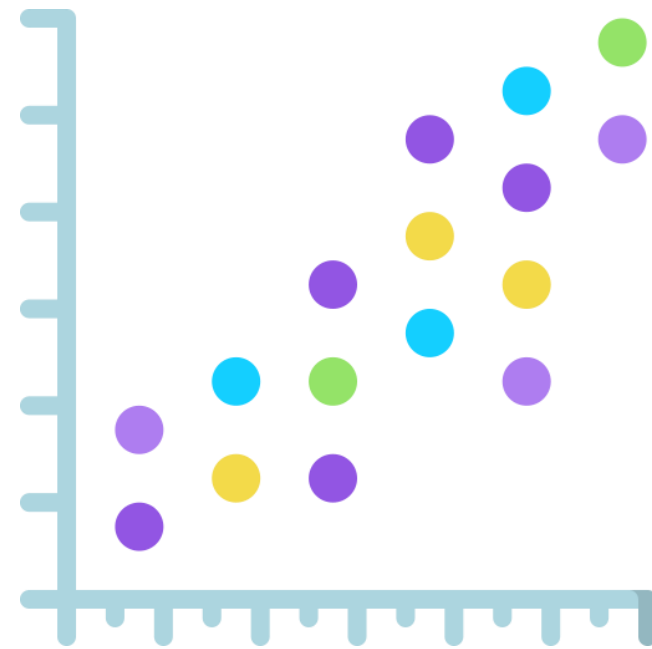
Seaborn specializes in visualizing statistical data and performs the necessary statistical calculations automatically.



While estimating statistical values, Seaborn uses bootstrapping to compute confidence intervals and draw error bars.

Seaborn's Visualization Features

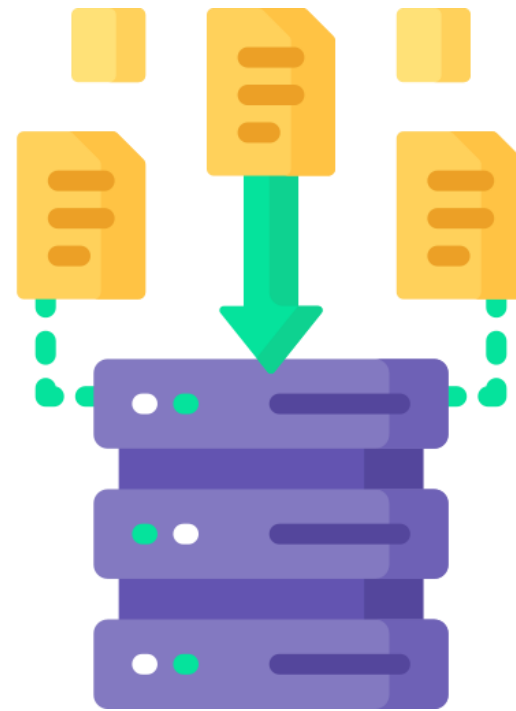
Seaborn can enhance a scatter plot by including a linear regression model (and its uncertainty) using `lmplot()`.



Seaborn offers the `displot()` function for automatically plotting and visualizing statistical distributions.

Seaborn's Visualization Features

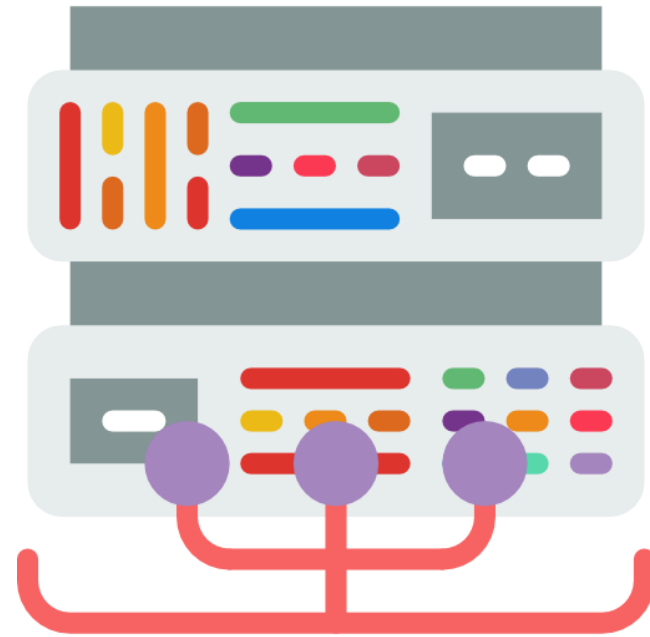
Category plots is a unique feature and is indispensable, especially when data sets include categories.



Seaborn provides the `Catplot()` method for this and the parameters can be adjusted according to the statistical visualization.

Seaborn's Visualization Features

Seaborn also offers composite views of multivariate datasets.



The `joinplot()` and `pairplot()` methods help visualize and understand complex statistical data.



Using Seaborn to Plot Graphs

Data for Seaborn

Seaborn is particularly helpful in statistical-related graphs and visualizations.



Let's examine a statistical estimation and error bars example from the Seaborn library's built-in fMRI data.

Data for Seaborn

The fMRI data is a data frame with 1064 rows and five columns.

The columns are:



To import the built-in fMRI data, use the `load_dataset()` method and pass **fMRI** as the parameter to create the fMRI dataframe.

Data for Seaborn

The plotting is essentially timepoint versus signal with respect to:

Different types of events

- Stim
- cue

Different regions

- parietal
- frontal

Plotting Using Seaborn

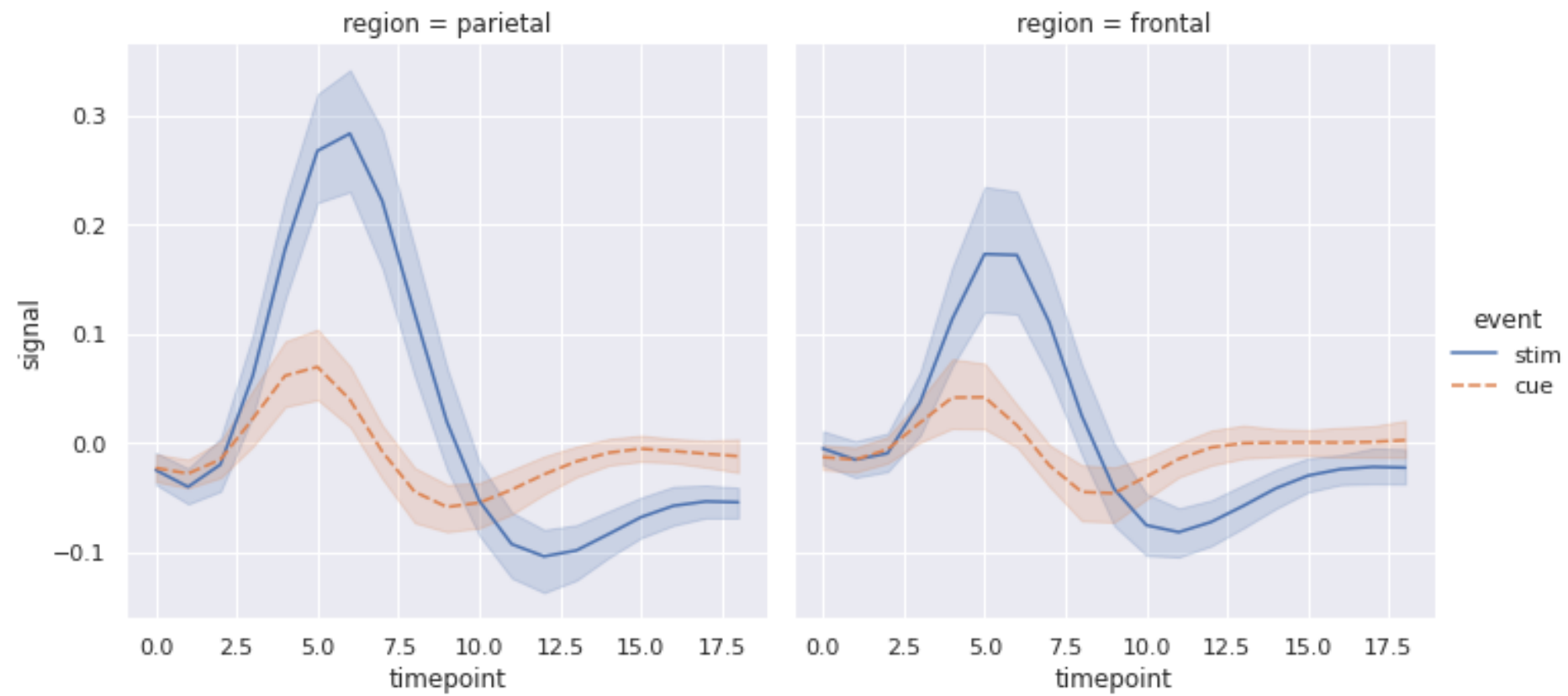
Seaborn takes the visualization of graphs and statistical performance to a new level.

For plotting with Seaborn, fire up JupyterLab Notebook or JupyterLab and perform the following:

```
import seaborn as sns
sns.set_theme()
fmri = sns.load_dataset("fmri")
sns.relplot(
    data=fmri, kind="line",
    x="timepoint", y="signal", col="region",
    hue="event", style="event",
)
```

Plotting Using Seaborn

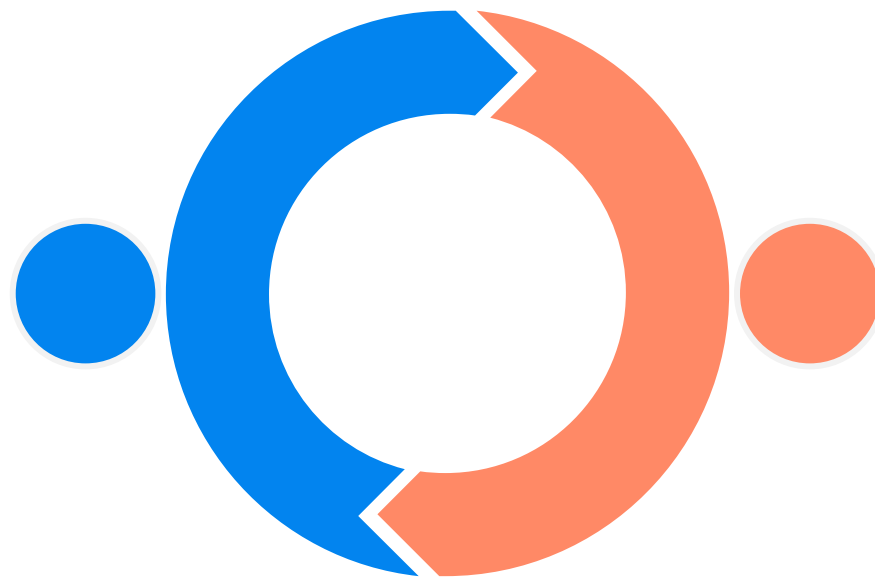
This will result in the plots shown below:



Analysis Using Seaborn Plots

There are two graphs because:

Two columns were
used as region



Two different colored lines
representing stim and cue
are events

For both plots, timepoint (x-axis) and signal (y-axis) are the same.

Analysis Using Seaborn Plots

Following are the inferences:

- Signals on the parietal are higher than those on the frontal.
- The stim line shows that it peaks or troughs around the timepoint value of 5.0.
- Both the signals are banded, indicating that Seaborn has already performed statistical estimation and error bars.
- Error bars are wider when the signal points are peaked or troughed.

Discussion: Data Visualization Libraries in Python

Duration: 10 minutes



- What is the difference between `plot()` and `show()` in Matplotlib?

Answer: The `plot()` function is used to create two-dimensional graphs, whereas the `show()` function is used to display and present these graphs.

- What are the advantages of Seaborn?

Answer: Some of the advantages of Seaborn include:

- Assisting developers in understanding and applying Seaborn's statistical plotting techniques
- Supporting multiple dataset formats and nearly all functions
- Utilizing a pivot feature for data manipulation, which transforms long-form and wide-form datasets to fine-tune visualization

Assisted Practices



Let's understand the topics below using Jupyter Notebooks.

- 12.9_Plotting 3D Graphs for Multiple Columns using Seaborn

Note: Please download the pdf files for each topics mentioned above from the Reference Material section.



Data Visualization Libraries in Python: Plotly

Discussion: Data Visualization Libraries in Python

Duration: 10 minutes



- What is Plotly?
- Define Bokeh and enumerate some of its advantages

Plotly

Plotly's Python graphing library helps analysts make interactive and publication-worthy graphs.

The library includes a rich set of features for making:



Line plots

Scatter plots

Heat maps

Histograms

Error bars

Box plots

Sub plots

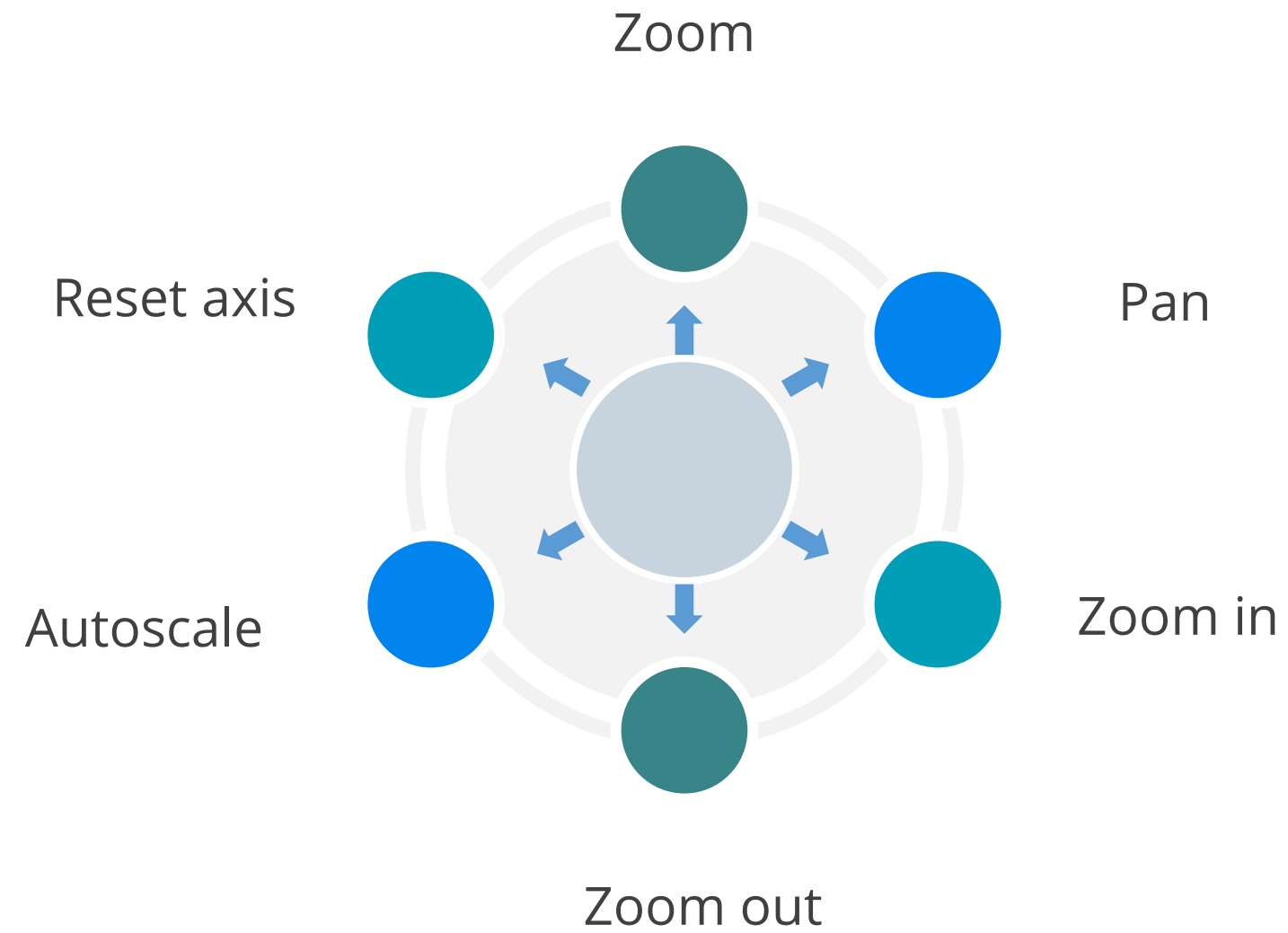
Bar and area charts

Multiple axes plots

Bubble and polar charts

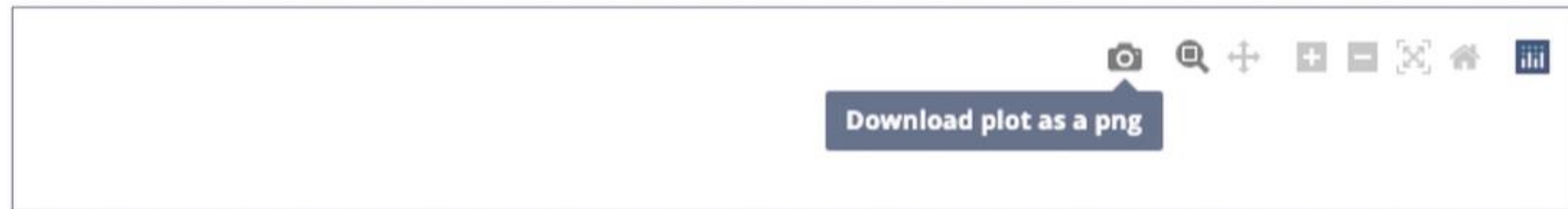
Plotly

Plotly's libraries specialize in interactivity such as:



Plotly

A PNG file can be downloaded for any Plotly-generated graph as shown below:



Plotly is rich in every aspect of information visualization and provides a variety of charts.

Plotly Features and Charts

Plotly's fundamental charts include:

Displaying figures

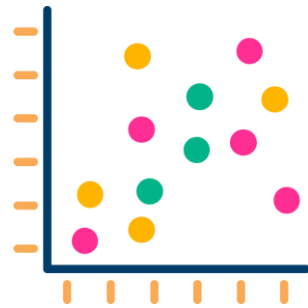
Creating and
updating figures



Plotly Express

Plotly Features and Charts

Its basic charts include:



