In [20]:
```python
import pandas as pd
import numpy as np
from sklearn.datasets import load_iris
from sklearn.linear_model import LogisticRegression
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler

from sklearn.metrics import classification_report
```

In [8]:
```python
df = pd.read_csv("C:\\Users\\kgnan\\OneDrive\\Desktop\\ANN\\ANN UNIT-1\\cod
print(df.head())
print(df['Species'].unique())
```

```
   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
['Iris-setosa' 'Iris-versicolor' 'Iris-virginica']
```

In [12]:
```python
#DATA PREPROCESSING
x = df.drop('Species', axis=1)
y = df['Species']

trainX, testX, trainY, testY = train_test_split(x, y, test_size = 0.2)
```

In [14]:
```python
sc=StandardScaler()

scaler = sc.fit(trainX)
trainX_scaled = scaler.transform(trainX)
testX_scaled = scaler.transform(testX)
```

In [22]:
```python
# Instantiate the model
cls = LogisticRegression(random_state=0,tol=0.01,penalty="l1",solver="saga"

# Train/Fit the model
cls.fit(trainX_scaled, trainY)
```

Out[22]:
```
LogisticRegression(multi_class='multinomial', penalty='l1', random_state=0,
                   solver='saga', tol=0.01)
```

In [23]:
```python
###Model Evaluation
y_pred = cls.predict(testX_scaled)
df_temp = pd.DataFrame({'Actual': testY, 'Predicted': y_pred})
df_temp.head()
```

Out[23]:

	Actual	Predicted
142	Iris-virginica	Iris-virginica
52	Iris-versicolor	Iris-versicolor
140	Iris-virginica	Iris-virginica
24	Iris-setosa	Iris-setosa
67	Iris-versicolor	Iris-versicolor

In [24]:
```python
classification_report(testY,y_pred )
```

Out[24]:
```
'                 precision    recall  f1-score   support\n\n    Iris-seto
sa       1.00      1.00      1.00        11\nIris-versicolor       0.89
1.00      0.94         8\n Iris-virginica       1.00      0.91      0.95
11\n\n       accuracy                           0.97        30\n      macr
o avg       0.96      0.97      0.96        30\n   weighted avg       0.97
0.97      0.97        30\n'
```

In [ ]:

In [ ]: