Comments on projects

- First, the necessary libraries are imported.
- The dataset was downloaded from Kaggle and studied.
- The data were read and exploratory analysis was performed.
- Then, the data was preprocessed and the data set was divided into X_train, X_test, y_train, y_test. The data set was normalized and scaled.
- Three types of models were determined and train operation was performed with these models.
- Linear regression and lasso regression scores were found to be very close to each other and continued with these two models.

- Random search cv is used instead of grid search cv for hyper-parameter optimization. Because it is desired to keep the parameters wide and optimize among more possibilities. However, hyper-parameter optimization has only been tested in Lasso regression. Linear regression has not been optimized because there are no parameters that can be used in Linear regression.
- But the best model of this training is again provided with linear regression.

- The model evaluation part is divided into three;
    - In the first, the models that emerged after the above training were evaluated.
    - In the second, the data was scaled with the standard scaler and put into the training phase again.
    - Finally, the data was scaled with polynomial and re-trained, and the alpha value obtained in parameter optimization was used while retraining the data.

Result:

"Linear Regression Predictions: [2081.40696945 12095.34272292 10543.98701922 2596.691265
  8389.43547912], Actual Values: 559 1646.4297
1087 11353.2276
1020 8798.5930
460 10381.4787
802 2103.0800
Name: charges, dtype: float64
Lasso Test Set RMSE: 6173.227036966229
Lasso Predictions: [2081.45126417 12095.33413749 10543.9996001 2596.71838527
  8389.44529213], Actual Values: 559 1646.4297
1087 11353.2276
1020 8798.5930
460 10381.4787
802 2103.0800
Name: charges, dtype: float64"

In the second training process, it is clear that there is an improvement in the scores and RMSE values.

There are train models optimized using polynomial below.

Result:
"Linear Regression Test Set RMSE: 4878.467344763226

Linear Regression Predictions: [6520.4552638 10414.50621999 3469.6320337 12348.53753618
 1442.86875623], Actual Values: 559 1646.4297
1087 11353.2276
1020 8798.5930
460 10381.4787
802 2103.0800
Name: charges, dtype: float64
Lasso Test Set RMSE: 4878.458569464854
Lasso Predictions: [6520.3145581 10414.47344181 3469.68397849 12348.67297522
 1442.44447573], Actual Values: 559 1646.4297
1087 11353.2276
1020 8798.5930
460 10381.4787
802 2103.0800
Name: charges, dtype: float64
C:\Users\ibrah\AppData\Roaming\Python\Python310\site-packages\sklearn\linear_model_coordinate_descent.py:648: ConvergenceWarning: Objective did not converge. You might want to increase the number of iterations, check the scale of the features or consider increasing regularisation. Duality gap: 1.026e+10, tolerance: 1.394e+07
 model = cd_fast.enet_coordinate_descent(
The best results were obtained from the training models applied after optimizing with polynomial."

Conclusion:
There are multiple regressions, multiple optimizations and parameters. We tried them one by one to find the most accurate method, and we got the best results in linear and lasso regression's alpha 0.01 parameter optimized with polynomial regression. Accurancy values reached 83% in both.