In [3]:
```python
import pandas as pd
```

In [4]:
```python
df = pd.read_csv('../Downloads/iris.csv')
```

In [5]:
```python
df.head()
```

Out[5]:

	x0	x1	x2	x3	x4	type
0	1	5.1	3.5	1.4	0.2	Iris-setosa
1	1	4.9	3.0	1.4	0.2	Iris-setosa
2	1	4.7	3.2	1.3	0.2	Iris-setosa
3	1	4.6	3.1	1.5	0.2	Iris-setosa
4	1	5.0	3.6	1.4	0.2	Iris-setosa

In [6]:
```python
df.shape
```

Out[6]: (150, 6)

In [7]:
```python
from sklearn.linear_model import LogisticRegression
```

In [8]:
```python
X = df[['x1','x2','x3','x4']]
y = df['type']
```

In [9]:
```python
model = LogisticRegression()
model.fit(X,y)
```

Out[9]: LogisticRegression()

In [10]:
```python
from sklearn.model_selection import train_test_split
```

In [11]:
```python
X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.20,random_state=42)
```

In [12]:
```python
X_train.shape
```

Out[12]: (120, 4)

In [13]:
```python
y_train.shape
```

Out[13]: (120,)

In [14]:
```python
from sklearn.preprocessing import MinMaxScaler
```

In [16]:
```python
Scaler = MinMaxScaler()
Scaler.fit(X_train)
X_train_Scaled = Scaler.transform(X_train)
```

In [17]:
```python
model = LogisticRegression()
model.fit(X_train_Scaled,y_train)
```

Out[17]: LogisticRegression()

In [21]:
```python
X_test_Scaled = Scaler.transform(X_test)
test_score = model.score(X_test_Scaled,y_test)
test_score
```

Out[21]: 0.9666666666666667

In [ ]: