Boston Housing Prices

PolyFeatures
2 Features LSTAT y RM: best correlated
lin_reg baseline
2 improves
3 improves
4 improves

LinReg

```
R2 score train 0.6363353379616082
MAE train 3.798063422689011
MSE train 27.98794118547948

R2 score test 0.6041142735229119
MAE test 4.582645892343419
MSE test 41.40202341152383
```

CrossValidation on the full dataset MAE

```
[-2.74282364 -2.66486168 -2.89248316 -5.15153202 -3.9402028  -
4.64564316
 -3.82923301 -8.69643448 -6.53638215 -2.94554412]
-MAE: -4.404514022690067
std: 1.8528516111956599
```

Pol2

```
R2 score train 0.7526744600340032
MAE train 3.0194990832234727
MSE train 19.034383564891193

R2 score test 0.7485007375000425
MAE test 3.5422408760579787
MSE test 26.301979732039786
```

CrossValidation on the full dataset MAE

```
[-2.18784704 -2.15758692 -2.38537873 -3.77741438 -3.78544616 -3.44238
 -3.6404846  -7.56599142 -3.93790228 -2.76106402]
-MAE: -3.5641495549969457
std: 1.4891814017406946
```

Pol3

```
R2 score test 0.7504651949495681
MAE test 3.5420021790793244
MSE test 26.09653531240899
R2 score train 0.7688165765778632
MAE train 2.908202831717163
MSE train 17.79207256907875
```

Pol4

```
R2 score test 0.6688023261862086
MAE test 3.8712993251952605
MSE test 34.636898801842555
R2 score train 0.8115227871427741
MAE train 2.700230995250689
MSE train 14.50536634129782
```

`cross_val_score` is a utility in scikit-learn that allows you to evaluate a model's performance using cross-validation. This function splits the dataset into multiple "folds," trains the model on a subset of these folds, and tests it on the remaining fold(s). The goal of cross-validation is to provide a more reliable estimate of the model's performance by testing it on different subsets of the data, reducing the risk of overfitting to a single train/test split.

The `neg_mean_absolute_error` is a scoring function commonly used in machine learning for evaluating models, particularly in regression tasks. It represents the negative of the Mean Absolute Error (MAE), where the MAE measures the average magnitude of errors in a set of predictions, without considering their direction (i.e., it calculates the absolute difference between predicted and actual values).

The reason for the "negative" sign is that many scikit-learn functions, like `cross_val_score`, by default expect higher scores to indicate better performance. Since a lower MAE indicates better performance (as it means smaller errors), using the negative sign allows scikit-learn to interpret larger values as better scores. This transformation enables consistent comparisons when using functions that minimize or maximize scores.