

Assignment No. 5

Aim :

1. Logistic Regression
2. Differentiate between Linear and Logistic Regression
3. Sigmoid Function
4. Types of LogisticRegression
5. Confusion Matrix Evaluation Metrics

Code:

```
In [7]: 1 import numpy as np
        2 import pandas as pd
```

```
In [8]: 1 data = pd.read_csv("https://raw.githubusercontent.com/selva86/datasets/master/BostonHousing.csv")
        2 data.head()
```

```
Out[8]:
```

	crim	zn	indus	chas	nox	rm	age	dis	rad	tax	ptratio	b	lstat	medv
0	0.00632	18.0	2.31	0	0.538	6.575	65.2	4.0900	1	296	15.3	396.90	4.98	24.0
1	0.02731	0.0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	396.90	9.14	21.6
2	0.02729	0.0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7
3	0.03237	0.0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4
4	0.06905	0.0	2.18	0	0.458	7.147	54.2	6.0622	3	222	18.7	396.90	5.33	36.2

```
In [9]: 1 data.tail()
```

```
Out[9]:
```

	crim	zn	indus	chas	nox	rm	age	dis	rad	tax	ptratio	b	lstat	medv
501	0.06263	0.0	11.93	0	0.573	6.593	69.1	2.4786	1	273	21.0	391.99	9.67	22.4
502	0.04527	0.0	11.93	0	0.573	6.120	76.7	2.2875	1	273	21.0	396.90	9.08	20.6
503	0.06076	0.0	11.93	0	0.573	6.976	91.0	2.1675	1	273	21.0	396.90	5.64	23.9
504	0.10959	0.0	11.93	0	0.573	6.794	89.3	2.3889	1	273	21.0	393.45	6.48	22.0
505	0.04741	0.0	11.93	0	0.573	6.030	80.8	2.5050	1	273	21.0	396.90	7.88	11.9

```
In [10]: 1 print("The shape of the data is: ")
        2 data.shape
```

The shape of the data is:

```
Out[10]: (506, 14)
```

```
In [22]: 1 print(data.isnull().sum())
```

```
crim      0
zn         0
indus      0
chas       0
nox        0
rm         0
age        0
dis        0
rad        0
tax        0
ptratio    0
b          0
lstat      0
medv       0
dtype: int64
```

```
In [13]: 1 X = data.iloc[:,0:13]
          2 y = data.iloc[:, -1]
          3 X
          4 y
```

```
Out[13]: 0      24.0
          1      21.6
          2      34.7
          3      33.4
          4      36.2
          ...
        501     22.4
        502     20.6
        503     23.9
        504     22.0
        505     11.9
        Name: medv, Length: 506, dtype: float64
```

```
In [15]: 1 from sklearn.model_selection import train_test_split
          2 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.20, random_state=42)
```

```
In [16]: 1 print(X_train.shape)
          2 print(X_test.shape)
          3 print(y_train.shape)
          4 print(y_test.shape)
```

```
(404, 13)
(102, 13)
(404,)
(102,)
```

```
In [17]: 1 from sklearn.linear_model import LinearRegression
```

```
In [18]: 1 from sklearn.preprocessing import StandardScaler
          2 from sklearn.pipeline import make_pipeline
          3 model = make_pipeline(StandardScaler(with_mean=False), LinearRegression())
          4 model.fit(X_train, y_train)
```

```
Out[18]: Pipeline
          Pipeline(steps=[('standardscaler', StandardScaler(with_mean=False)),
                           ('linearregression', LinearRegression())])
          StandardScaler
          StandardScaler(with_mean=False)
          LinearRegression
          LinearRegression()
```

```
In [19]: 1 model.score(X_test, y_test)
```

```
Out[19]: 0.668759493535632
```

In [20]: 1 X_test

Out[20]:

	crim	zn	indus	chas	nox	rm	age	dis	rad	tax	ptratio	b	lstat
173	0.09178	0.0	4.05	0	0.510	6.416	84.1	2.6463	5	296	16.6	395.50	9.04
274	0.05644	40.0	6.41	1	0.447	6.758	32.9	4.0776	4	254	17.6	396.90	3.53
491	0.10574	0.0	27.74	0	0.609	5.983	98.8	1.8681	4	711	20.1	390.11	18.07
72	0.09164	0.0	10.81	0	0.413	6.065	7.8	5.2873	4	305	19.2	390.91	5.52
452	5.09017	0.0	18.10	0	0.713	6.297	91.8	2.3682	24	666	20.2	385.09	17.27
...
412	18.81100	0.0	18.10	0	0.597	4.628	100.0	1.5539	24	666	20.2	28.79	34.37
436	14.42080	0.0	18.10	0	0.740	6.461	93.3	2.0026	24	666	20.2	27.49	18.05
411	14.05070	0.0	18.10	0	0.597	6.657	100.0	1.5275	24	666	20.2	35.05	21.22
86	0.05188	0.0	4.49	0	0.449	6.015	45.1	4.4272	3	247	18.5	395.99	12.86
75	0.09512	0.0	12.83	0	0.437	6.286	45.0	4.5026	5	398	18.7	383.23	8.94

102 rows × 13 columns

In [21]: 1 y_test

Out[21]:

```
173    23.6
274    32.4
491    13.6
72     22.8
452    16.1
...
412    17.9
436     9.6
411    17.2
86     22.5
75     21.4
```

Name: medv, Length: 102, dtype: float64

Name : Mohan Kadambande

Roll No. : 13212 (TECO-b1)