HW 5: Regression Report

Calicia Perea

In this document, we will implement several linear regression functions provided by the scikit-learn library on the California housing dataset which can be loaded using fetch_california_housing from sklearn.datasets.

The following regression functions will be used:

1. LinearRegression
2. RANSACRegressor
3. Ridge
4. Lasso
5. ElasticNet

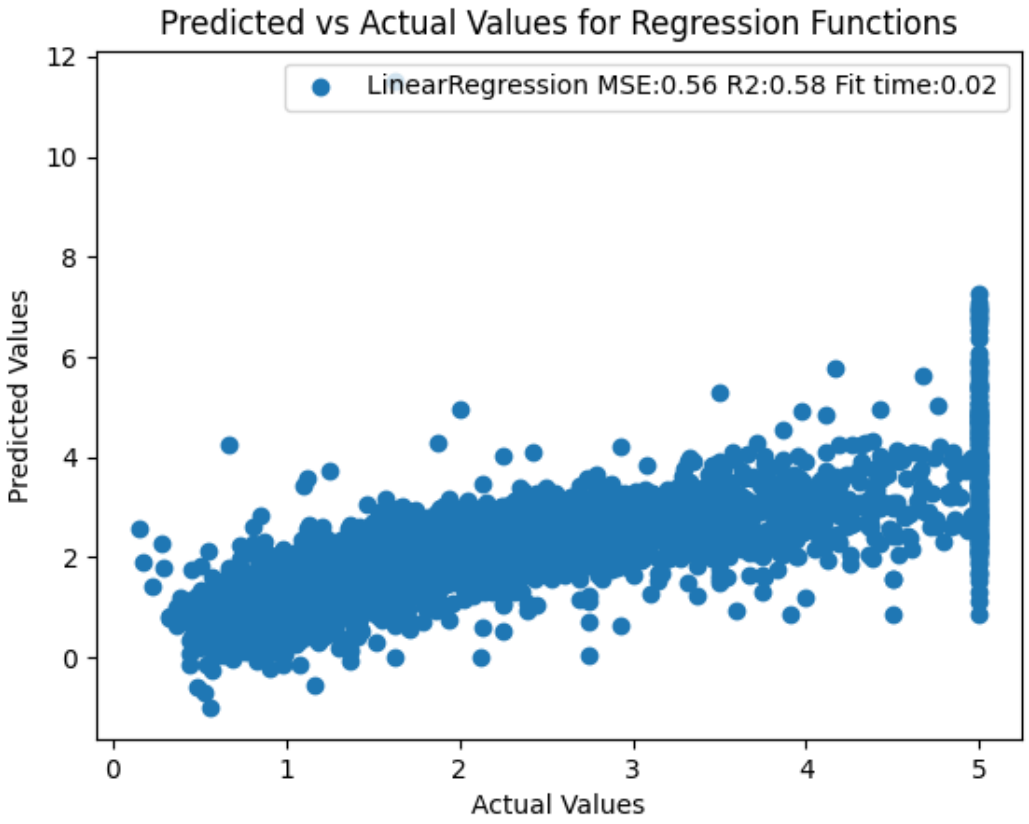
All the columns and instances in the dataset will be utilized for testing.

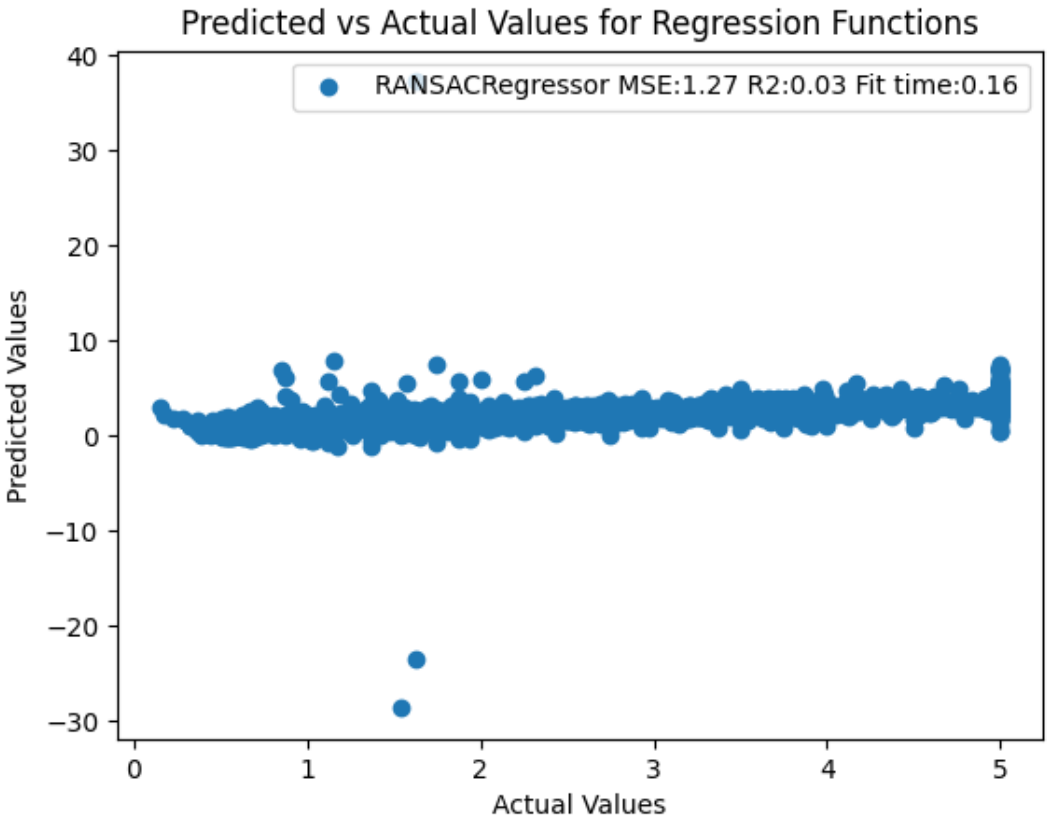
```
#Calicia Perea
#hw 5: Regression
# April 7, 2023

# Each regressor needs to be tested using the California housing dataset, which can
# be loaded using fetch_california_housing from sklearn.datasets. You need to use all the
# columns and all the instances in this dataset. Such analysis should include at least Mean squared
#error (MSE), R2 score, and the fitting (or training) time.
import numpy as np
import time
from sklearn.linear_model import LinearRegression, ElasticNet, Lasso, RANSACRegressor,Ridge
from sklearn.datasets import fetch_california_housing
from sklearn.metrics import mean_squared_error, r2_score
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt


# load the dataset
X,y = fetch_california_housing(return_X_y= True)

#split the dataset into traing and testing
X_train, X_test, y_train, y_test = train_test_split(
    X,y, test_size =0.2, random_state = 42
)
```

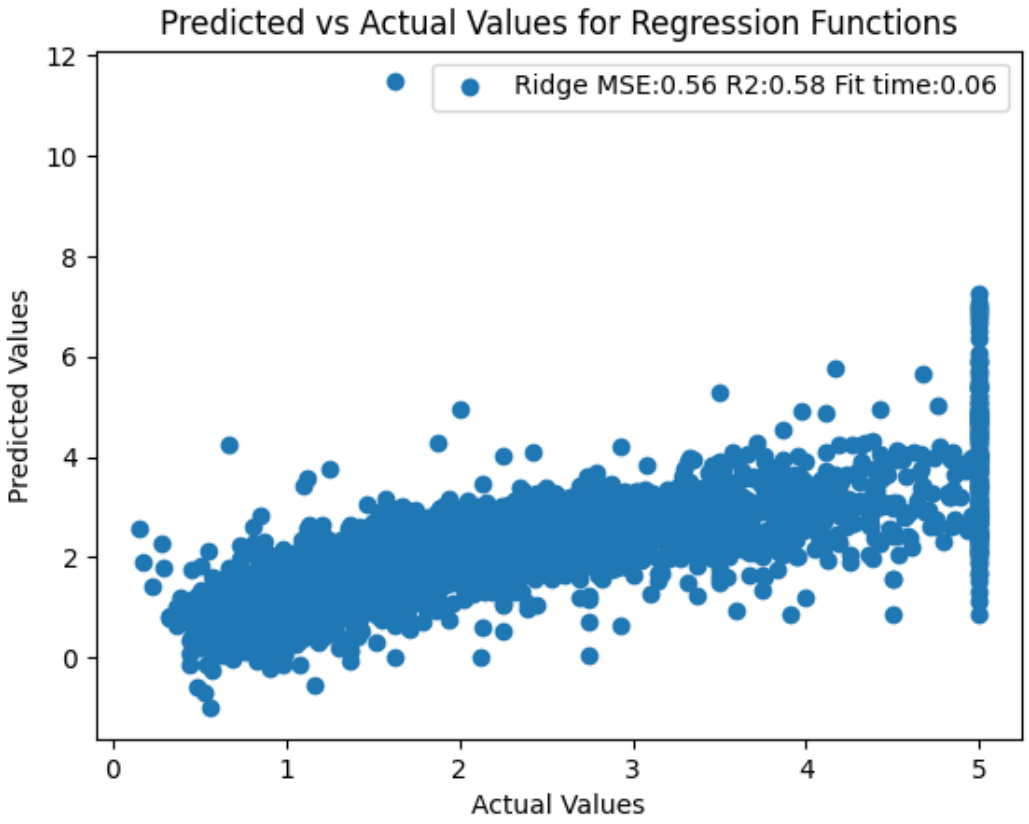
```
# create each regression function
regressor = [
    LinearRegression(),
    RANSACRegressor(),
    Ridge(),
    Lasso(),
    ElasticNet()
]

#Train and evaluate each regression function
for function in regressor:
    start = time.perf_counter()
    function.fit(X_train, y_train)
    stop = time.perf_counter()
    fit_time = stop - start
    y_pred = function.predict(X_test)
    mse = mean_squared_error(y_test,y_pred)
    r2 = r2_score(y_test, y_pred)
    plt.scatter(y_test, y_pred, label=function.__class__.__name__
                + " MSE:" + str(round(mse, 2)) + " R2:"
                + str(round(r2, 2)) + " Fit time:"
                + str(round(fit_time, 2)))
    plt.xlabel("Actual Values")
    plt.ylabel("Predicted Values")
    plt.title("Predicted vs Actual Values for Regression Functions")
    plt.legend()
    plt.show()
    print(function.__class__.__name__, " MSE:", mse," R2:",r2,
        " Fit time:", fit_time)
```

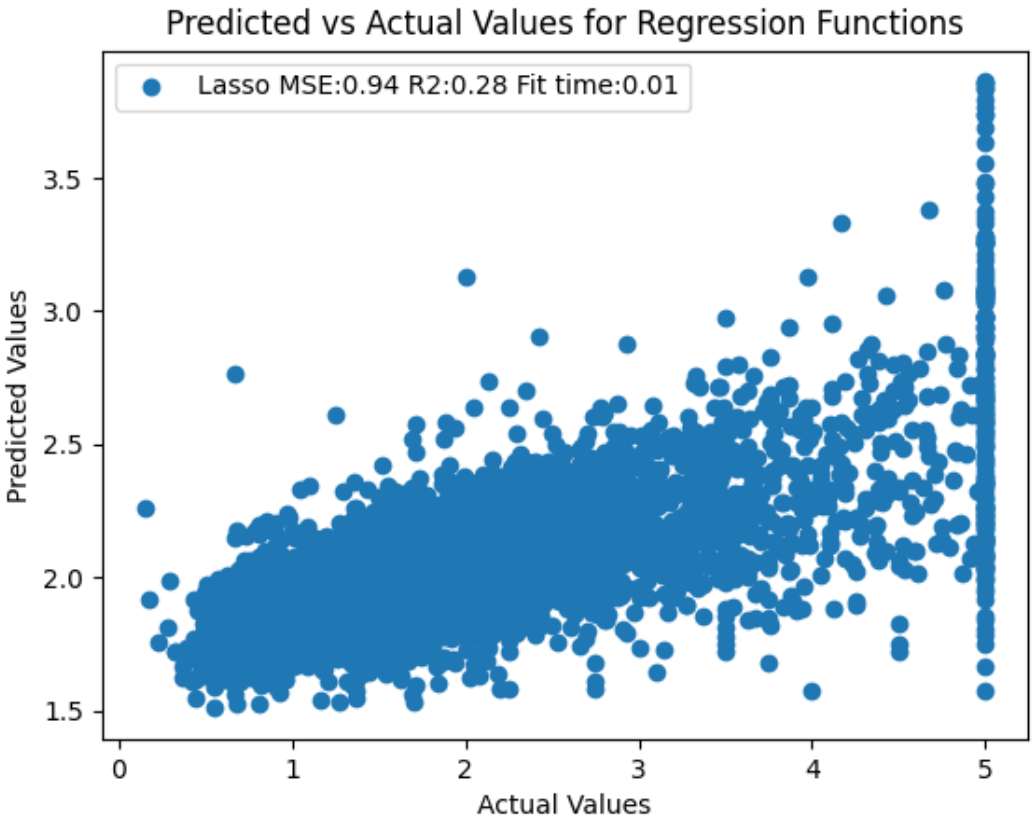
This code will output the mean squared error (MSE), R2 score, scatterplot, (showing the predicted values on the x-axis and the actual values on the y-axis. The closer the points are to the diagonal line, the better the model's predictions), and fit time for each regression function on the California housing dataset.

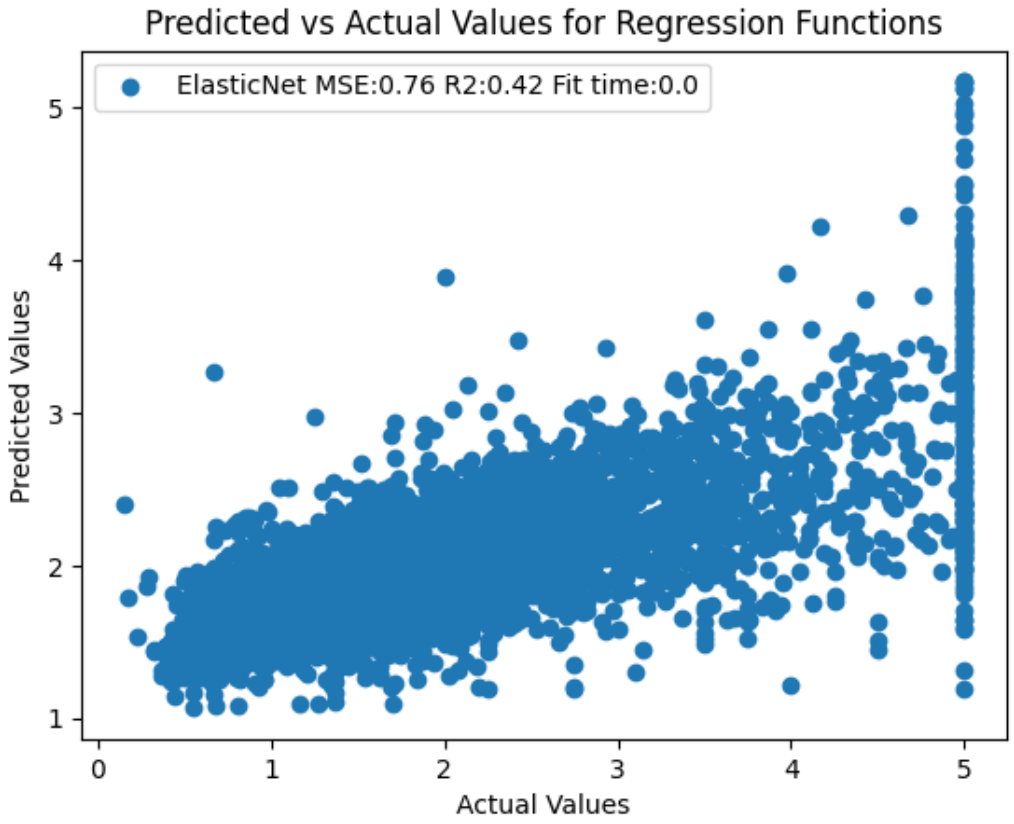
The scatter plots below show the predicted and actual values for each of the five regression functions used to model the California housing dataset. The closer the points are to the diagonal line, the better the model's predictions.


Predicted vs Actual Values for Regression Functions
LinearRegression MSE:0.56 R2:0.58 Fit time:0.02
Predicted Values
Actual Values
0
2
4
6
8
10
12
0
1
2
3
4
5


Predicted vs Actual Values for Regression Functions
RANSACRegressor MSE:1.27 R2:0.03 Fit time:0.16
Predicted Values
Actual Values
40
30
20
10
0
−10
−20
−30
0
1
2
3
4
5


Predicted vs Actual Values for Regression Functions
Ridge MSE:0.56 R2:0.58 Fit time:0.06
Predicted Values
Actual Values
0
2
4
6
8
10
12
0
1
2
3
4
5


Predicted vs Actual Values for Regression Functions
Lasso MSE:0.94 R2:0.28 Fit time:0.01
Predicted Values
Actual Values
3.5
3.0
2.5
2.0
1.5
0
1
2
3
4
5



The scatter plots show that LinearRegression and Ridge models have a better fit than the other models. The RANSACRegressor model has a few outliers, indicating that it did not perform well on those instances. The Lasso and ElasticNet models have a similar fit, but not as good as the LinearRegression and Ridge models.

The code output shows the mean squared error (MSE), R2 score, and fit time for each of the five regression functions used to model the California housing dataset.

```
(base) caliciaperea@Calicias-MacBook-Pro ~ % /Users/caliciaperea/Library/r-miniconda/bin/python /Users/caliciaperea/Deskt
op/spring23/MACHINELEARNING/hw5/HW5.py
LinearRegression  MSE: 0.5558915986952422  R2: 0.5757877060324524  Fit time: 0.025199421999786864
RANSACRegressor  MSE: 0.9158705275672965  R2: 0.30108039339232384  Fit time: 0.17180108200045652
Ridge  MSE: 0.5558034669932196  R2: 0.5758549611440138  Fit time: 0.010874333000174374
Lasso  MSE: 0.9380337514945429  R2: 0.28416718210083947  Fit time: 0.007418958999551251
ElasticNet  MSE: 0.7645556403971132  R2: 0.41655189098028234  Fit time: 0.00427364000097441
(base) caliciaperea@Calicias-MacBook-Pro ~ %
```

The LinearRegression and Ridge models have the lowest MSE and highest R2 score, indicating their better performance in predicting the target variable. RANSACRegressor has the highest MSE and negative R2 score, indicating poor performance in predicting the target variable.

The Lasso and ElasticNet models have an intermediate performance, with Lasso having a higher MSE and lower R2 score compared to ElasticNet.

The fit time for each model is also displayed, indicating how long it took to train the model.

Overall, LinearRegression and Ridge models are recommended for predicting the target variable in the California housing dataset.