

▾ Lecture-19 and LAB 8

Advanced Data Visualization with SEABORN (Part-II)



▾ REVIEW: Features of Seaborn

- Python Library for **STATISTICAL GRAPHICAL VISUALIZATION AND PLOTTING**
- **TYPES OF PLOTS** which can be obtained using Seaborn:

1. RELATIONAL PLOTS

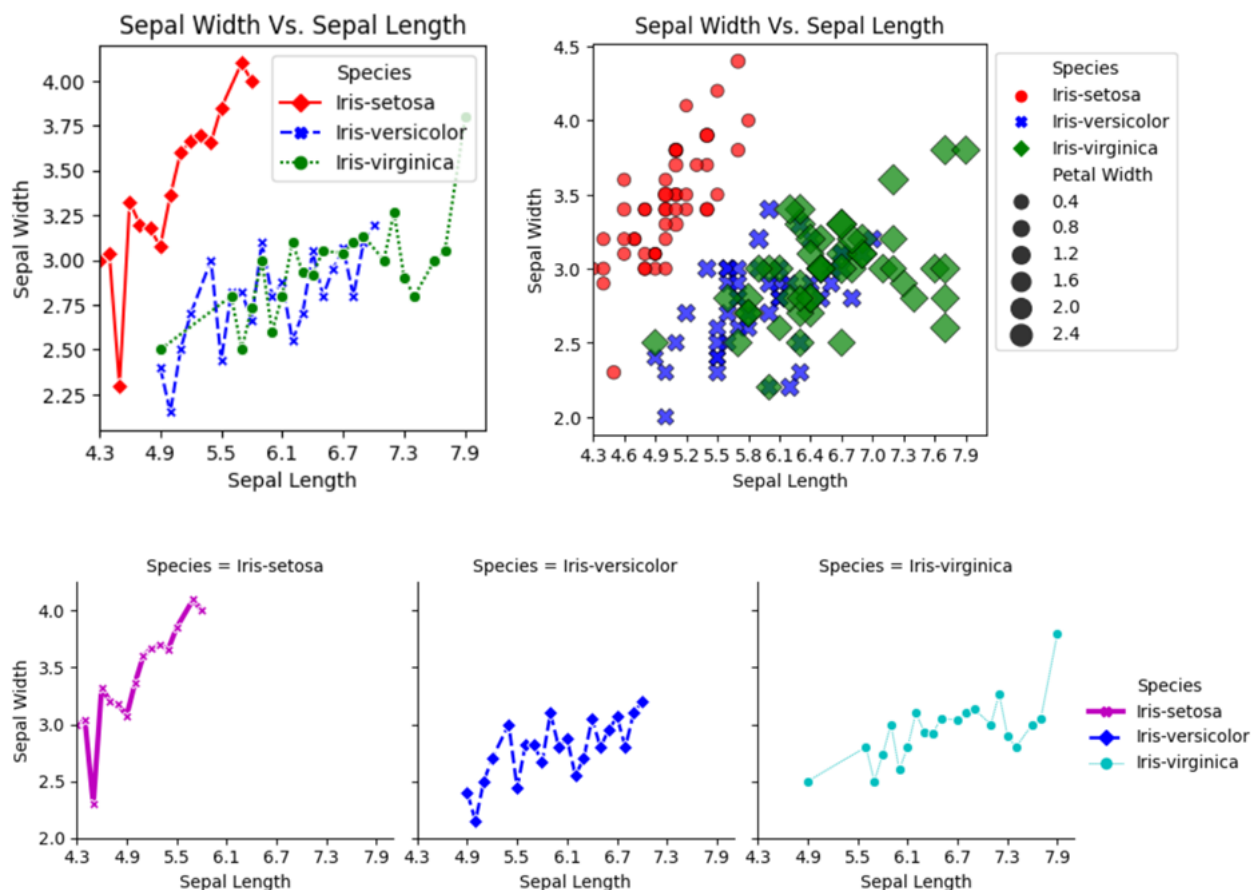
Relational plots are standard types of plots which are used for visualizing the statistical relationship between two quantities (variables)

In Relational Plots, the two quantities between which the relationship is studied are generally of 'NUMERICAL' type

- Types of Relational plots in Seaborn:
 - **LINE PLOT**: `sns.lineplot()`
 - **SCATTER PLOT & BUBBLE CHART**: `sns.scatterplot()`
 - **REL PLOT**: `sns.relplot()`

IRIS FLOWER DATASET (EXAMPLE):

	A	B	C	D	E	F
1	S.No.	Sepal Length	Sepal Width	Petal Length	Petal Width	Species
2	1	5.1	3.5	1.4	0.2	Iris-setosa
3	2	4.9	3	1.4	0.2	Iris-setosa
4	3	4.7	3.2	1.3	0.2	Iris-setosa
5	4	4.6	3.1	1.5	0.2	Iris-setosa
6	5	5	3.6	1.4	0.2	Iris-setosa
7	6	5.4	3.9	1.7	0.4	Iris-setosa
8	7	4.6	3.4	1.4	0.3	Iris-setosa
9	8	5	3.4	1.5	0.2	Iris-setosa
10	9	4.4	2.9	1.4	0.2	Iris-setosa
11	10	4.9	3.1	1.5	0.1	Iris-setosa



2. CATEGORICAL PLOTS

Categorical Plots are types of plots which are used for visualizing the statistical relationship between two quantities (variables) in which at least one variable is a **CATEGORICAL VARIABLE (or Qualitative Variable)**

- Types of Categorical plots in Seaborn:
 - BAR PLOT:** `sns.barplot()`
 - COUNT PLOT:** `sns.countplot()`
 - BOX PLOT:** `sns.boxplot()`
 - STRIP PLOT:** `sns.stripplot()`
 - SWARM PLOT:** `sns.swarmplot()`

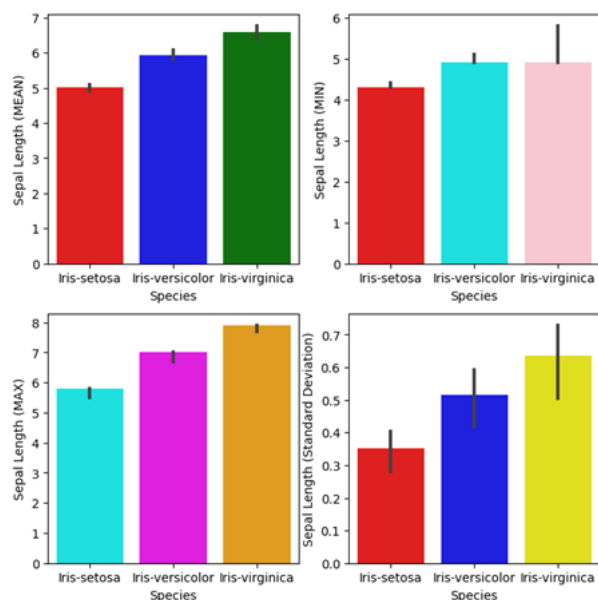
BAR PLOT

(Statistical BAR Plot using Seaborn)

`sns.barplot()`

Statistical Bar Plot (columnar plot) is used for analysis and visualization of variable in the form of VERTICAL BARS

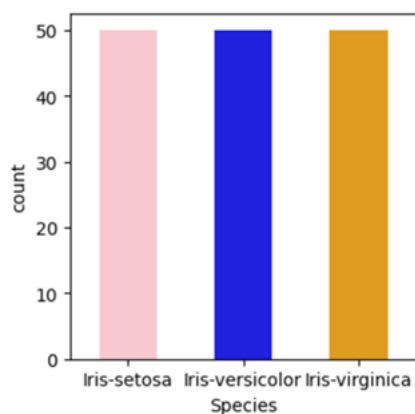
- Generally, a BAR PLOT is used to analyse and plot relationship between a **Numerical Type variable (along the y axis) Vs. a Categorical Type variable (along the x-axis)**
- In BAR PLOT, obtained using Seaborn Library, HEIGHT of the bar may represent COUNT, MEAN, MINIMUM VALUE, MAXIMUM VALUE or any other statistical value of a quantity



▼ COUNT PLOT

sns.countplot()

A countplot basically counts the number of occurrence of different values of a CATEGORICAL VARIABLE



▼ 6. Box Plot

sns.boxplot()

A **box and whisker plot** (also referred to as box plot) displays the five-number statistical summary of a set of data (variable).

The values of a given variable are summarised in the form of the following five statistical parameters :

- **Minimum**
- **First Quartile (25th Percentile)**
- **Median (50th Percentile)**
- **Third Quartile (75th Percentile)**
- **Maximum**

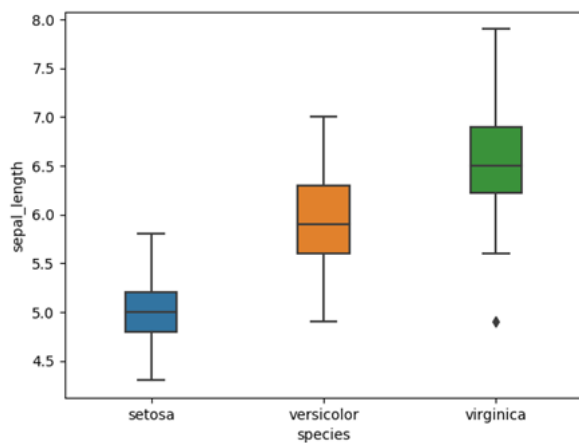
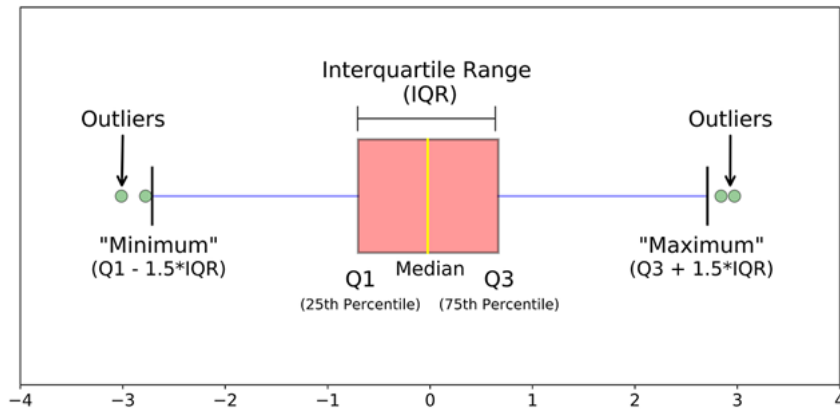
A box plot is plotted as follows:

- we draw a box from the first quartile to the third quartile.
- A vertical line goes through the box at the **median**.
- The whiskers go from each quartile to the **minimum or maximum**.

InterQuartile Range and Outliers in the BoxPlot

- **InterQuartile Range = IQR = Third Quartile - First Quartile**
- In the BOXPLOT, **the width of the BOX represents the Interquartile Range**

- Any datapoint whose value is 1.5 times the InterQuartile Range is referred to as an **Outlier**
- In a **BOXPLOT**, an **Outlier is represented by a marker (generally a CIRCLE)**



BOXPLOT 1

▼ sns.boxplot()

- Load **IRIS FLOWER** dataset
- BOXPLOT** for 'Sepal Length' for different species of **IRIS FLOWER**

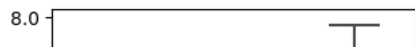
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

IRIS_DATA = pd.read_csv('Iris.csv')

plt.figure(figsize=(3.5,3.5))

sns.boxplot(x='Species',y='Sepal Length',data=IRIS_DATA,
            palette=['pink','cyan','blue'])

plt.show()
```



BOXPLOT 2

▼ sns.boxplot()

- Load IRIS FLOWER dataset
- BOXPLOT for 'Petal Length' for different species of IRIS FLOWER

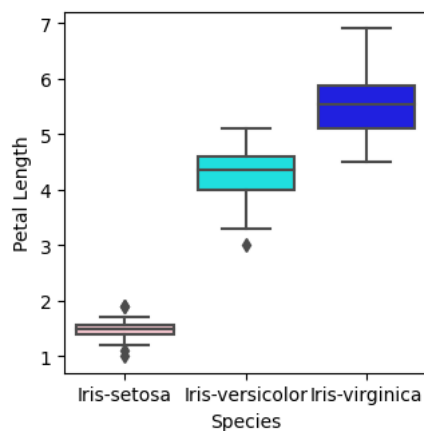
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

IRIS_DATA = pd.read_csv('Iris.csv')

plt.figure(figsize=(3.5,3.5))

sns.boxplot(x='Species',y='Petal Length',data=IRIS_DATA,
            palette=['pink','cyan','blue'])

plt.show()
```



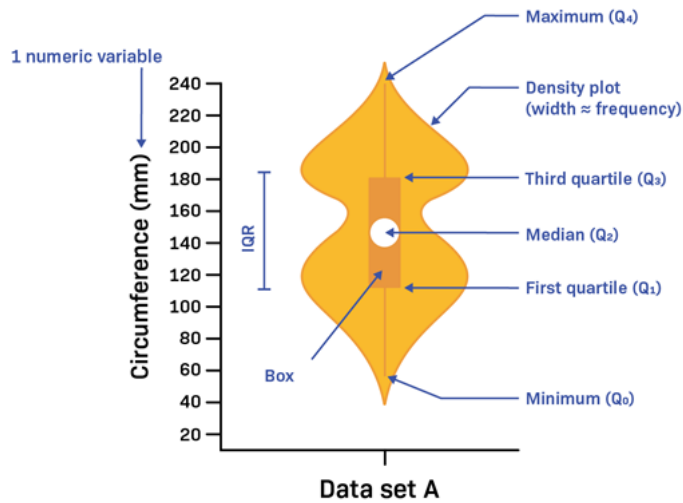
▼ 7. VIOLIN PLOT

sns.violinplot()

- A violin plot is a hybrid of a box plot and probability density plot. It displays the statistical summary of the dataset and also the probability distribution profile of the given data set
- Similar to a box plot, it shows statistical summary of a given variable in terms of
 - Minimum
 - First Quartile (25th Percentile)
 - Median (50th Percentile)
 - Third Quartile (75th Percentile)
 - Maximum

In addition to this, a violin plot also shows the probability distribution of a variable.

A Violin Plot is interpreted as shown below:



VIOLIN PLOT 1

▼ sns.violinplot()

- Load IRIS FLOWER dataset
- Display VIOLINPLOT for 'Sepal Length' for different species of IRIS FLOWER

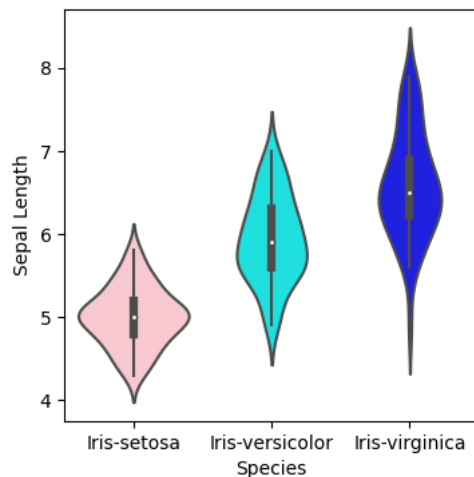
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

IRIS_DATA = pd.read_csv('Iris.csv')

plt.figure(figsize=(4,4))

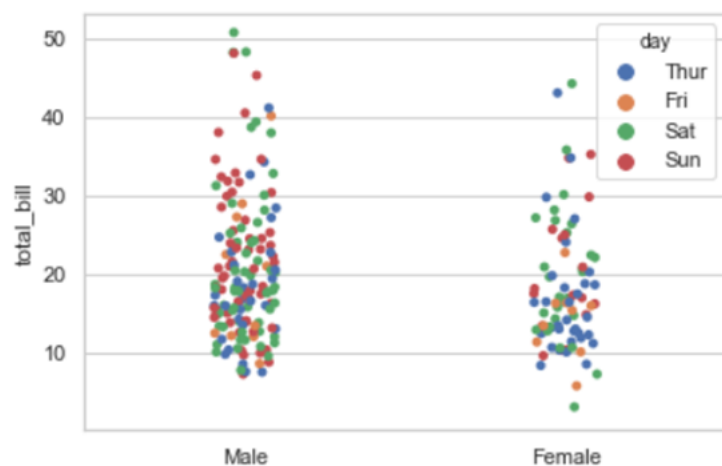
sns.violinplot(x='Species',y='Sepal Length',data=IRIS_DATA,
               palette=['pink','cyan','blue'])

plt.show()
```



▼ 8. STRIP PLOT

- Strip Plot (also referred to as Jitter Plot) is very similar to scatter plot
- In a Strip Plot, the collection (cloud of datapoints) representing a values of a variable appear to look like a 'STRIP' as shown below:



STRIP PLOT 1

▼ sns.stripplot()

- Load IRIS FLOWER dataset
- Display STRIP PLOT for 'Sepal Length' for different species of IRIS FLOWER

STRIP PLOT 1

STRIP PLOT for 'Sepal Length' for different species of IRIS FLOWER

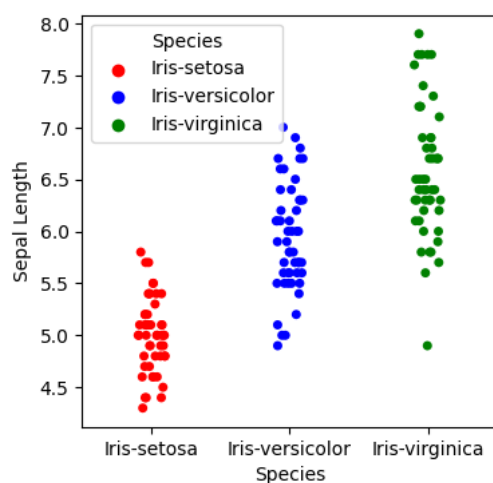
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

IRIS_DATA = pd.read_csv('Iris.csv')

plt.figure(figsize=(4,4))

sns.stripplot(x='Species',y='Sepal Length',data=IRIS_DATA,
             hue='Species',palette=['r','b','g'],size=5)

plt.show()
```



STRIP PLOT 2

STRIP PLOT for 'Sepal Length' for different species of IRIS FLOWER

- using 'size', 'edgecolor', 'linewidth' and 'alpha' parameters to format the datapoint markers
- using 'hue_order' parameter to adjust the order in which 'color' is assigned to different curves(plots)

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
```

```

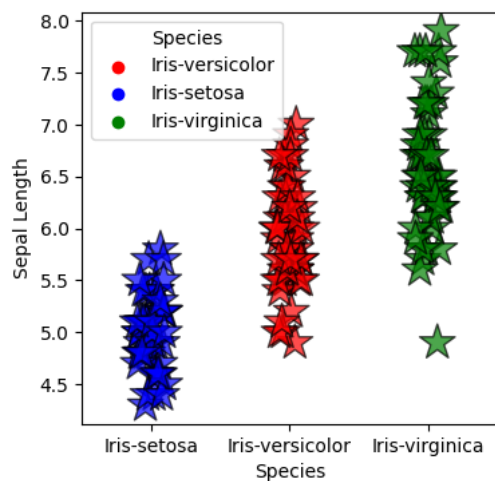
IRIS_DATA = pd.read_csv('Iris.csv')

plt.figure(figsize=(4,4))

sns.stripplot(x='Species',y='Sepal Length',data=IRIS_DATA,
              hue='Species',palette=['r','b','g'],
              hue_order=['Iris-versicolor','Iris-setosa','Iris-virginica'],
              marker='*',size=20,
              edgecolor='black',linewidth=1,
              alpha=0.7)

plt.show()

```



STRIP PLOT 3

STRIP PLOT for 'Sepal Length' for different species of IRIS FLOWER

- using 'jitter' parameter to align the datapoints

```

import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

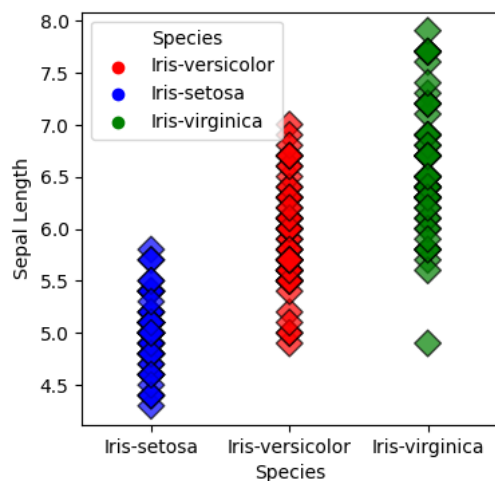
IRIS_DATA = pd.read_csv('Iris.csv')

plt.figure(figsize=(4,4))

sns.stripplot(x='Species',y='Sepal Length',data=IRIS_DATA,
              hue='Species',palette=['r','b','g'],
              hue_order=['Iris-versicolor','Iris-setosa','Iris-virginica'],
              marker='D',size=10,
              edgecolor='black',linewidth=1,
              alpha=0.7,
              jitter=False)

plt.show()

```



9. SWARM PLOT

sns.swarmplot()

A swarm plot is a type of scatter plot that is used for representing categorical values. It is very similar to the strip plot, but it avoids the overlapping of points.

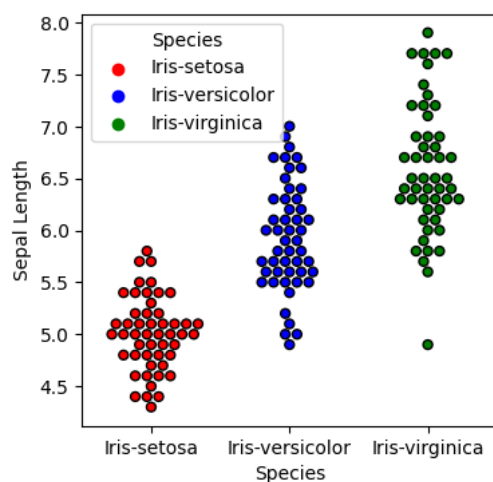
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

IRIS_DATA = pd.read_csv('Iris.csv')

plt.figure(figsize=(4,4))

sns.swarmplot(x='Species',y='Sepal Length',data=IRIS_DATA,
              hue='Species',palette=['r','b','g'],size=5,
              edgecolor='black', linewidth=1)

plt.show()
```



10. CAT PLOT

sns.catplot()

sns.catplot() is a seaborn library function which can be used to plot any of the following categorical plots:

- BAR Plot
- BOX Plot
- Strip Plot
- Swarm Plot

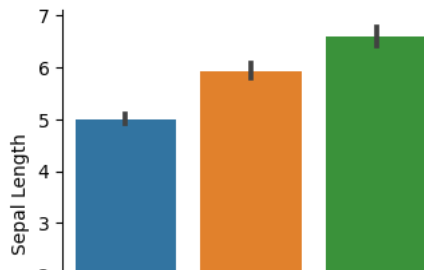
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

IRIS_DATA = pd.read_csv('Iris.csv')

sns.catplot(x='Species',y='Sepal Length',data=IRIS_DATA,
            kind='bar', estimator='mean',
            height=3.5,aspect=1)

plt.show()
```





```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
```

```
IRIS_DATA = pd.read_csv('Iris.csv')
```

```
plt.figure(figsize=(8,8))
```

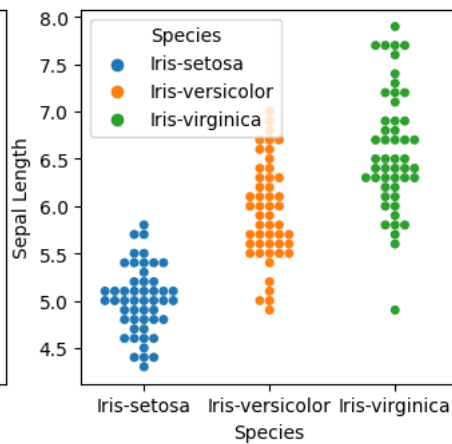
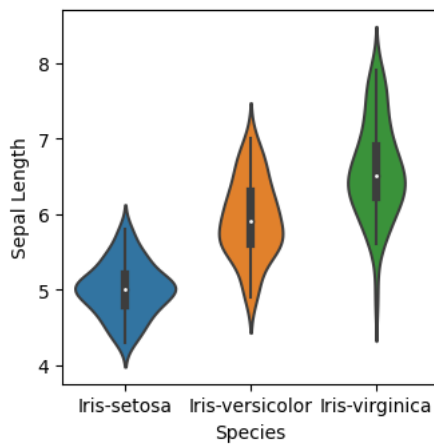
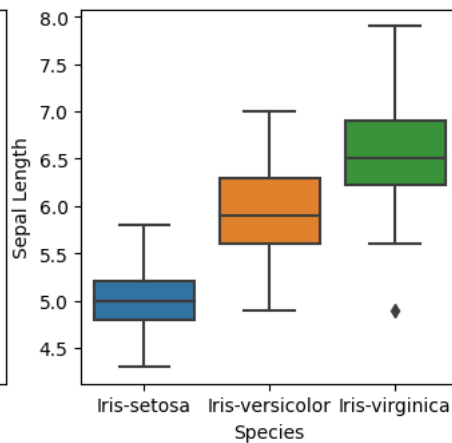
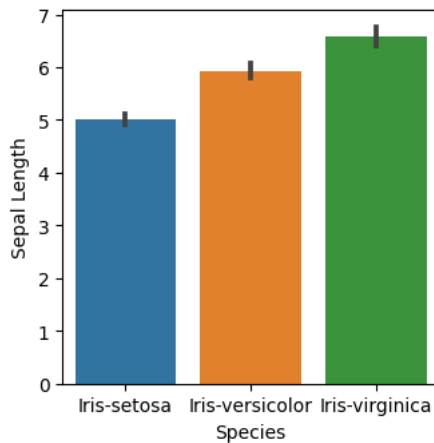
```
plt.subplot(2,2,1)
sns.barplot(x='Species',y='Sepal Length',data=IRIS_DATA,
            estimator='mean')
```

```
plt.subplot(2,2,2)
sns.boxplot(x='Species',y='Sepal Length',data=IRIS_DATA)
```

```
plt.subplot(2,2,3)
sns.violinplot(x='Species',y='Sepal Length',data=IRIS_DATA)
```

```
plt.subplot(2,2,4)
sns.swarmplot(x='Species',y='Sepal Length',data=IRIS_DATA,
              hue='Species',
              size=5)
```

<Axes: xlabel='Species', ylabel='Sepal Length'>



▼ DISTRIBUTION PLOTS

There are four types of distribution plots which can be plotted in seaborn:

- **Joinplot**