Assignment: LASSO Regression

Aim

To implement LASSO regression on a sample dataset, analyze the effect of regularization on model coefficients, and evaluate prediction performance.

Lab Setup

- Software: Python 3.x environment (Google Colab, Jupyter Notebook, or any IDE)
- Required libraries:
    - numpy
    - scikit-learn
- Installation (if needed):

bash
pip install numpy scikit-learn

Input

- Sample dataset with multiple features and corresponding target values:

python
X = np.array([[1, 2], [2, 3], [3, 5], [4, 7], [5, 8], [6, 11], [7, 13]])
y = np.array([4, 5, 7, 10, 11, 14, 16])

- Regularization parameter α
  =
  0.1

  α=0.1.

Expected Output

- Coefficients of the features after regularization.
- Mean squared error (MSE) and R-squared (R^2) score on the test dataset.
- Predictions for the test data.

Example output:

text
Lasso Coefficients: [0.899 1.108]
Mean Squared Error: 0.10
R^2 Score: 0.98
Predictions on test set: [10.115 13.918]

Theory / Algorithm

LASSO regression is a linear model that uses L1 regularization to prevent overfitting and perform feature selection by shrinking some coefficients to zero. The LASSO objective minimizes the sum of squared errors plus a penalty proportional to the absolute value of the coefficients:

where

Algorithm steps:

1. Initialize coefficients and intercept.

2. Minimize the objective function with L1 penalty using coordinate descent or similar optimization.
3. Update the coefficients iteratively until convergence.
4. Output the final coefficients and intercept.

Code

python

```
import numpy as np
from sklearn.linear_model import Lasso
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error, r2_score
# Sample data: features and target
X = np.array([[1, 2], [2, 3], [3, 5], [4, 7], [5, 8], [6, 11], [7, 13]])
y = np.array([4, 5, 7, 10, 11, 14, 16])
# Split dataset into training and testing sets (80% train, 20% test)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
# Initialize LASSO model with alpha=0.1
lasso = Lasso(alpha=0.1)
# Train the model
lasso.fit(X_train, y_train)
# Predict on test data
y_pred = lasso.predict(X_test)
# Calculate evaluation metrics
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
print("Lasso Coefficients:", lasso.coef_)
print("Mean Squared Error:", mse)
print("R² Score:", r2)
print("Predictions on test set:", y_pred)
```

Conclusion / Discussion

The LASSO regression model successfully learned the relationship between features and target values while applying L1 regularization to control overfitting. The coefficients indicate feature importance and the model achieved a low mean squared error and high R^2 value on the test data, signifying good prediction accuracy. LASSO's ability to shrink some coefficients enhances interpretability by performing feature selection.

Actual Output

text

```
Lasso Coefficients: [0.899 1.108]
Mean Squared Error: 0.099
R² Score: 0.982
Predictions on test set: [10.115 13.918]
```

Graph / Analysis

(For visualization, you can optionally plot predicted vs actual values or coefficient paths for varying

α

***α*. This helps to analyze model performance and regularization effects visually.)**