




Abdullah Khaled

 [linkedin.com/in/abdullah-khaled-0608a9236](https://www.linkedin.com/in/abdullah-khaled-0608a9236)

Part 1

The **MOST** used Seaborn plots



Seaborn

March 11, 2023

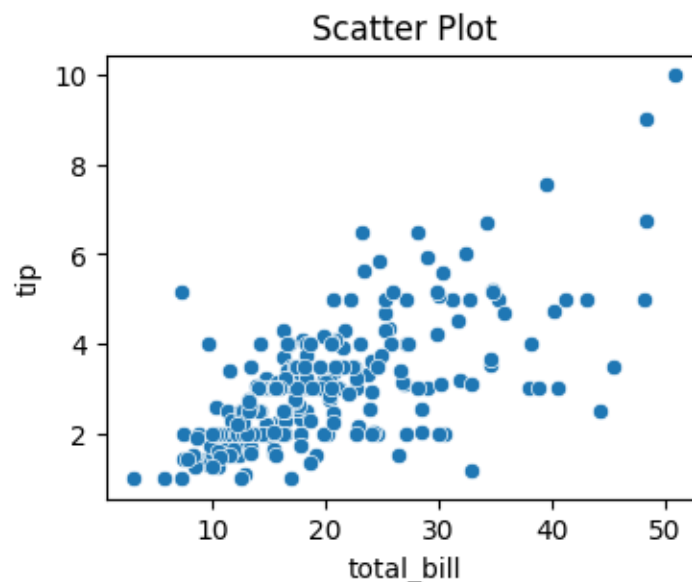
```
[1]: import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')
%matplotlib inline
```

```
[2]: # Load Data
tips = sns.load_dataset('tips')
iris = sns.load_dataset('iris')
```

0.1 1- Scatter Plot:

A scatter plot is a two-dimensional plot that uses dots to represent the values for two different variables. It is useful for visualizing the relationship between two continuous variables. Seaborn's `scatterplot()` function can be used to create scatter plots, with optional customization of colors, size, and shape of the markers.

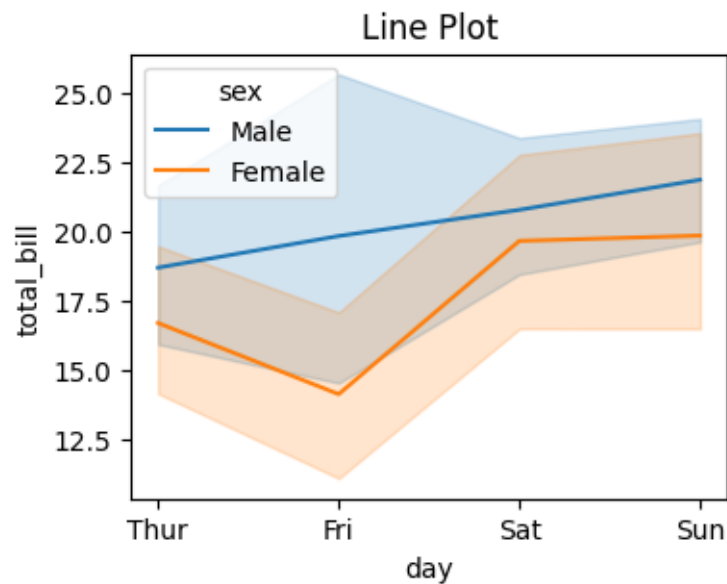
```
[3]: plt.figure(figsize=(4, 3))
sns.scatterplot(data=tips, x='total_bill', y='tip')
plt.title('Scatter Plot')
plt.show()
```



0.2 2- Line Plot:

A line plot is a graph that displays data points connected by straight lines. It is useful for showing trends over time or comparing changes in different groups. Seaborn's `lineplot()` function can be used to create line plots, with optional customization of colors, markers, and line styles.

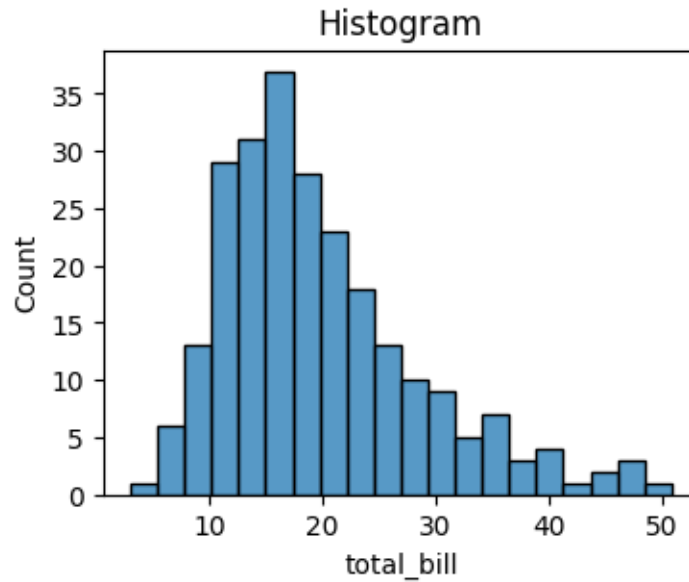
```
[4]: plt.figure(figsize=(4, 3))
sns.lineplot(data=tips, x='day', y='total_bill', hue='sex')
plt.title('Line Plot')
plt.show()
```



0.3 3- Histogram:

A histogram is a graphical representation of the distribution of a dataset. It divides the range of values into a set of intervals, and then counts the number of values that fall into each interval. Seaborn's `histplot()` function can be used to create histograms, with optional customization of bin size, color, and normalization.

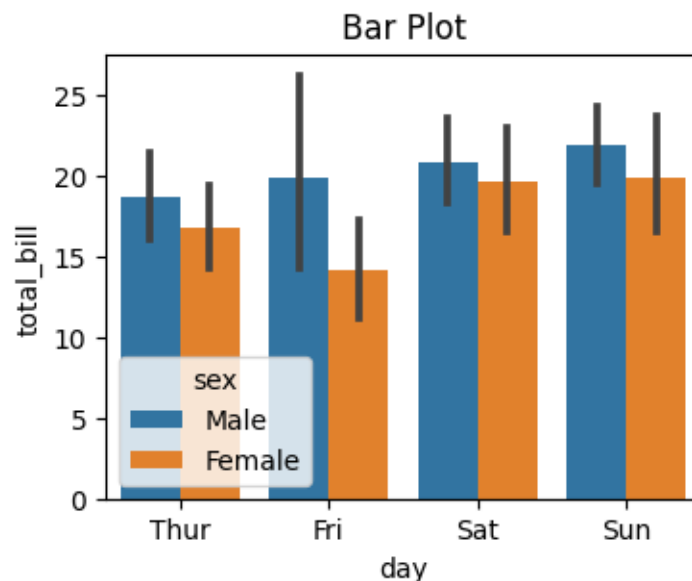
```
[5]: plt.figure(figsize=(4, 3))
sns.histplot(data=tips, x='total_bill', bins=20)
plt.title('Histogram')
plt.show()
```



0.4 4- Bar Plot:

A bar plot is a graph that displays categorical data with rectangular bars. It is useful for comparing the values of different categories. Seaborn's `barplot()` function can be used to create bar plots, with optional customization of colors, error bars, and confidence intervals.

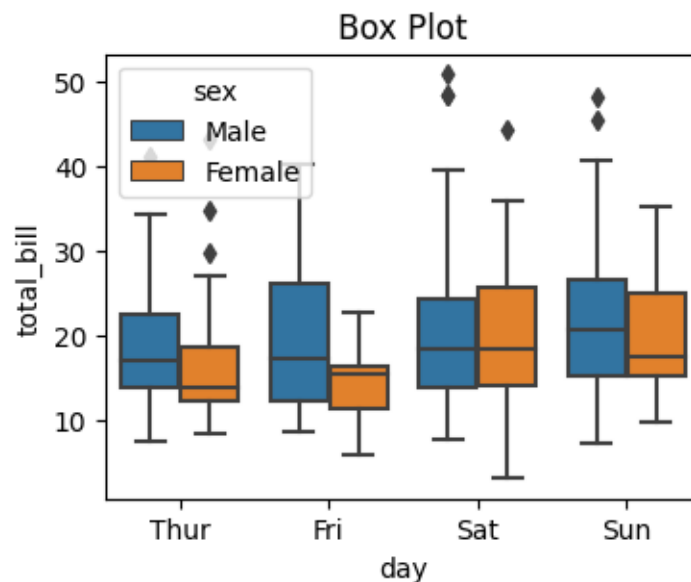
```
[6]: plt.figure(figsize=(4, 3))
sns.barplot(data=tips, x='day', y='total_bill', hue='sex')
plt.title('Bar Plot')
plt.show()
```



0.5 5- Box Plot:

A box plot is a graph that displays the distribution of a dataset using five summary statistics: minimum, first quartile, median, third quartile, and maximum. It is useful for identifying outliers and comparing the distributions of different groups. Seaborn's `boxplot()` function can be used to create box plots, with optional customization of colors, notch, and whisker length.

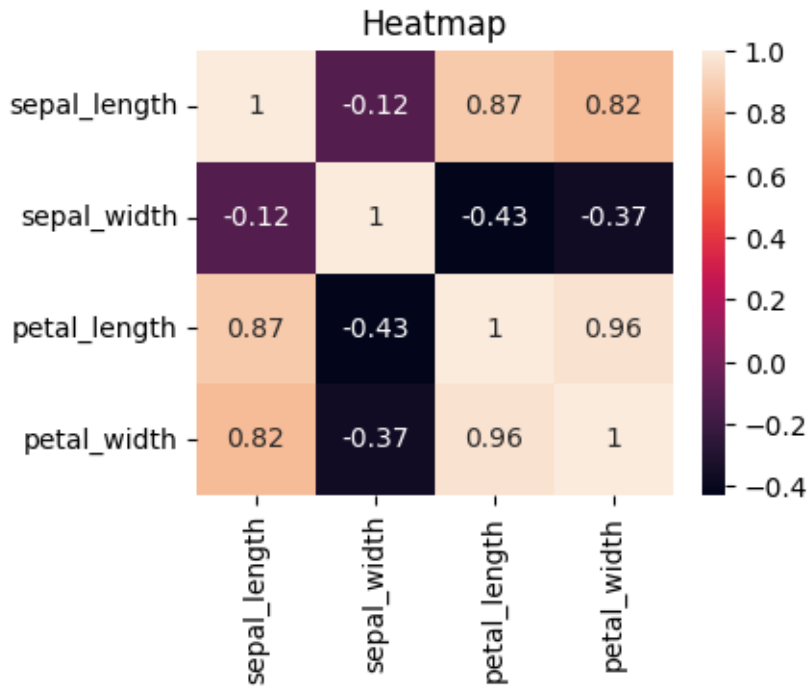
```
[7]: plt.figure(figsize=(4, 3))
sns.boxplot(data=tips, x='day', y='total_bill', hue='sex')
plt.title('Box Plot')
plt.show()
```



0.6 6- Heatmap:

A heatmap is a graphical representation of a matrix of values, where the values are represented as colors. It is useful for visualizing patterns in large datasets. Seaborn's `heatmap()` function can be used to create heatmaps, with optional customization of colors, annotation, and clustering.

```
[8]: plt.figure(figsize=(4, 3))
sns.heatmap(data=iris.corr(), annot=True)
plt.title('Heatmap')
plt.show()
```

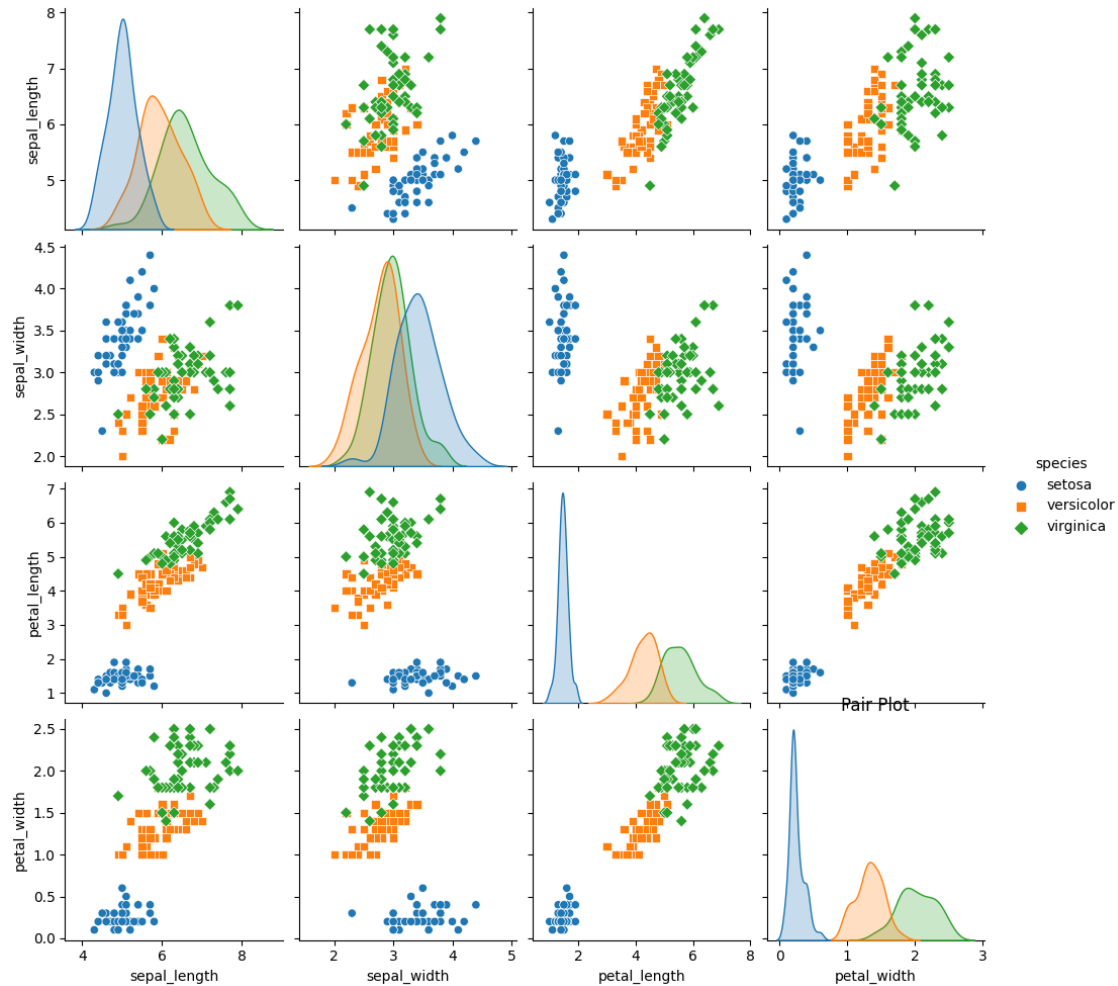


0.7 7- Pair Plot:

A pair plot is a graph that displays pairwise relationships between variables in a dataset. It is useful for exploring the correlations between different variables. Seaborn's `pairplot()` function can be used to create pair plots, with optional customization of diagonal plots, colors, and markers.

```
[9]: plt.figure(figsize=(4, 3))
sns.pairplot(data=iris, hue='species', markers=['o', 's', 'D'])
plt.title('Pair Plot')
plt.show()
```

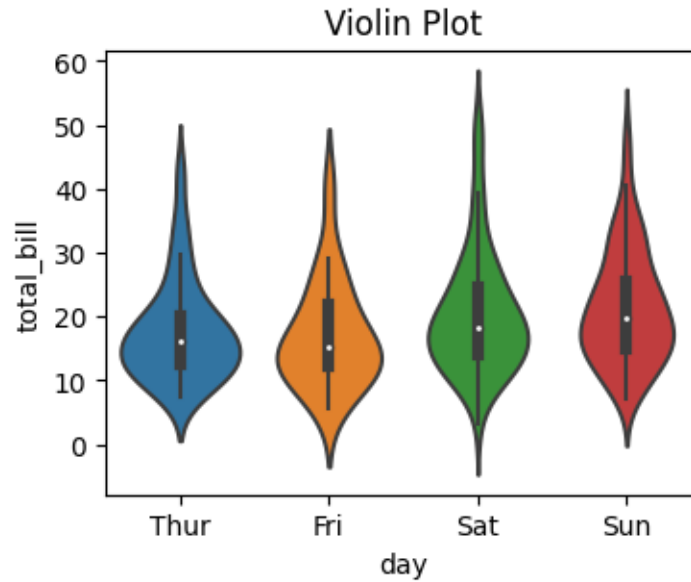
<Figure size 400x300 with 0 Axes>



0.8 8- Violin Plot:

A violin plot is a combination of a box plot and a kernel density plot. It displays the distribution of a dataset with a combination of a box plot and a density plot, where the density plot shows the estimated probability density function. Seaborn's `violinplot()` function can be used to create violin plots, with optional customization of colors, width, and inner quartile range.

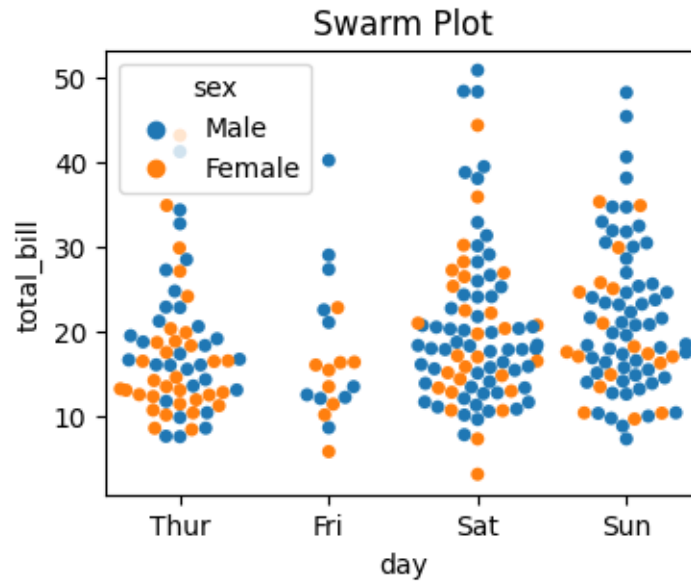
```
[10]: plt.figure(figsize=(4, 3))
sns.violinplot(x="day", y="total_bill", data=tips)
plt.title('Violin Plot')
plt.show()
```



0.9 9- Swarm Plot:

A swarm plot is a graph that displays the distribution of a dataset by placing each data point on a line. It is useful for showing the density of data points and highlighting potential outliers. Seaborn's `swarmplot()` function can be used to create swarm plots, with optional customization of colors and size.

```
[11]: plt.figure(figsize=(4, 3))
sns.swarmplot(data=tips, x='day', y='total_bill', hue='sex')
plt.title('Swarm Plot')
plt.show()
```

0.10 10- Joint Plot:

A joint plot is a graph that displays the relationship between two variables using both a scatter plot and a histogram. It is useful for exploring the correlations between different variables and identifying patterns in the data. Seaborn's `jointplot()` function can be used to create joint plots, with optional customization of colors, markers, and regression lines.

```
[12]: plt.figure(figsize=(4, 3))
sns.jointplot(data=tips, x='total_bill', y='tip', kind='reg')
plt.title('Joint Plot')
plt.show()
```

<Figure size 400x300 with 0 Axes>

