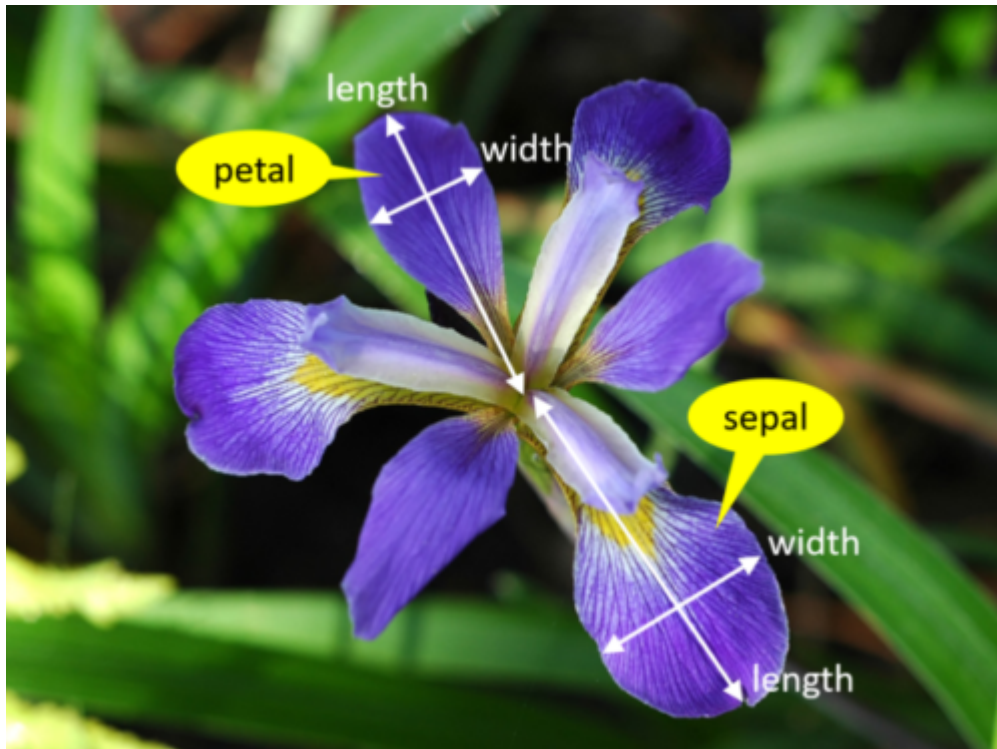


IRIS DATASET VISUALIZATION(SEABORN,MATPLOTLIB)



```
In [13]: import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
```

```
In [15]: # Importing pandas and Seaborn module

import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
#plt.style.use('fivethirtyeight')
import warnings
warnings.filterwarnings('ignore') #this will ignore the warnings.it wont displa
```

```
In [17]: #Importing Iris data set

iris=pd.read_csv(r'E:\Data Science & AI\Dataset files\Iris.csv')
```

```
In [19]: iris
```

Out[19]:

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

150 rows × 6 columns

In [21]: *# Displaying Data*
iris.head()

Out[21]:

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

In [23]: iris.drop('Id',axis=1,inplace=True)

In [25]: iris.head()

```
Out[25]:
```

	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 [27]: # Checking if there are any missing values
```

```
iris.info()
```

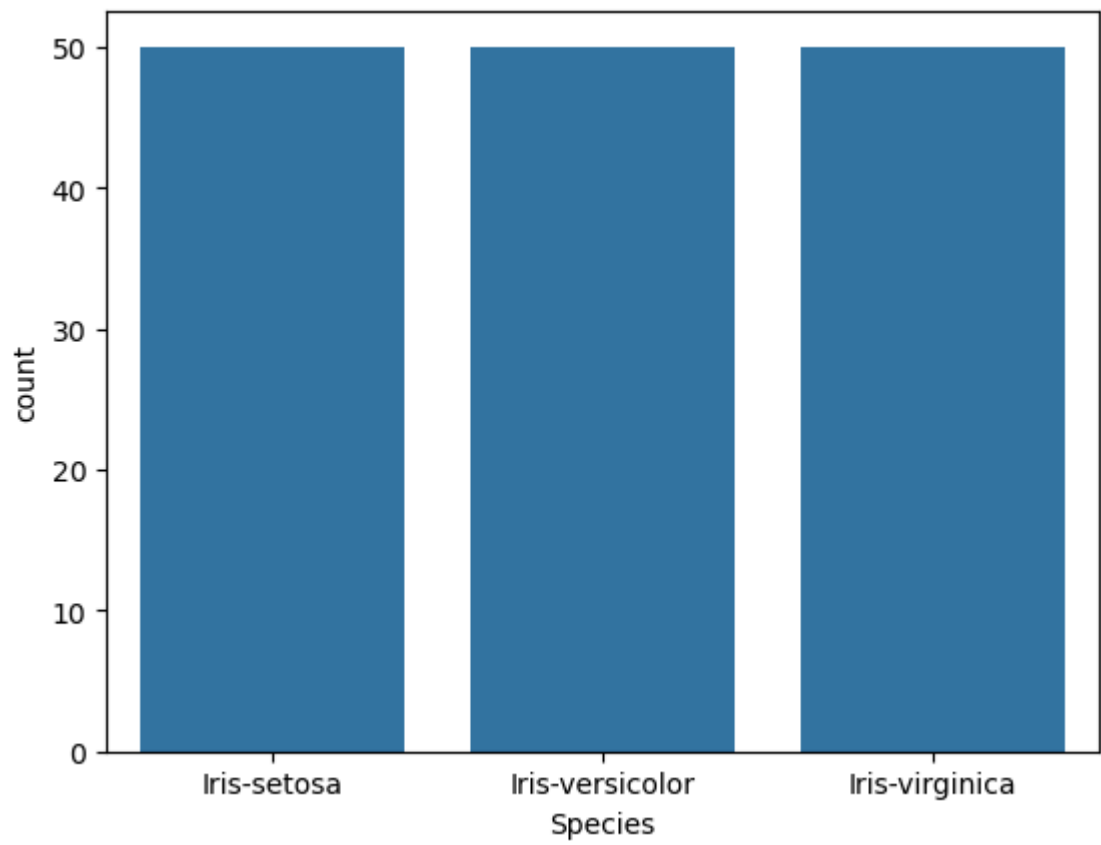
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 5 columns):
#   Column          Non-Null Count  Dtype
---  -
0   SepalLengthCm   150 non-null   float64
1   SepalWidthCm    150 non-null   float64
2   PetalLengthCm   150 non-null   float64
3   PetalWidthCm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

```
In [29]: iris['Species'].value_counts()
```

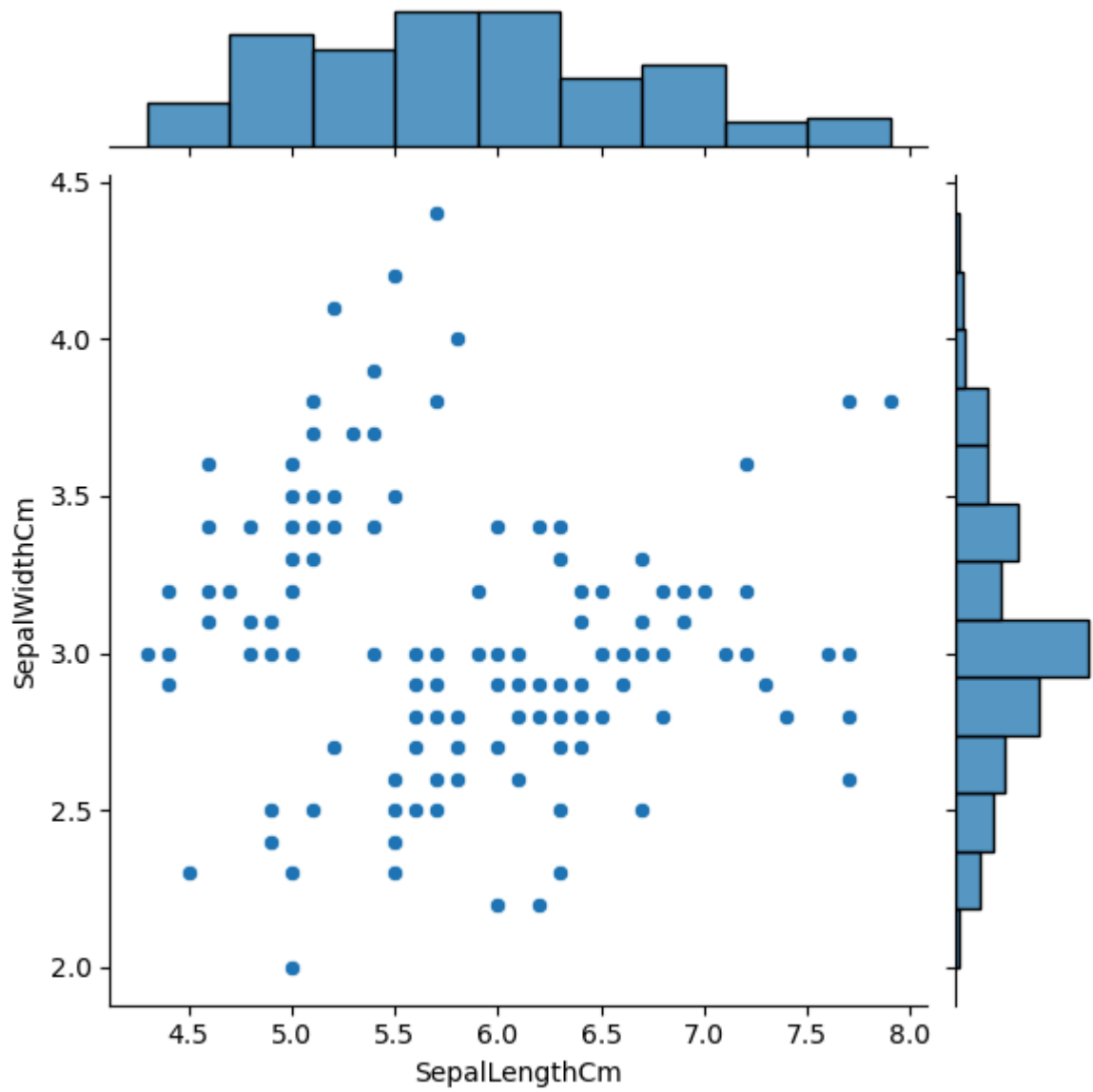
```
Out[29]: Species
Iris-setosa      50
Iris-versicolor  50
Iris-virginica   50
Name: count, dtype: int64
```

```
In [39]: # Bar Plot
```

```
sns.countplot(iris,x= 'Species')
plt.show()
```

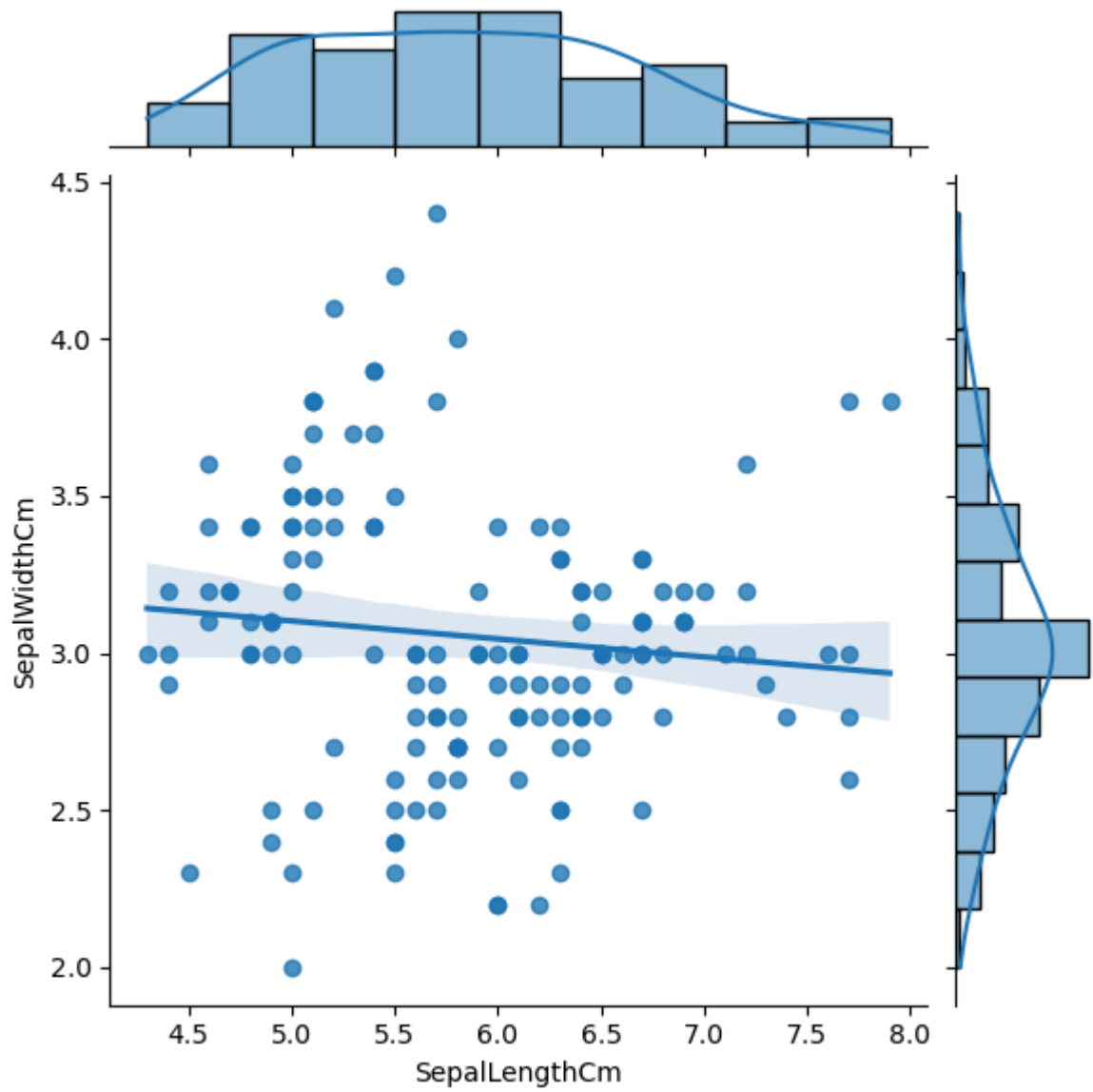


```
In [41]: # Jointplot
fig=sns.jointplot(x='SepalLengthCm',y='SepalWidthCm',data=iris)
```

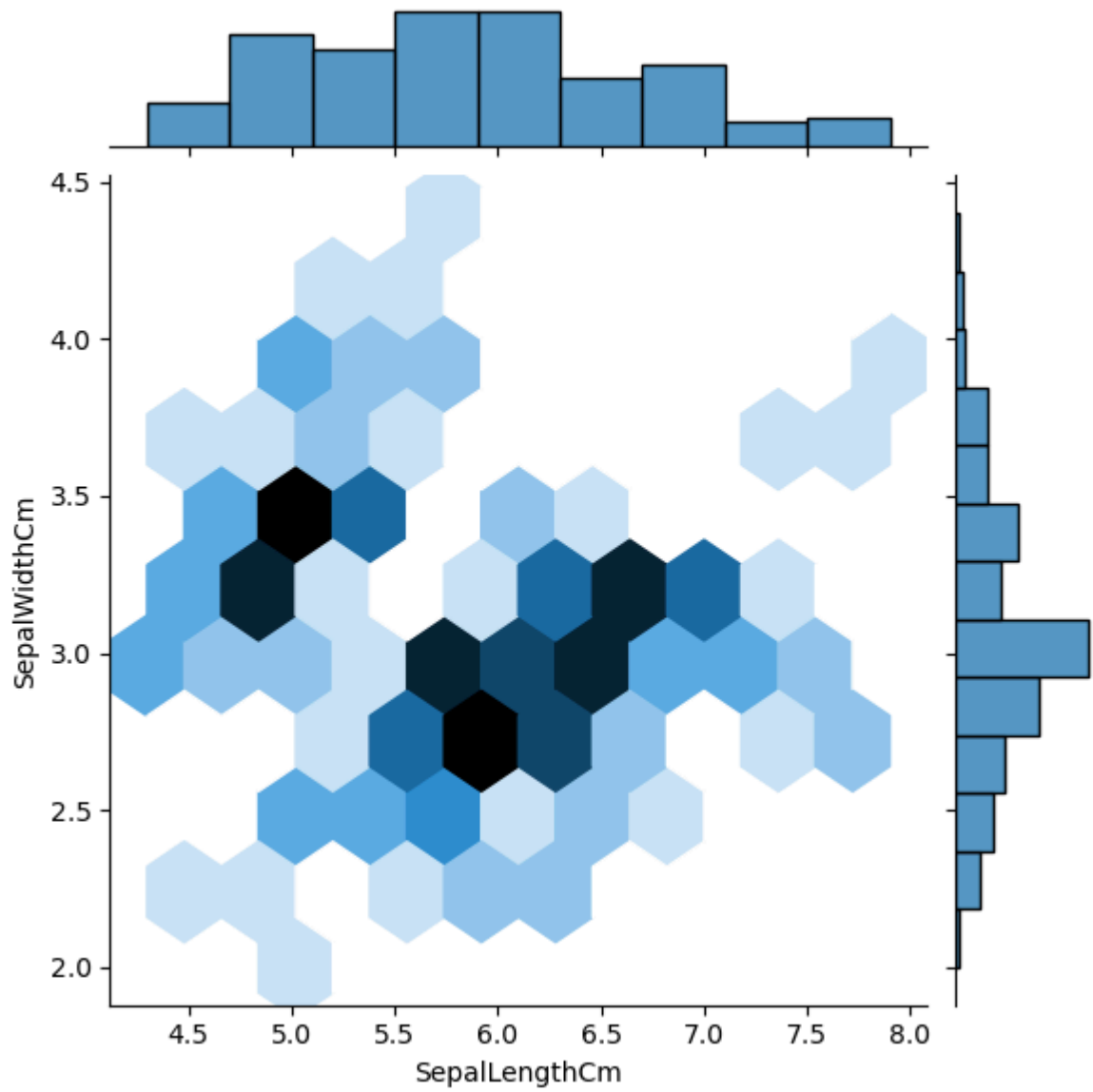


```
In [45]: sns.jointplot(iris,x="SepalLengthCm", y="SepalWidthCm", kind="reg")
```

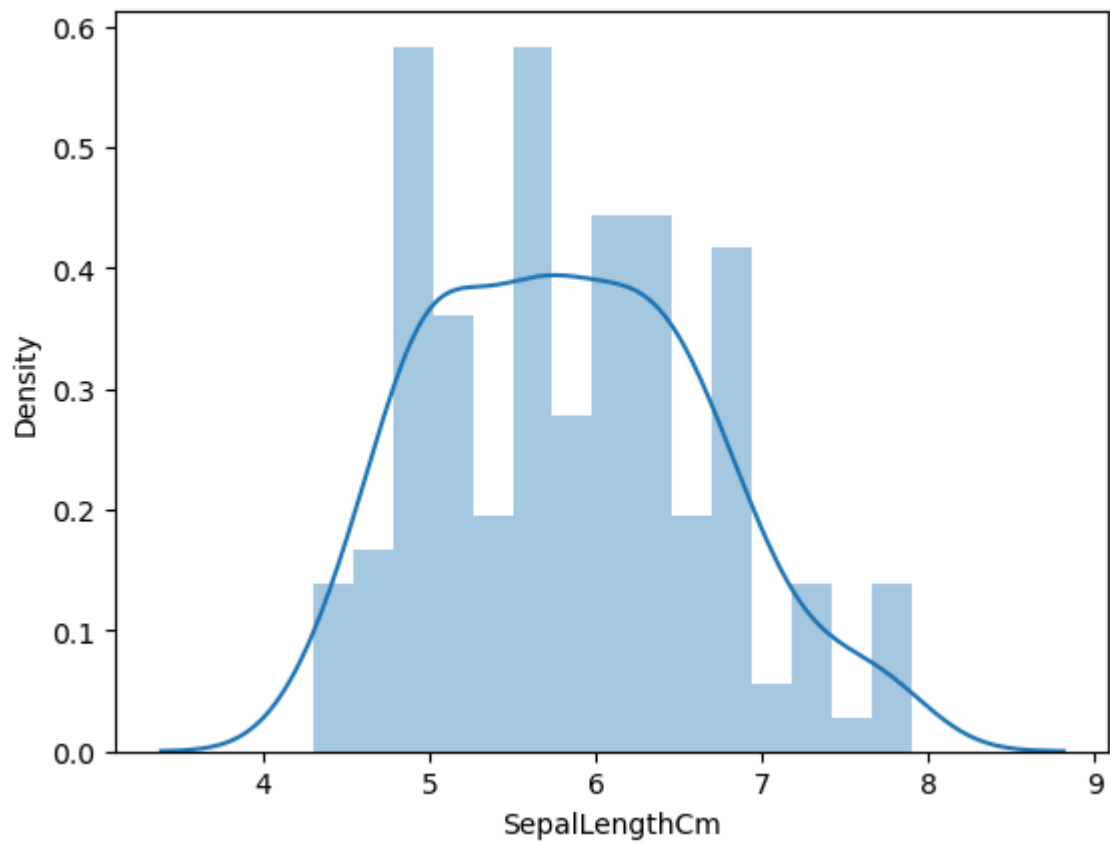
```
Out[45]: <seaborn.axisgrid.JointGrid at 0x27ebd340e60>
```



```
In [47]: fig=sns.jointplot(x='SepalLengthCm',y='SepalWidthCm',kind='hex',data=iris)
```



```
In [67]: #Distplot
sns.distplot(iris['SepalLengthCm'],kde = True,bins = 15);
```



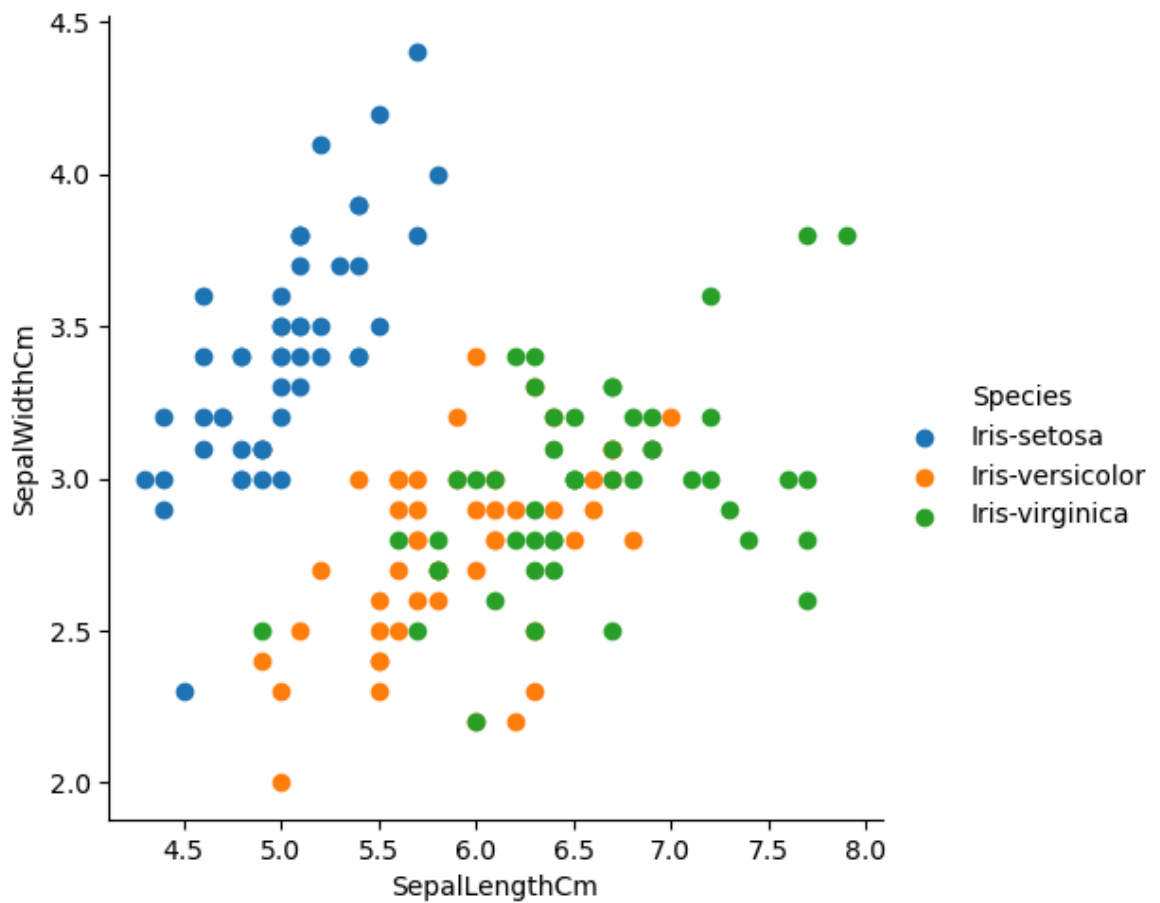
```
In [69]: plt.show()
```