Scatter plots



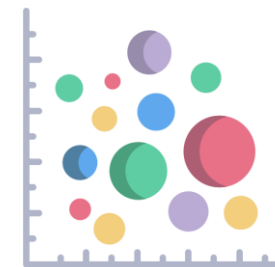
Line charts



Bar charts



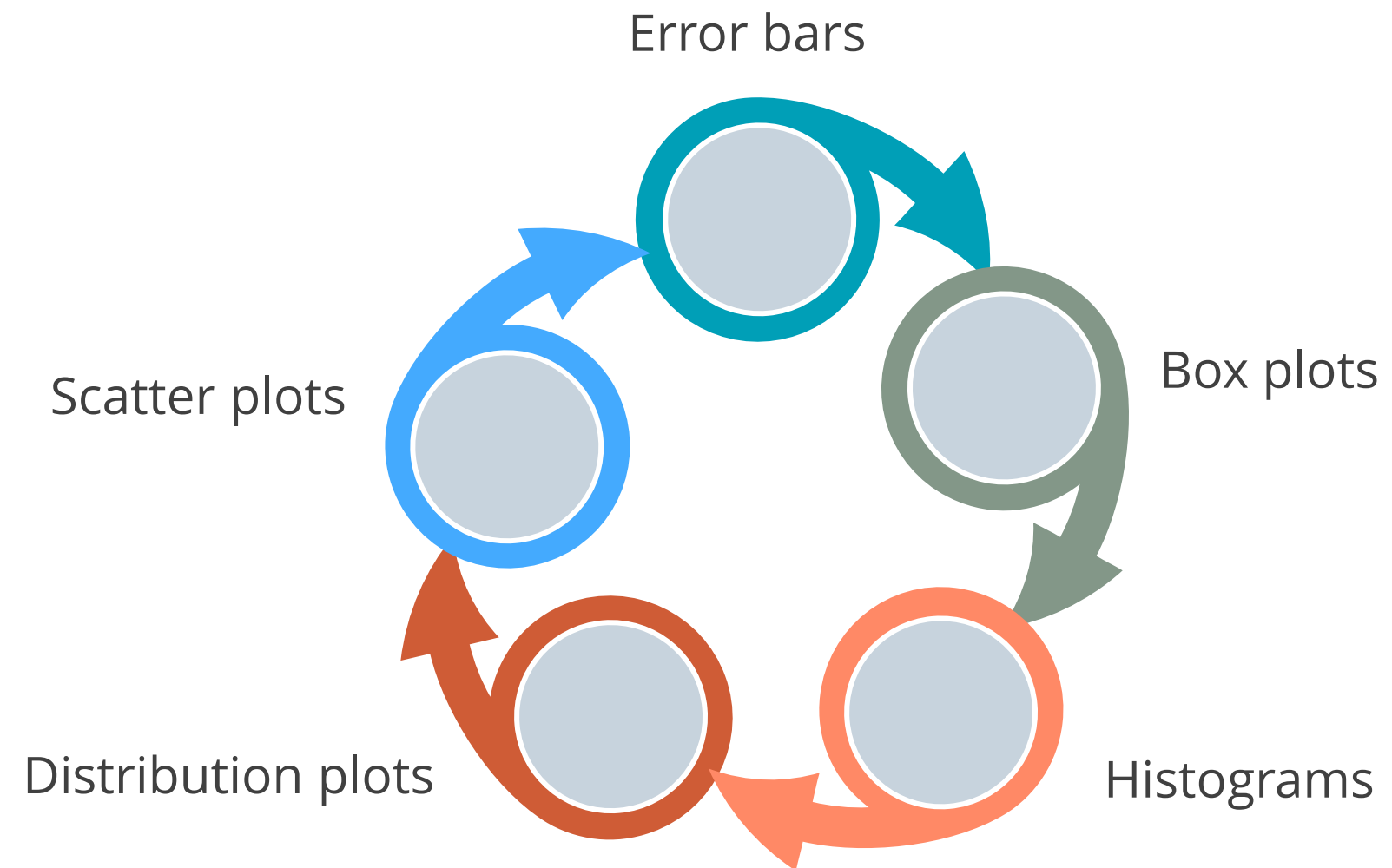
Pie charts



Bubble plots

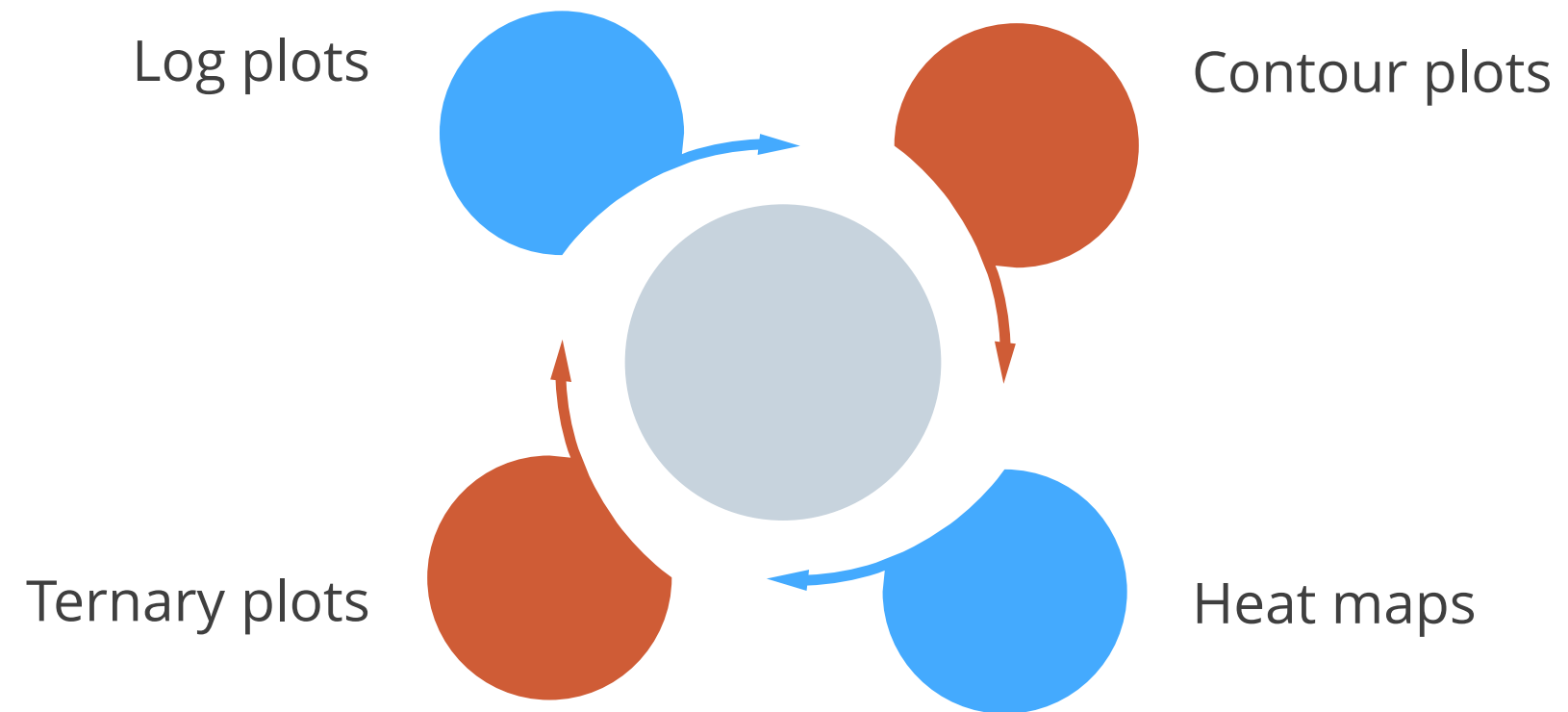
Plotly Features and Charts

Plotly's statistical charts include:



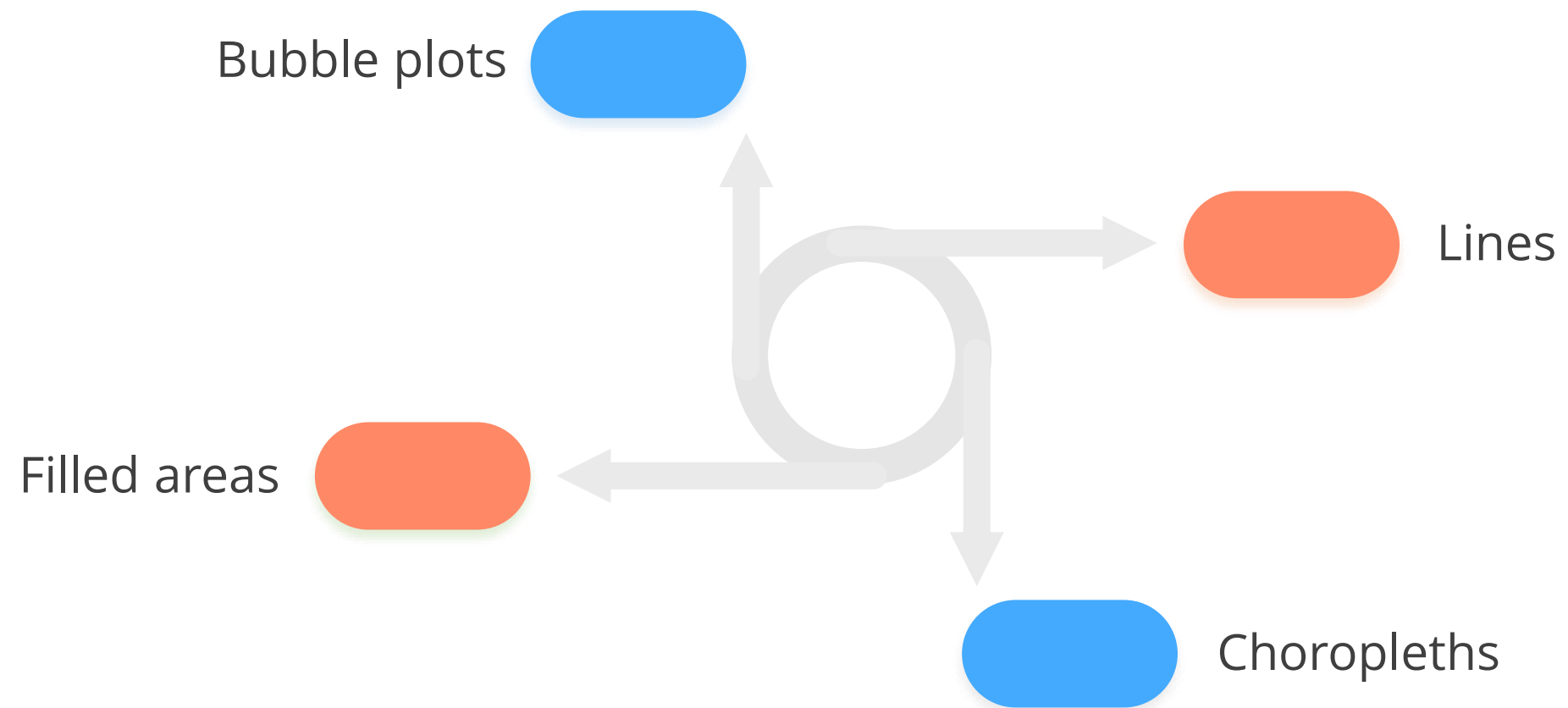
Plotly Features and Charts

Scientific charts include:



