

Let's Grow More Data Science Internship

Task :- 02

Task Name :- Iris Flowers Classification ML Project

Name :- Ishika Goel

IMPORTING LIBRARIES

```
In [1]:
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

LOADING DATASET

```
In [2]:
df = sns.load_dataset('iris')
df
```

Out[2]:

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	setosa
1	4.9	3.0	1.4	0.2	setosa
2	4.7	3.2	1.3	0.2	setosa
3	4.6	3.1	1.5	0.2	setosa
4	5.0	3.6	1.4	0.2	setosa
...
145	6.7	3.0	5.2	2.3	virginica
146	6.3	2.5	5.0	1.9	virginica
147	6.5	3.0	5.2	2.0	virginica
148	6.2	3.4	5.4	2.3	virginica
149	5.9	3.0	5.1	1.8	virginica

150 rows × 5 columns

CHECKING MISSING VALUES

```
In [3]:
df.isnull().sum()
```

Out[3]:

sepal_length	0
sepal_width	0
petal_length	0
petal_width	0
species	0
dtype:	int64

DATA SEPARATION

In [4]:

```
x = df.iloc[:,0:4]
x
```

Out[4]:

	sepal_length	sepal_width	petal_length	petal_width
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 [5]:

```
y =df.iloc[:,4]
y
```

Out[5]:

```
0      setosa
1      setosa
2      setosa
3      setosa
4      setosa
...
145  virginica
146  virginica
147  virginica
148  virginica
149  virginica
Name: species, Length: 150, dtype: object
```

TRAIN TEST SPLIT

In [6]:

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.2)
x_train
```

Out[6]:

	sepal_length	sepal_width	petal_length	petal_width
6	4.6	3.4	1.4	0.3
141	6.9	3.1	5.1	2.3
3	4.6	3.1	1.5	0.2
99	5.7	2.8	4.1	1.3
149	5.9	3.0	5.1	1.8
...
143	6.8	3.2	5.9	2.3
30	4.8	3.1	1.6	0.2
36	5.5	3.5	1.3	0.2
14	5.8	4.0	1.2	0.2
133	6.3	2.8	5.1	1.5

120 rows × 4 columns

KNN CLASSIFIER

In [7]:

```
from sklearn.neighbors import KNeighborsClassifier
from sklearn.metrics import accuracy_score
```

In [8]:

```
knn = KNeighborsClassifier(n_neighbors = 15)
knn.fit(x_train,y_train)
y_predict = knn.predict(x_test)
y_predict
```

C:\Users\Gyanender\anaconda3\lib\site-packages\sklearn\neighbors_classification.py:228: FutureWarning: Unlike other reduction functions (e.g. `skew`, `kurtosis`), the default behavior of `mode` typically preserves the axis it acts along. In SciPy 1.11.0, this behavior will change: the default value of `keepdims` will become False, the `axis` over which the statistic is taken will be eliminated, and the value None will no longer be accepted. Set `keepdims` to True or False to avoid this warning.

```
mode, _ = stats.mode(_y[neigh_ind, k], axis=1)
```

Out[8]:

```
array(['virginica', 'virginica', 'versicolor', 'virginica', 'virginica',
       'virginica', 'versicolor', 'setosa', 'setosa', 'virginica',
       'versicolor', 'setosa', 'setosa', 'setosa', 'setosa', 'virginica',
       'versicolor', 'versicolor', 'setosa', 'setosa', 'virginica',
       'virginica', 'virginica', 'virginica', 'setosa', 'virginica',
       'versicolor', 'virginica', 'versicolor', 'versicolor'],
      dtype=object)
```

ACCURACY

In [9]:

```
accuracy_score(y_test,y_predict)
```

Out[9]:

```
0.9333333333333333
```

CONFUSION MATRIX

In [10]:

```
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test,y_predict)
cm
```

Out[10]:

```
array([[ 9,  0,  0],
       [ 0,  8,  2],
       [ 0,  0, 11]], dtype=int64)
```

CLASSIFICATION REPORT

In [11]:

```
from sklearn.metrics import classification_report
print(classification_report(y_test,y_predict))
```

	precision	recall	f1-score	support
setosa	1.00	1.00	1.00	9
versicolor	1.00	0.80	0.89	10
virginica	0.85	1.00	0.92	11
accuracy			0.93	30
macro avg	0.95	0.93	0.94	30
weighted avg	0.94	0.93	0.93	30

CONCLUSION

I have performed Iris Flower Classification using kNN Classifier. Accuracy of kNN Classifier is 93.33%.

Thankyou!!

In []: