

# 5\_multiple\_linear\_regression\_with\_sklearn

May 24, 2019

## 0.0.1 The Boston Housing Dataset

<https://archive.ics.uci.edu/ml/datasets/Housing>

```
In [ ]: from sklearn.datasets import load_boston
import matplotlib.pyplot as plt
import numpy as np
```

```
In [ ]: boston = load_boston()
boston["data"]
```

```
In [ ]: x_data = boston.data
y_data = boston.target.reshape(boston.target.size,1)
```

```
In [ ]: y_data.shape
```

```
In [ ]: from sklearn import preprocessing
```

```
minmax_scale = preprocessing.MinMaxScaler(feature_range=(0,5)).fit(x_data)
# standard_scale = preprocessing.StandardScaler().fit(x_data)
x_scaled_data = minmax_scale.transform(x_data)

x_scaled_data[:3]
```

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

```
X_train, X_test, y_train, y_test = train_test_split(x_scaled_data, y_data, test_size=0
```

```
In [ ]: X_train.shape, X_test.shape, y_train.shape, y_test.shape
```

```
In [ ]: from sklearn import linear_model
```

```
regr = linear_model.LinearRegression(fit_intercept=True,
                                     normalize=False,
                                     copy_X=True,
                                     n_jobs=8)

regr.fit(X_train, y_train)
regr
```

```

In [ ]: regr.coef_ , regr.intercept_

In [ ]: ## The coefficients
        print('Coefficients: ', regr.coef_)
        print('intercept: ', regr.intercept_)

In [ ]: regr.predict(x_data[:5])

In [ ]: x_data[:5].dot(regr.coef_.T) + regr.intercept_

In [ ]: from sklearn.metrics import r2_score
        from sklearn.metrics import mean_absolute_error
        from sklearn.metrics import mean_squared_error

In [ ]: y_true = y_test
        y_hat = regr.predict(X_test)

        r2_score(y_true, y_hat), mean_absolute_error(y_true, y_hat), mean_squared_error(y_true

In [ ]: y_true = y_train
        y_hat = regr.predict(X_train)

        r2_score(y_true, y_hat), mean_absolute_error(y_true, y_hat), mean_squared_error(y_true

In [ ]: regr.score(X_test, y_test)

In [ ]:

```