Plotly Features and Charts

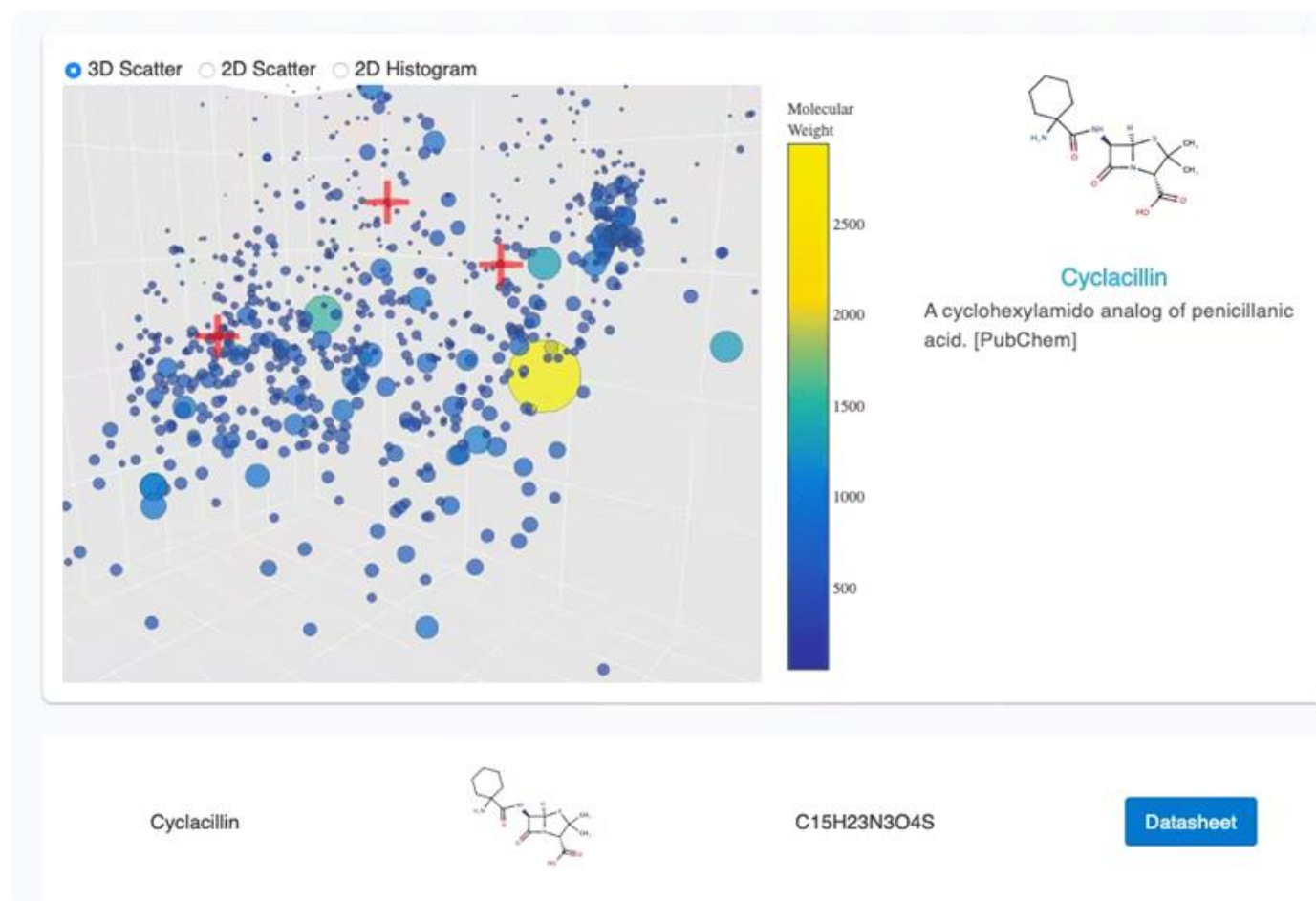
Maps are another specialty of Plotly and are used to generate:



These can be generated directly on geographical maps.

Plotly Features and Charts

Plotly also provides specialized plotting for artificial intelligence, machine learning, and bioinformatics.

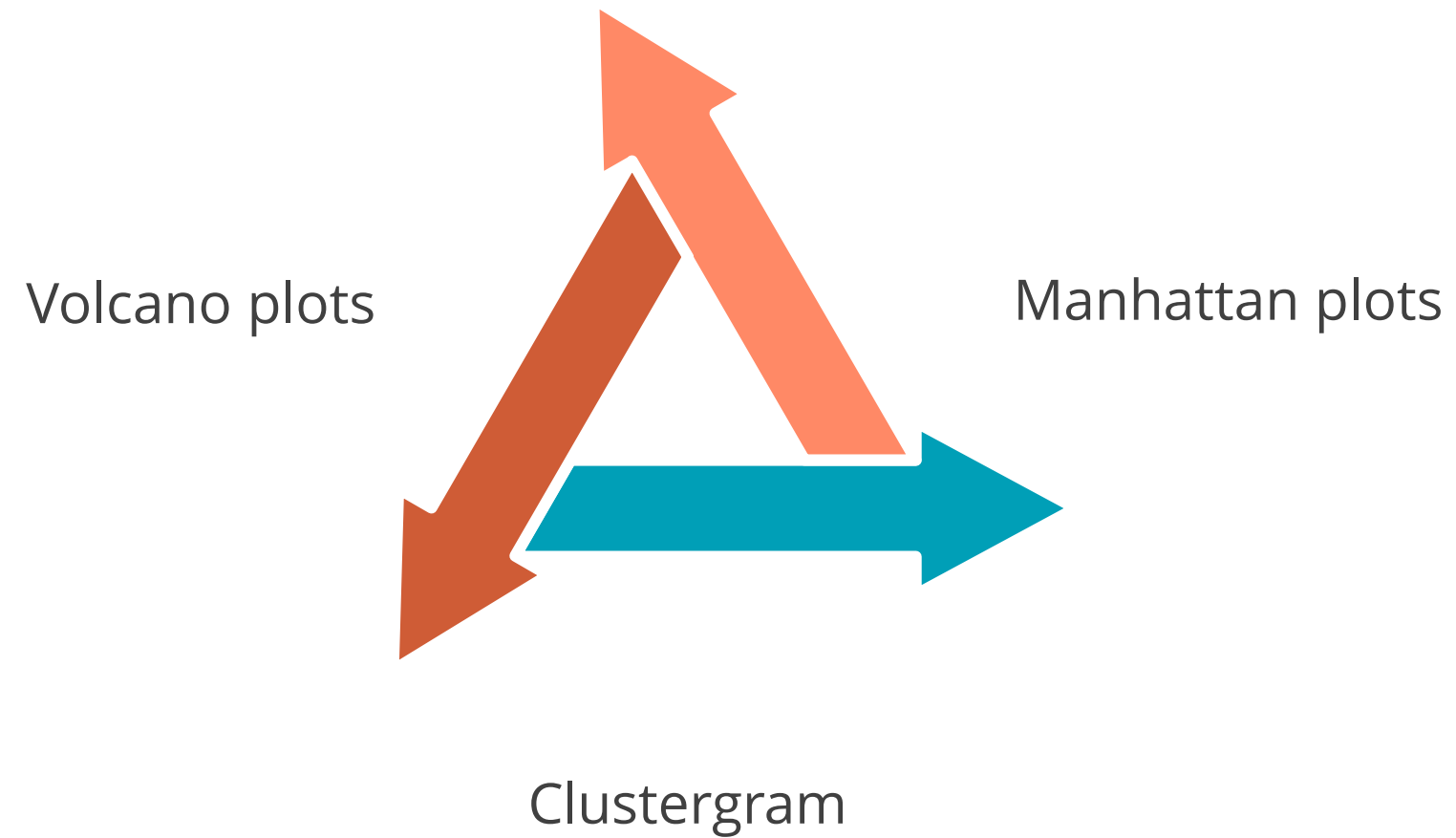


Example

Regression plots, classification plots, and other related plots can be generated based on the AI and ML datasets.

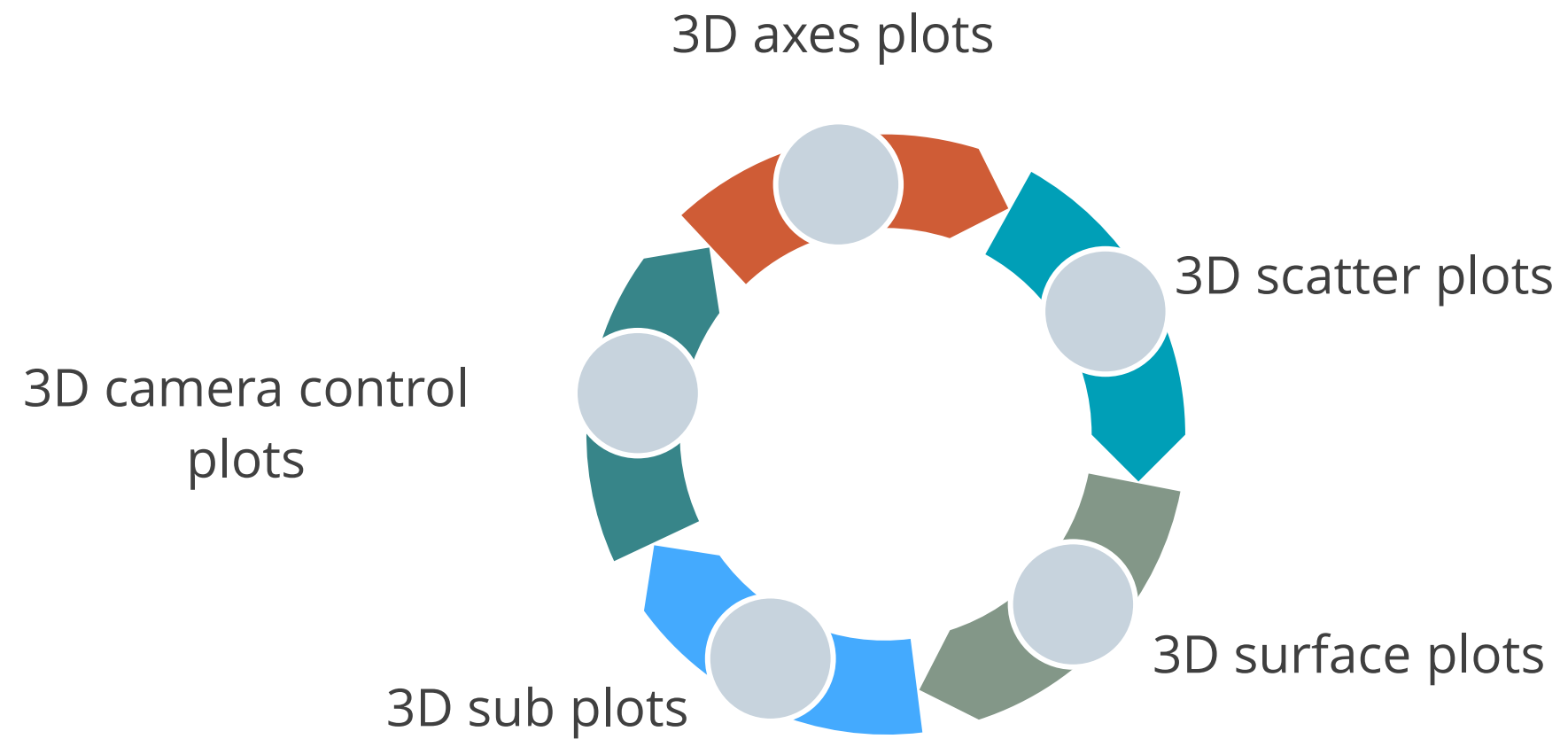
Plotly Features and Charts

Plotly libraries are used to easily sketch plots useful for bioinformatics such as:



Plotly Features and Charts

Plotly's other specialties include:



Plotly Features and Charts

Plotly's subplot features include:

Table and chart sub-plots

Figure factory sub plots

Map sub-plots

Mixed sub-plots



Plotly Features and Charts

Plotly allows several graphic-based transformations such as:

- ◆ Perform filter, groupby, aggregate, and multiple transformations
- ◆ Add custom control to graphs and figures
- ◆ Add animations to graphs, both two-dimensional and three-dimensional




Plotly also helps generate dashboards and other prerequisites for complex data analysis.



Data Visualization Libraries in Python: Bokeh

Bokeh

According to
<http://bokeh.org>:

- 1  Bokeh is a Python library for creating interactive visualizations for modern web browsers.
- 2  It is used to build graphics, ranging from simple plots to complex dashboards, with streaming datasets.
- 3  It can create JavaScript-powered visualizations without writing any JavaScript.

Bokeh

Bokeh is an interactive visualization library in Python, which provides graphs and plots on web browsers like:



Mozilla Firefox



Google Chrome



Apple Safari

It helps generate a wide range of elegant and concise charts.

Bokeh

Bokeh is especially suitable for elegant web app development and web-based dashboards.



Before use, install Bokeh using Anaconda or pip.

Bokeh is supported and tested on Python 3.7 and above.

Steps for Building Simple Bokeh Graphs

Follow these steps to build simple Bokeh graphs:

STEP 1: Prepare data in Python;
for simple graphs, use lists

STEP 2: Call the `figure()`
method; customize
properties like title, tools,
axes labels, etc.

STEP 3: Add renderers to the
plot; use the `line()` method to
plot line plots

STEP 4: Use the `show()` or
`save()` method

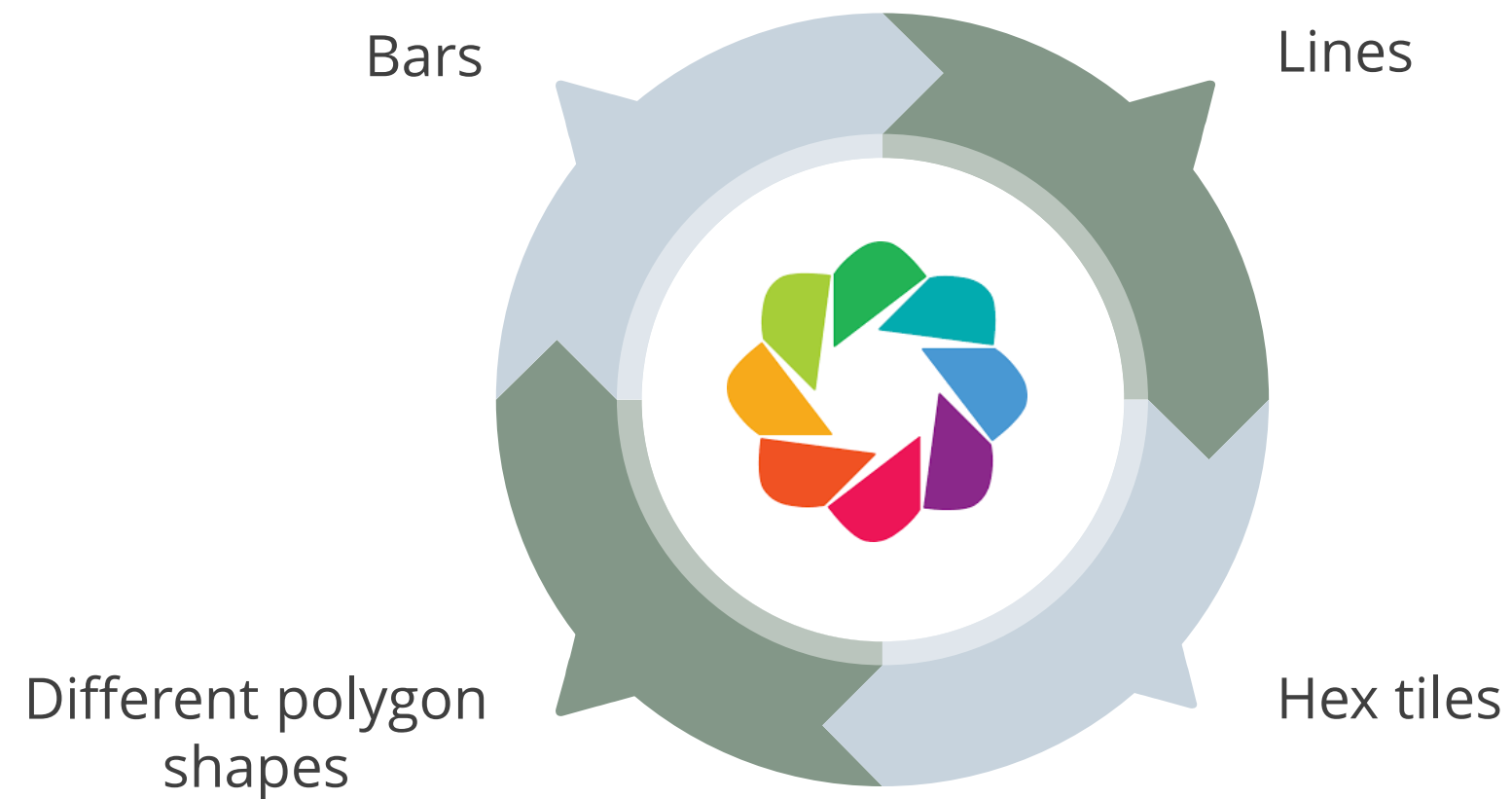


In step 3, we can use attributes like legends, widths, colors, etc. We can combine multiple graphs with different properties using the same renderer.

Customizing and Rendering in Bokeh

Render functions can be used to create Glyphs in Bokeh.

Bokeh's plotting interface supports many different Glyphs such as:



Customizing and Rendering in Bokeh

circle()

The method is used for rendering circles.
Color or size can be used to modify the Glyph property.

vbar()

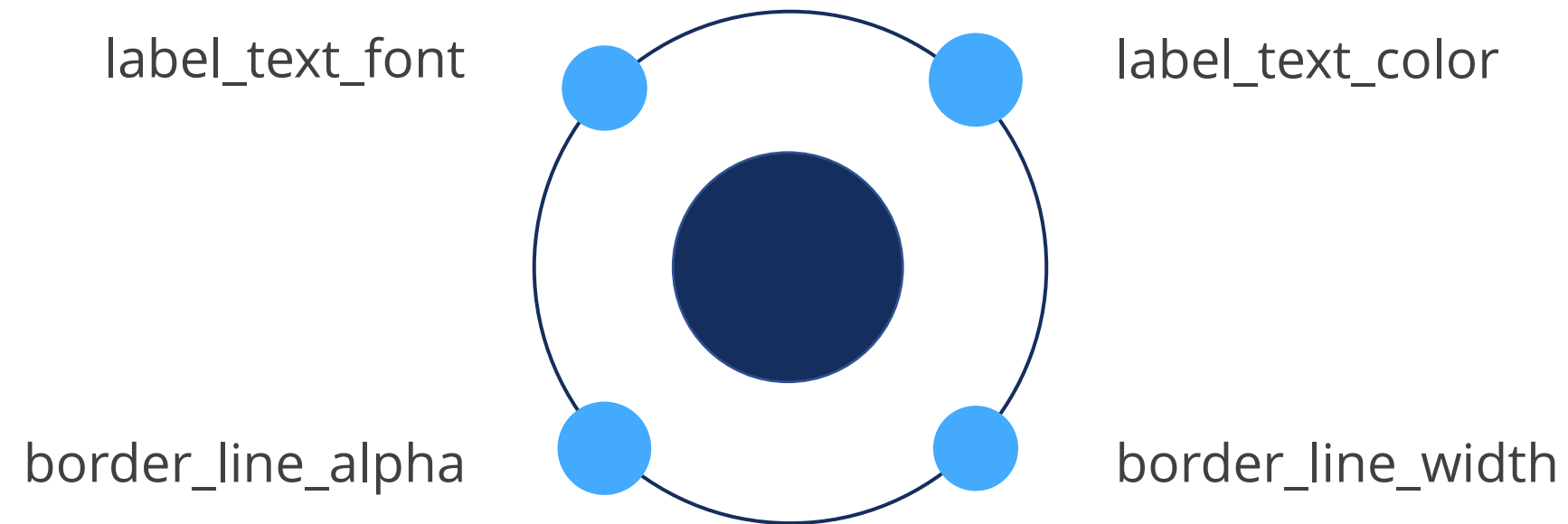
The function is used for rendering bars.
Property can be used to modify the Glyph property.

Multiple renderers can be included in the same plot to get customized graphs and rendering.

Adding Legends, Text, and Annotations in Plots

Bokeh, along with other attributes, provides the legend object to modify the generated Glyph's properties.

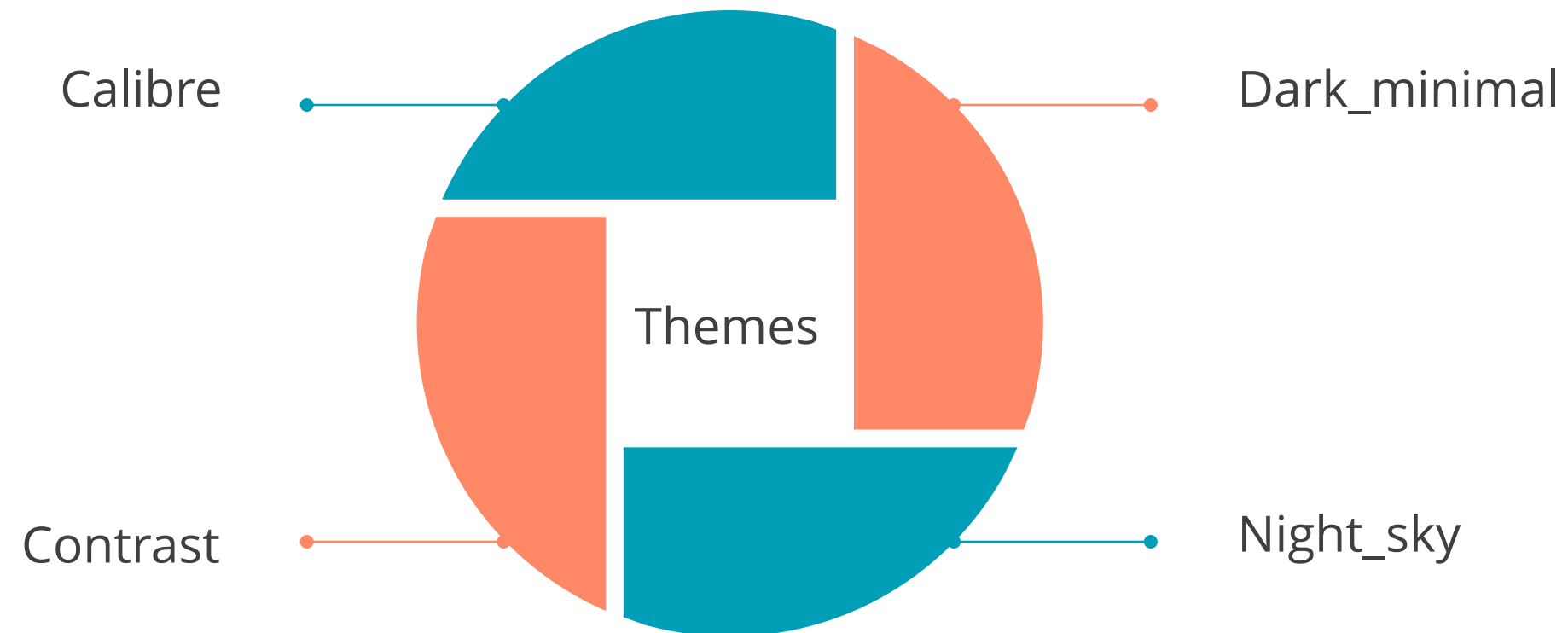
Some common legend attributes include:



Additionally, import BoxAnnotation for including annotations.

Adding Legends, Text, and Annotations in Plots

Bokeh provides some built-in themes for further customization of plots.



Additionally, the `sizing_mode` attribute can be used to enable responsive sizing.

Discussion: Data Visualization Libraries in Python

Duration: 10 minutes



- What is Plotly?

Answer: Plotly is a Python graphing library that enables analysts to produce interactive and visually engaging graphs appropriate for publication.

- Define Bokeh and enumerate some of its advantages

Answer: Bokeh is a Python library that facilitates the creation of interactive visualizations designed for modern web browsers.

Some of its advantages include:

- Supporting the creation of a variety of graphics, from simple plots to complex dashboards, capable of handling streaming datasets
- Allowing for the generation of JavaScript-powered visualizations without the need for manual JavaScript coding

Assisted Practices



Let's understand the topics below using Jupyter Notebooks.

- 12.12 Introduction to Matplotlib
- 12.13_Matplotlib for 3D Visualization
- 12.14_Using Matplotlib with Other Python Packages
- 12.15_Introduction to Plotly
- 12.16_Introduction to Bokeh

Note: Please download the pdf files for each topics mentioned above from the Reference Material section.

Key Takeaways

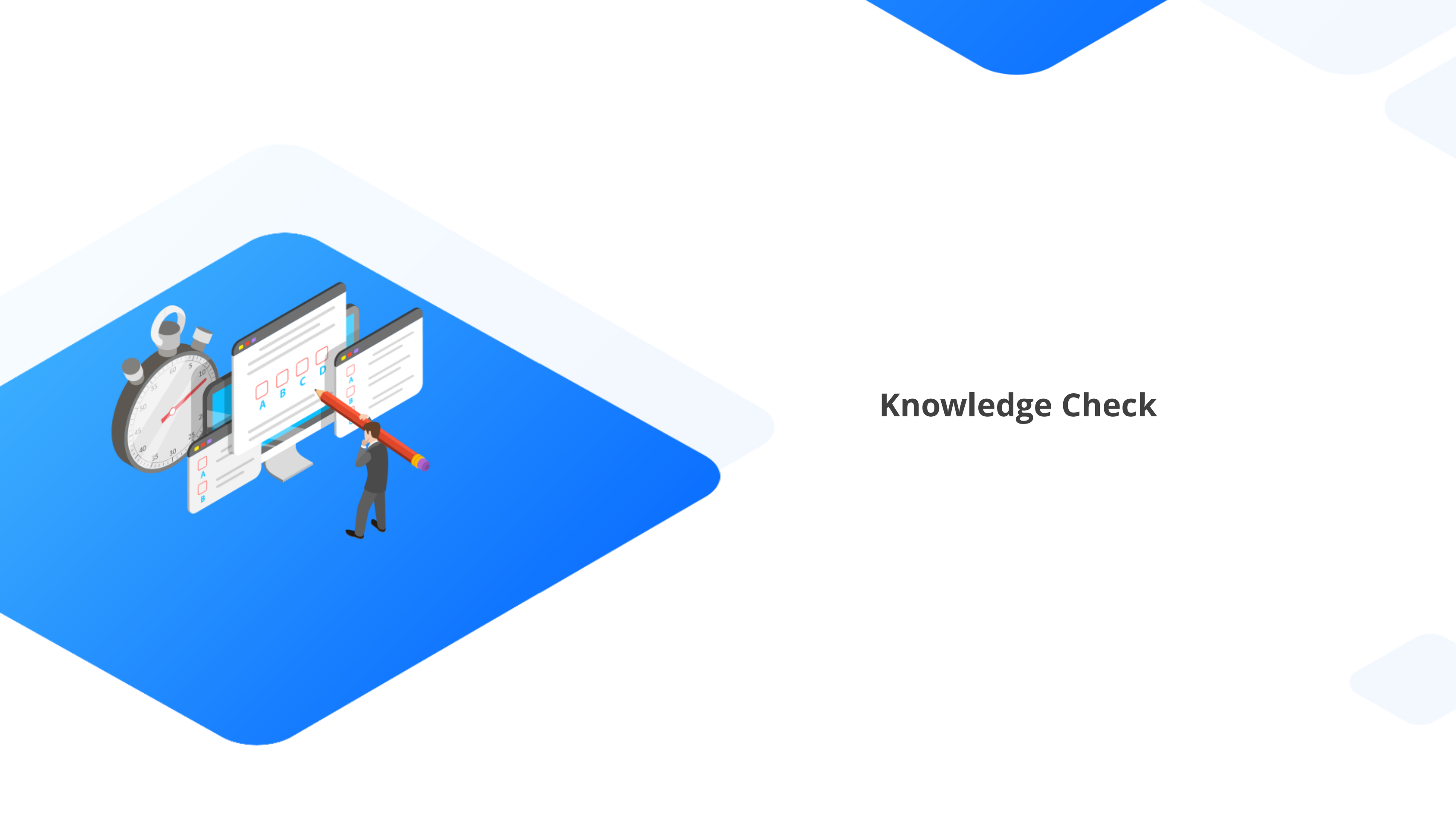
- Information visualization principles are categorized as graphical integrity, data-ink, chart junk, data density and small multiples.
- Matplotlib is one of the earliest and most comprehensive libraries for developing static, animated, and interactive visualizations in Python.
- Matplotlib helps the user in drawing Scatter plots, Bar charts, Histograms, and Pie charts.
- Plotly allows us to perform several graphic-based transformations.



Key Takeaways

- Seaborn is a high-level library for preparing statistical graphics and makes visualization a central part of exploring and understanding complex data sets.
- Bokeh is an interactive visualization library in Python that provides graphs on modern web browsers.
- Bokeh's plotting interface supports many different Glyphs, like bars, lines, hex tiles, or different polygon shapes.





Knowledge Check

Knowledge Check

1

Which of the following pertains to excessive and unnecessary use of graphical effects?

- A. Data-Ink
- B. Chart Junk
- C. Data Density
- D. Small Multiples



Knowledge Check

1

Which of the following pertains to excessive and unnecessary use of graphical effects?

- A. Data-Ink
- B. Chart Junk
- C. Data Density
- D. Small Multiples

The correct answer is **B**

Chart Junk pertains to excessive and unnecessary use of graphical effects.



**Knowledge
Check**
2

Which of the following helps to build graphics, ranging from simple plots to complex dashboards, with streaming datasets?

- A. Plotly
- B. Bokeh
- C. Seaborn
- D. None of the above



**Knowledge
Check**
2

Which of the following helps to build graphics, ranging from simple plots to complex dashboards, with streaming datasets?

- A. Plotly
- B. Bokeh
- C. Seaborn
- D. None of the above

The correct answer is **B**

Bokeh helps to build graphics, ranging from simple plots to complex dashboards, with streaming datasets.



Knowledge Check

3

What is the most important feature of Seaborn?

- A. Providing example datasets for learning Seaborn's statistical plotting techniques
- B. Creating interactive visualizations for modern web browsers
- C. Specializing in plotting statistical data and providing statistical estimation while plotting
- D. Providing a variety of charts for building interactive and publication-worthy graphs



Knowledge Check

3

What is the most important feature of Seaborn?

- A. Providing example datasets for learning Seaborn's statistical plotting techniques
- B. Creating interactive visualizations for modern web browsers
- C. Specializing in plotting statistical data and providing statistical estimation while plotting
- D. Providing a variety of charts for building interactive and publication-worthy graphs

The correct answer is **C**

Seaborn's most important feature is its specialization in plotting statistical data.





Thank You