

p4

March 28, 2024

```
[4]: import pandas as pd

df = pd.read_csv("https://raw.githubusercontent.com/selva86/datasets/master/
↳BostonHousing.csv")
df
```

```
[4]:
```

	crim	zn	indus	chas	nox	rm	age	dis	rad	tax	\
0	0.00632	18.0	2.31	0	0.538	6.575	65.2	4.0900	1	296	
1	0.02731	0.0	7.07	0	0.469	6.421	78.9	4.9671	2	242	
2	0.02729	0.0	7.07	0	0.469	7.185	61.1	4.9671	2	242	
3	0.03237	0.0	2.18	0	0.458	6.998	45.8	6.0622	3	222	
4	0.06905	0.0	2.18	0	0.458	7.147	54.2	6.0622	3	222	
..	...	...	...	...	...	...	...	...	...	...	
501	0.06263	0.0	11.93	0	0.573	6.593	69.1	2.4786	1	273	
502	0.04527	0.0	11.93	0	0.573	6.120	76.7	2.2875	1	273	
503	0.06076	0.0	11.93	0	0.573	6.976	91.0	2.1675	1	273	
504	0.10959	0.0	11.93	0	0.573	6.794	89.3	2.3889	1	273	
505	0.04741	0.0	11.93	0	0.573	6.030	80.8	2.5050	1	273	

  

	ptratio	b	lstat	medv
0	15.3	396.90	4.98	24.0
1	17.8	396.90	9.14	21.6
2	17.8	392.83	4.03	34.7
3	18.7	394.63	2.94	33.4
4	18.7	396.90	5.33	36.2
..	...	...	...	...
501	21.0	391.99	9.67	22.4
502	21.0	396.90	9.08	20.6
503	21.0	396.90	5.64	23.9
504	21.0	393.45	6.48	22.0
505	21.0	396.90	7.88	11.9

[506 rows x 14 columns]

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

```
[5]:      crim    zn  indus  chas    nox    rm   age    dis  rad  tax  ptratio  \
0  0.00632  18.0   2.31    0  0.538  6.575  65.2  4.0900   1  296    15.3
1  0.02731   0.0   7.07    0  0.469  6.421  78.9  4.9671   2  242    17.8
2  0.02729   0.0   7.07    0  0.469  7.185  61.1  4.9671   2  242    17.8
3  0.03237   0.0   2.18    0  0.458  6.998  45.8  6.0622   3  222    18.7
4  0.06905   0.0   2.18    0  0.458  7.147  54.2  6.0622   3  222    18.7

      b  lstat  medv
0  396.90   4.98  24.0
1  396.90   9.14  21.6
2  392.83   4.03  34.7
3  394.63   2.94  33.4
4  396.90   5.33  36.2
```

```
[6]: x = df.drop("medv", axis = 1)
```

```
#output data
y = df["medv"]
```

```
[7]: print("Shape of medv: ", x.shape)
```

```
Shape of medv: (506, 13)
```

```
[9]: from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, random_state = 0,
↳ test_size = 0.25)
x_train
```

```
[9]:      crim    zn  indus  chas    nox    rm   age    dis  rad  tax  \
245  0.19133  22.0   5.86    0  0.431  5.605  70.2  7.9549   7  330
59   0.10328  25.0   5.13    0  0.453  5.927  47.2  6.9320   8  284
276  0.10469  40.0   6.41    1  0.447  7.267  49.0  4.7872   4  254
395  8.71675   0.0  18.10    0  0.693  6.471  98.8  1.7257  24  666
416 10.83420   0.0  18.10    0  0.679  6.782  90.8  1.8195  24  666
..      ...    ...    ...    ...    ...    ...    ...    ...
323  0.28392   0.0   7.38    0  0.493  5.708  74.3  4.7211   5  287
192  0.08664  45.0   3.44    0  0.437  7.178  26.3  6.4798   5  398
117  0.15098   0.0  10.01    0  0.547  6.021  82.6  2.7474   6  432
47   0.22927   0.0   6.91    0  0.448  6.030  85.5  5.6894   3  233
172  0.13914   0.0   4.05    0  0.510  5.572  88.5  2.5961   5  296

      ptratio    b  lstat
245    19.1  389.13  18.46
59     19.7  396.90   9.22
276    17.6  389.25   6.05
395    20.2  391.98  17.12
416    20.2  21.57  25.79
```

```

..      ...      ...      ...
323      19.6      391.13      11.74
192      15.2      390.49      2.87
117      17.8      394.51      10.30
47       17.9      392.74      18.80
172      16.6      396.90      14.69

```

[379 rows x 13 columns]

```
[10]: x_train.head()
```

```

[10]:      crim      zn  indus  chas      nox      rm      age      dis  rad  tax  \
245    0.19133  22.0   5.86     0  0.431  5.605  70.2  7.9549   7  330
59     0.10328  25.0   5.13     0  0.453  5.927  47.2  6.9320   8  284
276    0.10469  40.0   6.41     1  0.447  7.267  49.0  4.7872   4  254
395    8.71675   0.0  18.10     0  0.693  6.471  98.8  1.7257  24  666
416   10.83420   0.0  18.10     0  0.679  6.782  90.8  1.8195  24  666

      ptratio      b  lstat
245      19.1  389.13  18.46
59      19.7  396.90   9.22
276      17.6  389.25   6.05
395      20.2  391.98  17.12
416      20.2   21.57  25.79

```

```

[12]: print("shape of train data: ", x_train.shape)
      print("Shape of the test data: ", x_test.shape)

```

```

shape of train data: (379, 13)
Shape of the test data: (127, 13)

```

```

[17]: from sklearn.linear_model import LinearRegression
      reg = LinearRegression()
      reg.fit(x_train, y_train)
      print("Regression coeff: \n", reg.coef_)
      print("\n Regression intercept: \n", reg.intercept_)

```

```

Regression coeff:
[-1.17735289e-01  4.40174969e-02 -5.76814314e-03  2.39341594e+00
-1.55894211e+01  3.76896770e+00 -7.03517828e-03 -1.43495641e+00
 2.40081086e-01 -1.12972810e-02 -9.85546732e-01  8.44443453e-03
-4.99116797e-01]

```

```

Regression intercept:
36.93325545711923

```

```
[18]: # predictions
y_pred = reg.predict(x_test)
print("Shape of y_pred: ", y_pred.shape)
```

Shape of y\_pred: (127,)

```
[20]: result = pd.DataFrame({"Actual": y_test, "Produced": y_pred})
print("Comparing Results: \n\n", result)
```

Comparing Results:

	Actual	Produced
329	22.6	24.952333
371	50.0	23.616997
219	23.0	29.205886
403	8.3	11.960705
78	21.2	21.333620
..	...	...
49	19.4	17.538048
498	21.2	21.502223
309	20.3	23.632813
124	18.8	20.282598
306	33.4	35.179734

[127 rows x 2 columns]

```
[24]: residual_err = abs(y_test - y_pred)
print("Residual Error \n\n", residual_err)
print("\n \n Mean Absolute Error: ", sum(residual_err) / len(residual_err))

from sklearn.metrics import mean_absolute_error
meanAbs = mean_absolute_error(y_pred, y_test)
print("\n\nMean Absolute Error from SkLearn: ", meanAbs)
```

Residual Error

329	2.352333
371	26.383003
219	6.205886
403	3.660705
78	0.133620
..	...
49	1.861952
498	0.302223
309	3.332813
124	1.482598
306	1.779734

Name: medv, Length: 127, dtype: float64

Mean Absolute Error: 3.66833014813572

Mean Absolute Error from SkLearn: 3.668330148135719

```
[26]: from sklearn.metrics import mean_absolute_percentage_error
print("Mean absolute percentage error: ",
      ↪mean_absolute_percentage_error(y_test, y_pred))
```

Mean absolute percentage error: 0.1754993780061571

```
[28]: from sklearn.metrics import r2_score
print("R2 Score from SK_learn: ", r2_score(y_test, y_pred))
```

R2 Score from SK\_learn: 0.6354638433202128

```
[ ]:
```