Oasis Infobyte - OIBSIP - Data Science

Task 1 : Iris Flower Classification

Intern - Shravani Mahesuni

Problem Statement : Measurements of Iris flower is given according to their species, and the moto is to train a machine learning model that can learn from the measurements of the iris species and their classification .

In [16]:
```
#importing libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report,confusion_matrix
```

Loading dataset

In [2]:
```
#load dataset/read dataset
df = pd.read_csv('Iris.csv')
df.head()
```

Out[2]:

	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 [3]:
```
#calculating numeric & object series
df.describe()
```

Out[3]:

	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 [4]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 150 entries, 0 to 149
Data columns (total 6 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   Id             150 non-null    int64  
 1   SepalLengthCm  150 non-null    float64
 2   SepalWidthCm   150 non-null    float64
 3   PetalLengthCm  150 non-null    float64
 4   PetalWidthCm   150 non-null    float64
 5   Species        150 non-null    object 
dtypes: float64(4), int64(1), object(1)
memory usage: 7.2+ KB
```

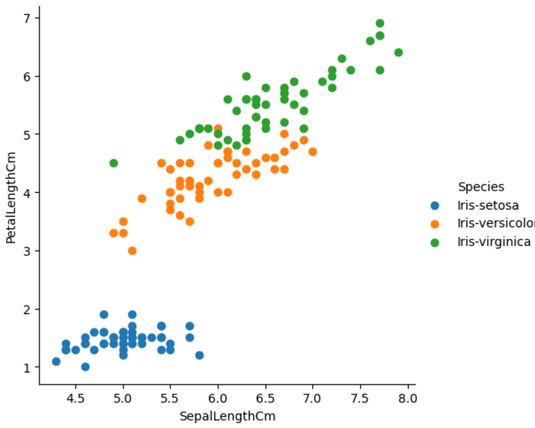
In [5]:
```
df["Species"].value_counts() #count sequences
```

Out[5]:
```
Iris-setosa        50
Iris-versicolor    50
Iris-virginica     50
Name: Species, dtype: int64
```

Splitting the data up by categories or different groups

In [6]:
```
sns.FacetGrid(df,hue="Species",height=5).map(plt.scatter,"SepalLengthCm","PetalLengthCm").add_legend() #data visualization
```

Out[6]:
```
<seaborn.axisgrid.FacetGrid at 0x24a1390f880>
```



In [7]:
```
x = df[["SepalLengthCm","SepalWidthCm","PetalLengthCm","PetalWidthCm"]].values
y = df[["Species"]].values
```

creating a logistic regression classifier object which can fit model using fit function

In [9]:
```
Model = LogisticRegression()
Model.fit(x,y)
```

```
C:\Users\DELL\anaconda3\lib\site-packages\sklearn\utils\validation.py:1143: DataConversionWarning: A column-vector y was passed when a 1d array was expected.
Please change the shape of y to (n_samples, ), for example using ravel().
  y = column_or_1d(y, warn=True)
```

Out[9]:
```
LogisticRegression()
```

In [10]:
```
Model.score(x,y).round(2) #accuracy
```

Out[10]:
```
0.97
```

In [11]:
```
#Prediction
Actual = y
predicted = Model.predict(x)
```

In [14]:
```
# classification report analysis
cr = classification_report(Actual,predicted)
print(cr)
```

```
                 precision    recall  f1-score   support

    Iris-setosa       1.00      1.00      1.00        50
Iris-versicolor       0.98      0.94      0.96        50
 Iris-virginica       0.94      0.98      0.96        50

       accuracy                           0.97       150
      macro avg       0.97      0.97      0.97       150
   weighted avg       0.97      0.97      0.97       150
```

In [17]:
```
cf = confusion_matrix(Actual,predicted)
```

In [18]:
```
print(cf) # printing confusion matrix
```

```
[[50  0  0]
 [ 0 47  3]
 [ 0  1 49]]
```

In [19]:
```
predicted = Model.predict([[5.1,3.5,1.4,0.2]])
predicted
```

Out[19]:
```
array(['Iris-setosa'], dtype=object)
```

Conclusion : Successfully completed the task of classifying iris flowers