

# Iris Dataset Visualization using Seaborn

Functions in this Code: Countplot

Joinplot

FacetGrid

BoxPlot

Stripplot

Violinplot

Pairplot

heatmap

hist

Implot

FacetGrid

factorplot

poinplot

boxenplot

kdeplot

boxplot,violinplot,stripplot,hist

Stacked histogram

Area plot - plot.area

distplot

```
In [3]: import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import warnings
warnings.filterwarnings('ignore')
```

```
In [4]: iris=pd.read_csv(r"C:\Users\ymani\OneDrive\Desktop\NIT_Files\27th - Project\27th
iris
```

Out[4]:

	<b>Id</b>	<b>SepalLengthCm</b>	<b>SepalWidthCm</b>	<b>PetalLengthCm</b>	<b>PetalWidthCm</b>	<b>Species</b>
<b>0</b>	1	5.1	3.5	1.4	0.2	Iris-setosa
<b>1</b>	2	4.9	3.0	1.4	0.2	Iris-setosa
<b>2</b>	3	4.7	3.2	1.3	0.2	Iris-setosa
<b>3</b>	4	4.6	3.1	1.5	0.2	Iris-setosa
<b>4</b>	5	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...	...
<b>145</b>	146	6.7	3.0	5.2	2.3	Iris-virginica
<b>146</b>	147	6.3	2.5	5.0	1.9	Iris-virginica
<b>147</b>	148	6.5	3.0	5.2	2.0	Iris-virginica
<b>148</b>	149	6.2	3.4	5.4	2.3	Iris-virginica
<b>149</b>	150	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 6 columns

In [5]: `iris.head()`

Out[5]:

	<b>Id</b>	<b>SepalLengthCm</b>	<b>SepalWidthCm</b>	<b>PetalLengthCm</b>	<b>PetalWidthCm</b>	<b>Species</b>
<b>0</b>	1	5.1	3.5	1.4	0.2	Iris-setosa
<b>1</b>	2	4.9	3.0	1.4	0.2	Iris-setosa
<b>2</b>	3	4.7	3.2	1.3	0.2	Iris-setosa
<b>3</b>	4	4.6	3.1	1.5	0.2	Iris-setosa
<b>4</b>	5	5.0	3.6	1.4	0.2	Iris-setosa

In [6]: `iris.drop('Id',axis=1,inplace=True)`

In [7]: `iris.head()`

```
Out[7]:
```

	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa

```
In [8]: iris.columns=['SepalLength', 'SepalWidth', 'PetalLength', 'PetalWidth', 'Specie']
```

```
In [9]: iris.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
#   Column          Non-Null Count  Dtype
---  -
0   SepalLength     150 non-null   float64
1   SepalWidth      150 non-null   float64
2   PetalLength     150 non-null   float64
3   PetalWidth      150 non-null   float64
4   Specie          150 non-null   object
dtypes: float64(4), object(1)
memory usage: 6.0+ KB
```

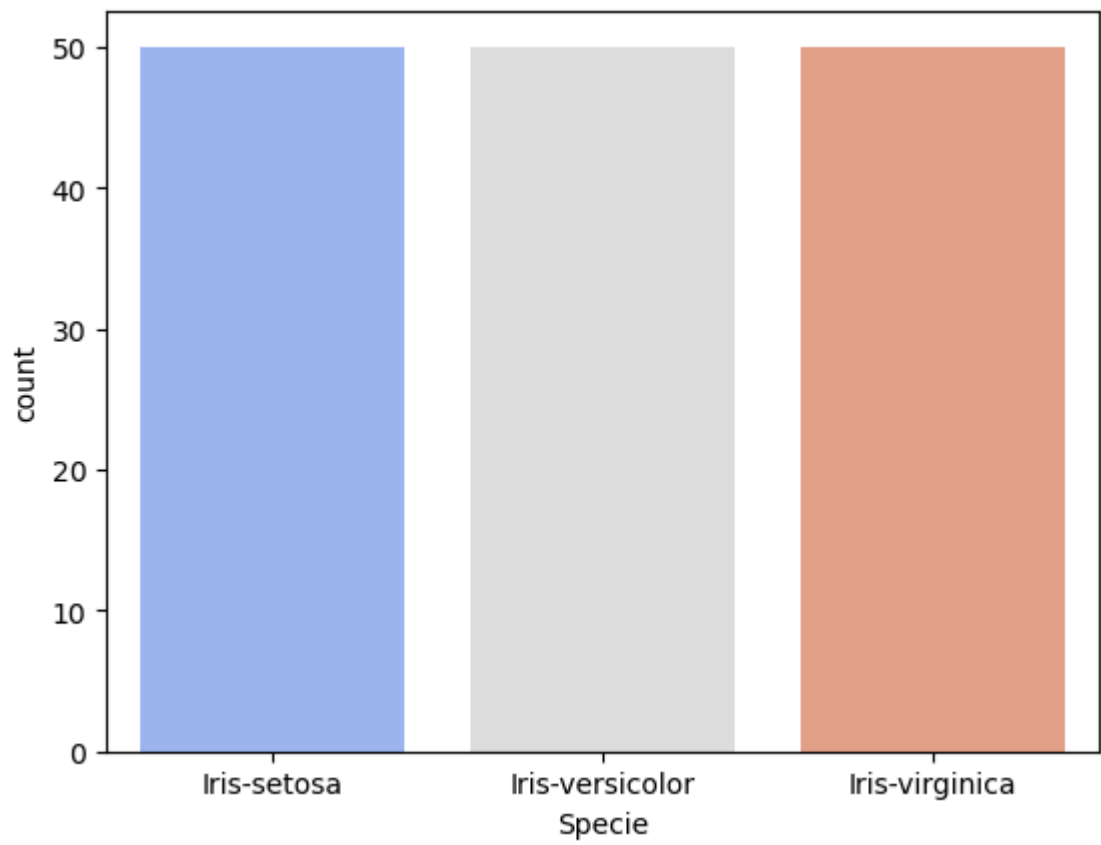
```
In [10]: iris['Specie'].value_counts()
```

```
Out[10]: Specie
Iris-setosa      50
Iris-versicolor  50
Iris-virginica   50
Name: count, dtype: int64
```

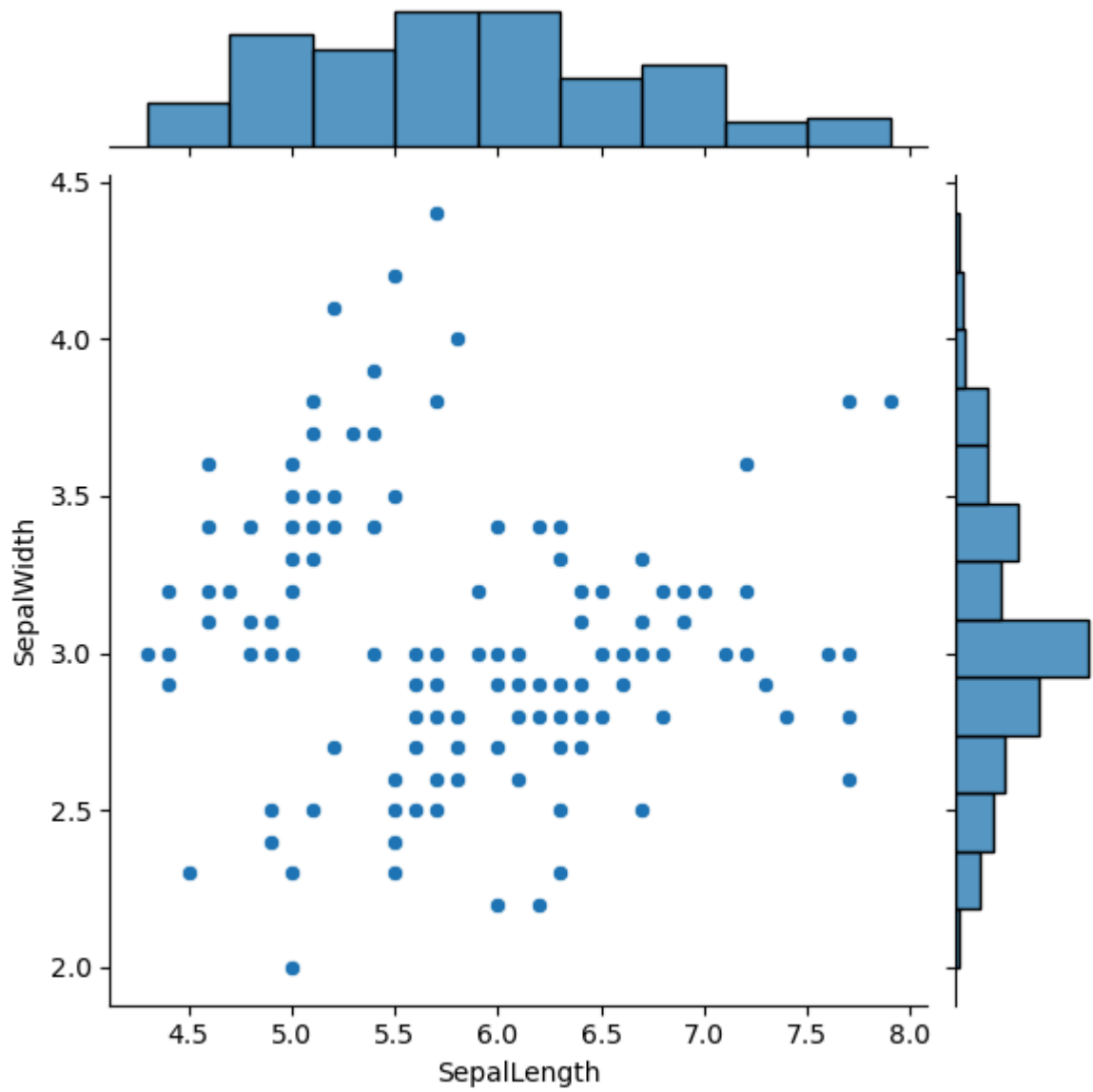
Bar Plot

```
In [12]: sns.countplot(x='Specie', data=iris, palette='coolwarm')
```

```
Out[12]: <Axes: xlabel='Specie', ylabel='count'>
```

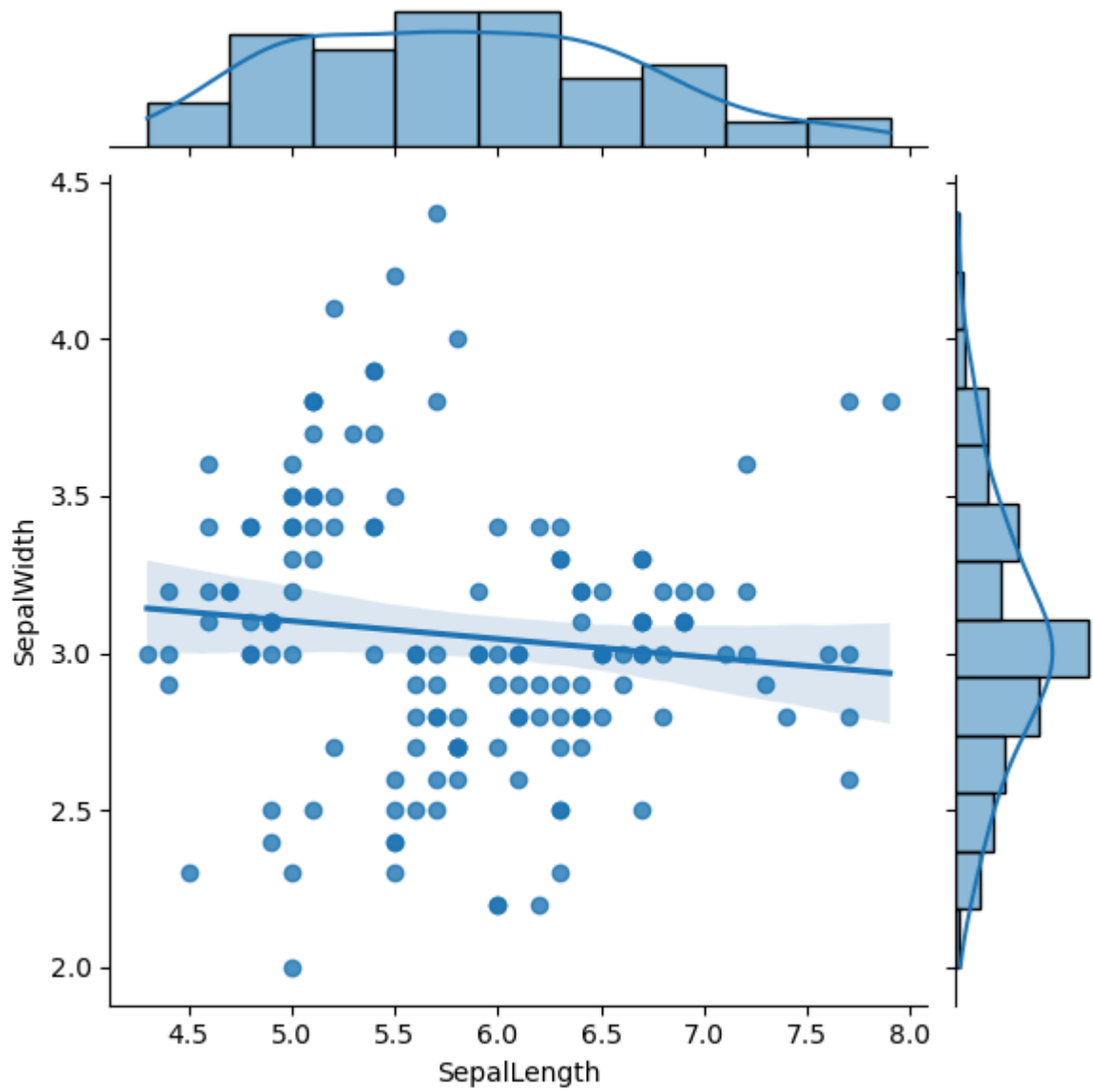


```
In [13]: fig=sns.jointplot(x='SepalLength',y='SepalWidth',data=iris)
plt.show()
```

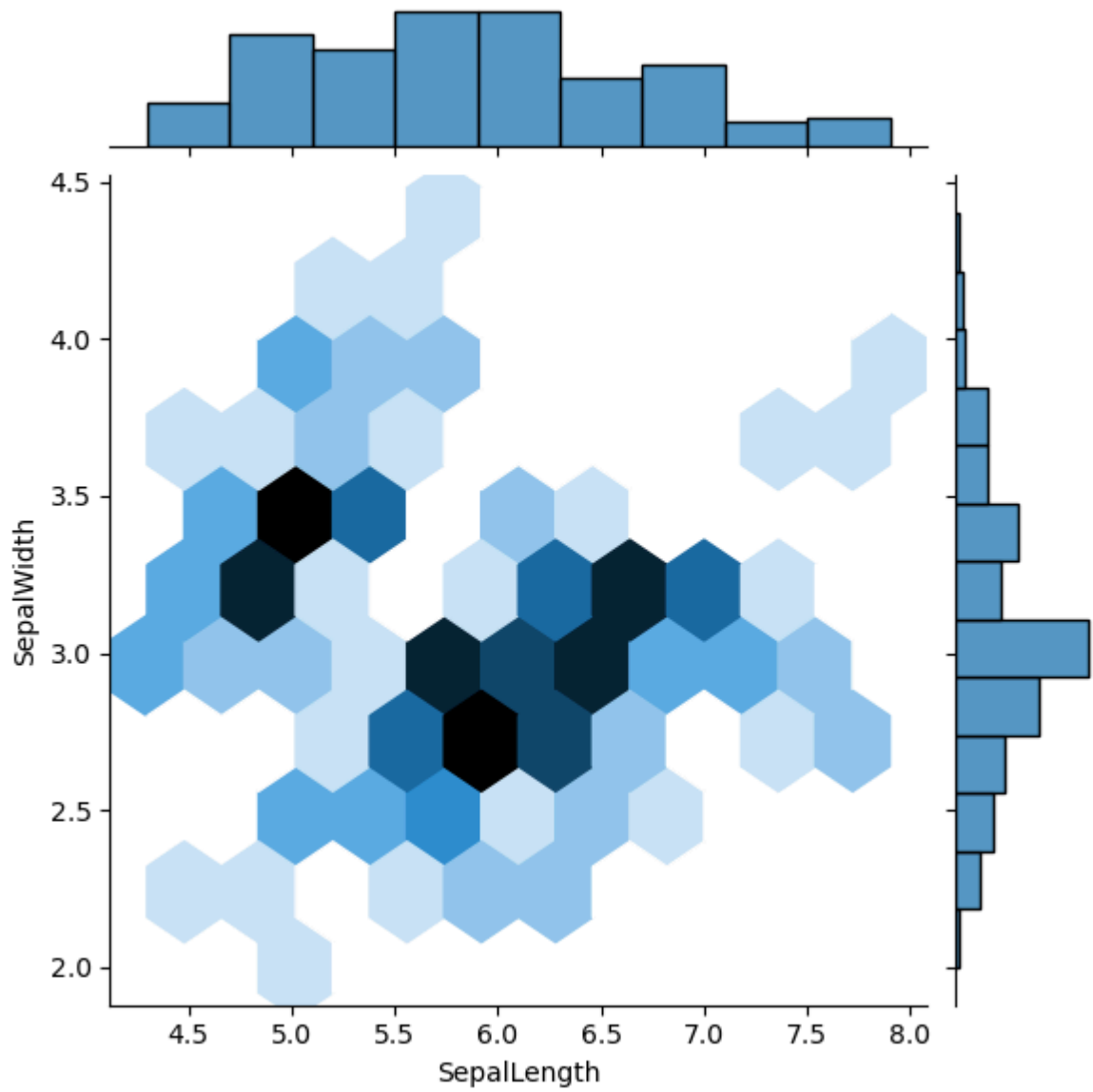


```
In [14]: sns.jointplot(x='SepalLength',y='SepalWidth',data=iris,kind='reg')
```

```
Out[14]: <seaborn.axisgrid.JointGrid at 0x15dc47bbda0>
```

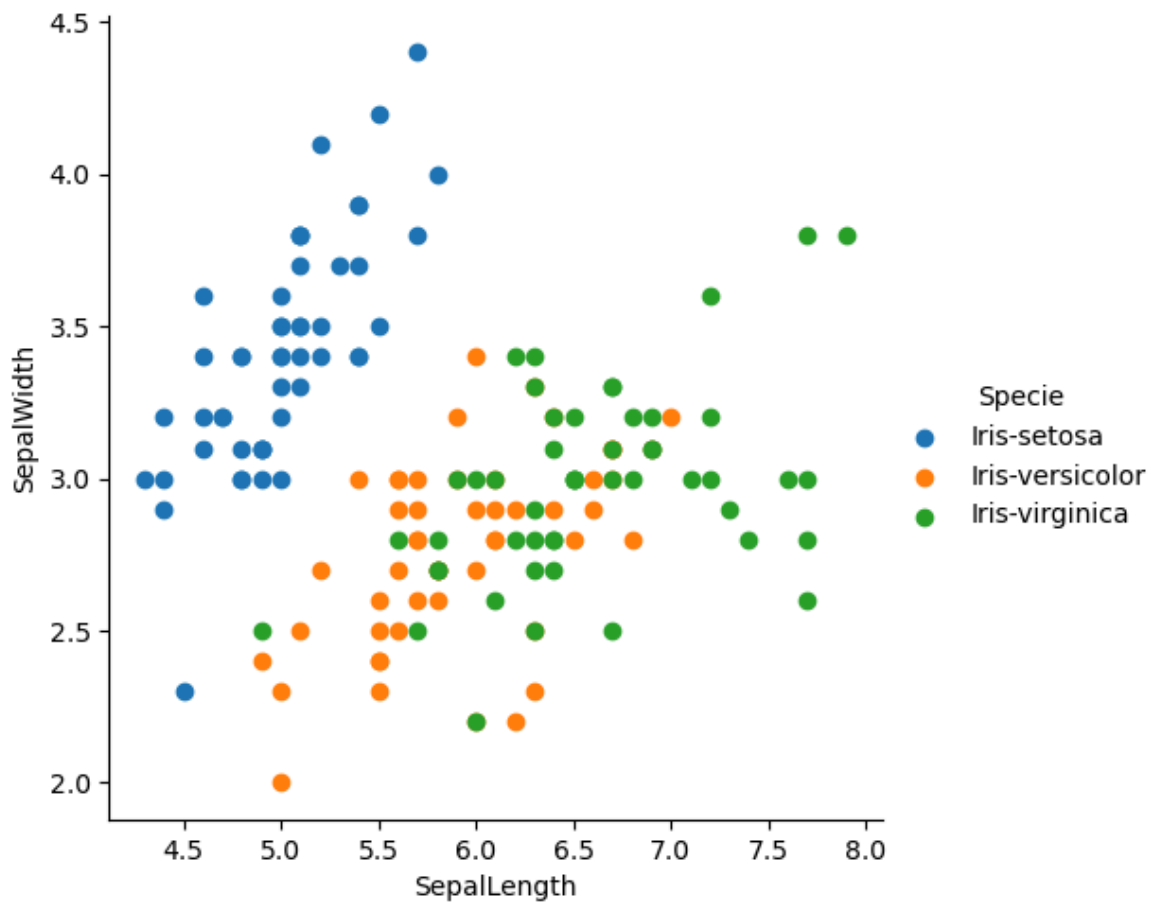


```
In [15]: fig=sns.jointplot(x='SepalLength',y='SepalWidth',kind='hex',data=iris)
plt.show()
```



```
In [16]: %matplotlib inline
```

```
In [17]: sns.FacetGrid(iris,hue='Specie',height=5).map(plt.scatter,'SepalLength','SepalWi  
plt.show()
```



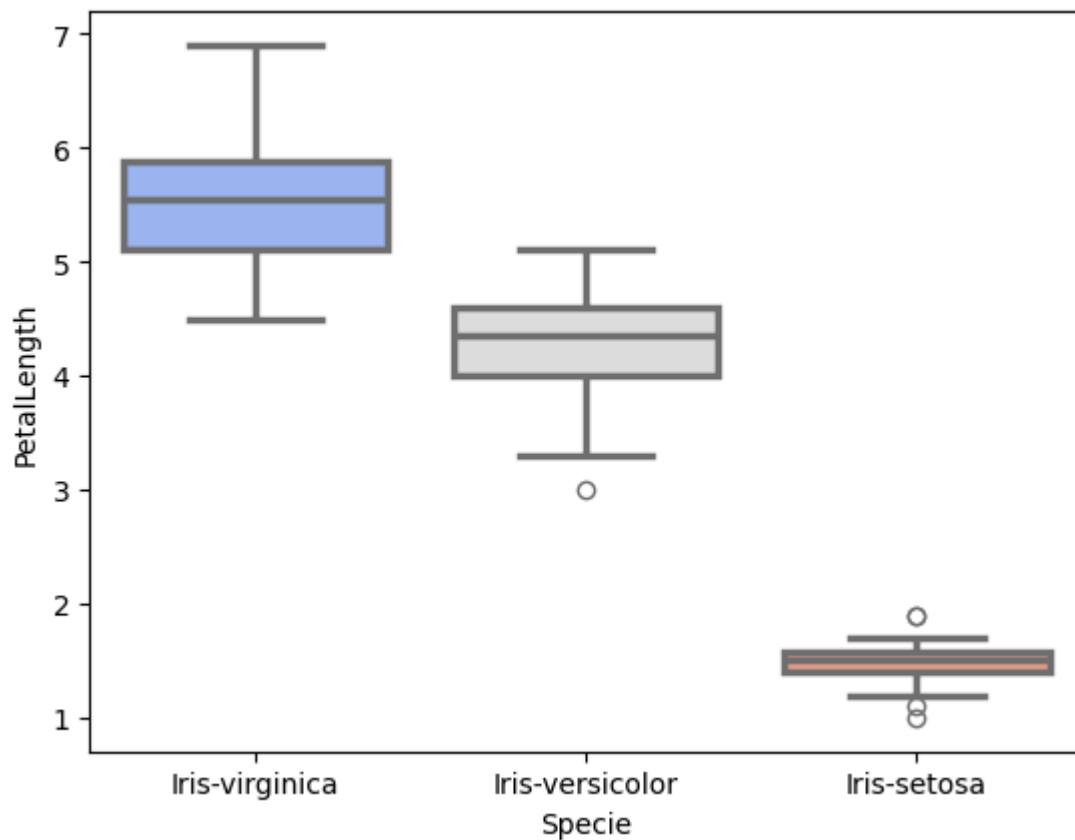
```
In [18]: iris.head()
```

```
Out[18]:
```

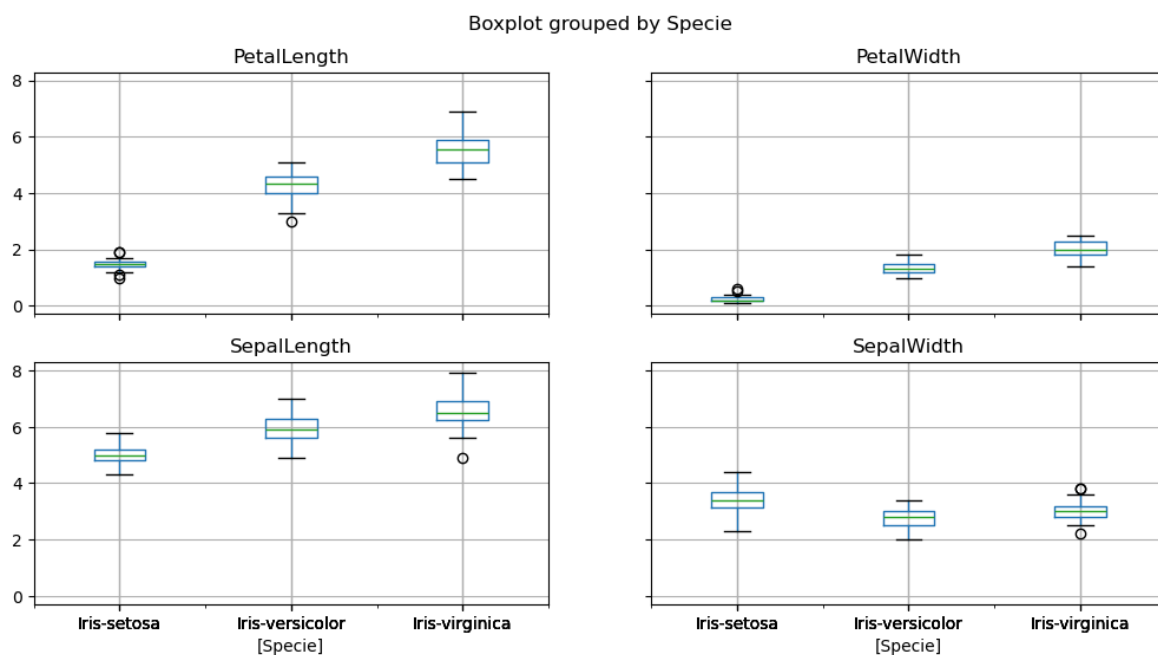
	SepalLength	SepalWidth	PetalLength	PetalWidth	Specie
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa

```
In [19]: fig=plt.gcf()
fig.set_size_inches=(10,7)
fig=sns.boxplot(x='Specie',y='PetalLength',data=iris,order=['Iris-virginica','Ir
plt.show()
```

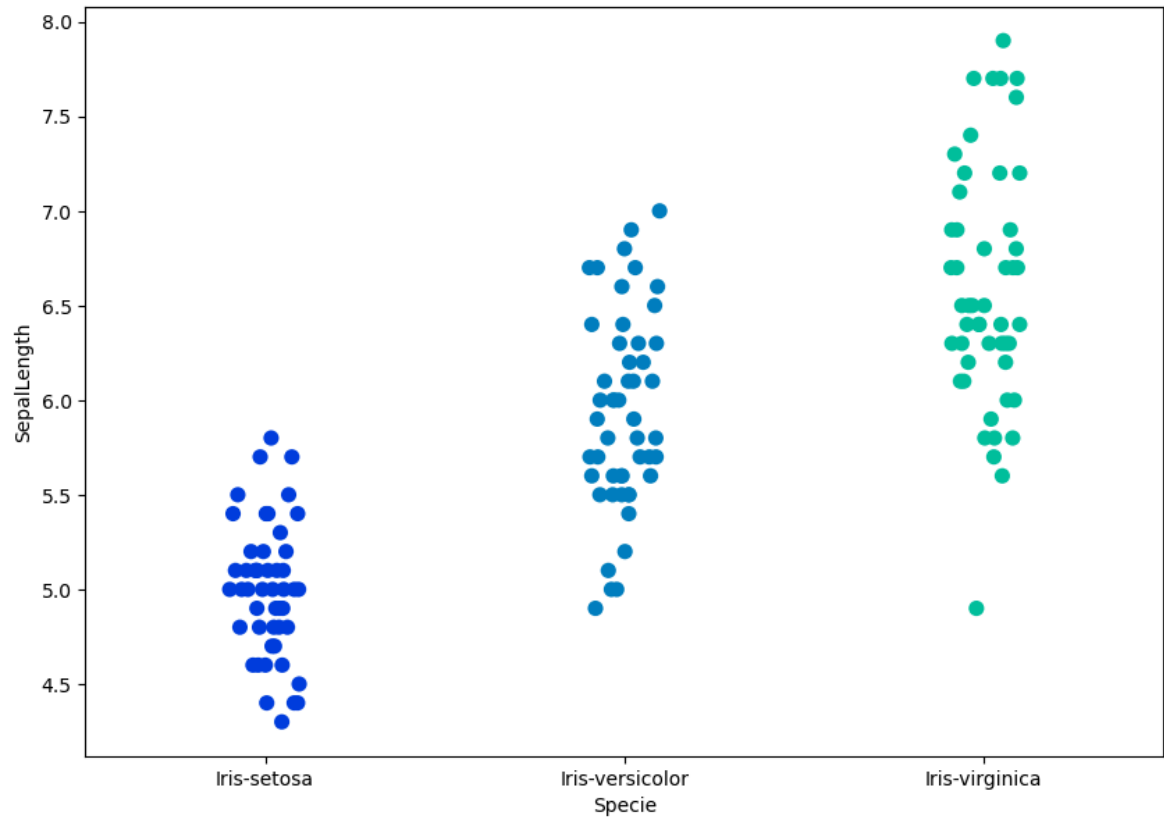




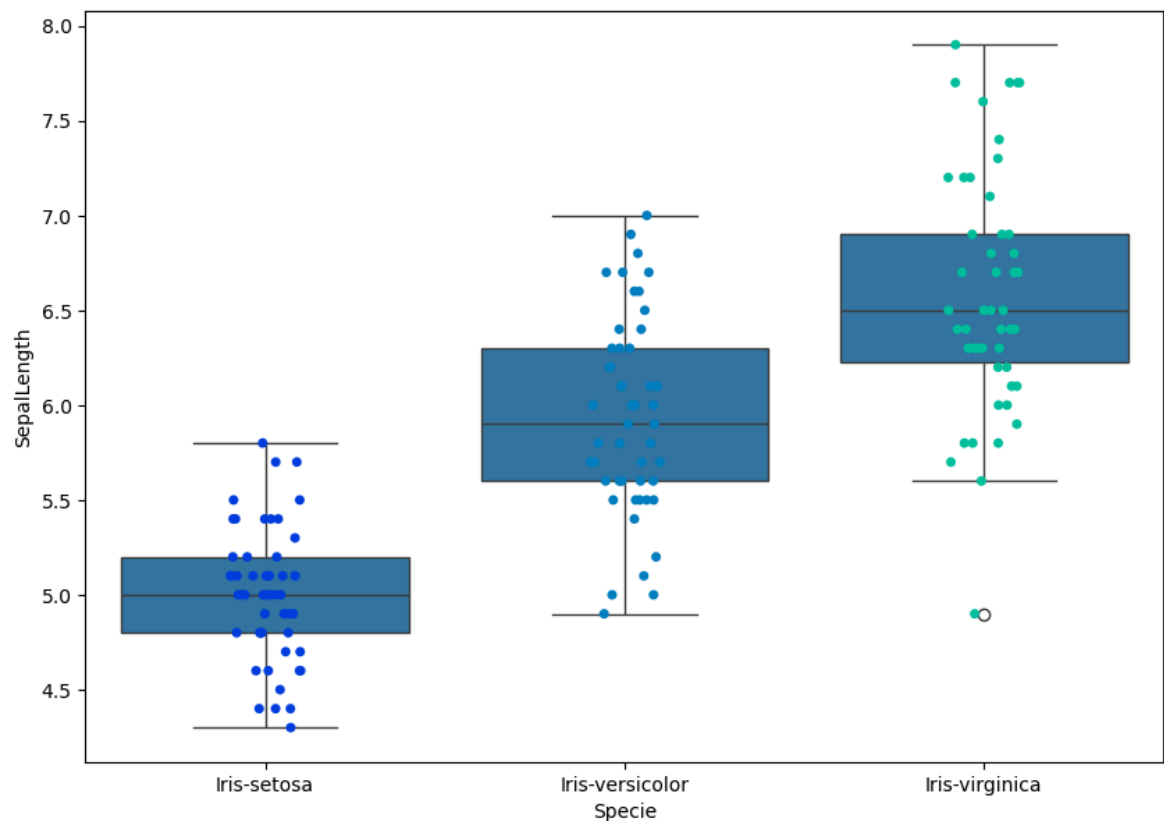
```
In [20]: iris.boxplot(by='Specie',figsize=(12,6))
plt.show()
```



```
In [21]: fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.stripplot(x='Specie',y='SepalLength',data=iris,jitter=True,edgecolor='gr')
plt.show()
```



```
In [22]: fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.boxplot(x='Specie',y='SepalLength',data=iris)
fig=sns.stripplot(x='Specie',y='SepalLength',data=iris,jitter=True,edgecolor='gray')
plt.show()
```



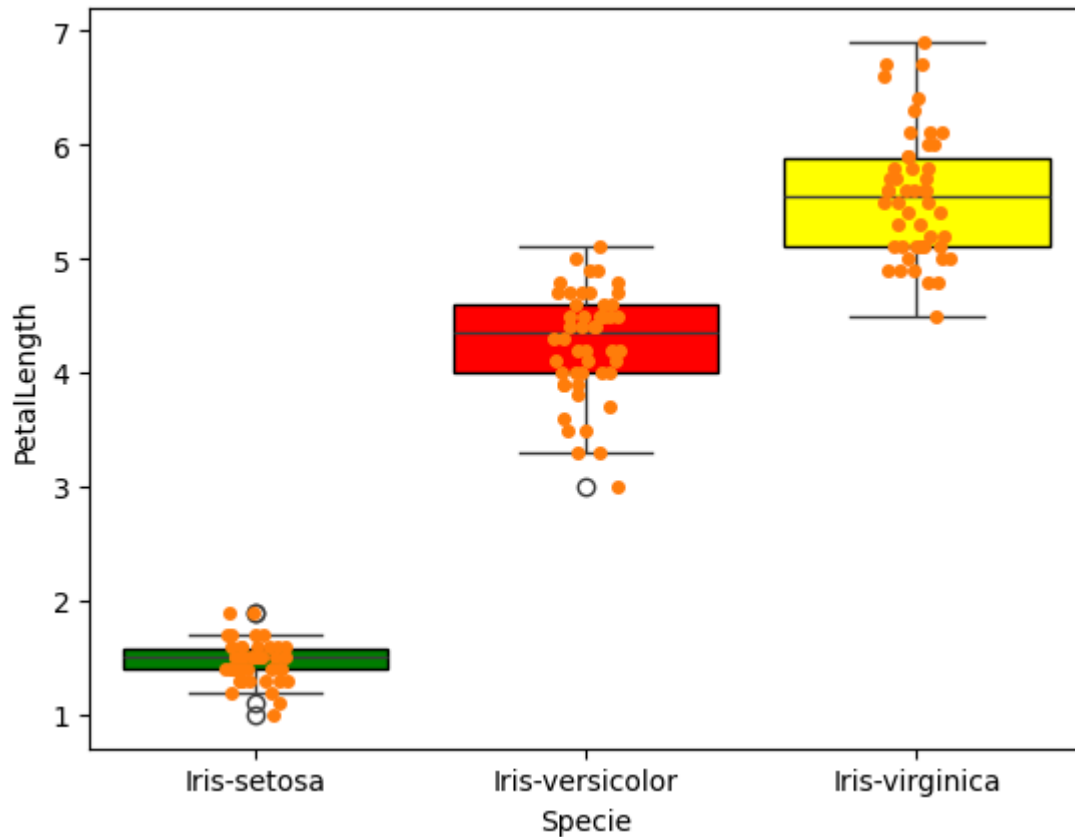
```
In [23]: ax=sns.boxplot(x='Specie',y='PetalLength',data=iris)
ax=sns.stripplot(x='Specie',y='PetalLength',data=iris,jitter=True,edgecolor='gray')
```

```

boxtwo=ax.patches[2]
boxtwo.set_facecolor('yellow')
boxtwo.set_edgecolor('black')
boxthree=ax.patches[1]
boxthree.set_facecolor('red')
boxthree.set_edgecolor('black')
boxthree=ax.patches[0]
boxthree.set_facecolor('green')
boxthree.set_edgecolor('black')

plt.show()

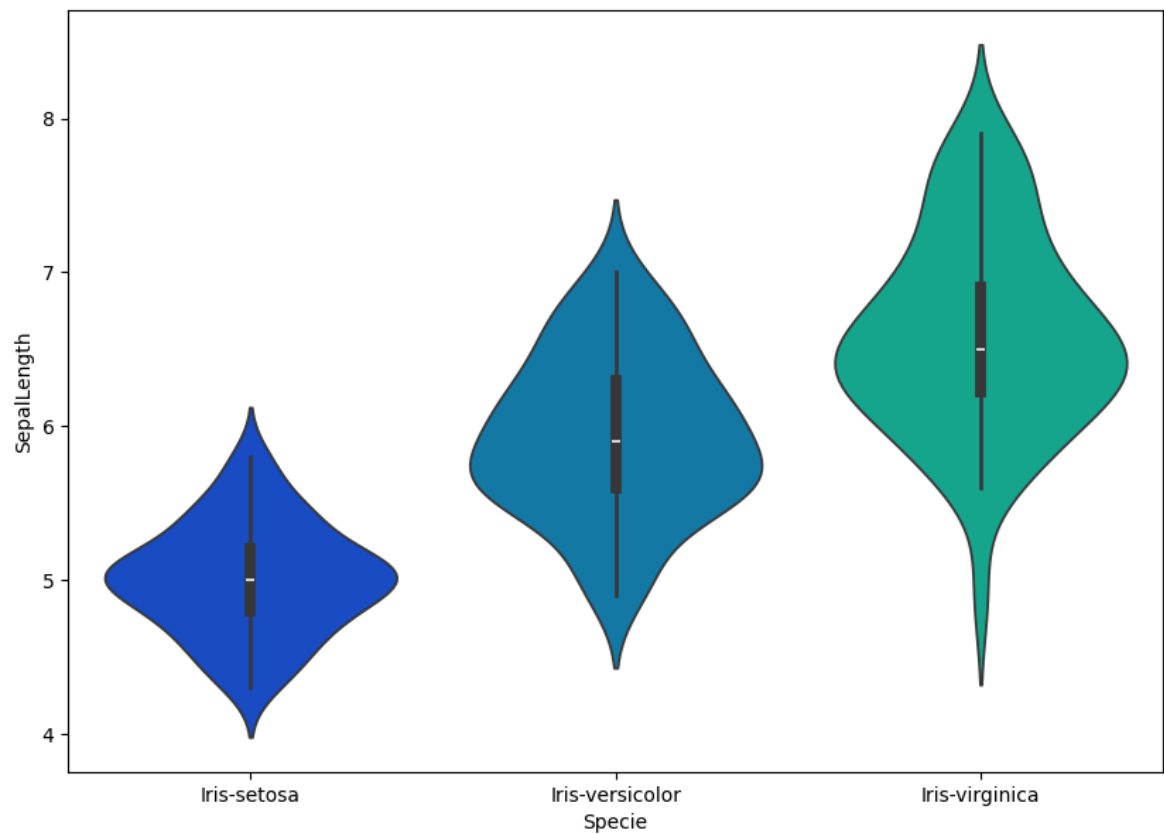
```



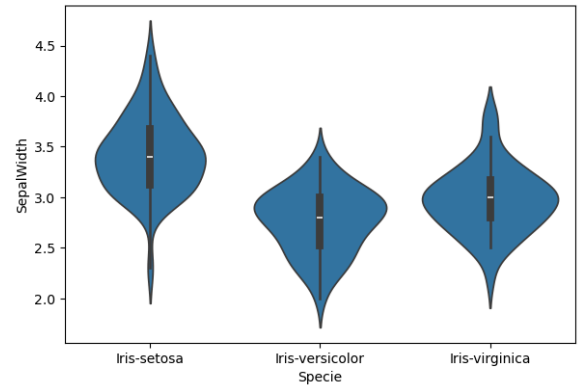
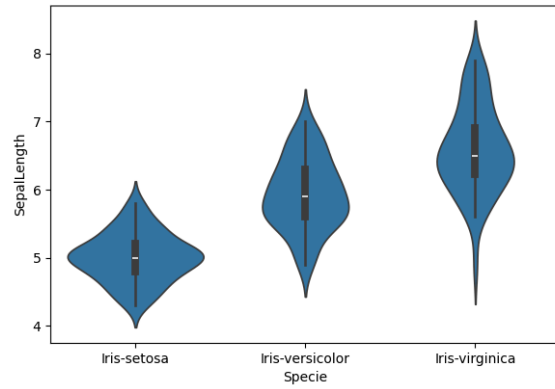
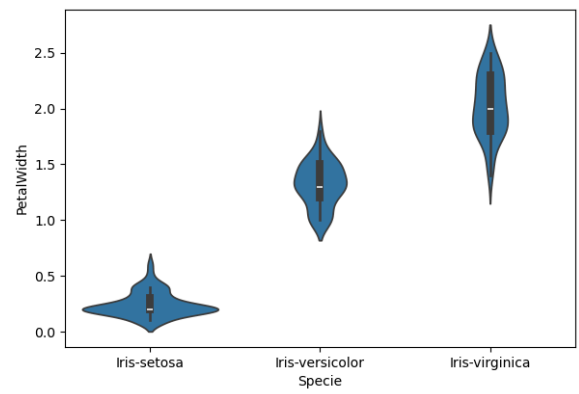
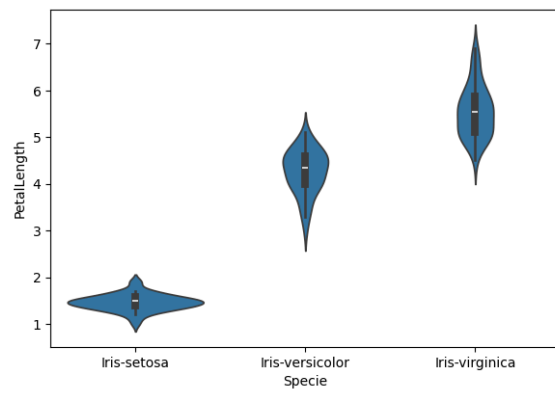
```

In [24]: fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.violinplot(x='Specie',y='SepalLength',data=iris,palette='winter')
plt.show()

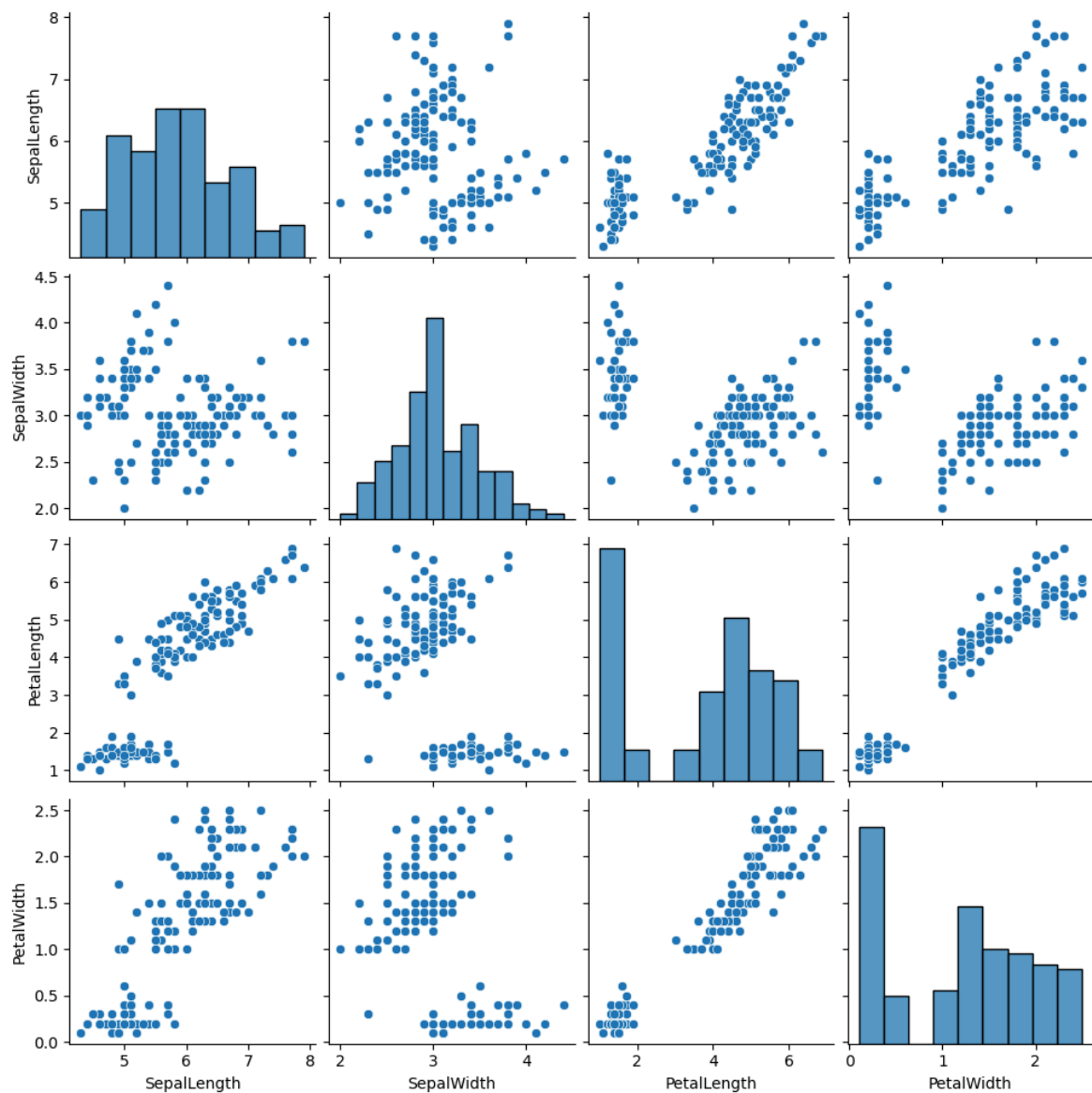
```



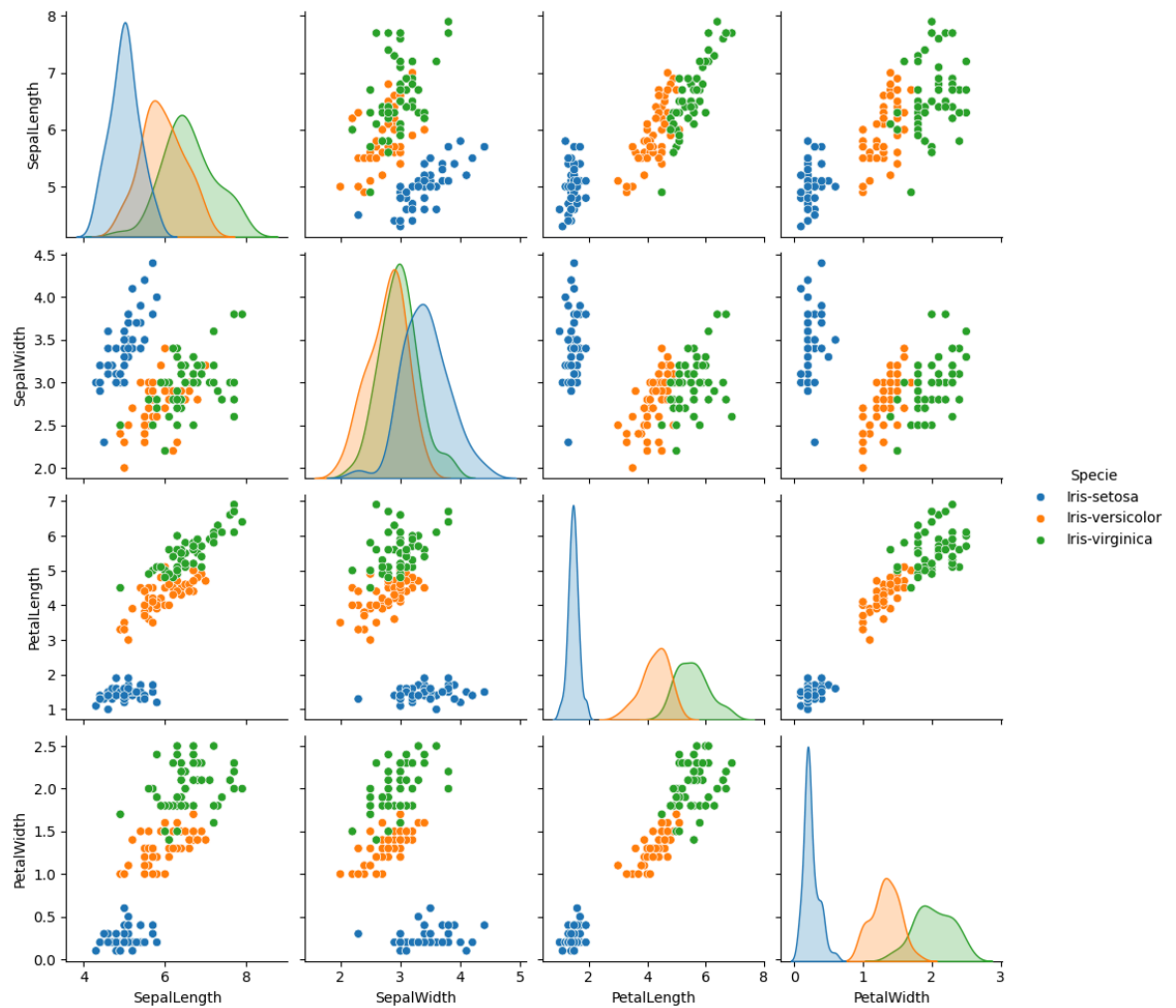
```
In [25]: plt.figure(figsize=(15,10))
plt.subplot(2,2,1)
sns.violinplot(x='Specie',y='PetalLength',data=iris)
plt.subplot(2,2,2)
sns.violinplot(x='Specie',y='PetalWidth',data=iris)
plt.subplot(2,2,3)
sns.violinplot(x='Specie',y='SepalLength',data=iris)
plt.subplot(2,2,4)
sns.violinplot(x='Specie',y='SepalWidth',data=iris)
plt.show()
```



```
In [26]: sns.pairplot(data=iris,kind='scatter')  
plt.show()
```

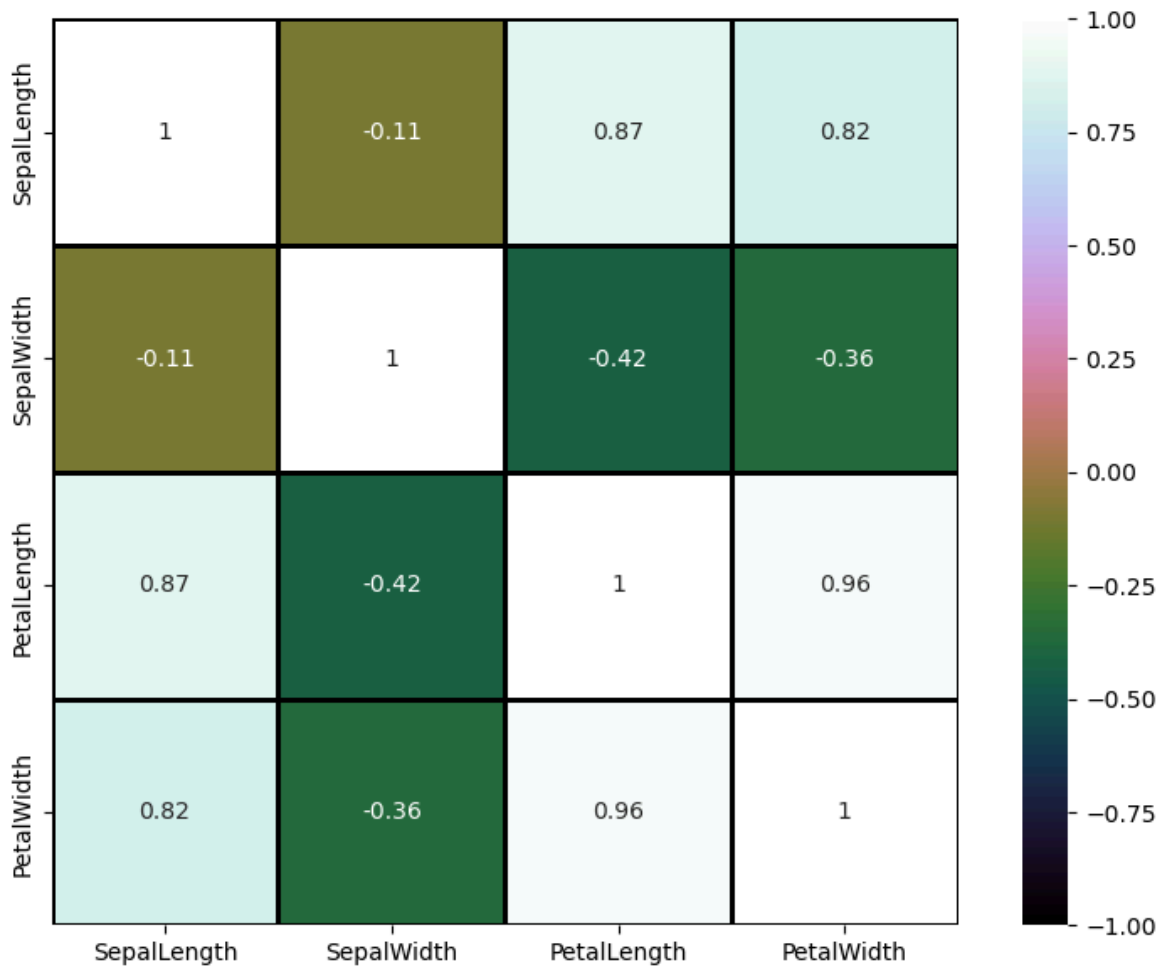


```
In [27]: sns.pairplot(iris,hue='Specie')
plt.show()
```

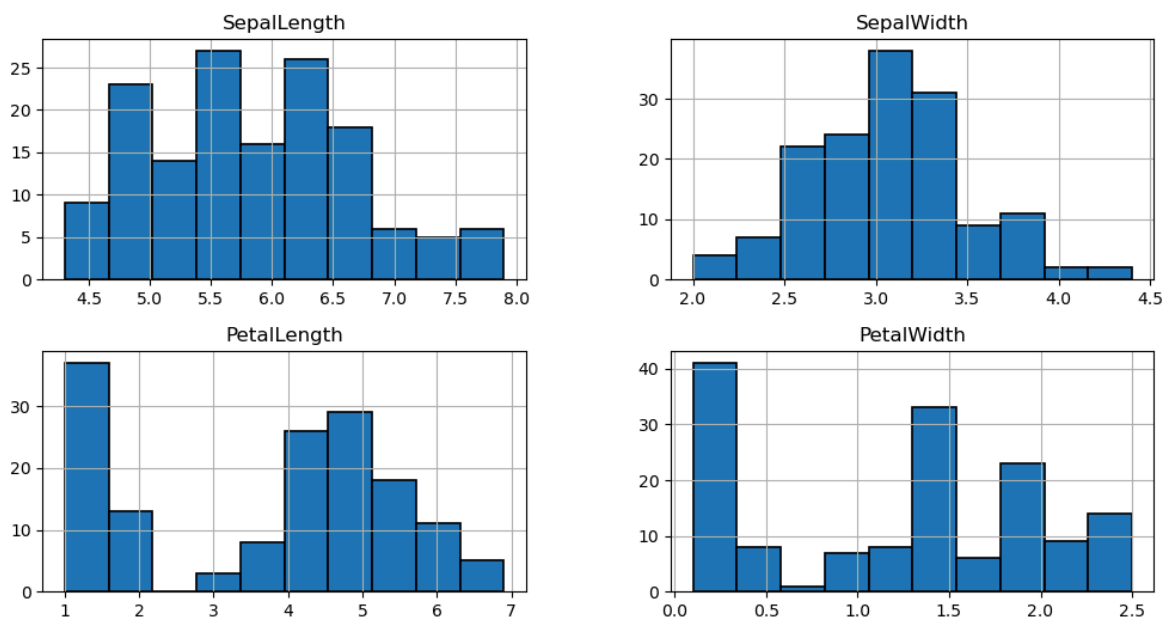


```
In [28]: iris_numeric=iris.drop(columns=['Specie'])
```

```
In [29]: fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.heatmap(iris_numeric.corr(),annot=True,cmap='cubehelix',linewidths=1,lin
plt.show())
```

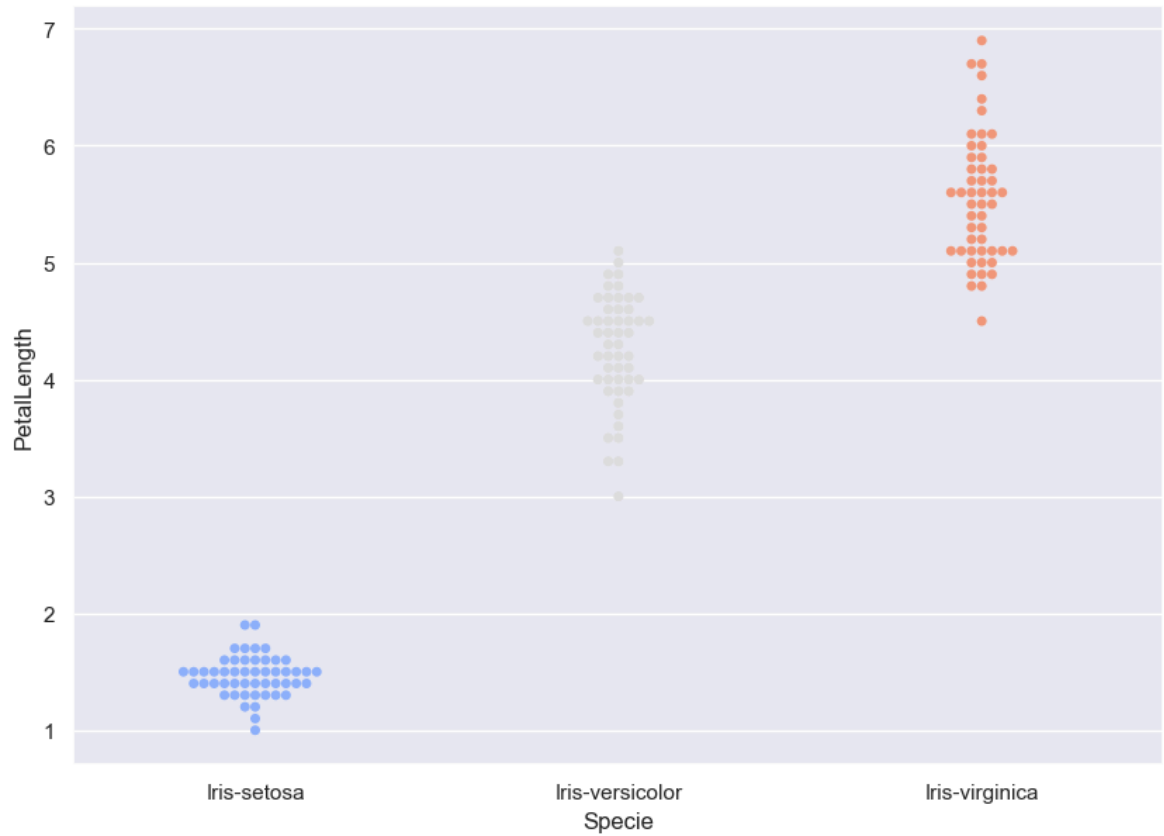


```
In [30]: iris.hist(edgecolor='black',linewidth=1.2)
fig=plt.gcf()
fig.set_size_inches(12,6)
plt.show()
```

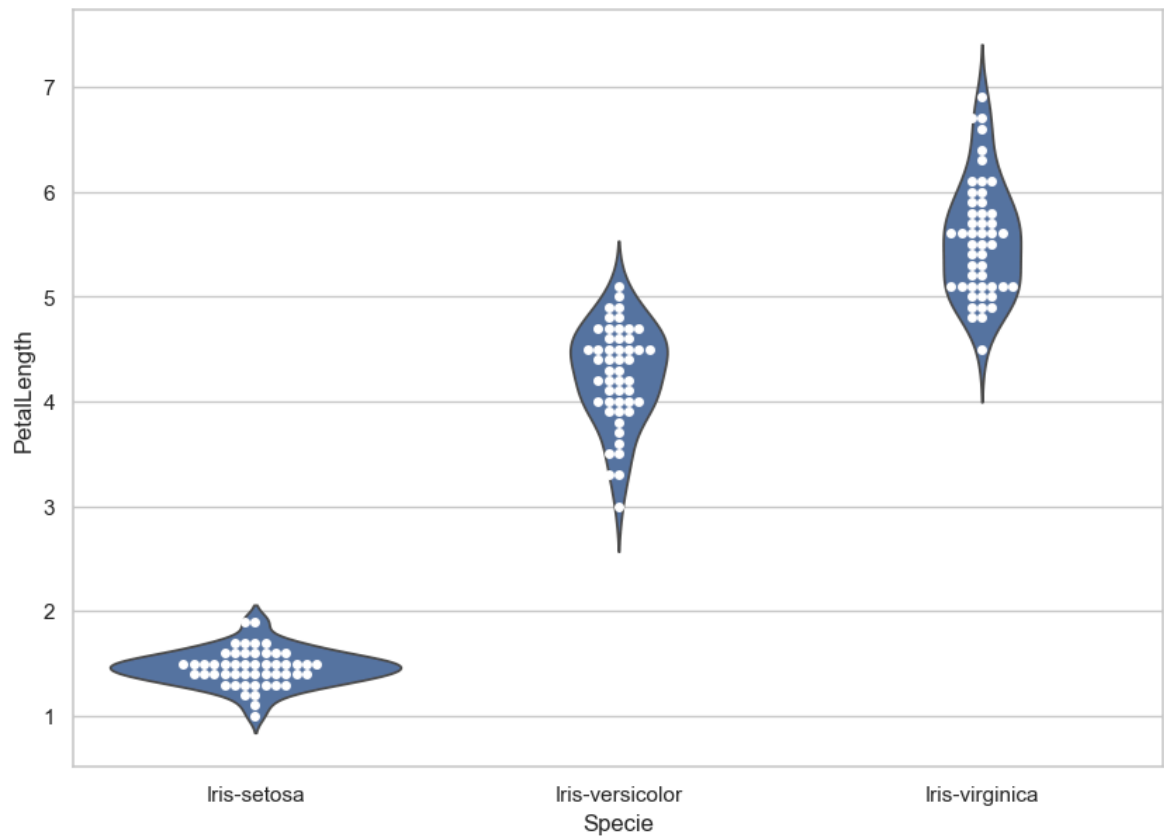


```
In [31]: sns.set(style="darkgrid")
fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.swarmplot(x="Specie",y='PetalLength',data=iris,palette='coolwarm')
plt.show()
```

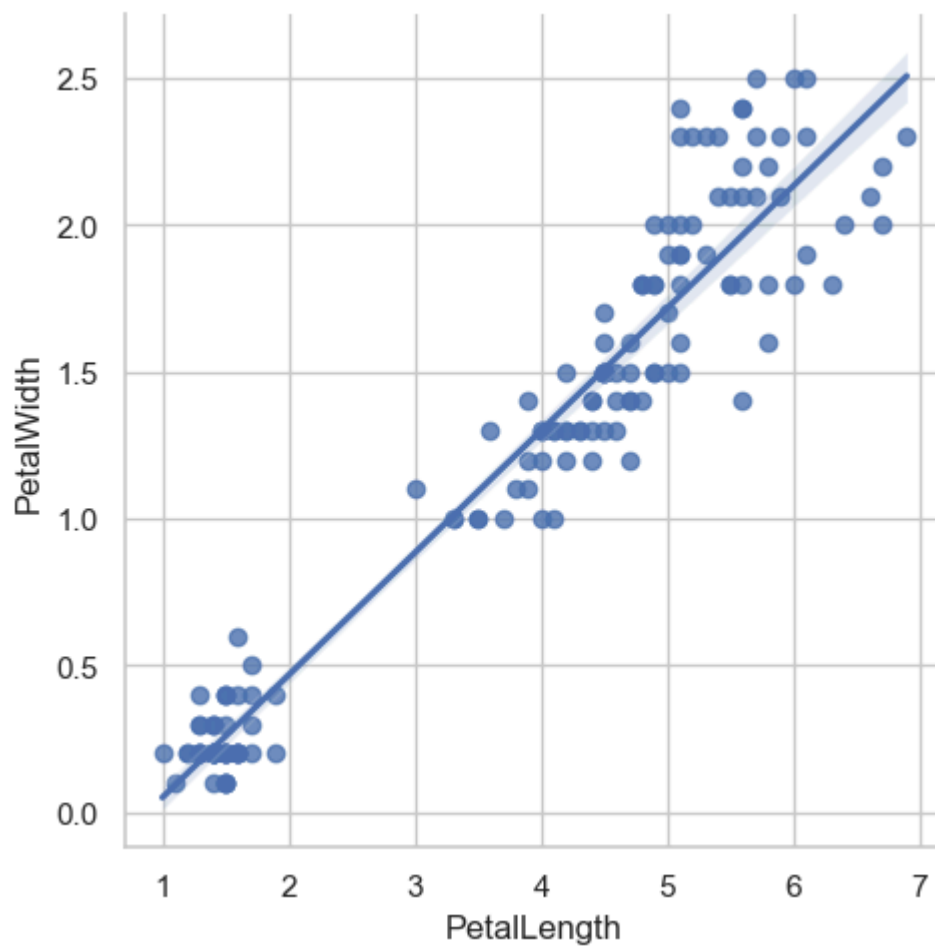




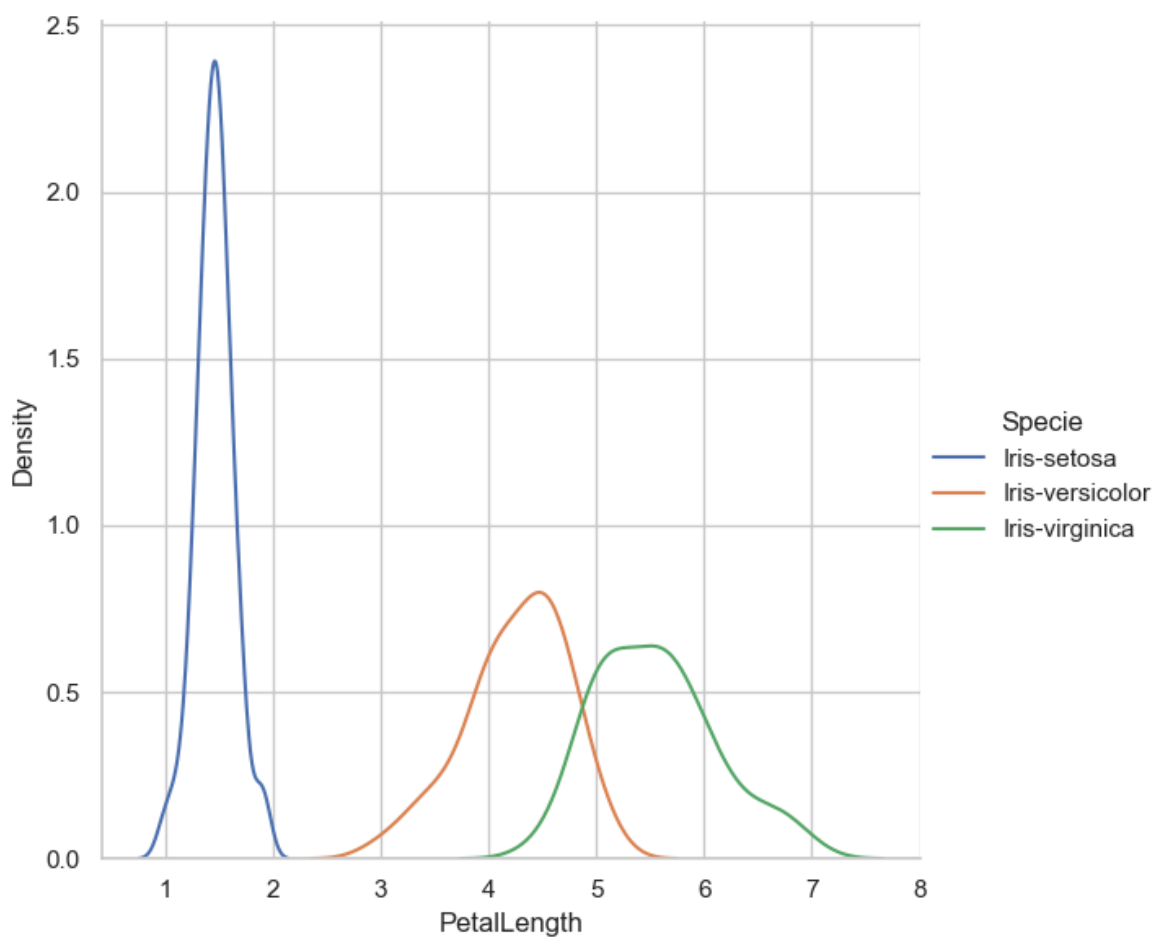
```
In [32]: sns.set(style='whitegrid')
fig=plt.gcf()
fig.set_size_inches(10,7)
ax=sns.violinplot(x='Specie',y='Petal.Length',data=iris,inner=None)
ax=sns.swarmplot(x='Specie',y='Petal.Length',data=iris,color='white',edgecolor='b')
plt.show()
```



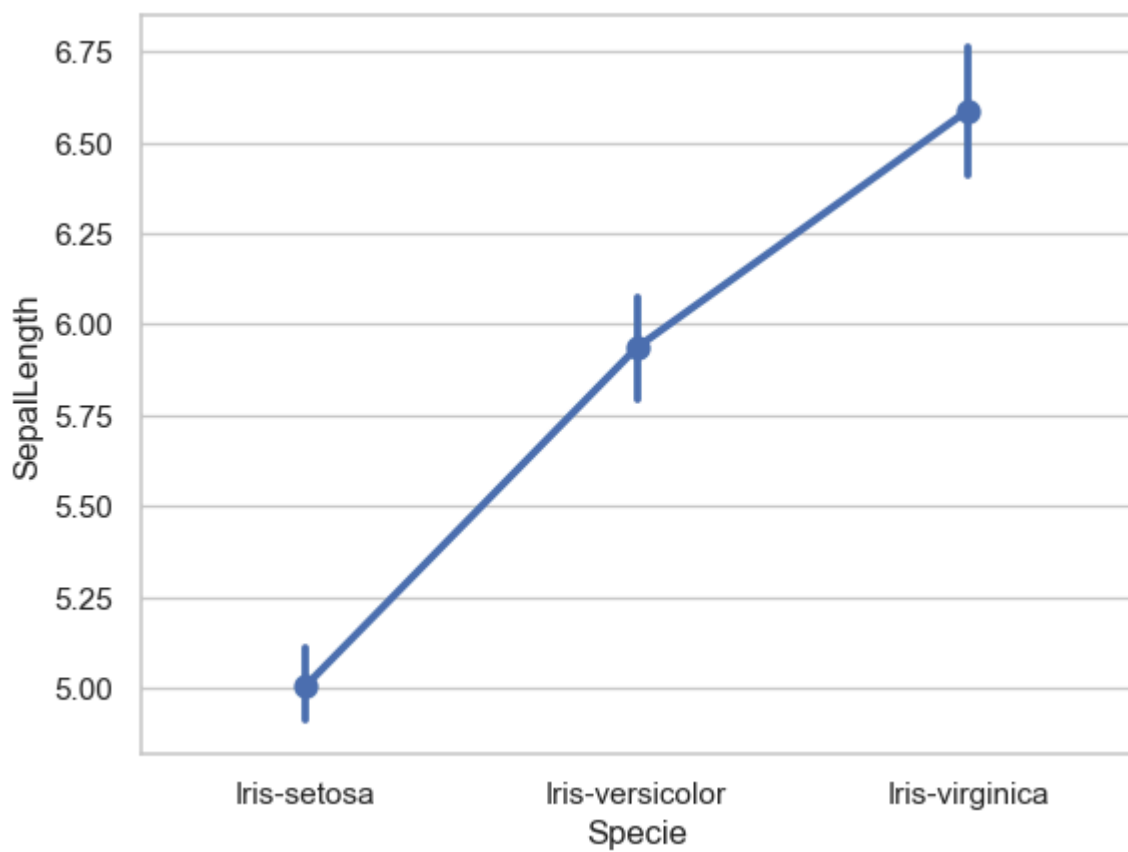
```
In [33]: fig=sns.lmplot(x='PetalLength',y='PetalWidth',data=iris)
plt.show()
```



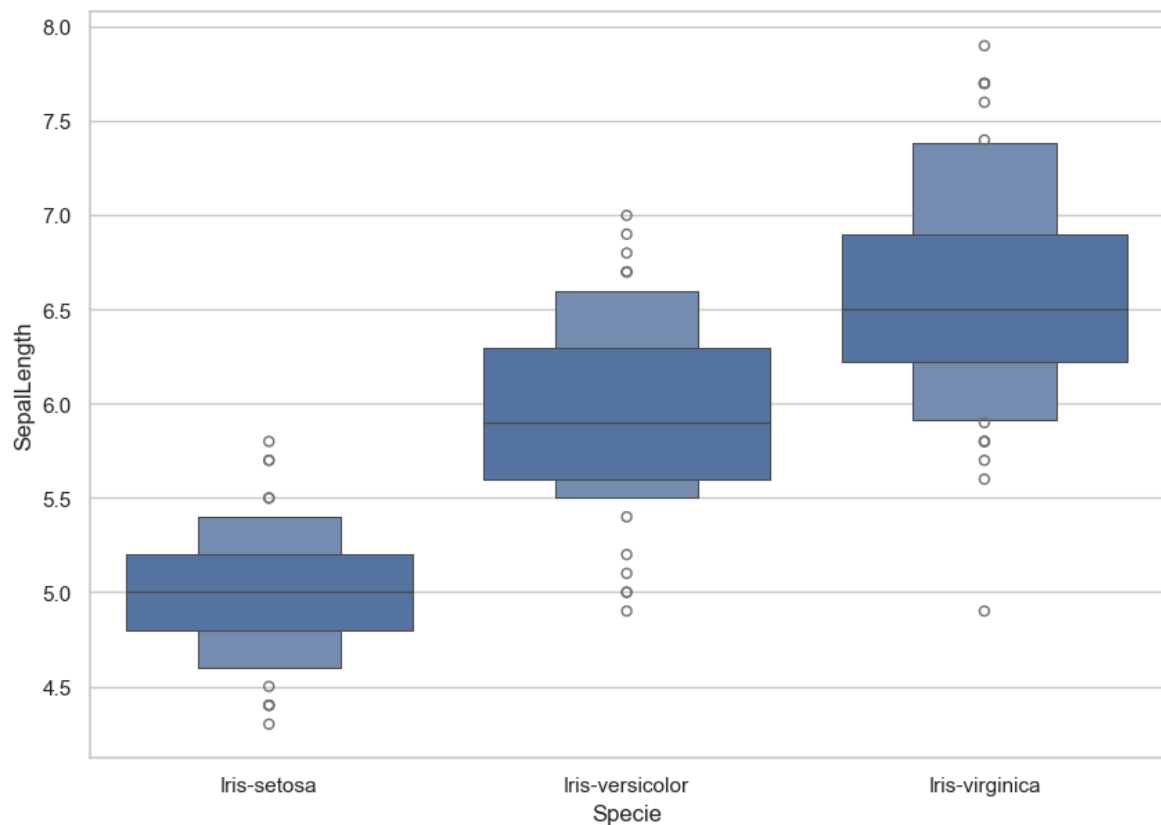
```
In [34]: sns.FacetGrid(iris,hue="Specie",height=6).map(sns.kdeplot,'PetalLength').add_leg
plt.ioff()
plt.show()
```



```
In [35]: sns.pointplot(x='Specie',y='SepallLength',data=iris)
plt.ioff()
plt.show()
```



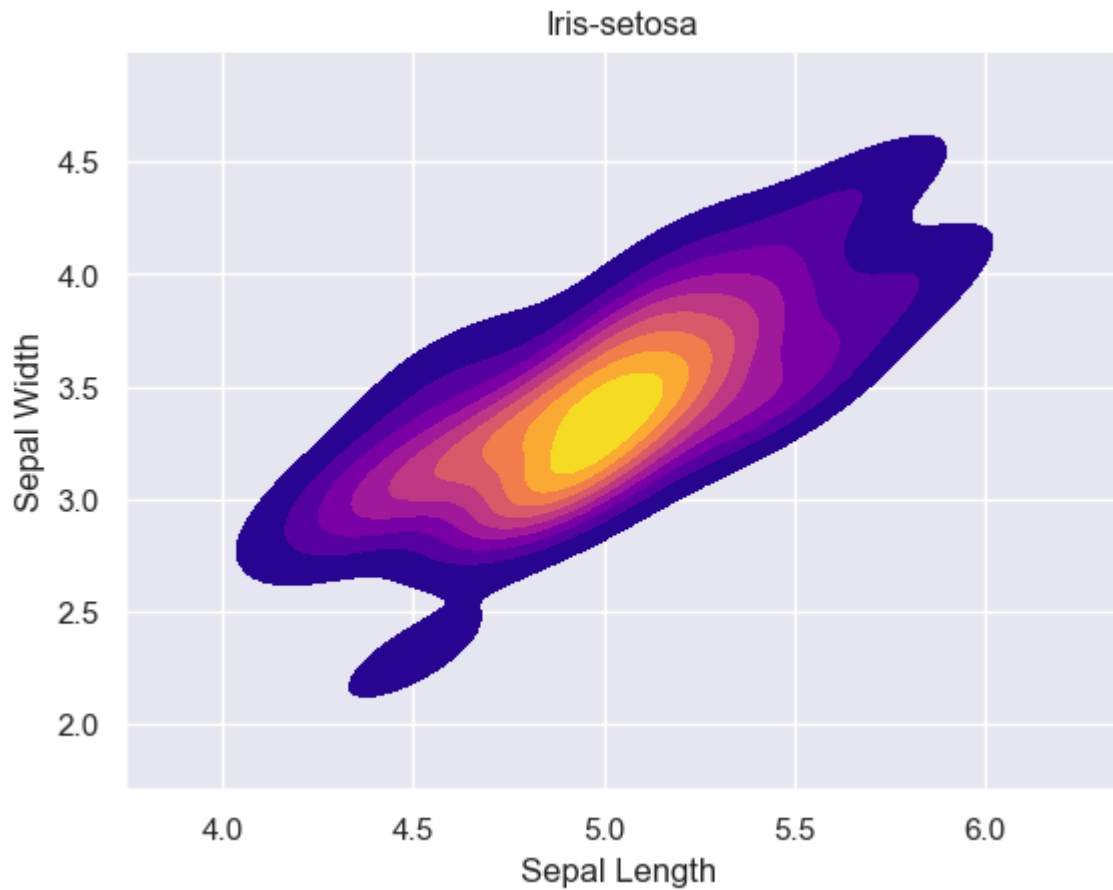
```
In [36]: fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.boxenplot(x='Specie',y='SepallLength',data=iris)
plt.show()
```



```
In [37]: iris.info()
```

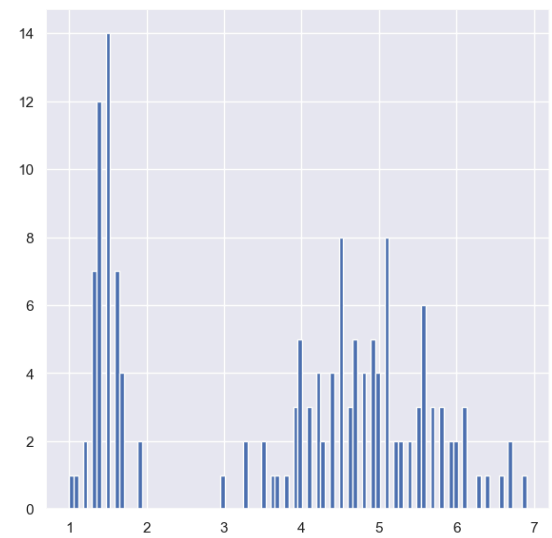
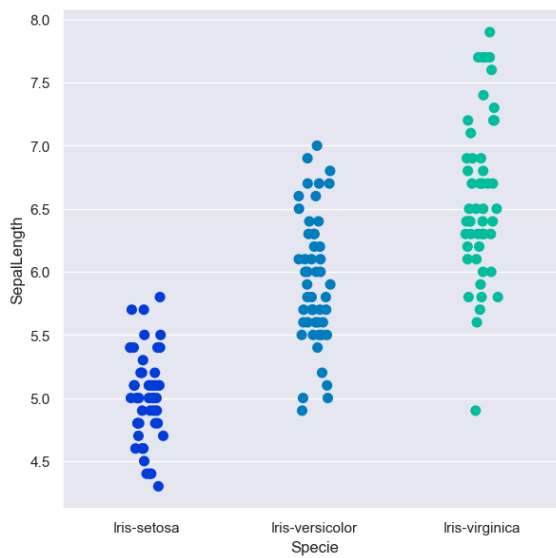
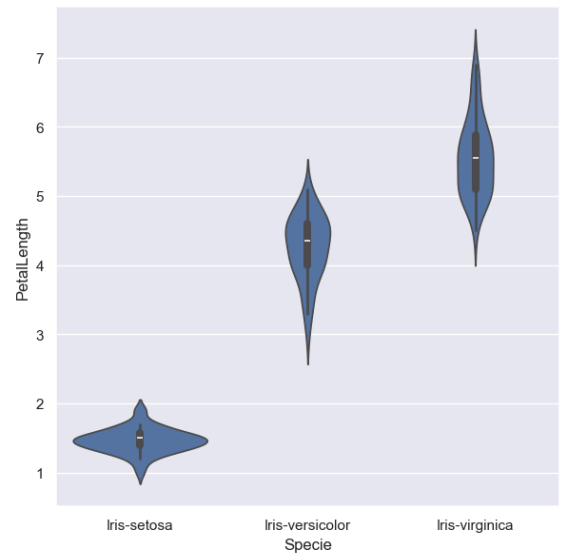
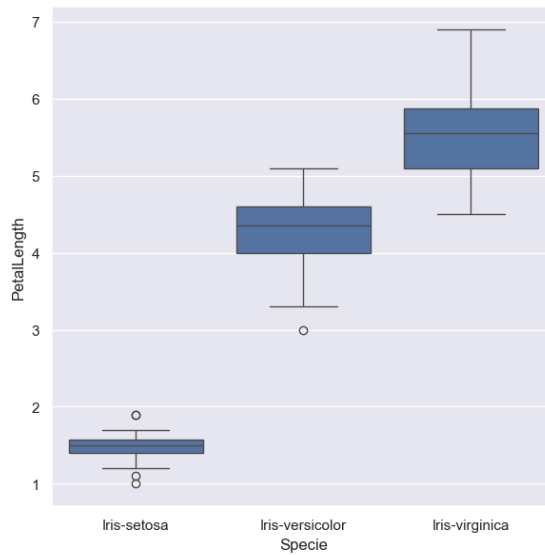
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
#   Column          Non-Null Count  Dtype
---  -
0   SepalLength     150 non-null   float64
1   SepalWidth      150 non-null   float64
2   PetalLength     150 non-null   float64
3   PetalWidth      150 non-null   float64
4   Specie          150 non-null   object
dtypes: float64(4), object(1)
memory usage: 6.0+ KB
```

```
In [56]: sub=iris[iris['Specie']=='Iris-setosa']
sns.kdeplot(x=sub['SepalLength'],y=sub['SepalWidth'],fill=True,shade=True,shade_
plt.title('Iris-setosa')
plt.xlabel('Sepal Length ')
plt.ylabel('Sepal Width')
plt.show()
```



```
In [58]: sns.set_style('darkgrid')
f, axes = plt.subplots(2, 2, figsize=(15, 15))

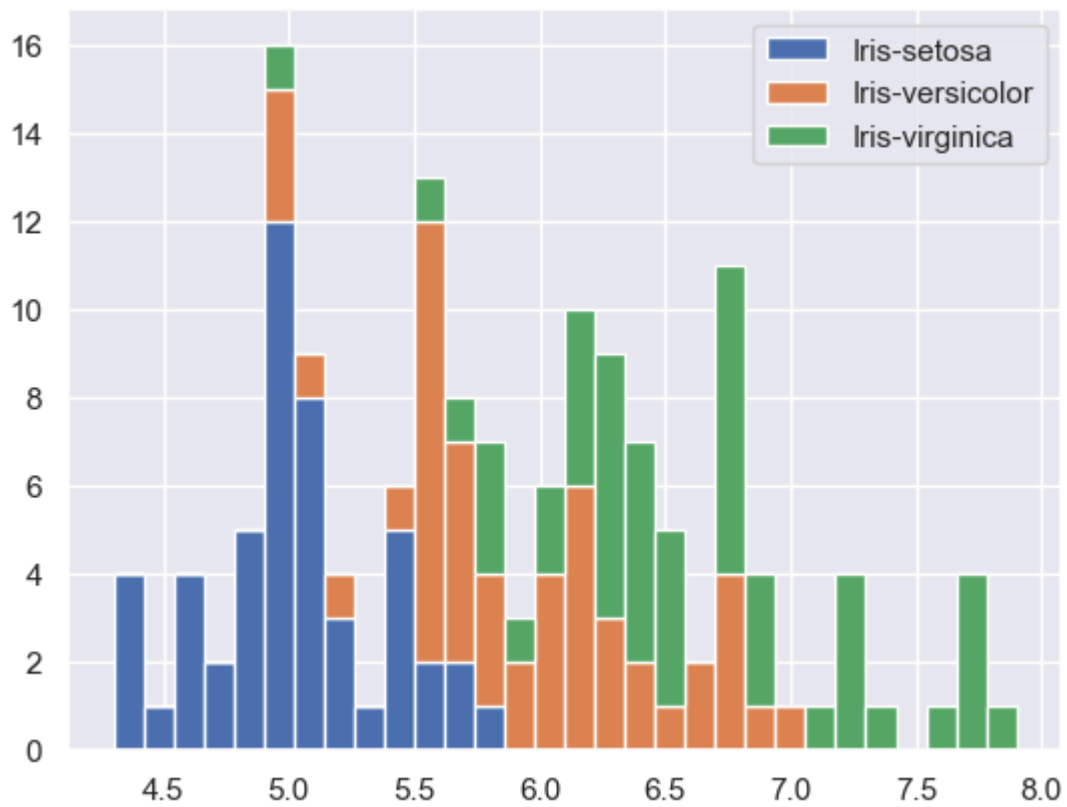
k1 = sns.boxplot(x='Species', y='PetalLength', data=iris, ax=axes[0, 0])
k2 = sns.violinplot(x='Species', y='PetalLength', data=iris, ax=axes[0, 1])
k3 = sns.stripplot(x='Species', y='SepalLength', data=iris, jitter=True, edgecolor='gray')
axes[1, 1].hist(iris.PetalLength, bins=100)
plt.show()
```



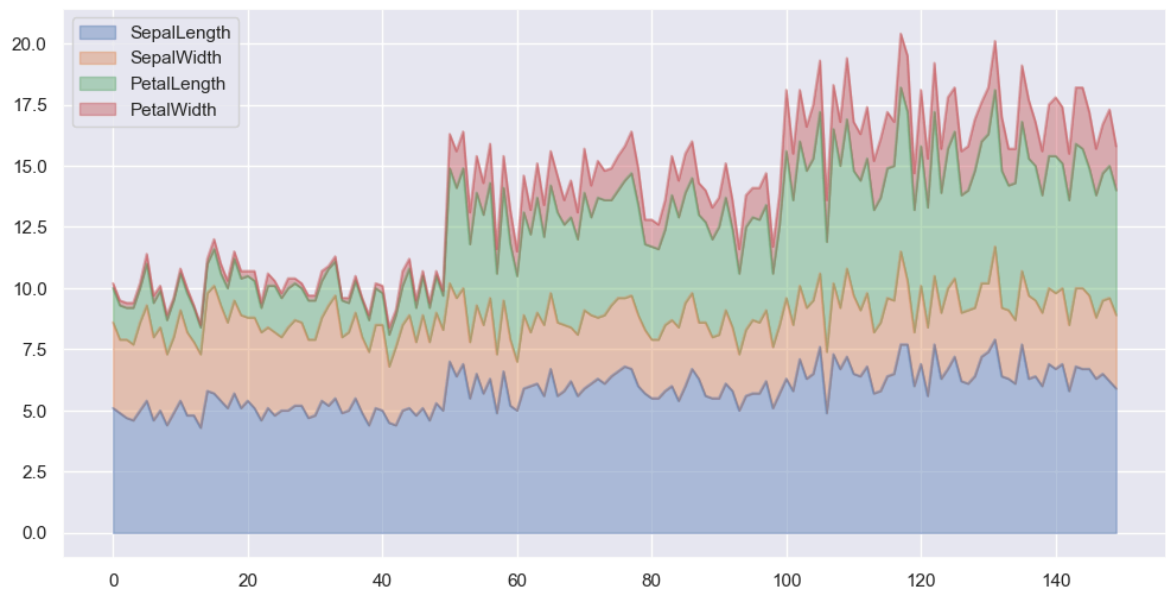
```
In [43]: iris['Specie']=iris['Specie'].astype('category')
```

```
In [45]: list1=list()
mylabels=list()
for gen in iris.Specie.cat.categories:
    list1.append(iris[iris.Specie==gen].Sepal.Length)
    mylabels.append(gen)

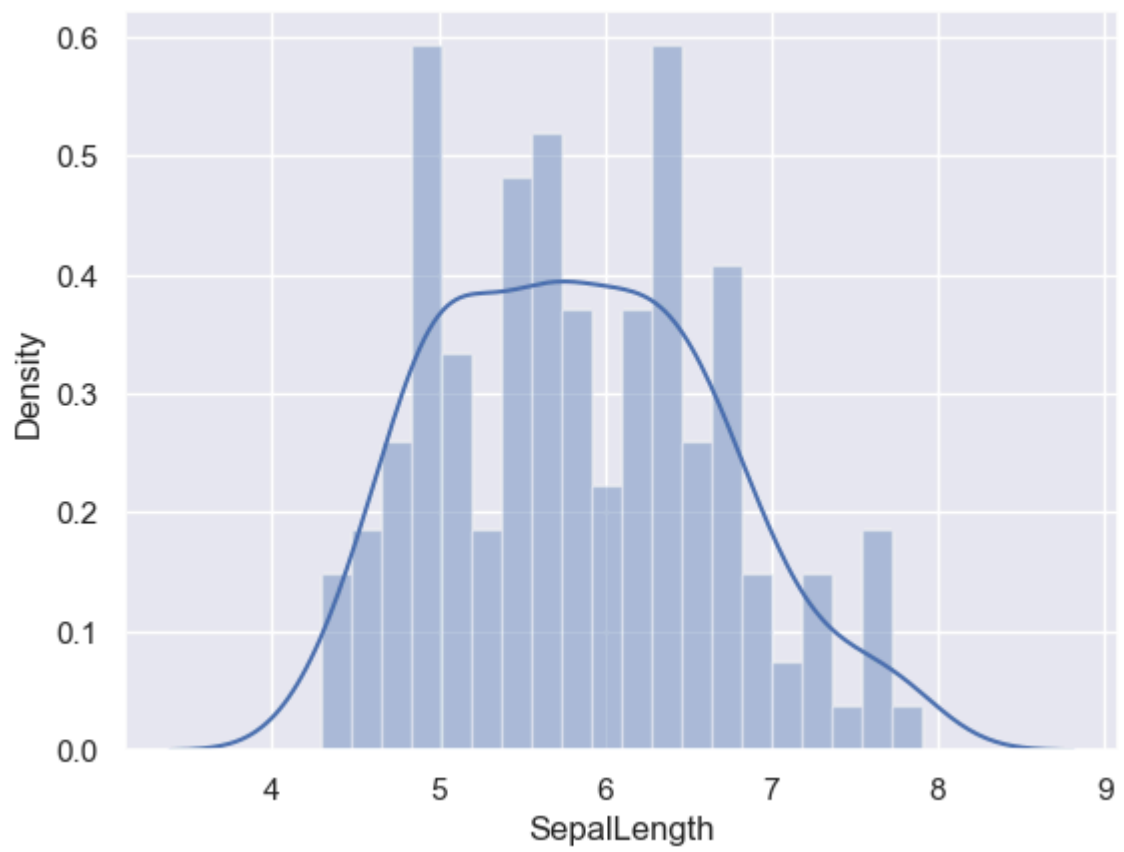
h=plt.hist(list1,bins=30,stacked=True,rwidth=1,label=mylabels)
plt.legend()
plt.show()
```



```
In [47]: iris.plot.area(y=['SepalLength', 'SepalWidth', 'PetalLength', 'PetalWidth'], alpha=0.5,
plt.show())
```



```
In [49]: sns.distplot(iris['SepalLength'], kde=True, bins=20)
plt.show()
```



Completed