```
In [63]: # Facetgrid plot

import matplotlib.pyplot as plt
%matplotlib inline
```

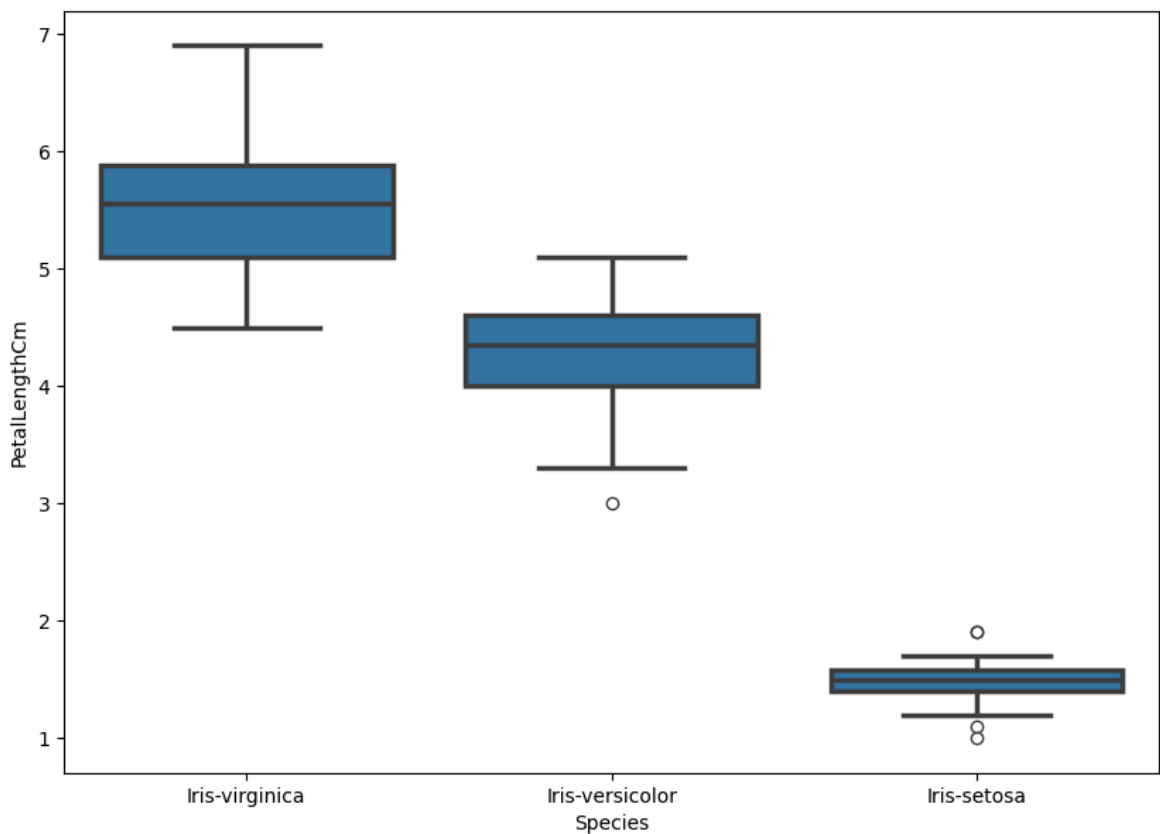
```
In [65]: sns.FacetGrid(iris , hue='Species' ,height=5)\
        .map(plt.scatter, 'SepalLengthCm', 'SepalWidthCm')\
        .add_legend()
```

```
Out[65]: <seaborn.axisgrid.FacetGrid at 0x27ebe3d9a30>
```

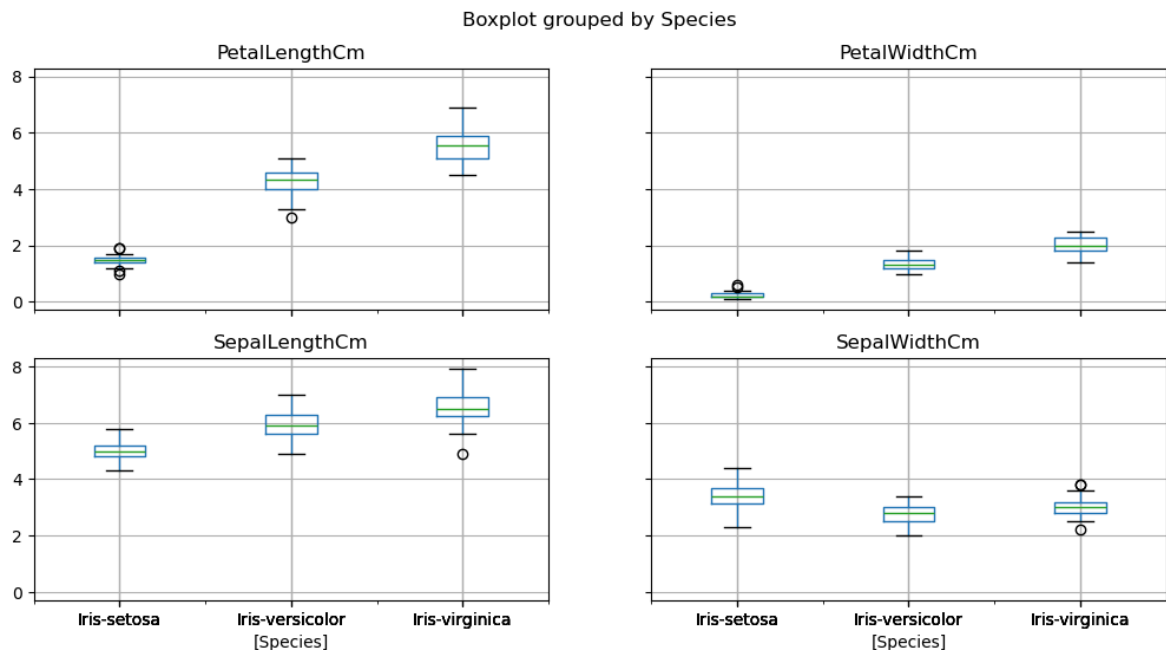
```
In [71]: # Boxplot or Whisker plot

fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.boxplot(x='Species',y='PetalLengthCm',data=iris,order=['Iris-virginica',
```



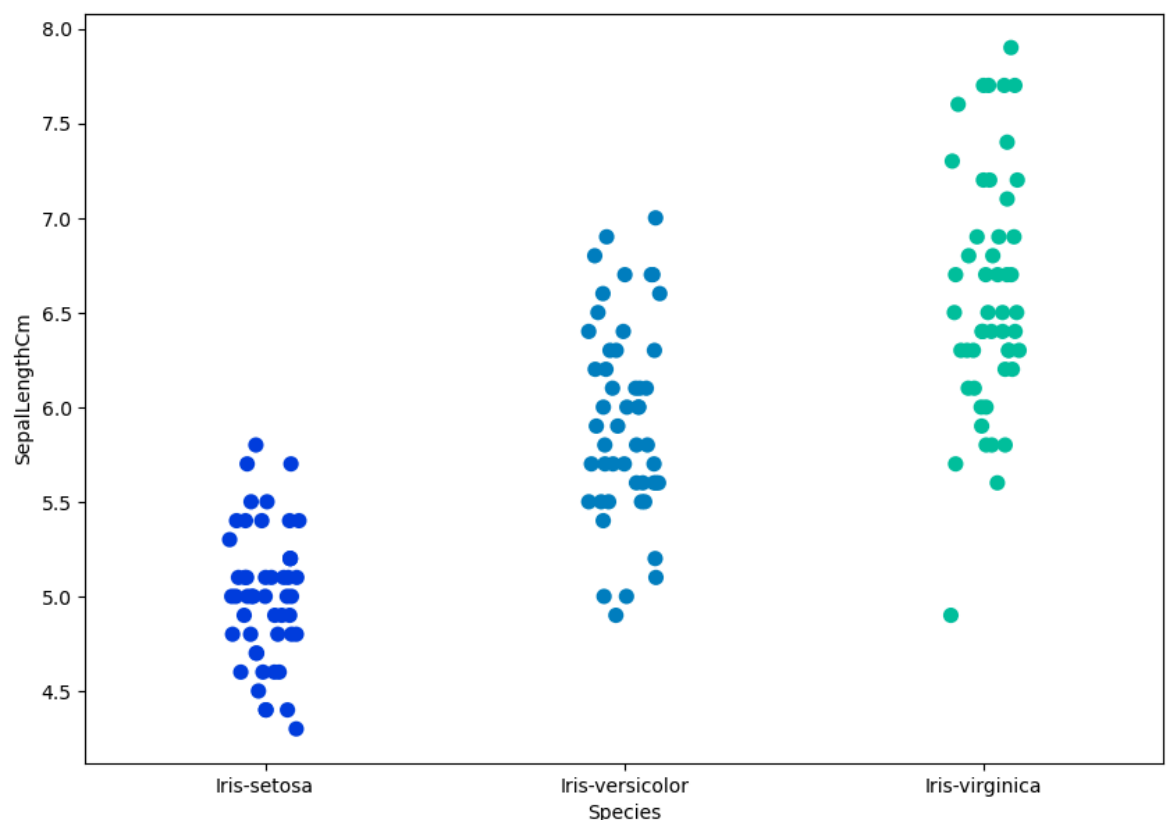
```
In [73]: #iris.drop("Id", axis=1).boxplot(by="Species", figsize=(12, 6))
iris.boxplot(by="Species", figsize=(12, 6))
```

```
Out[73]: array([[<Axes: title={'center': 'PetalLengthCm'}, xlabel='[Species]>',
  <Axes: title={'center': 'PetalWidthCm'}, xlabel='[Species]>',
  <Axes: title={'center': 'SepalLengthCm'}, xlabel='[Species]>',
  <Axes: title={'center': 'SepalWidthCm'}, xlabel='[Species]>]],
  dtype=object)
```



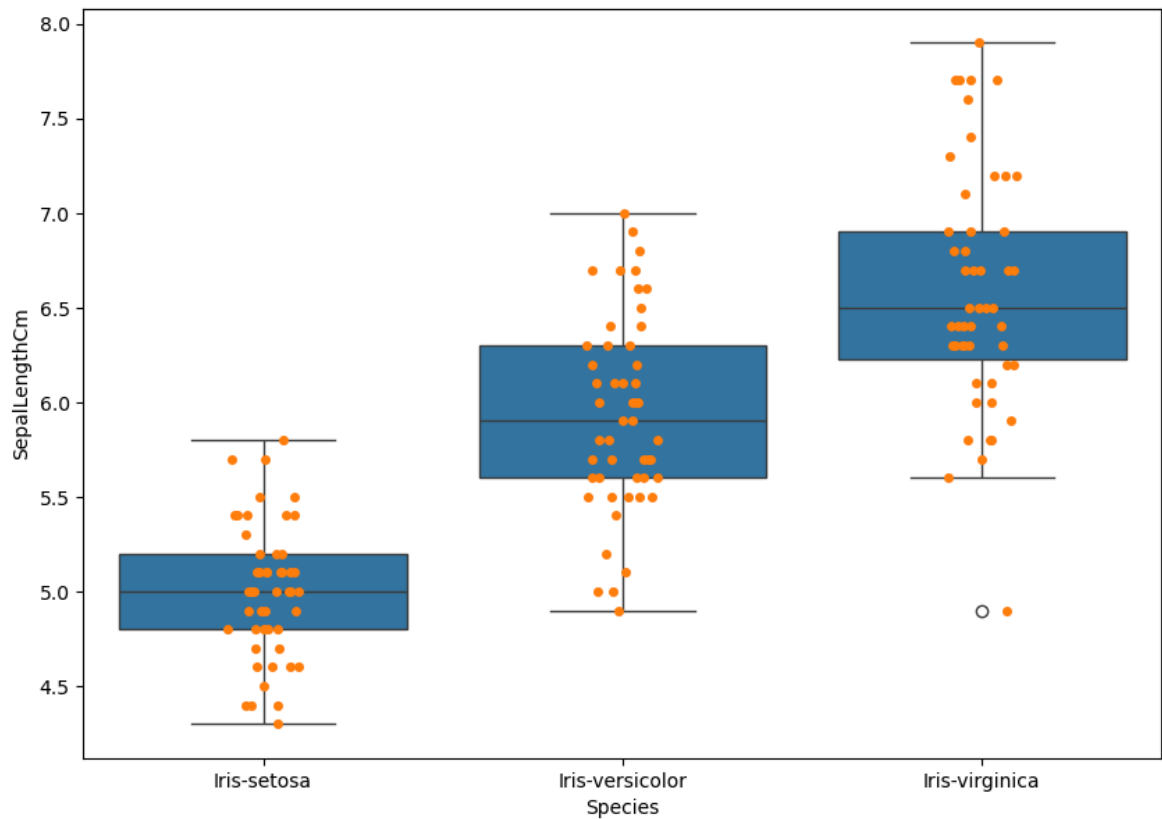
```
In [75]: # Strip plot

fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.stripplot(x='Species',y='SepalLengthCm',data=iris,jitter=True,edgecolor=
```

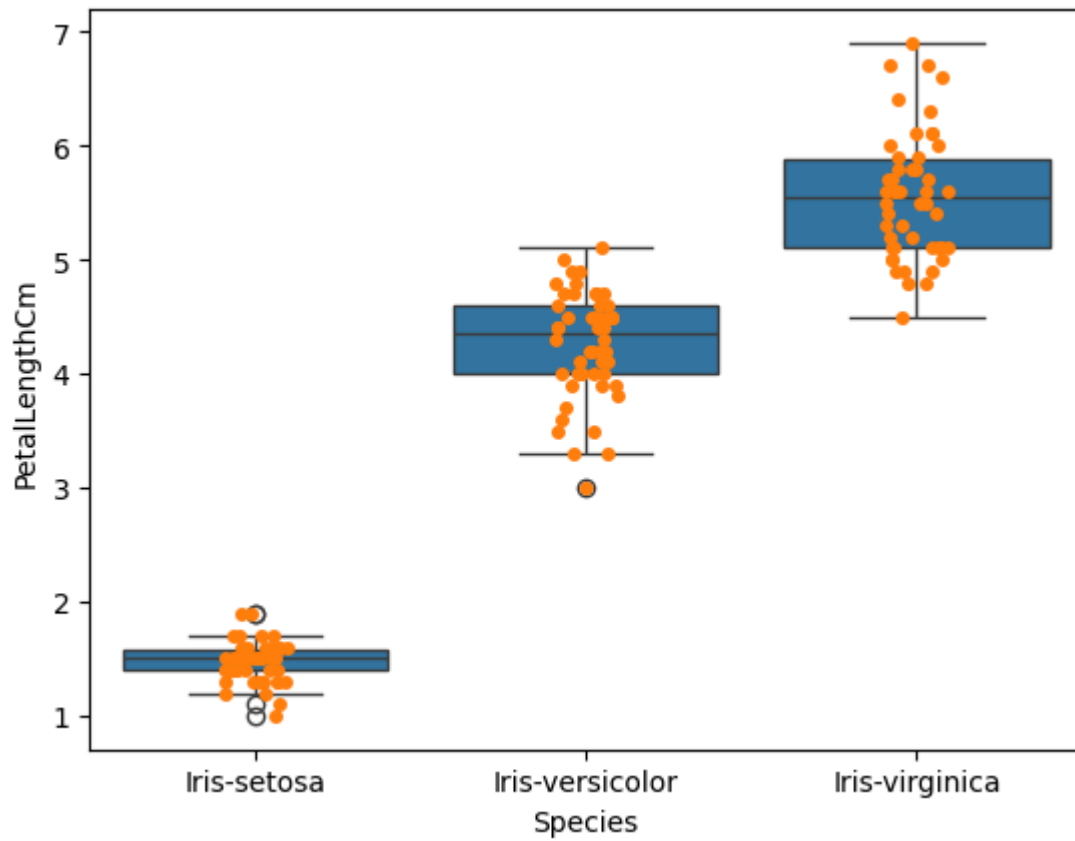


```
In [77]: # Combining Box and Strip Plots
```

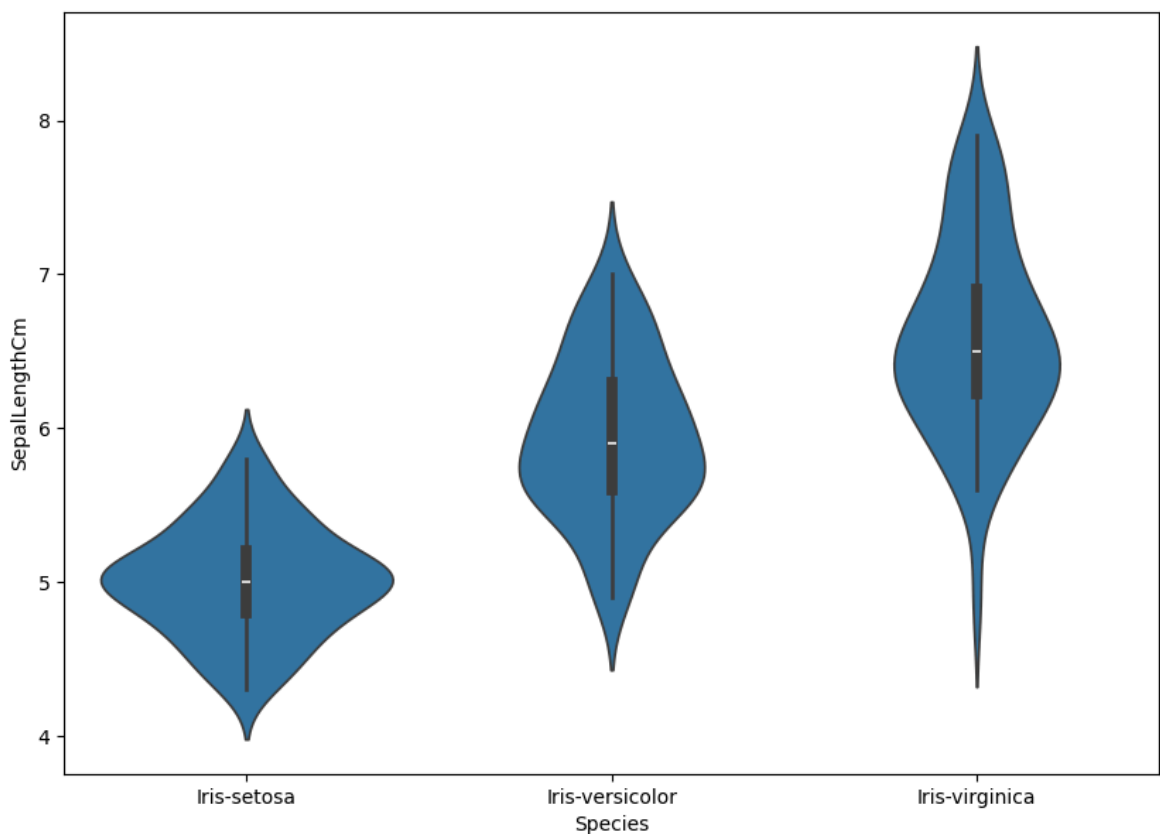
```
fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.boxplot(x='Species',y='SepalLengthCm',data=iris)
fig=sns.stripplot(x='Species',y='SepalLengthCm',data=iris,jitter=True,edgecolor=
```



```
In [83]: ax= sns.boxplot(x="Species", y="PetalLengthCm", data=iris)
ax= sns.stripplot(x="Species", y="PetalLengthCm", data=iris, jitter=True, edgeco
```



```
In [87]: # Violin plot
fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.violinplot(x='Species',y='SepalLengthCm',data=iris)
```



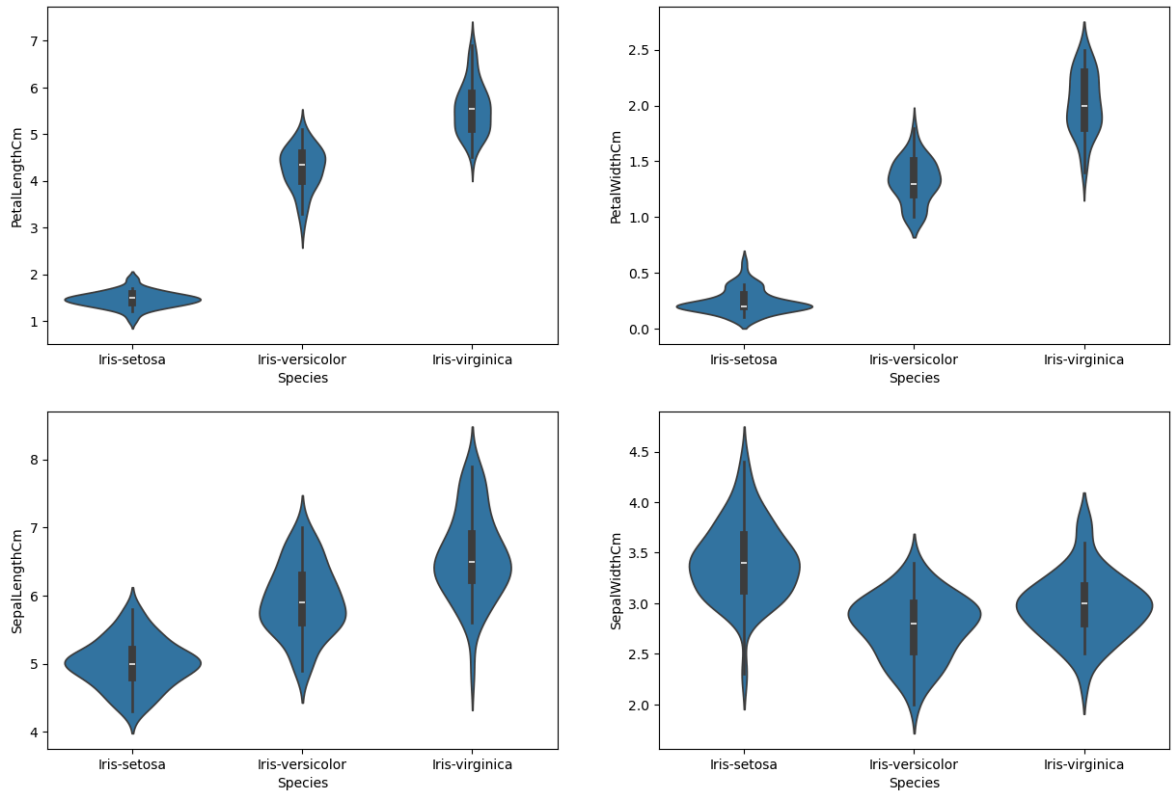
```
In [89]: plt.figure(figsize=(15,10))
plt.subplot(2,2,1)
```

```

sns.violinplot(x='Species',y='PetalLengthCm',data=iris)
plt.subplot(2,2,2)
sns.violinplot(x='Species',y='PetalWidthCm',data=iris)
plt.subplot(2,2,3)
sns.violinplot(x='Species',y='SepalLengthCm',data=iris)
plt.subplot(2,2,4)
sns.violinplot(x='Species',y='SepalWidthCm',data=iris)

```

Out[89]: <Axes: xlabel='Species', ylabel='SepalWidthCm'>



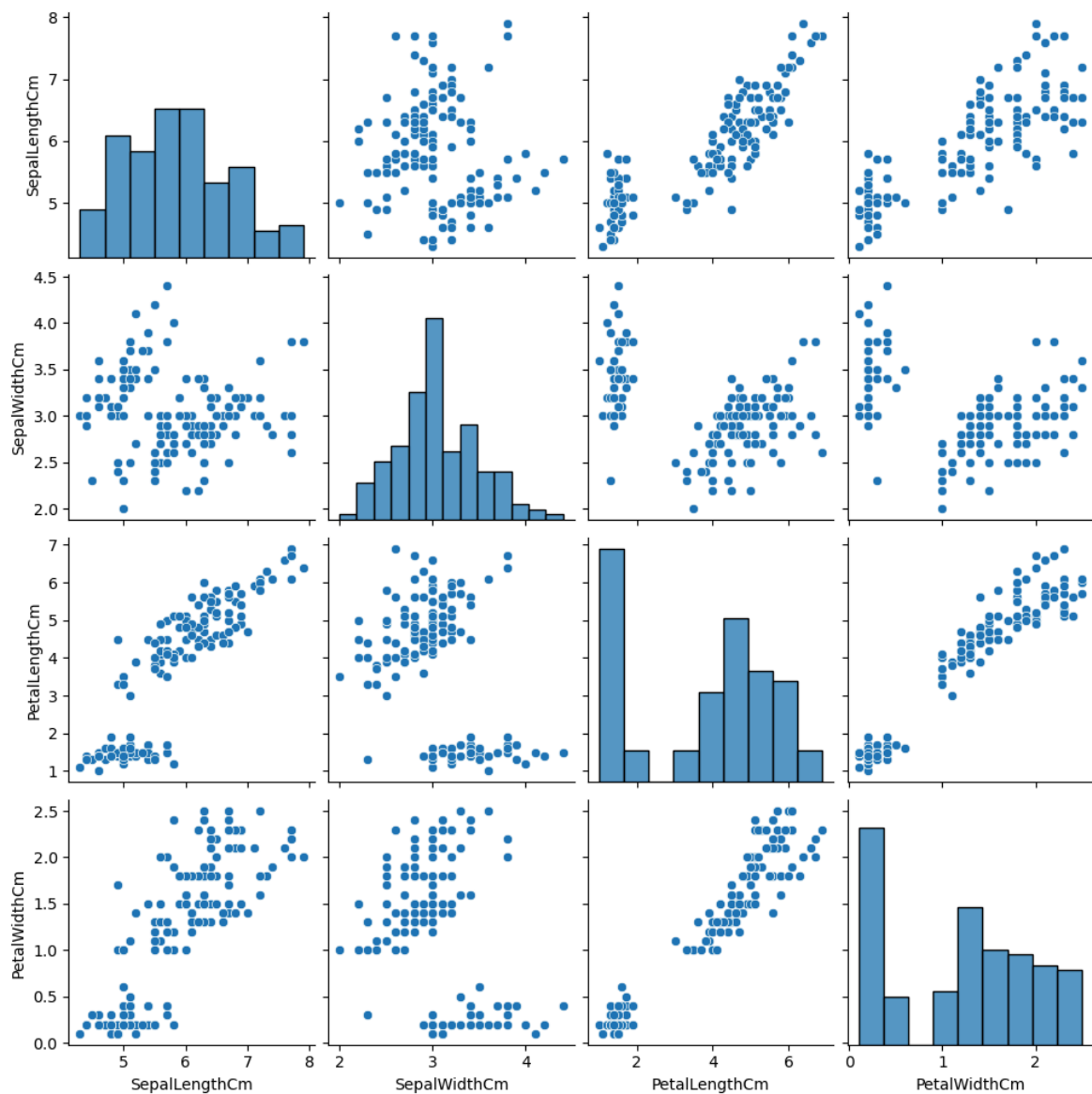
```

In [91]: # Pair plot

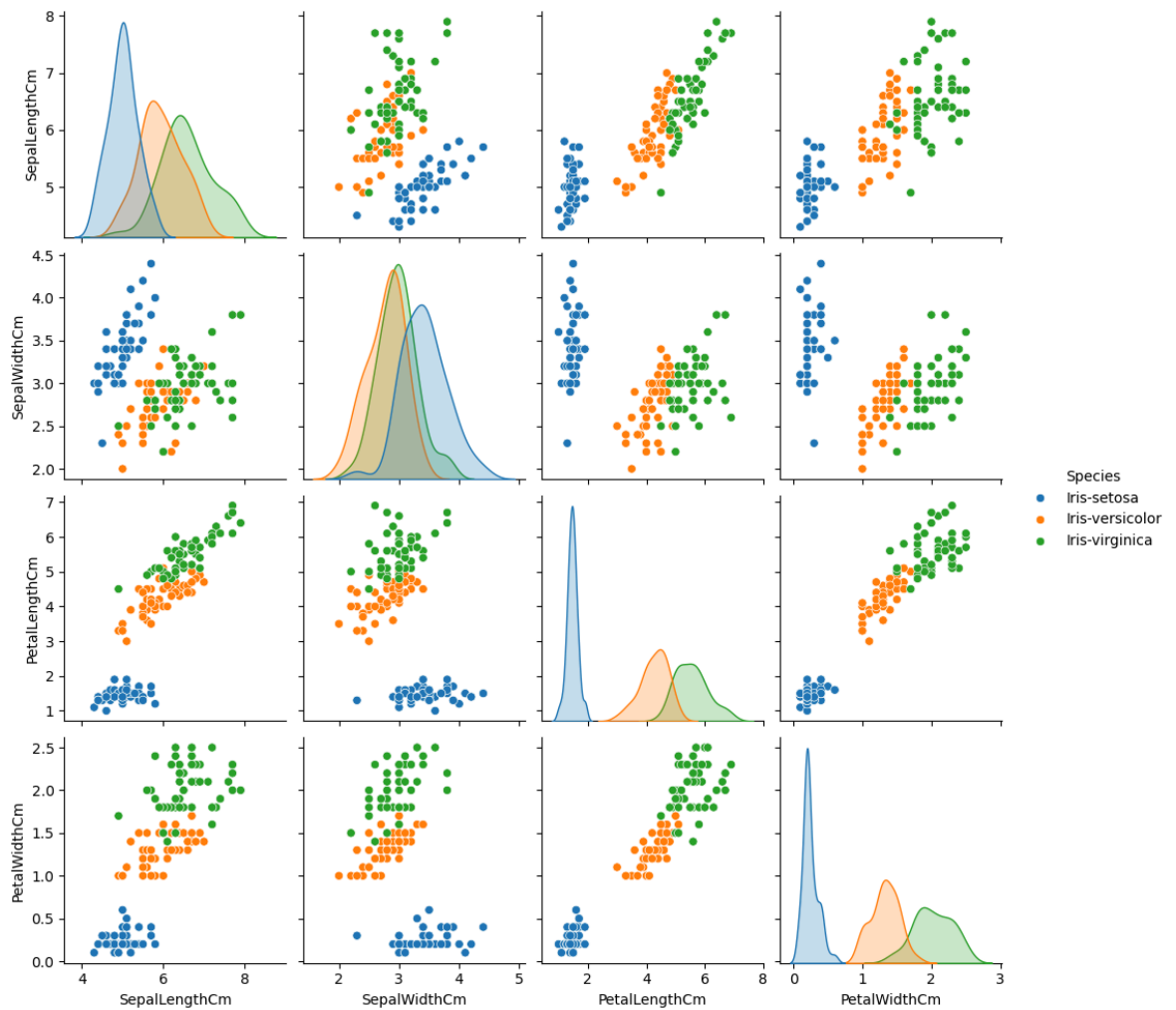
sns.pairplot(data=iris,kind='scatter')

```

Out[91]: <seaborn.axisgrid.PairGrid at 0x27ec09ae000>



```
In [93]: sns.pairplot(iris,hue='Species');
```



```
In [103... iris1 = iris.copy()
```

```
In [105... iris1
```

```
Out[105...
```

	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
...
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
147	6.5	3.0	5.2	2.0	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 5 columns

```
In [107... iris1.drop('Species',axis=1, inplace =True)
```

```
In [109... iris
```

```
Out[109... 
```

	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
...
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
147	6.5	3.0	5.2	2.0	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 5 columns

```
In [111... iris1
```

```
Out[111... 
```

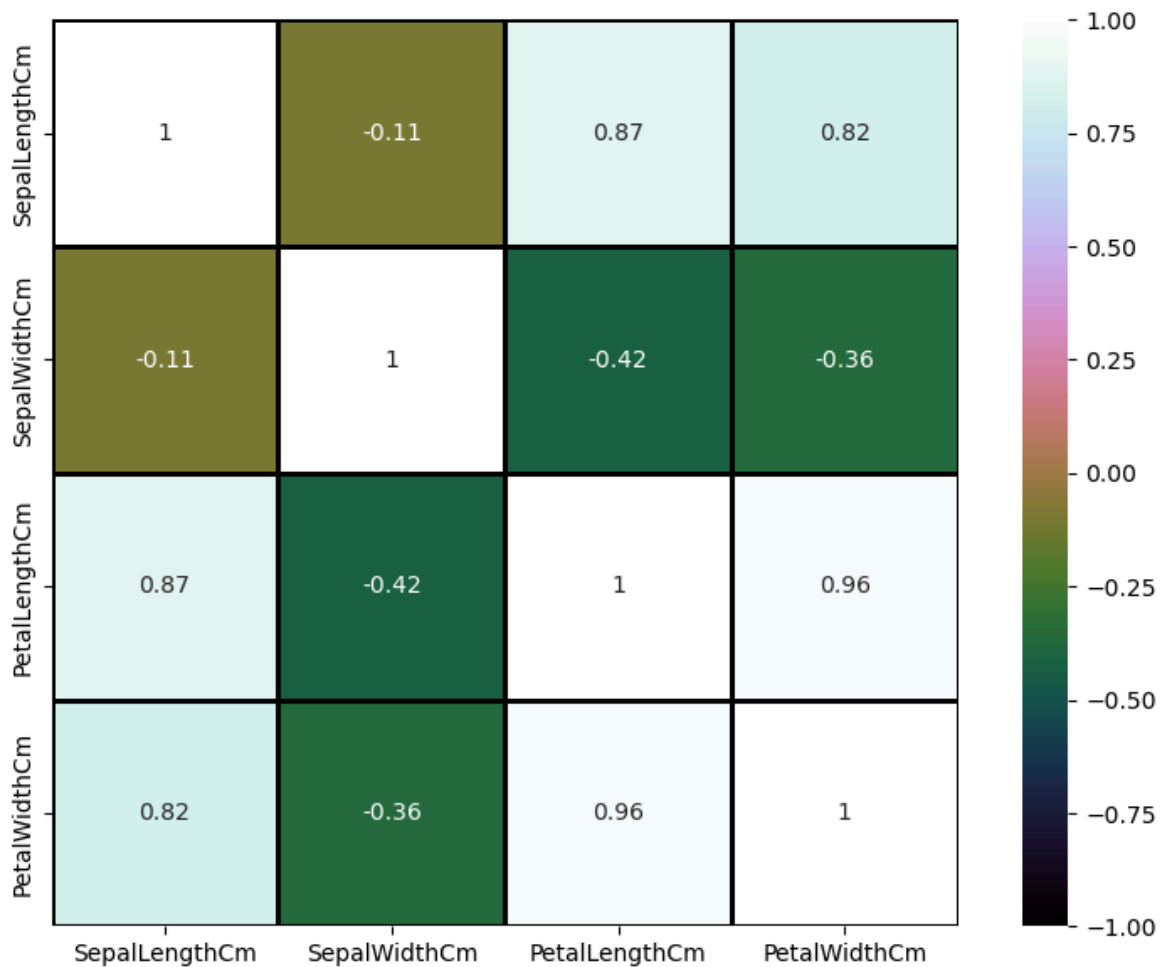
	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2
...
145	6.7	3.0	5.2	2.3
146	6.3	2.5	5.0	1.9
147	6.5	3.0	5.2	2.0
148	6.2	3.4	5.4	2.3
149	5.9	3.0	5.1	1.8

150 rows × 4 columns

```
In [113... # Heat map  
fig=plt.gcf()
```

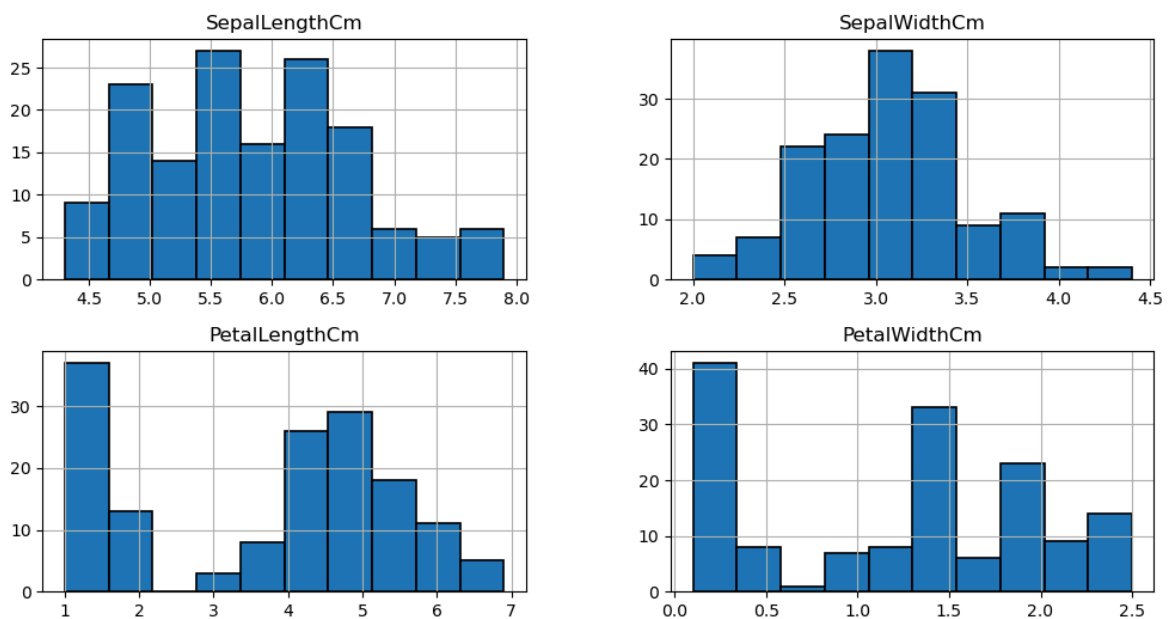


```
fig.set_size_inches(10,7)
fig=sns.heatmap(iris1.corr(),annot=True,cmap='cubehelix',linewidths=1,linewidth=
```



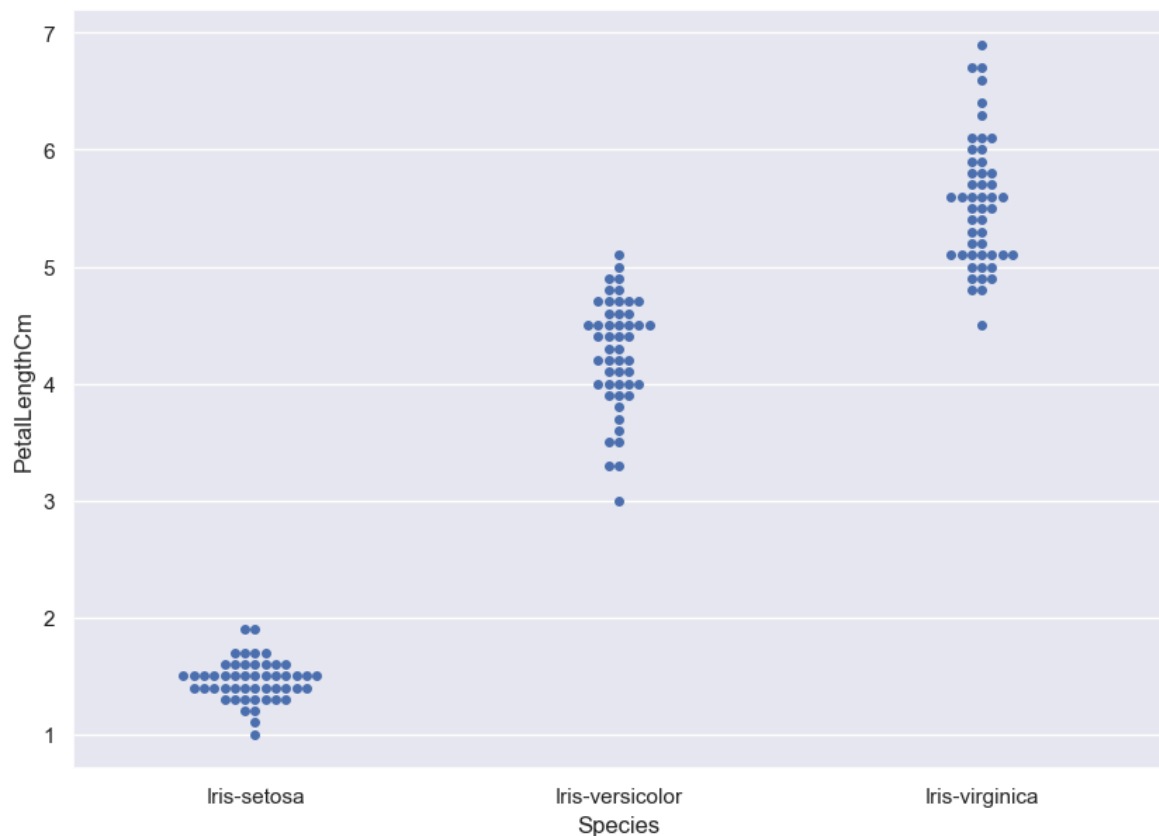
```
In [115... # Distribution plot

iris.hist(edgecolor='black', linewidth=1.2)
fig=plt.gcf()
fig.set_size_inches(12,6)
```



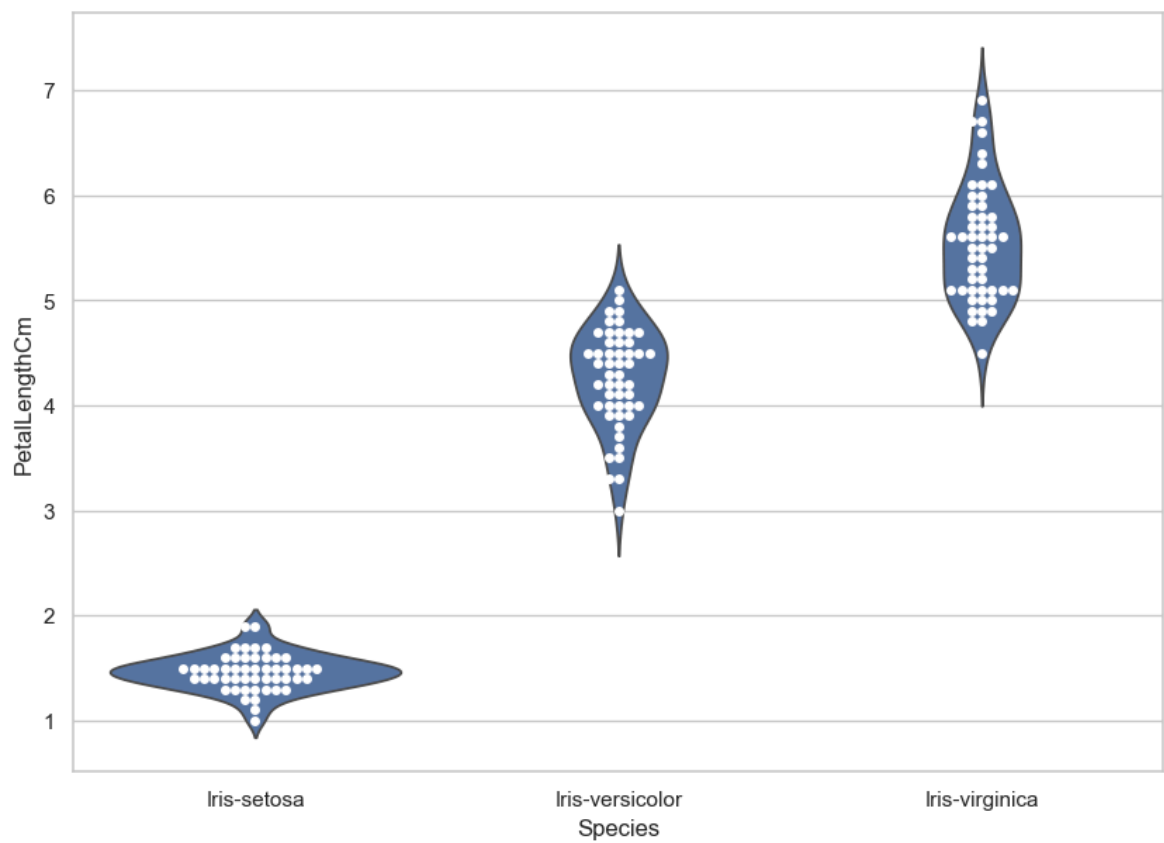
```
In [117... # Swarm plot
```

```
sns.set(style="darkgrid")
fig=plt.gcf()
fig.set_size_inches(10,7)
fig = sns.swarmplot(x="Species", y="PetalLengthCm", data=iris)
```



In [119...

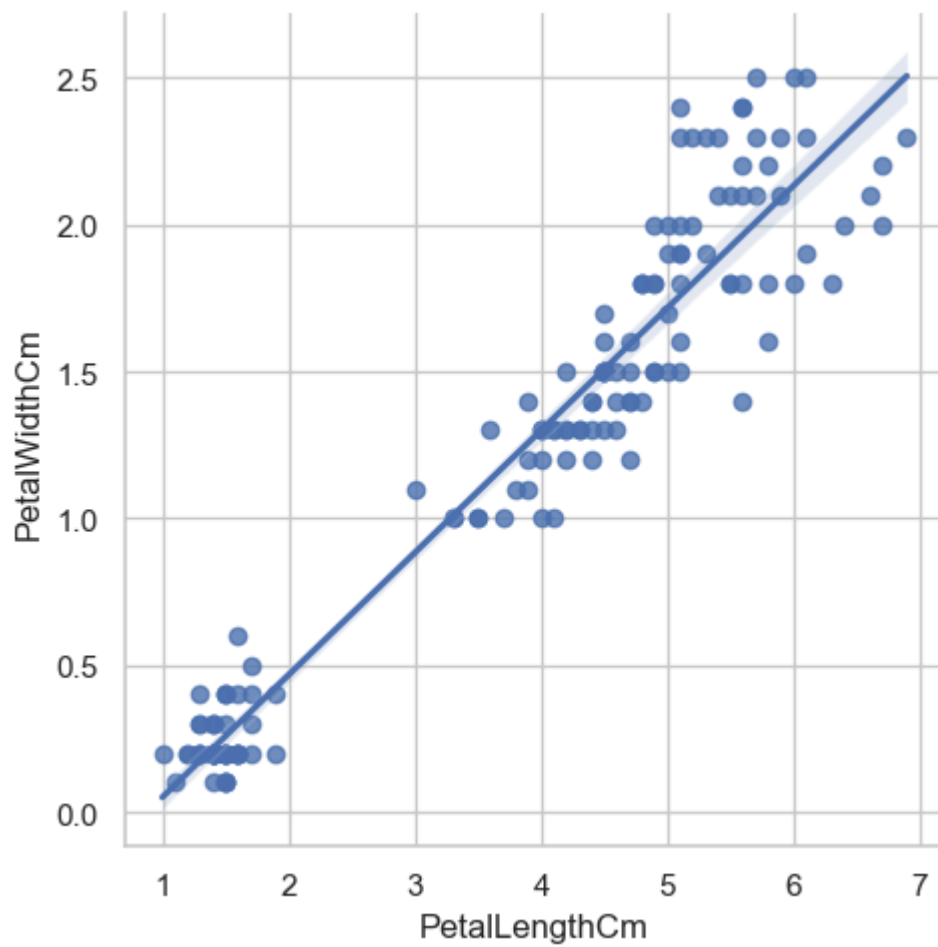
```
sns.set(style="whitegrid")
fig=plt.gcf()
fig.set_size_inches(10,7)
ax = sns.violinplot(x="Species", y="PetalLengthCm", data=iris, inner=None)
ax = sns.swarmplot(x="Species", y="PetalLengthCm", data=iris,color="white", edge
```



In [121...

```
# LM Plot
```

```
fig=sns.lmplot(x="PetalLengthCm", y="PetalWidthCm",data=iris)
```

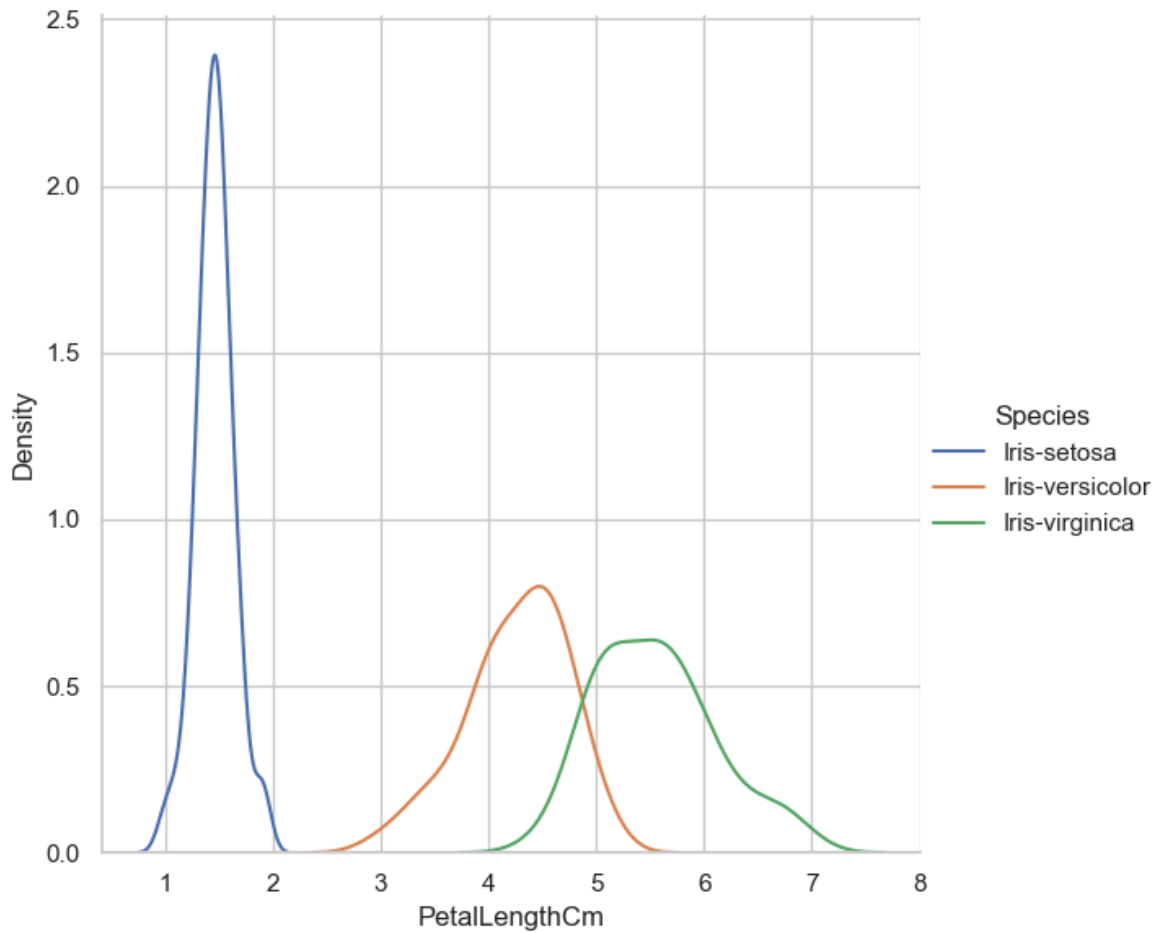


In [127...

```
# Facetgrid

sns.FacetGrid(iris, hue = 'Species' , height = 6)\
.map(sns.kdeplot, 'PetalLengthCm')\
.add_legend()\
plt.ioff()
```

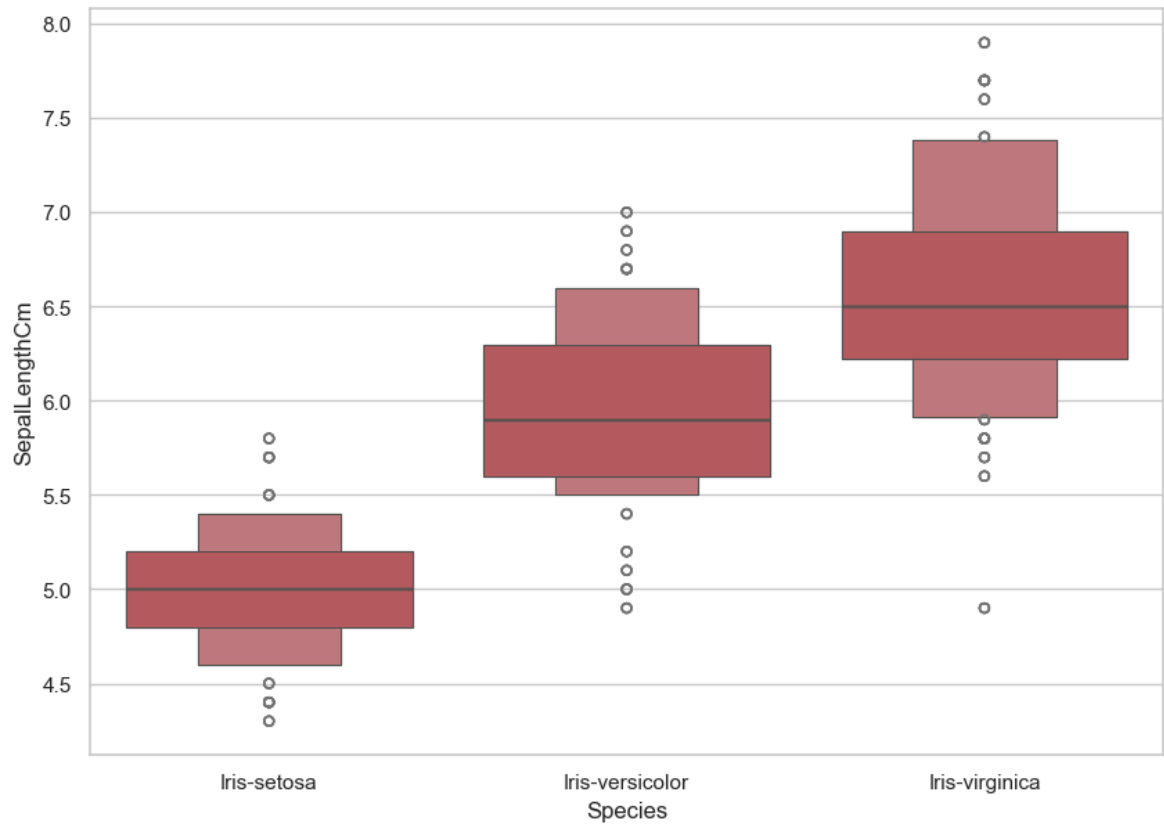
Out[127... <contextlib.ExitStack at 0x27ec7701220>



In [143...

```
# Boxenplot

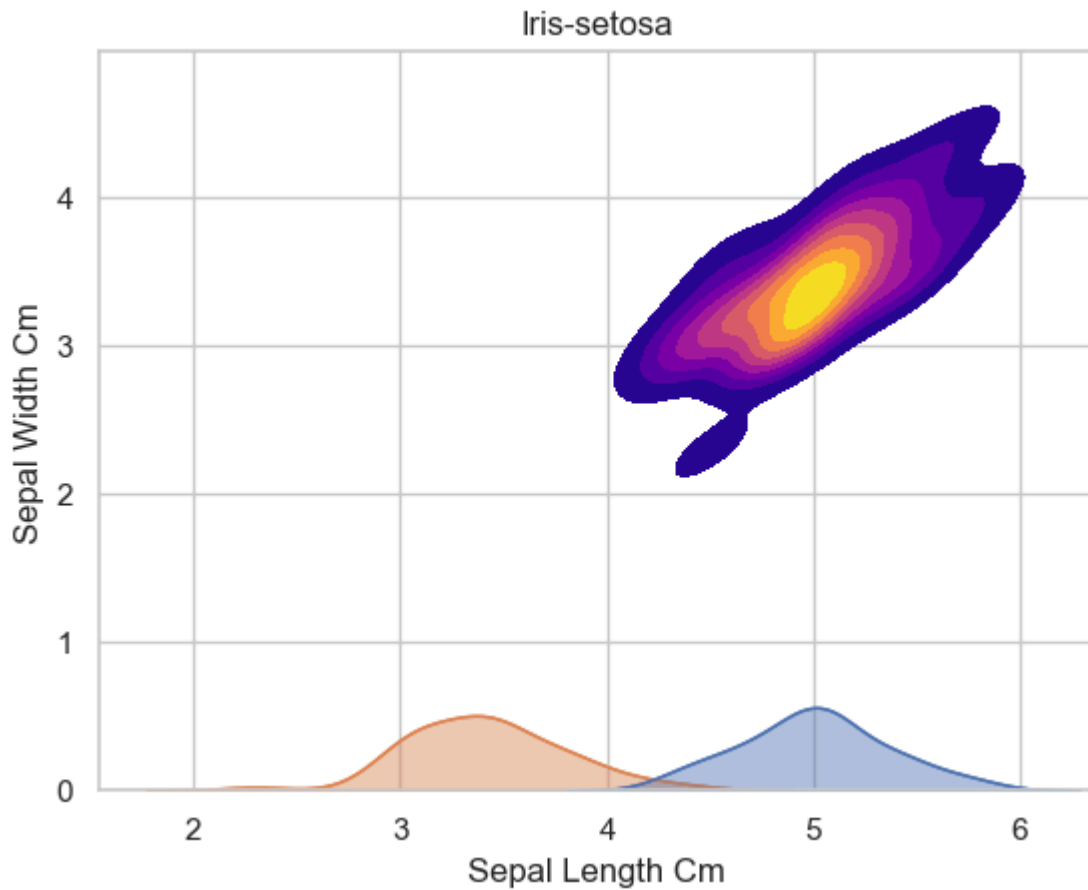
fig=plt.gcf()
fig.set_size_inches(10,7)
fig=sns.boxenplot(iris,x='Species',y='SepalLengthCm')
plt.show()
```



In [149...

```
# KDE plot

sub=iris[iris['Species']=='Iris-setosa']
sns.kdeplot(data=sub,x='SepalLengthCm',y='SepalWidthCm',cmap="plasma", shade=True)
plt.title('Iris-setosa')
plt.xlabel('Sepal Length Cm')
plt.ylabel('Sepal Width Cm')
plt.show()
```

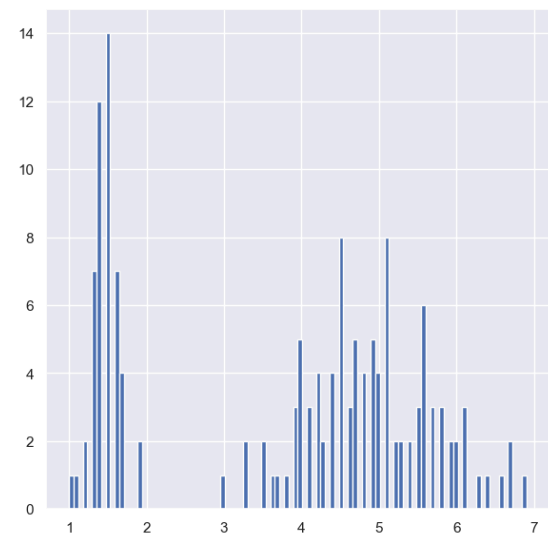
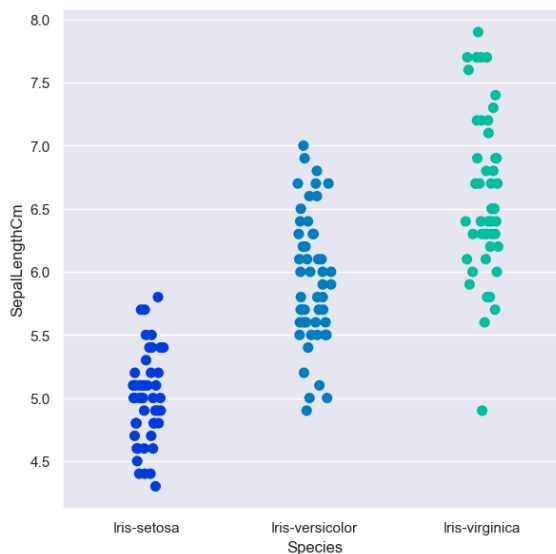
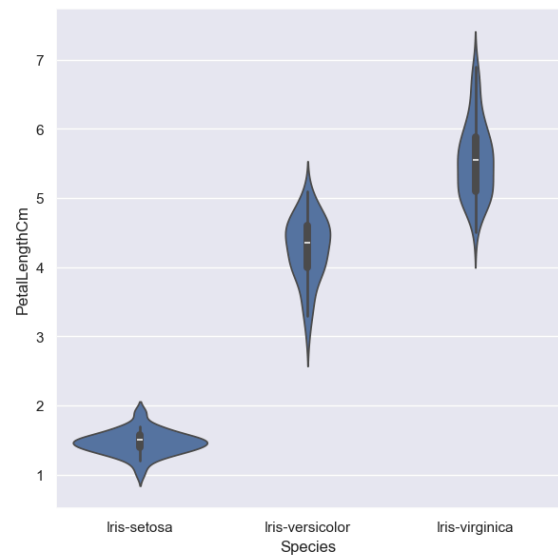
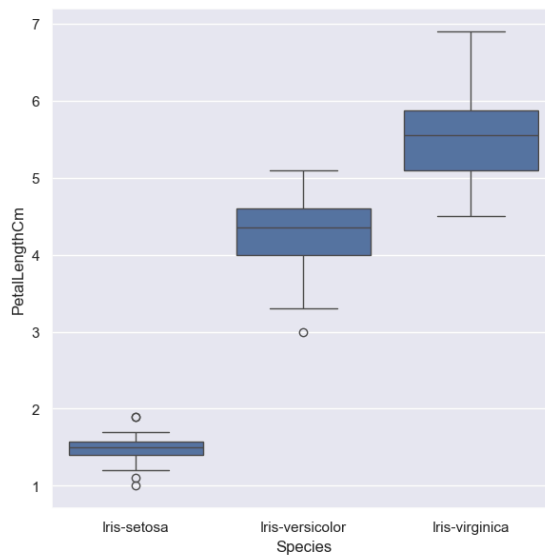


In [151...

```
# Dashboard

sns.set_style('darkgrid')
f, axes = plt.subplots(2, 2, figsize=(15, 15))

k1 = sns.boxplot(x="Species", y="PetalLengthCm", data=iris, ax=axes[0, 0])
k2 = sns.violinplot(x='Species', y='PetalLengthCm', data=iris, ax=axes[0, 1])
k3 = sns.stripplot(x='Species', y='SepalLengthCm', data=iris, jitter=True, edgecolor='
#axes[1, 1].hist(iris.hist, bin=10)
axes[1, 1].hist(iris.PetalLengthCm, bins=100)
#k2.set(xlim=(-1, 0.8))
plt.show()
```



In [157...

```
# Stacked Histogram

iris['Species'] = iris['Species'].astype('category')
iris.head()
```

Out[157...

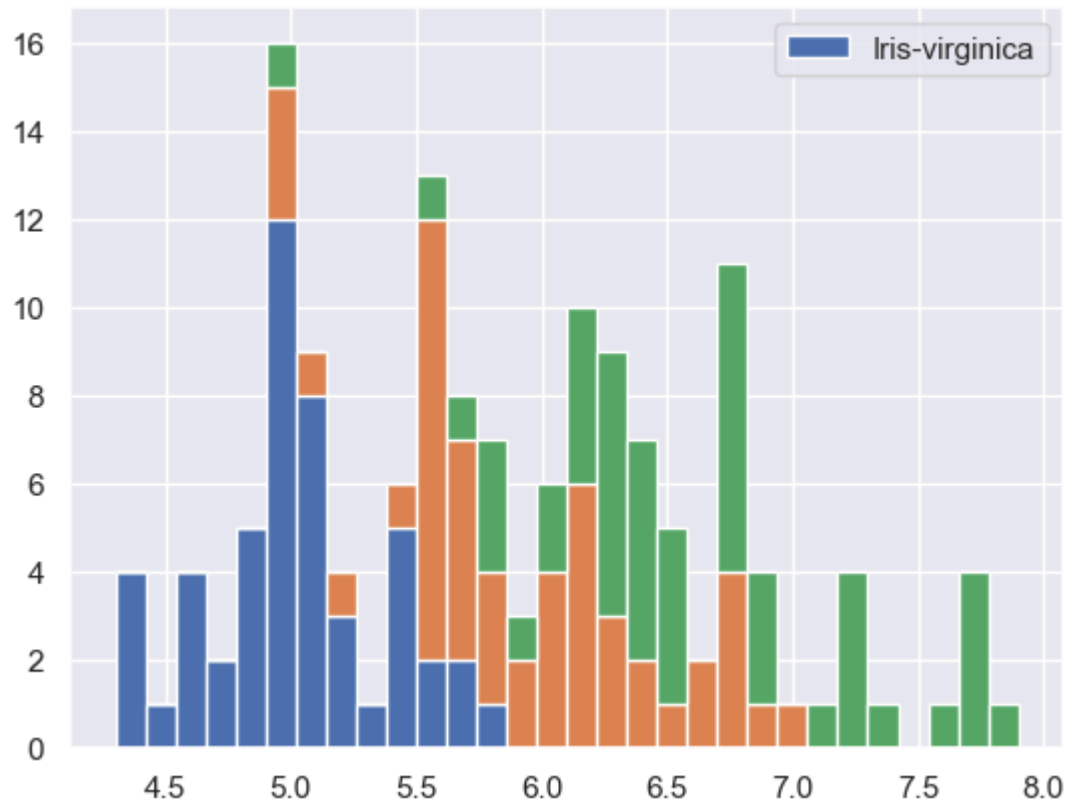
	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa

In [161...

```
list1=list()
mylabels=list()
for gen in iris.Species.cat.categories:
    list1.append(iris[iris.Species==gen].SepalLengthCm)
    mylabels.append(gen)

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

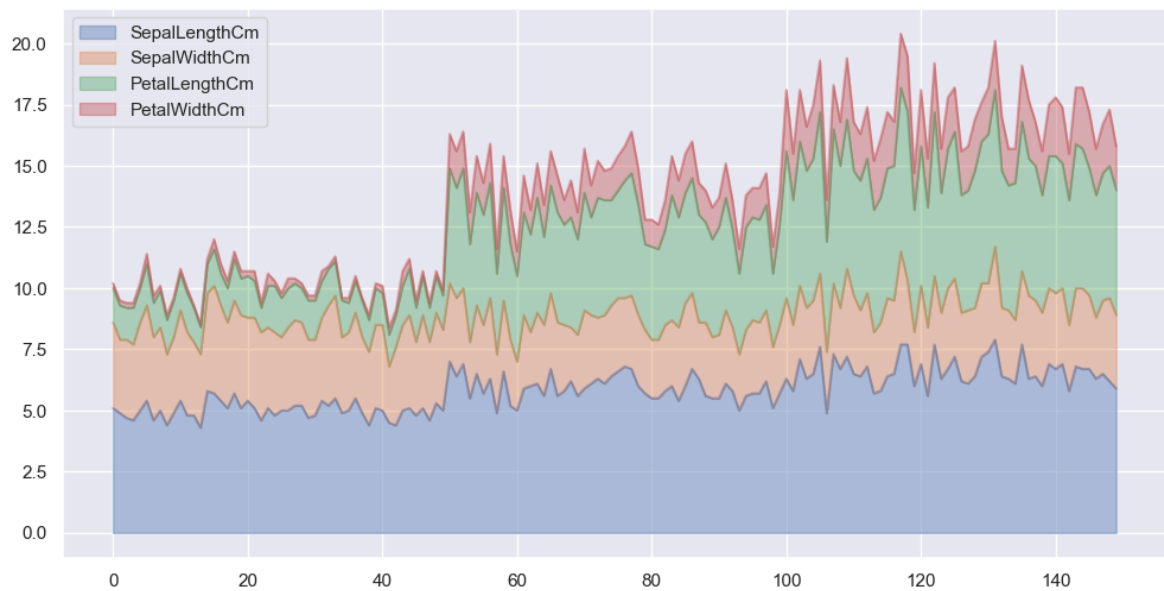
```
plt.legend()
plt.show()
```

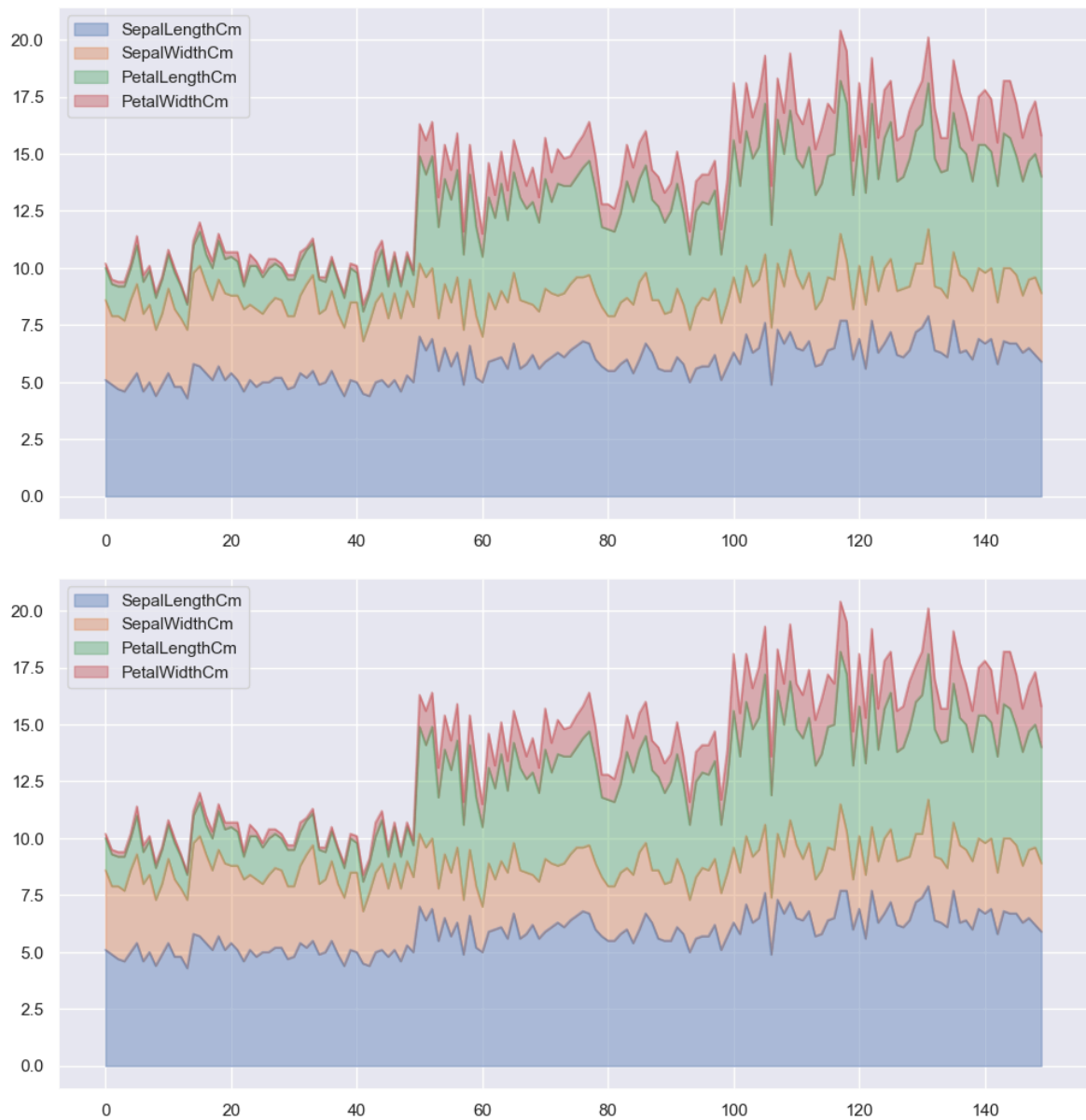


In [167...

```
# Area Plot
```

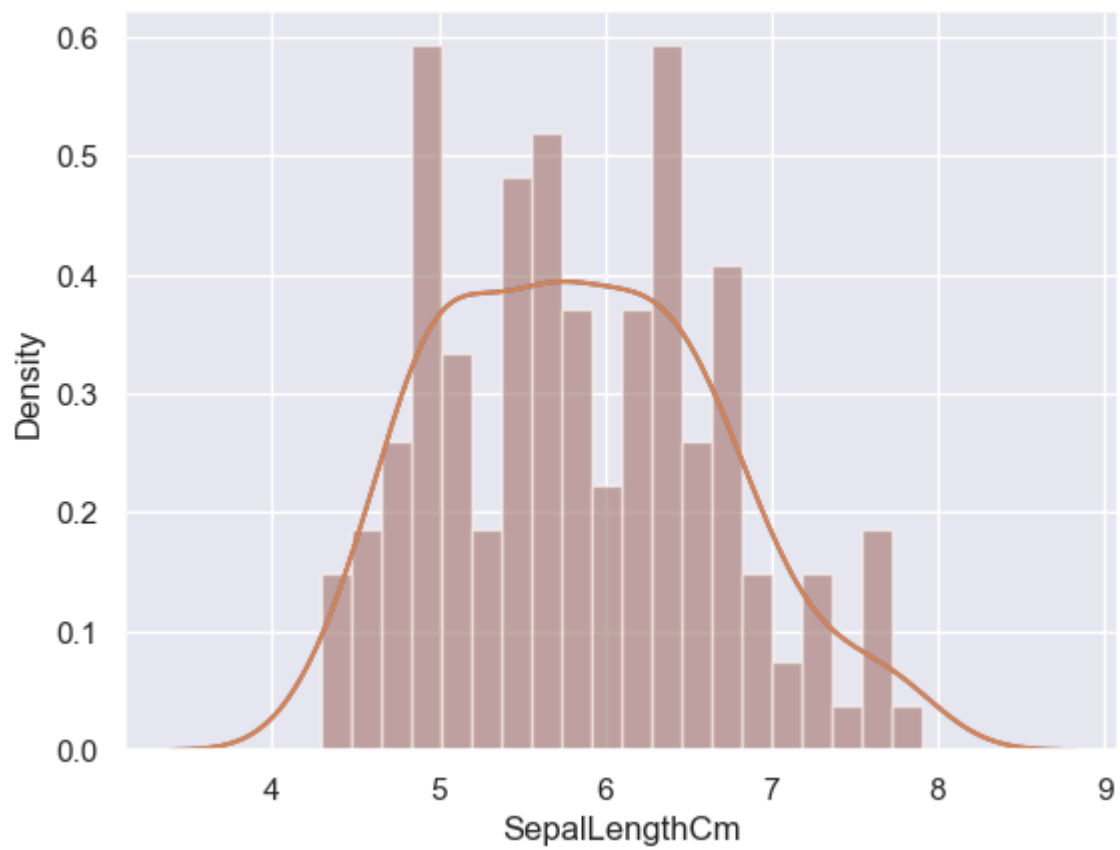
```
iris.plot.area(y=['SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm'])
plt.show()
```





```
In [171... # Distplot

sns.distplot(iris['SepalLengthCm'],kde=True,bins=20);
plt.show()
```



In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []: