

LOCHITA GUPTA

OASIS INFOBYTE - TASK 1

IRIS FLOWER CLASSIFICATION

Installed various libraries

In [1]: !pip install pandas

```
Requirement already satisfied: pandas in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (2.0.2)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from pandas) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from pandas) (2023.3)
Requirement already satisfied: tzdata>=2022.1 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from pandas) (2023.3)
Requirement already satisfied: numpy>=1.21.0 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from pandas) (1.23.5)
Requirement already satisfied: six>=1.5 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from python-dateutil>=2.8.2->pandas) (1.16.0)
```

```
In [2]: !pip install seaborn
```

Collecting seaborn

```
  Downloading seaborn-0.12.2-py3-none-any.whl (293 kB)
    0.0/293.3 kB ? eta -:-:--
  ----- 41.0/293.3 kB 653.6 kB/s eta 0:00:
01 ----- 112.6/293.3 kB 1.1 MB/s eta 0:00:
01 ----- 225.3/293.3 kB 1.7 MB/s eta 0:00:
01 ----- 293.3/293.3 kB 1.6 MB/s eta 0:00:
00
Requirement already satisfied: numpy!=1.24.0,>=1.17 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from seaborn) (1.23.5)
Requirement already satisfied: pandas>=0.25 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from seaborn) (2.0.2)
Requirement already satisfied: matplotlib!=3.6.1,>=3.1 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from seaborn) (3.7.1)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (1.0.7)
Requirement already satisfied: cycler>=0.10 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (4.39.4)
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (1.4.4)
Requirement already satisfied: packaging>=20.0 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (23.1)
Requirement already satisfied: pillow>=6.2.0 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (9.5.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (3.0.9)
Requirement already satisfied: python-dateutil>=2.7 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from matplotlib!=3.6.1,>=3.1->seaborn) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from pandas>=0.25->seaborn) (2023.3)
Requirement already satisfied: tzdata>=2022.1 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from pandas>=0.25->seaborn) (2023.3)
Requirement already satisfied: six>=1.5 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from python-dateutil>=2.7->matplotlib!=3.6.1,>=3.1->seaborn) (1.16.0)
Installing collected packages: seaborn
Successfully installed seaborn-0.12.2
```

```
In [3]: !pip install scikit-plot
```

Collecting scikit-plot

Downloading scikit_plot-0.3.7-py3-none-any.whl (33 kB)

Requirement already satisfied: matplotlib>=1.4.0 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from scikit-plot)

(3.7.1)

Requirement already satisfied: scikit-learn>=0.18 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from scikit-plot)

(1.2.2)

Requirement already satisfied: scipy>=0.9 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from scikit-plot) (1.10.1)

Requirement already satisfied: joblib>=0.10 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from scikit-plot) (1.2.0)

Requirement already satisfied: contourpy>=1.0.1 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from matplotlib>=1.4.0->scikit-plot) (1.0.7)

Requirement already satisfied: cyclor>=0.10 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from matplotlib>=1.4.0->scikit-plot) (0.11.0)

Requirement already satisfied: fonttools>=4.22.0 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from matplotlib>=1.4.0->scikit-plot) (4.39.4)

Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from matplotlib>=1.4.0->scikit-plot) (1.4.4)

Requirement already satisfied: numpy>=1.20 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from matplotlib>=1.4.0->scikit-plot) (1.23.5)

Requirement already satisfied: packaging>=20.0 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from matplotlib>=1.4.0->scikit-plot) (23.1)

Requirement already satisfied: pillow>=6.2.0 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from matplotlib>=1.4.0->scikit-plot) (9.5.0)

Requirement already satisfied: pyparsing>=2.3.1 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from matplotlib>=1.4.0->scikit-plot) (3.0.9)

Requirement already satisfied: python-dateutil>=2.7 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from matplotlib>=1.4.0->scikit-plot) (2.8.2)

Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from scikit-learn>=0.18->scikit-plot) (3.1.0)

Requirement already satisfied: six>=1.5 in c:\users\lochita gupta\appdata\local\programs\python\python311\lib\site-packages (from python-dateutil>=2.7->matplotlib>=1.4.0->scikit-plot) (1.16.0)

Installing collected packages: scikit-plot

Successfully installed scikit-plot-0.3.7

Import Libraries

```
In [5]: import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
import scikitplot as skplt
```

Importing Dataset

```
In [8]: df = pd.read_csv('Iris.csv')
```

```
In [9]: df.head() #top 5 values
```

```
Out[9]:
```

	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 [10]: df.tail() #last 5 values
```

```
Out[10]:
```

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
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

```
In [11]: df.shape #no. of rows and columns
```

```
Out[11]: (150, 6)
```

In [12]: `df.isnull()` *#returns a dataframe object where all the values are replaced with*

Out[12]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm	Species
0	False	False	False	False	False	False
1	False	False	False	False	False	False
2	False	False	False	False	False	False
3	False	False	False	False	False	False
4	False	False	False	False	False	False
...
145	False	False	False	False	False	False
146	False	False	False	False	False	False
147	False	False	False	False	False	False
148	False	False	False	False	False	False
149	False	False	False	False	False	False

150 rows × 6 columns

In [13]: `df.isnull().sum()` *#returns the number of missing values in data set.*

Out[13]:

Id	0
SepalLengthCm	0
SepalWidthCm	0
PetalLengthCm	0
PetalWidthCm	0
Species	0
dtype: int64	

In [14]: `df.describe()` *#used to view some basic statistical details like percentile, me*

Out[14]:

	Id	SepalLengthCm	SepalWidthCm	PetalLengthCm	PetalWidthCm
count	150.000000	150.000000	150.000000	150.000000	150.000000
mean	75.500000	5.843333	3.054000	3.758667	1.198667
std	43.445368	0.828066	0.433594	1.764420	0.763161
min	1.000000	4.300000	2.000000	1.000000	0.100000
25%	38.250000	5.100000	2.800000	1.600000	0.300000
50%	75.500000	5.800000	3.000000	4.350000	1.300000
75%	112.750000	6.400000	3.300000	5.100000	1.800000
max	150.000000	7.900000	4.400000	6.900000	2.500000

```
In [15]: df.columns
```

```
Out[15]: Index(['Id', 'SepalLengthCm', 'SepalWidthCm', 'PetalLengthCm', 'PetalWidthCm',  
              'Species'],  
              dtype='object')
```

```
In [16]: df.nunique() #return number of unique elements in the object
```

```
Out[16]: Id                150  
         SepalLengthCm      35  
         SepalWidthCm       23  
         PetalLengthCm      43  
         PetalWidthCm       22  
         Species           3  
         dtype: int64
```

```
In [18]: df.Species.nunique()
```

```
Out[18]: 3
```

```
In [19]: df.Species.value_counts()
```

```
Out[19]: Species  
         Iris-setosa      50  
         Iris-versicolor  50  
         Iris-virginica   50  
         Name: count, dtype: int64
```

```
In [20]: df.max()
```

```
Out[20]: Id                150  
         SepalLengthCm      7.9  
         SepalWidthCm       4.4  
         PetalLengthCm      6.9  
         PetalWidthCm       2.5  
         Species      Iris-virginica  
         dtype: object
```

```
In [21]: df.min()
```

```
Out[21]: Id                1  
         SepalLengthCm      4.3  
         SepalWidthCm       2.0  
         PetalLengthCm      1.0  
         PetalWidthCm       0.1  
         Species      Iris-setosa  
         dtype: object
```

```
In [22]: #drop the value of id form dataset  
df.drop('Id',axis=1,inplace=True)  
df.head()
```

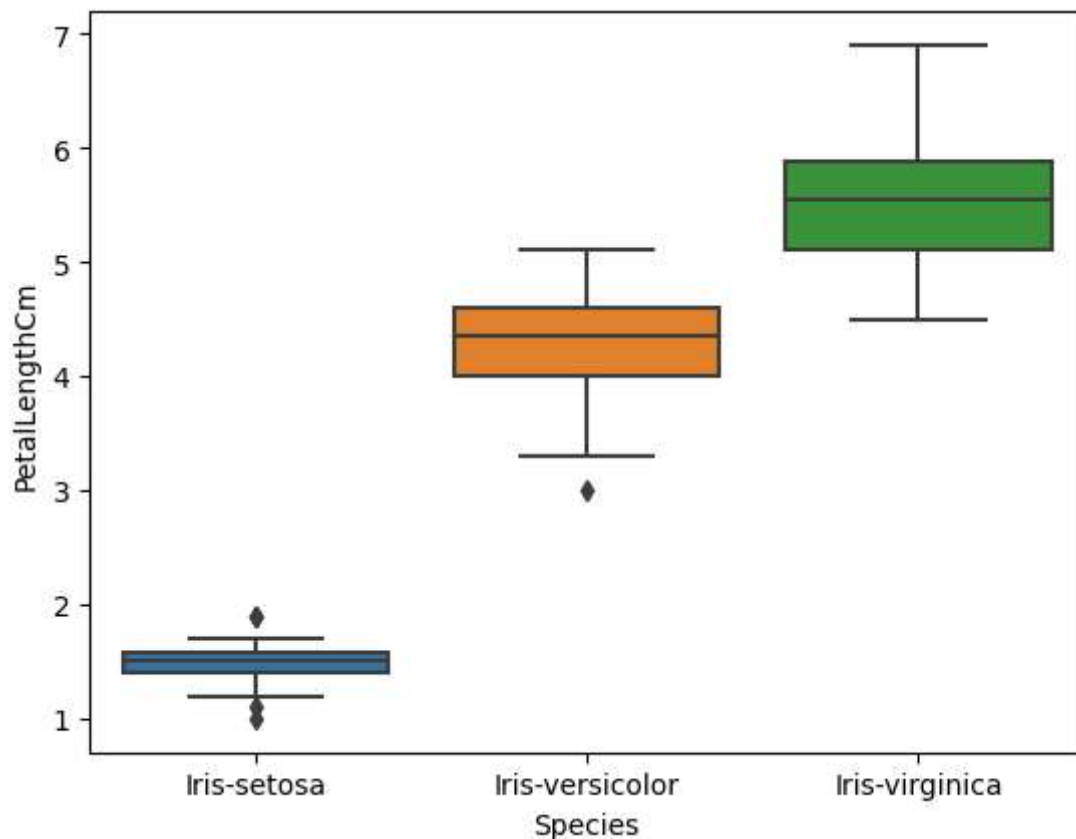
```
Out[22]:
```

	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

Visualization

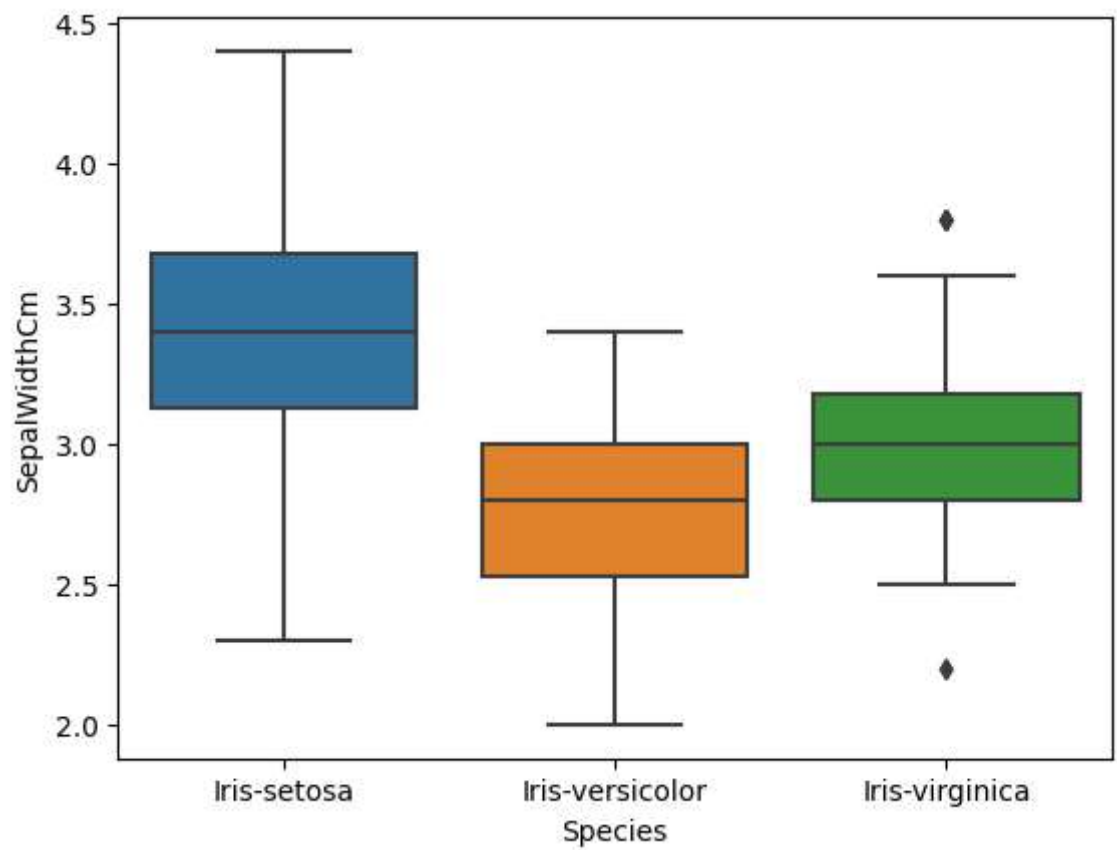
```
In [23]: sns.boxplot(x="Species", y='PetalLengthCm', data=df)  
plt.show
```

```
Out[23]: <function matplotlib.pyplot.show(close=None, block=None)>
```



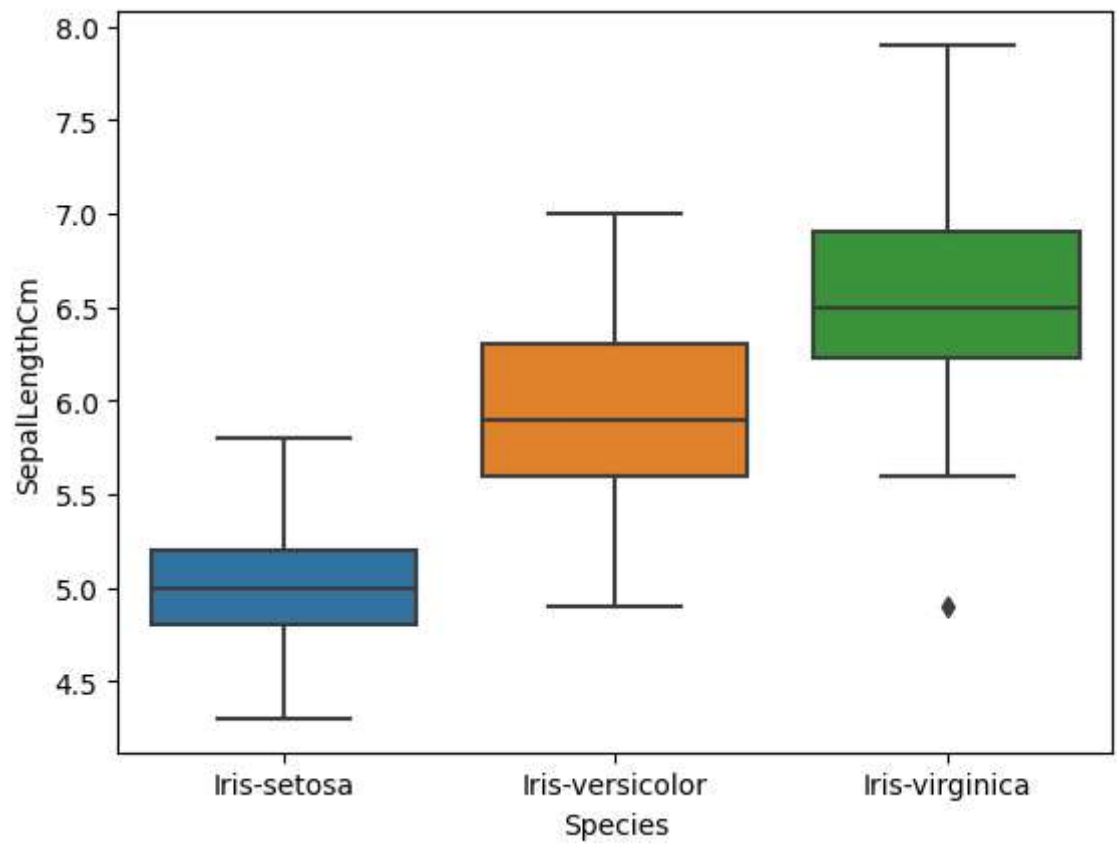
```
In [24]: sns.boxplot(x="Species", y="SepalWidthCm", data=df)
```

```
Out[24]: <Axes: xlabel='Species', ylabel='SepalWidthCm'>
```



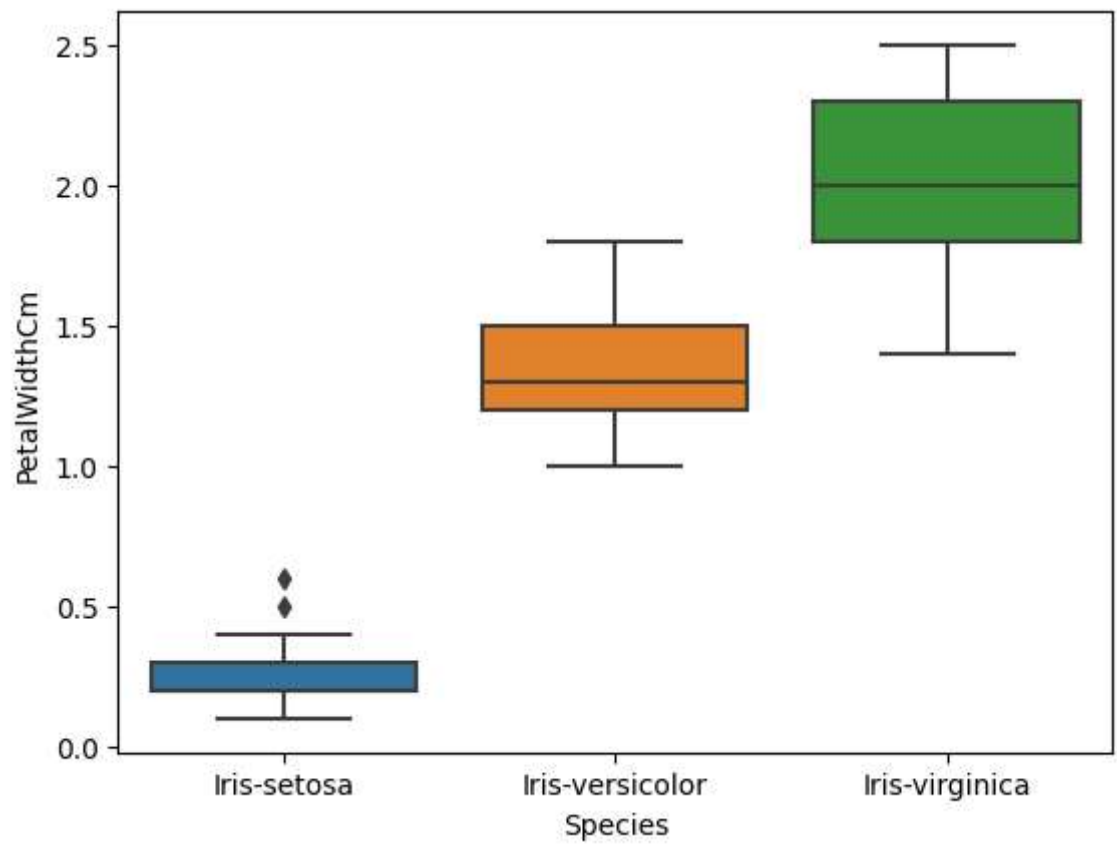

```
In [25]: sns.boxplot(x="Species", y="SepalLengthCm", data=df)
```

```
Out[25]: <Axes: xlabel='Species', ylabel='SepalLengthCm'>
```

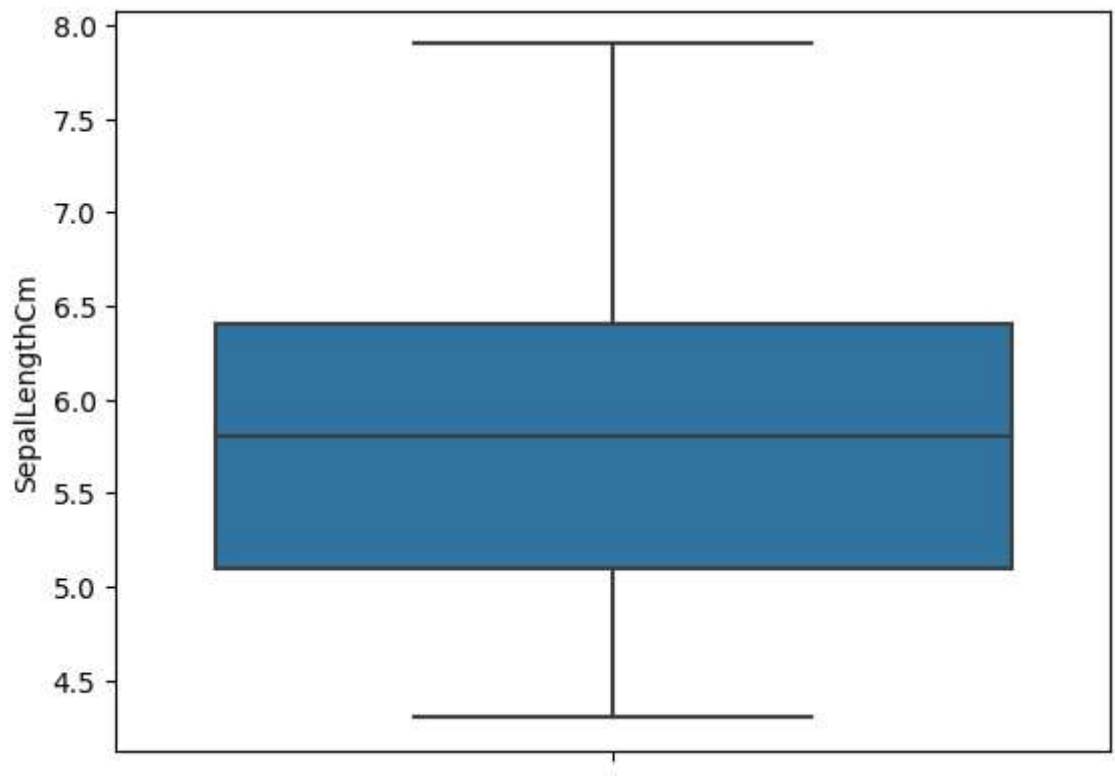


```
In [26]: sns.boxplot(x="Species", y="PetalWidthCm", data=df)
```

```
Out[26]: <Axes: xlabel='Species', ylabel='PetalWidthCm'>
```

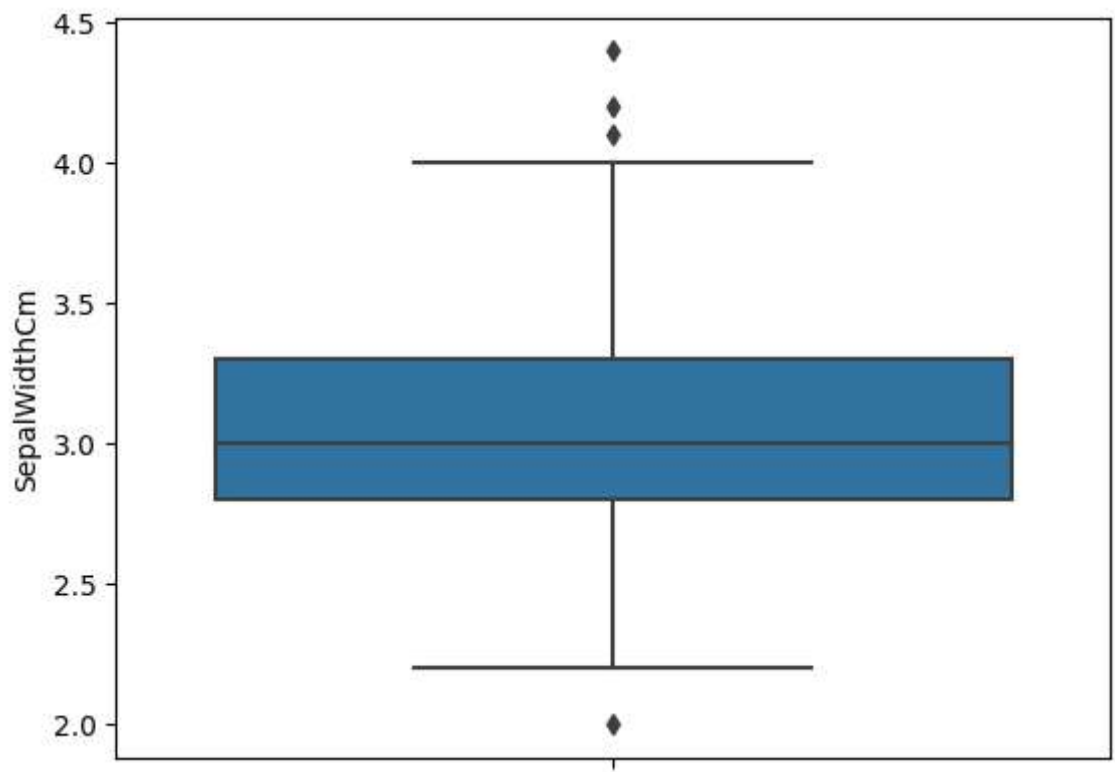


```
In [27]: sns.boxplot(y="SepalLengthCm" , data=df);  
plt.show()
```



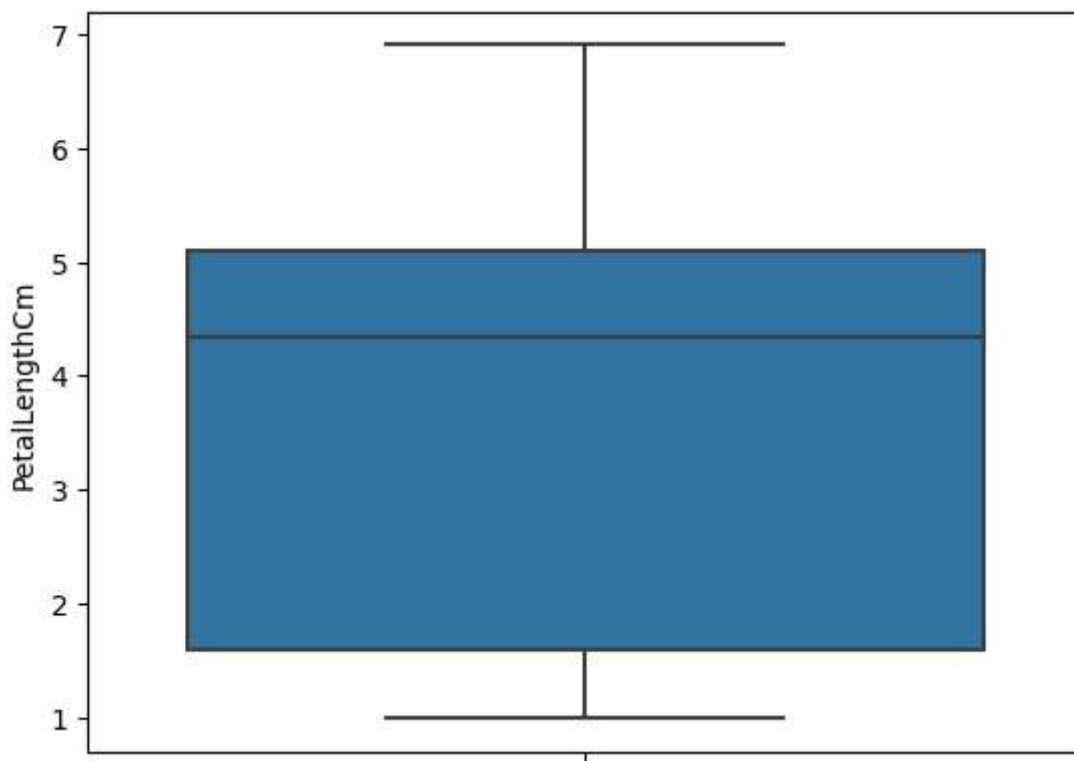
```
In [28]: sns.boxplot(y="SepalWidthCm", data=df);  
plt.show
```

```
Out[28]: <function matplotlib.pyplot.show(close=None, block=None)>
```

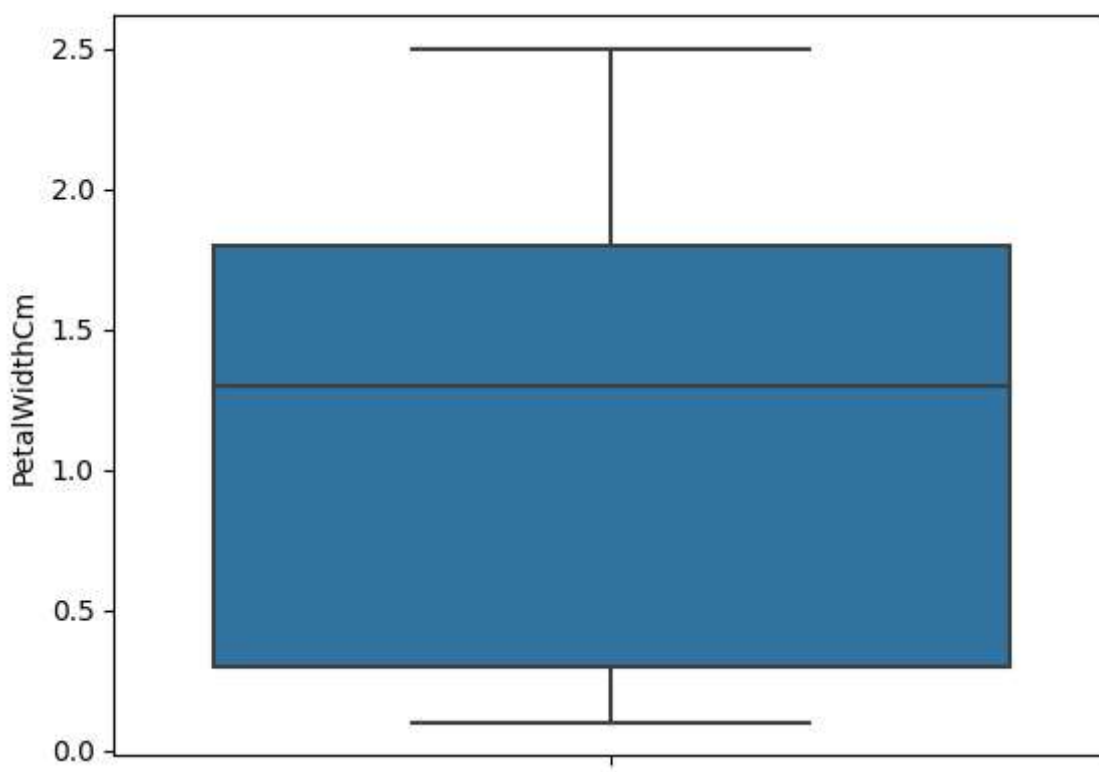


```
In [29]: sns.boxplot(y="PetalLengthCm", data=df);  
plt.show
```

```
Out[29]: <function matplotlib.pyplot.show(close=None, block=None)>
```

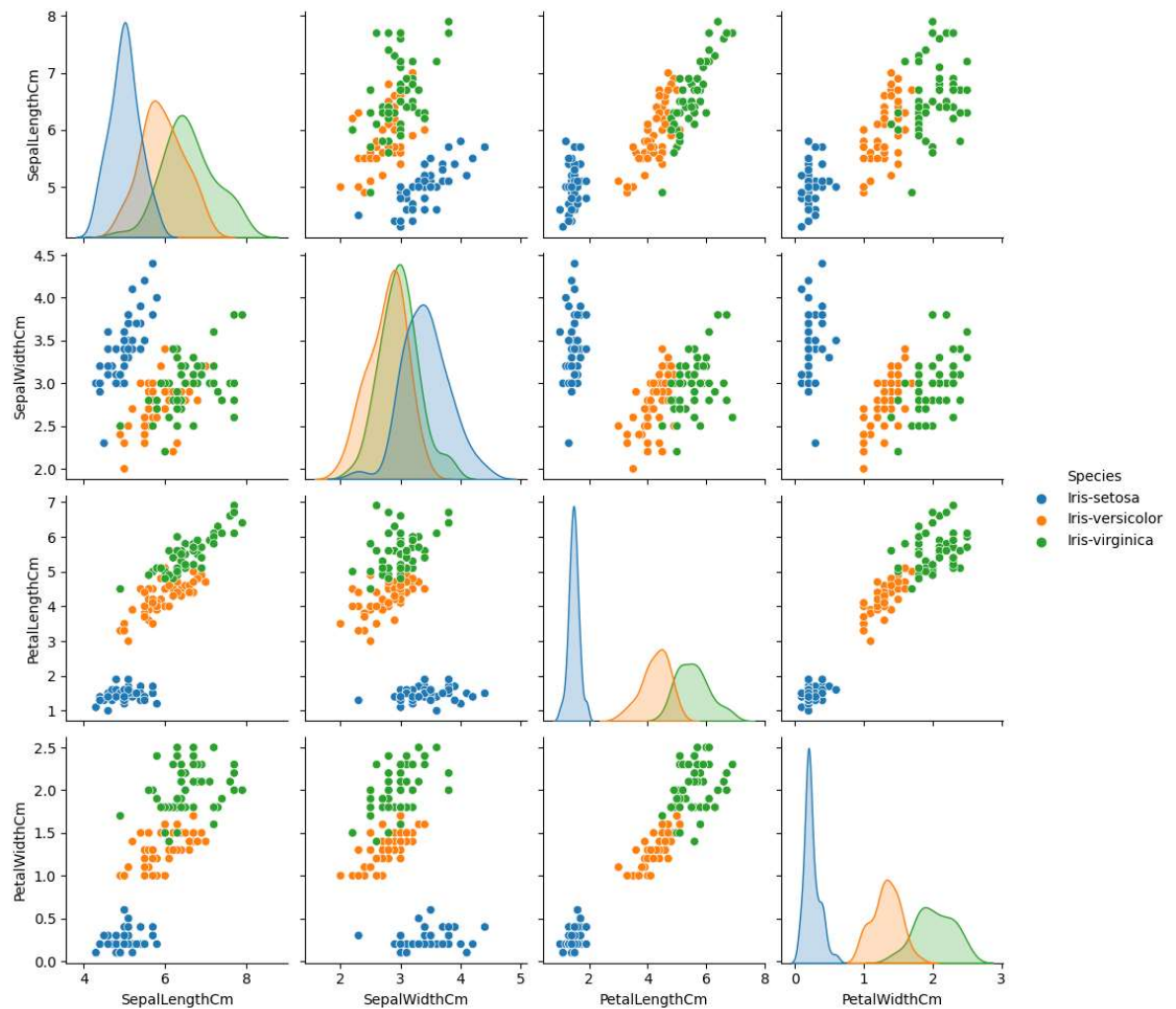


```
In [30]: sns.boxplot(y="PetalWidthCm", data=df);  
plt.show()
```



```
In [31]: sns.pairplot(df, hue = 'Species') # A pairplot plot a pairwise relationships in
```

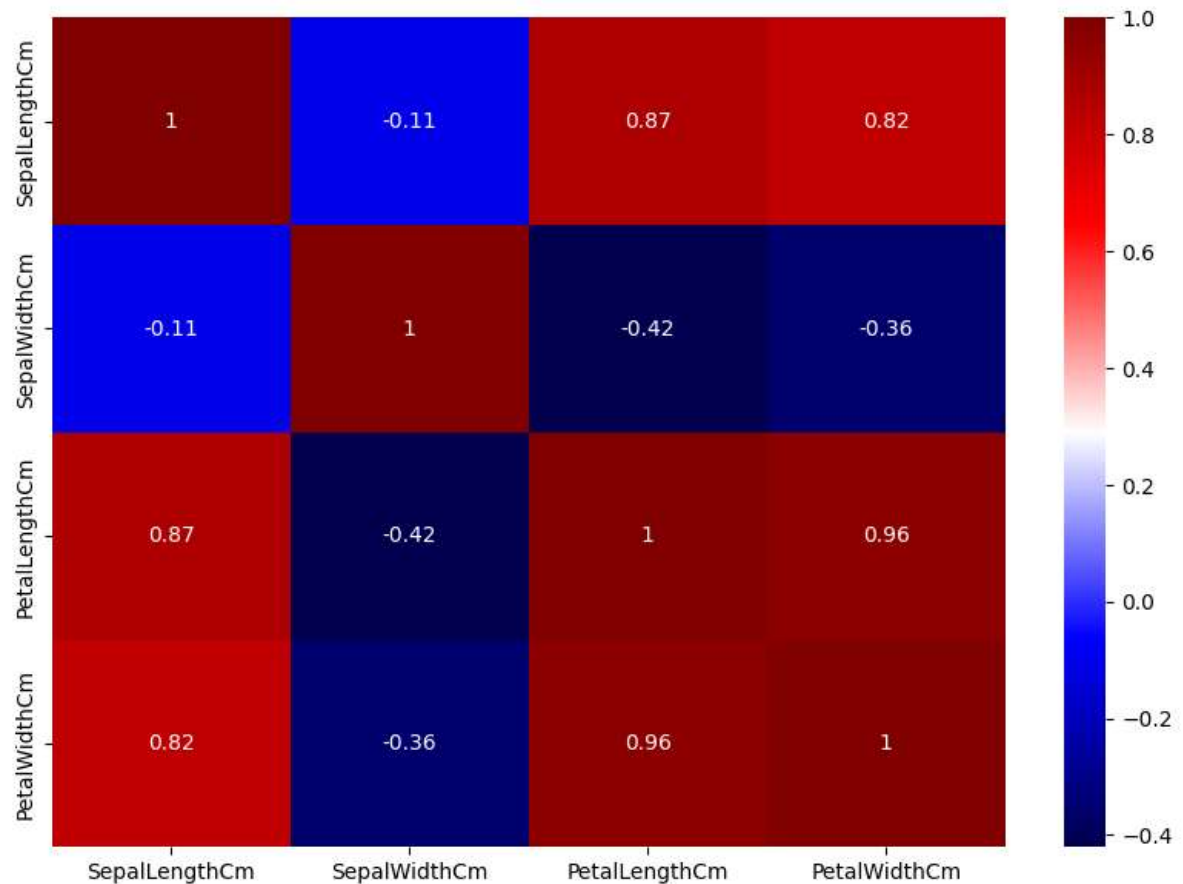
```
Out[31]: <seaborn.axisgrid.PairGrid at 0x2210029ac10>
```



Data PreProcessing or Correlation Matrix

```
In [37]: numeric_columns = df.select_dtypes(include=['number']) # Select only numeric
correlation_matrix = numeric_columns.corr()

plt.figure(figsize=(10, 7))
sns.heatmap(correlation_matrix, annot=True, cmap="seismic")
plt.show()
```



Label Encoder

```
In [41]: from sklearn.preprocessing import LabelEncoder
```

```
In [42]: le = LabelEncoder()
df['Species'] = le.fit_transform(df['Species'])
df.head()
```

```
Out[42]:
```

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

```
In [43]: X = df.drop(columns=['Species']) #drop column  
Y = df['Species']  
X[:5] #return list from beginning until index 5
```

```
Out[43]:
```

	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

```
In [44]: Y[:5]
```

```
Out[44]: 0    0  
1    0  
2    0  
3    0  
4    0  
Name: Species, dtype: int32
```

Splitting the Dataset into the Training set and Test set

```
In [46]: from sklearn.model_selection import train_test_split  
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.3, random_state=42)
```

Selecting the Models and Metrics (SML Models)

```
In [47]: from sklearn.linear_model import LogisticRegression  
from sklearn.neighbors import KNeighborsClassifier  
from sklearn.svm import SVC  
from sklearn.naive_bayes import GaussianNB  
from sklearn.tree import DecisionTreeClassifier  
from sklearn.ensemble import RandomForestClassifier  
from sklearn.metrics import accuracy_score
```



```
In [48]: lr = LogisticRegression()
knn = KNeighborsClassifier()
svm = SVC()
nb = GaussianNB()
dt = DecisionTreeClassifier()
rf = RandomForestClassifier()
```

Training and Evaluating the Models

```
models = [lr, knn, svm, nb, dt, rf] scores = []
```

```
for model in models: model.fit(X_train, Y_train) Y_pred = model.predict(X_test)
scores.append(accuracy_score(Y_test, Y_pred)) print("Accuracy of" + type(model).name + "is",
accuracy_score(Y_test, Y_pred))
```

```
In [51]: results = pd.DataFrame({
    'Models': ['Logistic Regression', 'K-Nearest Neighbors', 'Support Vector M
    'Random Forest'], 'Accuracy': scores})
results = results.sort_values(by='Accuracy', ascending=False)
print(results)
```

	Models	Accuracy
0	Logistic Regression	0.977778
1	K-Nearest Neighbors	0.977778
2	Support Vector Machine	0.977778
4	Decision Tree	0.955556
5	Random Forest	0.955556
3	Naive Bayes	0.933333

THANK YOU