

Seaborn

What is Seaborn

- Seaborn is a Python data visualization library based on Matplotlib that provides a high-level interface for creating attractive and informative statistical graphics.
- It is particularly useful for creating complex visualizations with concise syntax.
- Seaborn comes with several built-in themes and color palettes to make it easy to create aesthetically pleasing plots.

Installation of seaborn library for Python

```
In [1]: !pip install seaborn
```

```
Requirement already satisfied: seaborn in c:\users\teks108\anaconda3\lib\site-packages (0.12.2)
Requirement already satisfied: numpy!=1.24.0,>=1.17 in c:\users\teks108\anaconda3\lib\site-packages (from seaborn) (1.24.3)
Requirement already satisfied: pandas>=0.25 in c:\users\teks108\anaconda3\lib\site-packages (from seaborn) (2.0.3)
Requirement already satisfied: matplotlib!=3.6.1,>=3.1 in c:\users\teks108\anaconda3\lib\site-packages (from seaborn) (3.7.2)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\teks108\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (1.0.5)
Requirement already satisfied: cycler>=0.10 in c:\users\teks108\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\teks108\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (4.25.0)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\teks108\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (1.4.4)
Requirement already satisfied: packaging>=20.0 in c:\users\teks108\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (23.1)
Requirement already satisfied: pillow>=6.2.0 in c:\users\teks108\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (9.4.0)
Requirement already satisfied: pyparsing<3.1,>=2.3.1 in c:\users\teks108\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\teks108\anaconda3\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\teks108\anaconda3\lib\site-packages (from pandas>=0.25->seaborn) (2023.3.post1)
Requirement already satisfied: tzdata>=2022.1 in c:\users\teks108\anaconda3\lib\site-packages (from pandas>=0.25->seaborn) (2023.3)
Requirement already satisfied: six>=1.5 in c:\users\teks108\anaconda3\lib\site-packages (from python-dateutil>=2.7->matplotlib!=3.6.1,>=3.1->seaborn) (1.16.0)
```

```
In [15]: #Loading the dataset in seaborn package
import seaborn as sns

# Get a list of available datasets in Seaborn
available_datasets = sns.get_dataset_names()
```

```
# Print the list of datasets
print("Available Seaborn Datasets:")
print(available_datasets)
```

Available Seaborn Datasets:

```
['anagrams', 'anscombe', 'attention', 'brain_networks', 'car_crashes', 'diamonds', 'd
ots', 'dowjones', 'exercise', 'flights', 'fmri', 'geyser', 'glue', 'healthexp', 'iri
s', 'mpg', 'penguins', 'planets', 'seance', 'taxis', 'tips', 'titanic']
```

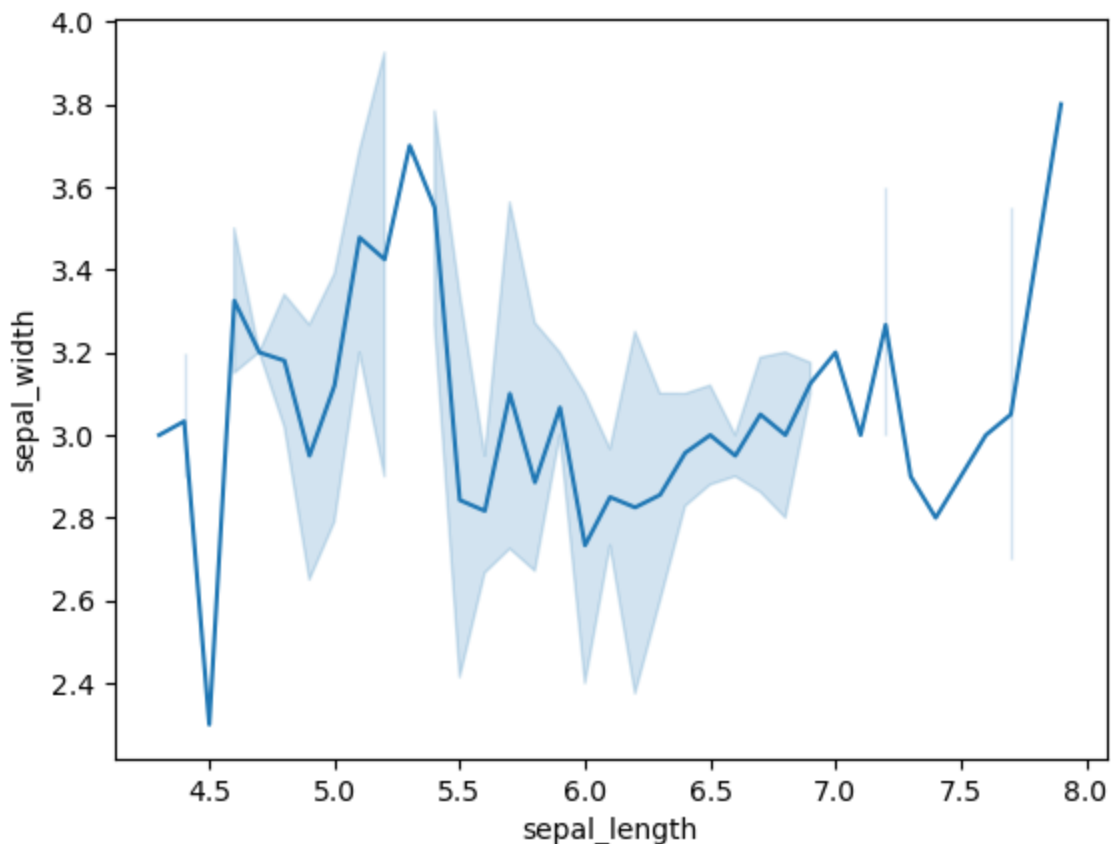
lineplot

```
In [2]: # importing packages
import seaborn as sns

# Loading dataset
data = sns.load_dataset("iris")

# draw lineplot
sns.lineplot(x="sepal_length", y="sepal_width", data=data)
```

```
Out[2]: <Axes: xlabel='sepal_length', ylabel='sepal_width'>
```



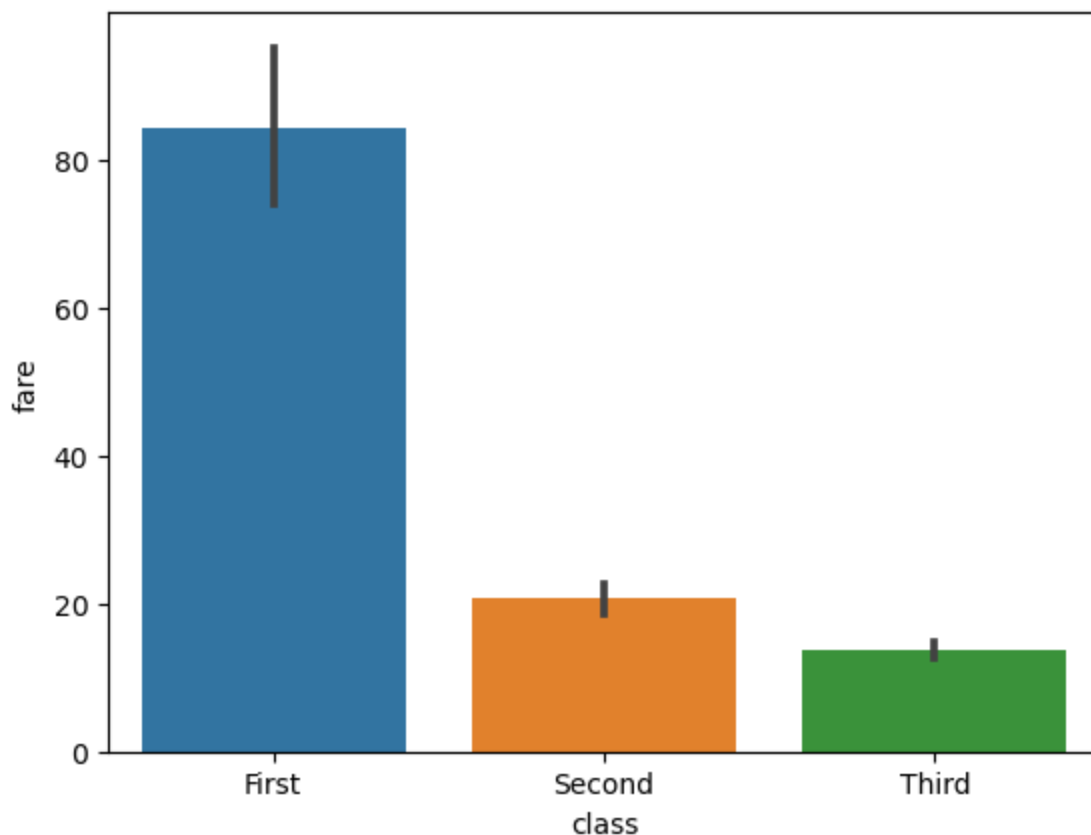
Barplot

```
In [3]: # importing the required library
import seaborn as sns
import matplotlib.pyplot as plt
```

```
# read a titanic.csv file
# from seaborn library
df = sns.load_dataset('titanic')

# class v / s fare barplot
sns.barplot(x = 'class', y = 'fare', data = df)

# Show the plot
plt.show()
```



Countplot()

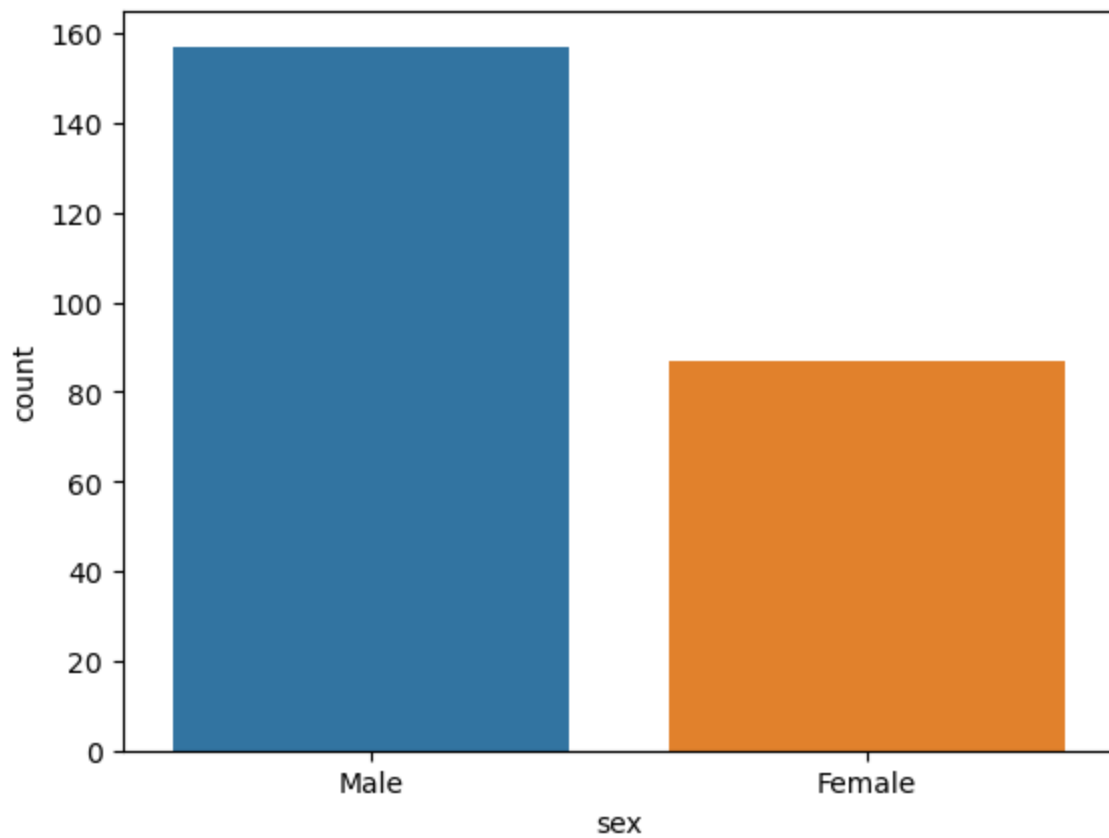
```
In [4]: # importing the required library

import seaborn as sns
import matplotlib.pyplot as plt

# read a tips.csv file from seaborn library
df = sns.load_dataset('tips')

# count plot on single categorical variable
sns.countplot(x = 'sex', data = df)

# Show the plot
plt.show()
```



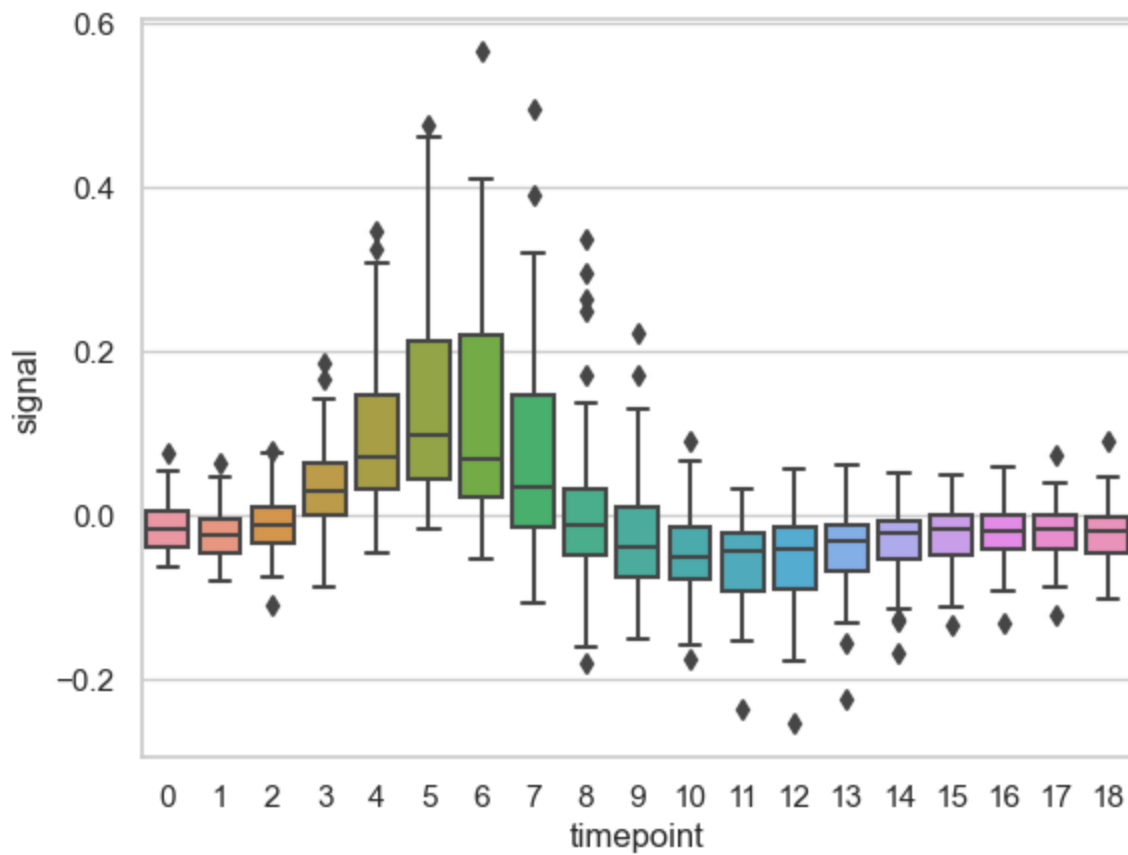
Boxplot

```
In [6]: import seaborn

seaborn.set(style='whitegrid')
fmri = seaborn.load_dataset("fmri")

seaborn.boxplot(x="timepoint", y="signal", data=fmri)

Out[6]: <Axes: xlabel='timepoint', ylabel='signal'>
```



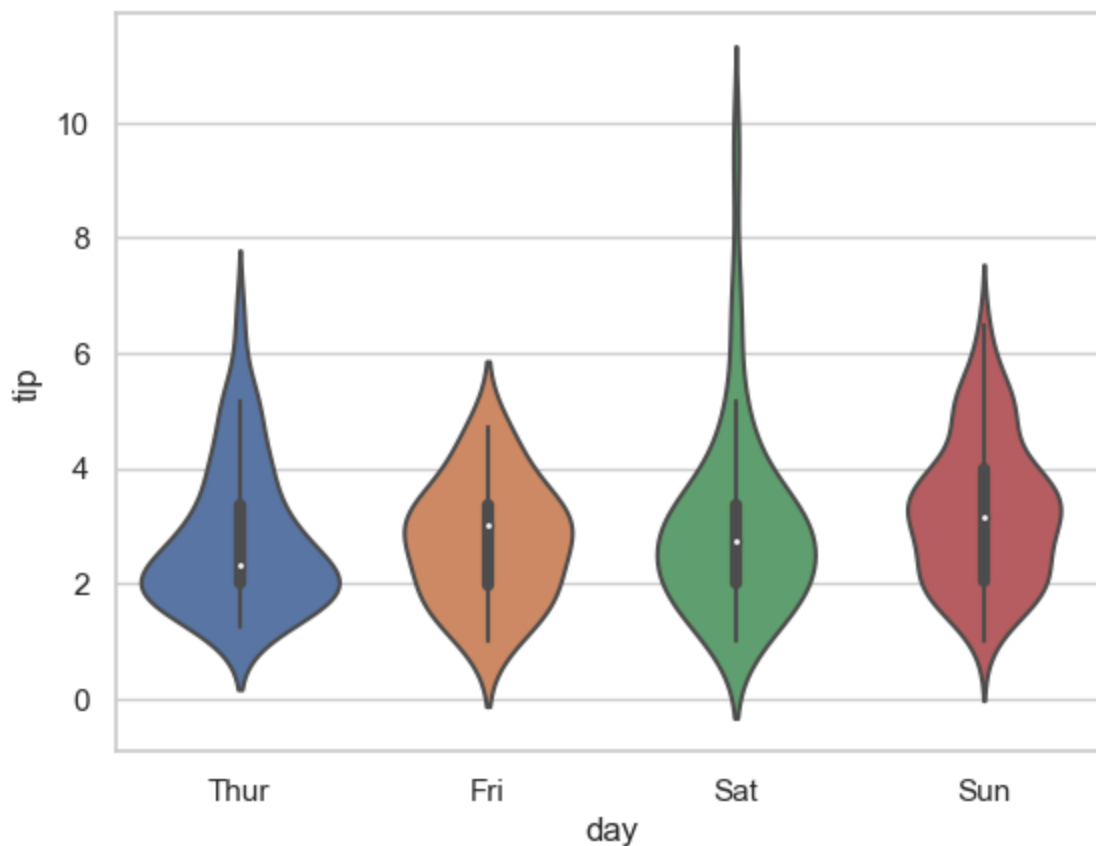
Violin plot

In [7]: `import seaborn`

```
seaborn.set(style = 'whitegrid')  
tip = seaborn.load_dataset('tips')
```

```
seaborn.violinplot(x = 'day', y = 'tip', data = tip)
```

Out[7]: `<Axes: xlabel='day', ylabel='tip'>`



In []:

seaborn.factorplot()

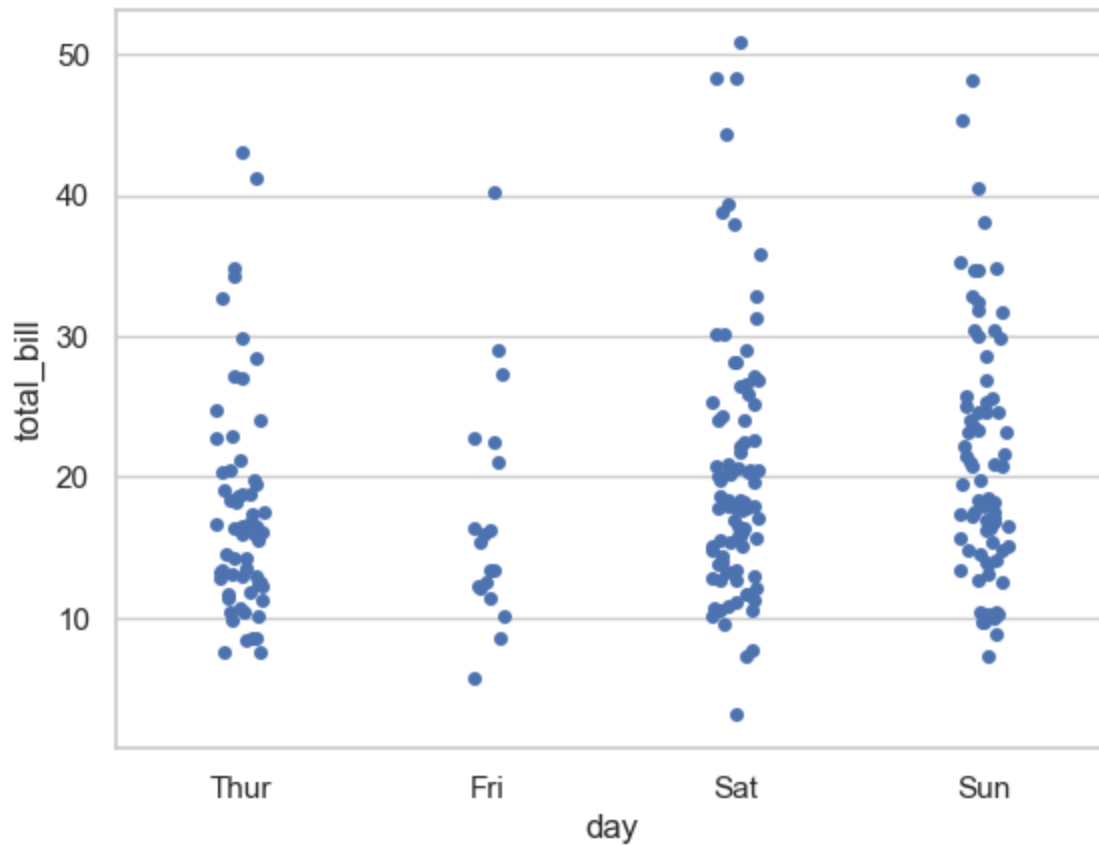
- This method is used to draw a categorical plot onto a FacetGrid.

```
In [8]: import seaborn
import matplotlib.pyplot as plt

seaborn.set(style = 'whitegrid')
tip = seaborn.load_dataset("tips")

seaborn.stripplot(x="day", y="total_bill", data=tip)

plt.show()
```



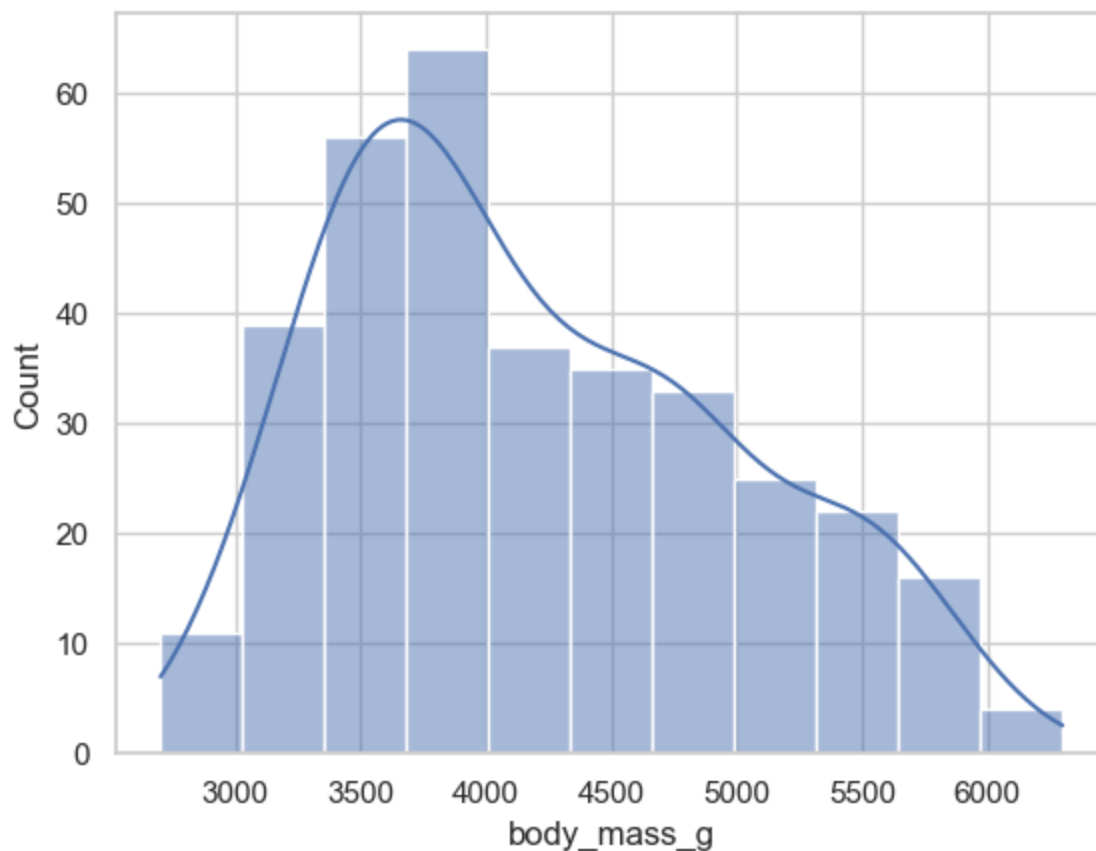
In []:

Histplot

```
In [10]: # Load dataset
penguins = sns.load_dataset("penguins")

# Plot histogram
sns.histplot(data = penguins, x = "body_mass_g", kde = True)
```

```
Out[10]: <Axes: xlabel='body_mass_g', ylabel='Count'>
```



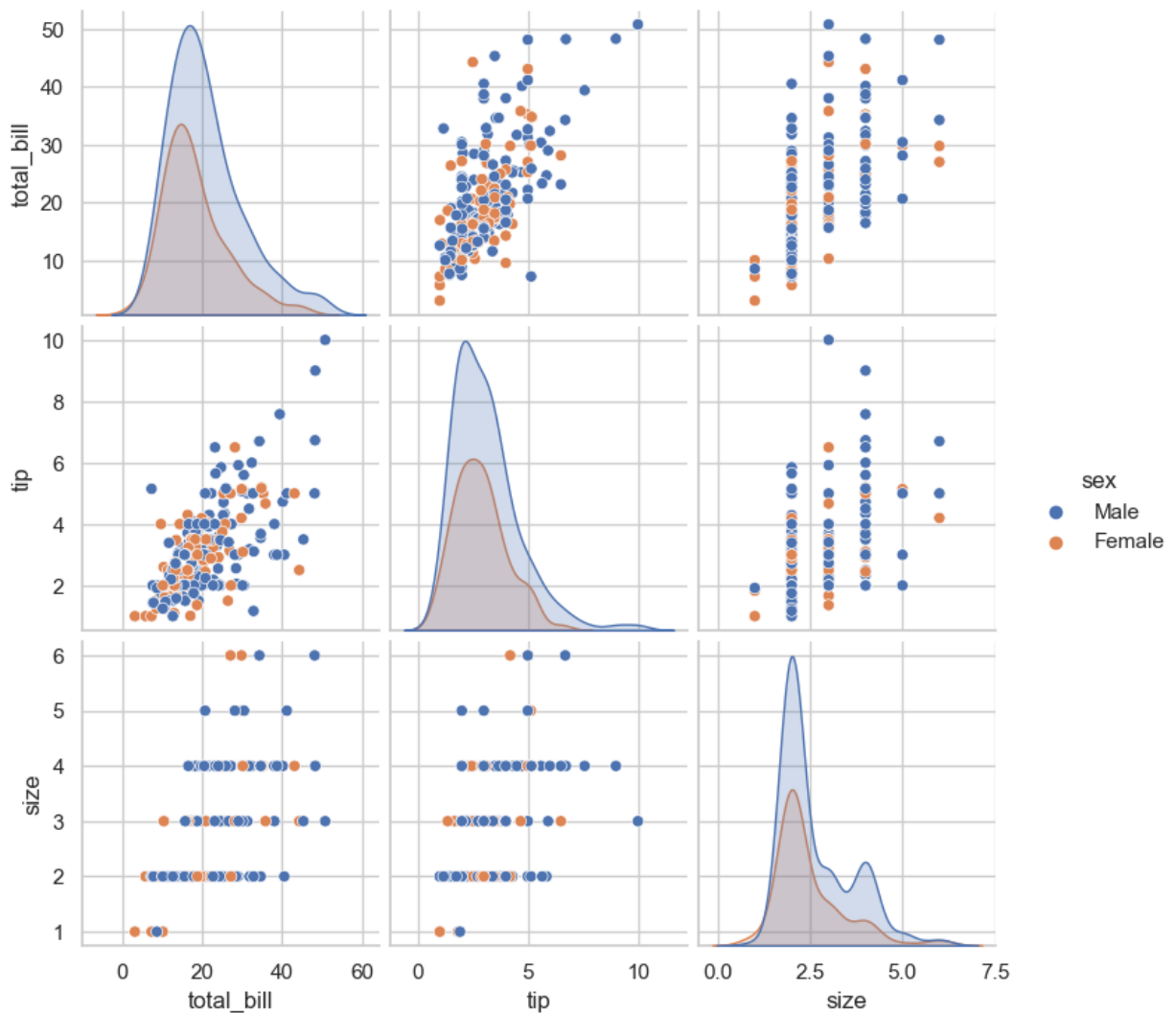
In []:

seaborn.pairplot() :

- To plot multiple pairwise bivariate distributions in a dataset,

```
In [11]: df = seaborn.load_dataset('tips')
# pairplot with hue sex
seaborn.pairplot(df, hue='sex')
# to show
plt.show()
```

C:\Users\TEKS108\anaconda3\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight
self._figure.tight_layout(*args, **kwargs)



Heatmap

- Heatmap is defined as a graphical representation of data using colors to visualize the value of the matrix.

Important Parameters:

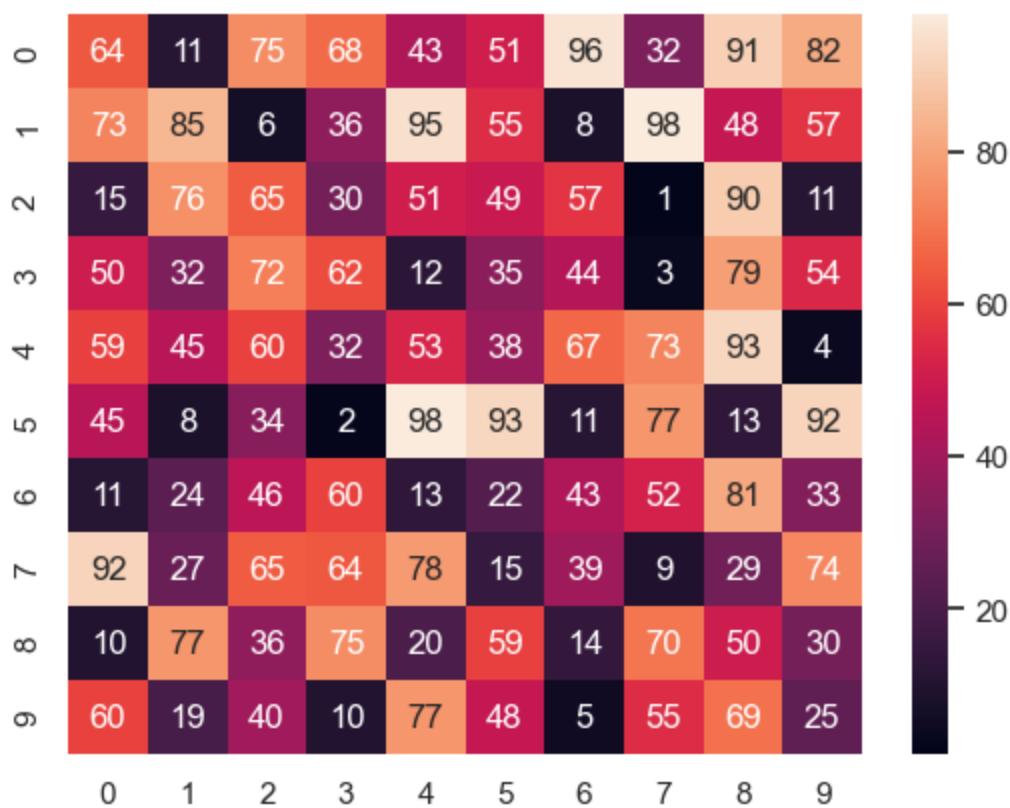
- data: 2D dataset that can be coerced into an ndarray.
- vmin, vmax: Values to anchor the colormap, otherwise they are inferred from the data and other keyword arguments.
- cmap: The mapping from data values to color space.
- center: The value at which to center the colormap when plotting divergent data.
- annot: If True, write the data value in each cell.
- fmt: String formatting code to use when adding annotations.
- linewidths: Width of the lines that will divide each cell.
- linecolor: Color of the lines that will divide each cell.
- cbar: Whether to draw a colorbar.

```
In [13]: # generating 2-D 10x10 matrix of random numbers
# from 1 to 100
data = np.random.randint(low=1, high=100, size=(10, 10))

# setting the parameter values
annot = True

# plotting the heatmap
hm = sns.heatmap(data=data, annot=annot)

# displaying the plotted heatmap
plt.show()
```



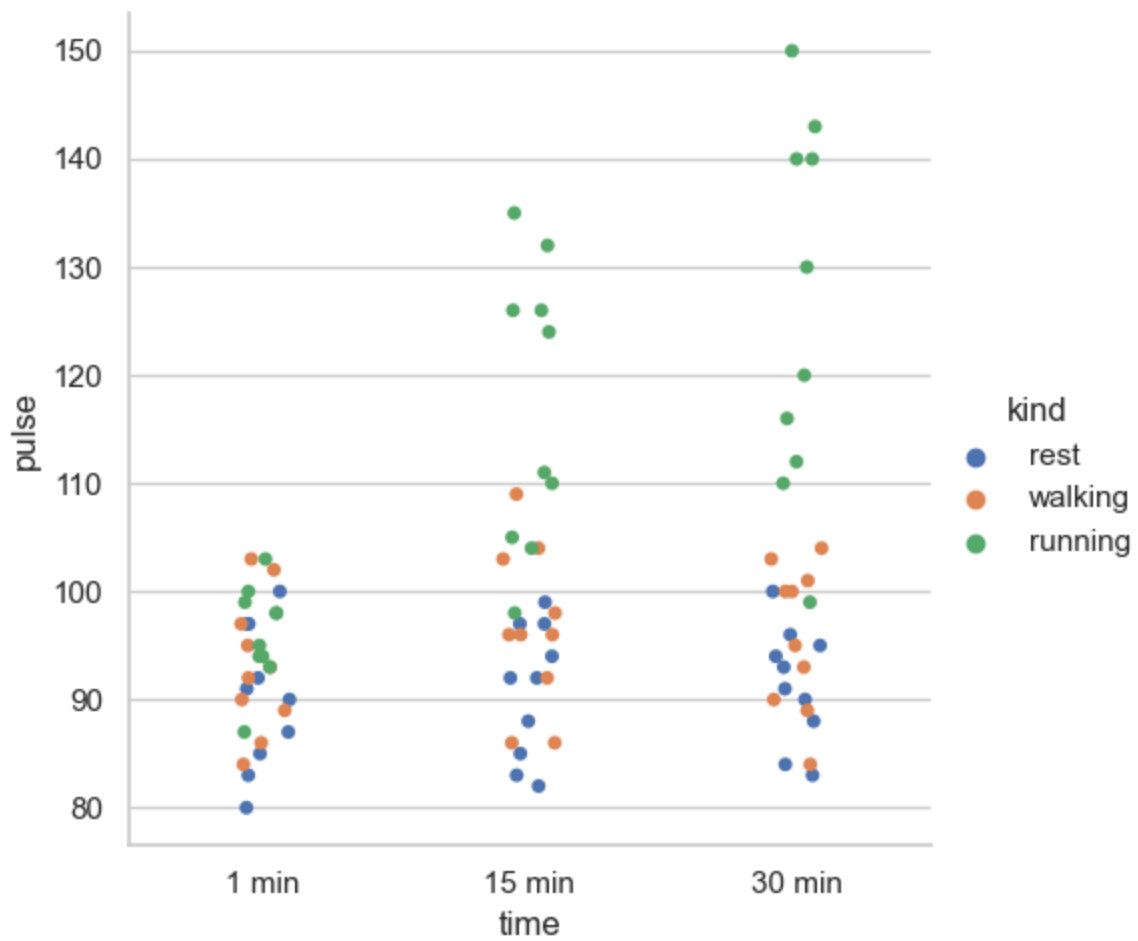
Catplot()

- The relationship between multiple variables in a dataset that has been categorized in some way.

```
In [14]: import seaborn as sns

exercise = sns.load_dataset("exercise")
g = sns.catplot(x="time", y="pulse", hue="kind", data=exercise)
```

C:\Users\TEKS108\anaconda3\Lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight
self._figure.tight_layout(*args, **kwargs)



In []: