**Ex.No. Build-in Dataset**

**Aim:**

To load build-in dataset through sklearn packages

**Description:**

Through Scikit (sklearn) packages, we can able to download different build-in dataset, like

Iris dataset

Diabetes dataset

Breast cancer dataset

**Program:**

from sklearn import datasets

import pandas as pd

iris = datasets.load\_iris()

print(iris)

# Prints the type/type object of iris

print("\ntype:\n",type(iris))

# prints the dictionary keys of iris data

print("\nkeys:\n",iris.keys())

# prints the type/type object of given attributes

print("\ntype of data and target:\n",type(iris.data), type(iris.target))

# prints the no of rows and columns in the dataset

print("\ndata shape:\n",iris.data.shape)

# prints the target set of the data

print("\ntarget names:\n",iris.target\_names)

# Load iris training dataset

X = iris.data

# Load iris target set

Y = iris.target

#print("\ntarget:\n",Y)

# Convert datasets' type into dataframe

df = pd.DataFrame(X, columns=iris.feature\_names)

# Print the first five tuples of dataframe.

print("\nIris dataframe:\n",df.head())

print("--------------------------------------------------------------\n")

diabetes = datasets.load\_diabetes()

print("\ndiabetes dataset:\n",diabetes);

X = diabetes.data

# Load iris target set

Y = diabetes.target

#print("\ntarget:\n",Y)

# Convert datasets' type into dataframe

df = pd.DataFrame(X, columns=diabetes.feature\_names)

# Print the first five tuples of dataframe.

print("\nDiabetes dataframe:\n",df.head())

print("--------------------------------------------------------------\n")

data = datasets.load\_breast\_cancer()

label\_names = data['target\_names']

labels = data['target']

feature\_names = data['feature\_names']

features = data['data']

print("Breast Cancer data:\n", data);

print("\nLabel names:\n", label\_names)

print("\nLabels:\n", labels)

print("\nFeature names:\n", feature\_names)

print("\nFeatures:\n", features)

**Output**

type:

<class 'sklearn.utils.\_bunch.Bunch'>

keys:

dict\_keys(['data', 'target', 'frame', 'target\_names', 'DESCR', 'feature\_names', 'filename', 'data\_module'])

type of data and target:

<class 'numpy.ndarray'> <class 'numpy.ndarray'>

data shape:

(150, 4)

target names:

['setosa' 'versicolor' 'virginica']

Iris dataframe:

sepal length (cm) sepal width (cm) petal length (cm) petal width (cm)

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

--------------------------------------------------------------

diabetes dataset:

Diabetes dataframe:

age sex bmi ... s4 s5 s6

0 0.038076 0.050680 0.061696 ... -0.002592 0.019907 -0.017646

1 -0.001882 -0.044642 -0.051474 ... -0.039493 -0.068332 -0.092204

2 0.085299 0.050680 0.044451 ... -0.002592 0.002861 -0.025930

3 -0.089063 -0.044642 -0.011595 ... 0.034309 0.022688 -0.009362

4 0.005383 -0.044642 -0.036385 ... -0.002592 -0.031988 -0.046641

[5 rows x 10 columns]

--------------------------------------------------------------

Breast Cancer data:

Label names:

['malignant' 'benign']

Labels:

[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

1 0 0 0 0 0 0 0 0 1 0 1 1 1 1 1 0 0 1 0 0 1 1 1 1 0 1 0 0 1 1 1 1 0 1 0 0

1 0 1 0 0 1 1 1 0 0 1 0 0 0 1 1 1 0 1 1 0 0 1 1 1 0 0 1 1 1 1 0 1 1 0 1 1

1 1 1 1 1 1 0 0 0 1 0 0 1 1 1 0 0 1 0 1 0 0 1 0 0 1 1 0 1 1 0 1 1 1 1 0 1

1 1 1 1 1 1 1 1 0 1 1 1 1 0 0 1 0 1 1 0 0 1 1 0 0 1 1 1 1 0 1 1 0 0 0 1 0

1 0 1 1 1 0 1 1 0 0 1 0 0 0 0 1 0 0 0 1 0 1 0 1 1 0 1 0 0 0 0 1 1 0 0 1 1

1 0 1 1 1 1 1 0 0 1 1 0 1 1 0 0 1 0 1 1 1 1 0 1 1 1 1 1 0 1 0 0 0 0 0 0 0

0 0 0 0 0 0 0 1 1 1 1 1 1 0 1 0 1 1 0 1 1 0 1 0 0 1 1 1 1 1 1 1 1 1 1 1 1

1 0 1 1 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 0 1 0 1 1 1 1 0 0 0 1 1

1 1 0 1 0 1 0 1 1 1 0 1 1 1 1 1 1 1 0 0 0 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0 0

0 1 0 0 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 0 1 1 0 0 1 1 1 1 1 1 0 1 1 1 1 1 1

1 0 1 1 1 1 1 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 0 1 0 0 1 0 1 1 1 1 1 0 1 1

0 1 0 1 1 0 1 0 1 1 1 1 1 1 1 1 0 0 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 0 1

1 1 1 1 1 1 0 1 0 1 1 0 1 1 1 1 1 0 0 1 0 1 0 1 1 1 1 1 0 1 1 0 1 0 1 0 0

1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 0 1 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 0 0 0 0 0 0 1]

Feature names:

['mean radius' 'mean texture' 'mean perimeter' 'mean area'

'mean smoothness' 'mean compactness' 'mean concavity'

'mean concave points' 'mean symmetry' 'mean fractal dimension'

'radius error' 'texture error' 'perimeter error' 'area error'

'smoothness error' 'compactness error' 'concavity error'

'concave points error' 'symmetry error' 'fractal dimension error'

'worst radius' 'worst texture' 'worst perimeter' 'worst area'

'worst smoothness' 'worst compactness' 'worst concavity'

'worst concave points' 'worst symmetry' 'worst fractal dimension']

Features:

[[1.799e+01 1.038e+01 1.228e+02 ... 2.654e-01 4.601e-01 1.189e-01]

[2.057e+01 1.777e+01 1.329e+02 ... 1.860e-01 2.750e-01 8.902e-02]

[1.969e+01 2.125e+01 1.300e+02 ... 2.430e-01 3.613e-01 8.758e-02]

...

[1.660e+01 2.808e+01 1.083e+02 ... 1.418e-01 2.218e-01 7.820e-02]

[2.060e+01 2.933e+01 1.401e+02 ... 2.650e-01 4.087e-01 1.240e-01]

[7.760e+00 2.454e+01 4.792e+01 ... 0.000e+00 2.871e-01 7.039e-02]]