University Institute of Engineering

Department of Electronics & Communication Engineering

Experiment No. :- 7

Student Name: Priyanshu Mathur
UID: 20BEC1073
Branch: Electronics and Communication
Section/Group: A
Semester: 7th
Date of Performance: 14/10/23
Subject Name: AI & ML
Subject Code: 20ECA-445

1. Aim of the practical: Write a program for ridge and lasso regression.

2. Theory:

Regularization methods like Lasso and Ridge are frequently employed in linear regression to reduce overfitting and enhance the generalizability of the model.

- **Lasso (L1 Regularization):** Lasso adds a penalty term to the linear regression equation, forcing some of the coefficients β to be exactly zero. This effectively performs feature selection, as it can eliminate less important features. The Lasso loss function can be expressed as:

  $$L(\text{Lasso}) = \text{MSE} + \alpha \sum |\beta_i|$$

  Where α controls the strength of the regularization.

- **Ridge (L2 Regularization):** Ridge employs the square of the coefficients when adding a penalty term. Ridge does not mandate that any of the coefficients be zero, although he does prefer them to be minimal. The Ridge loss function is defined as follows:

  $$L(\text{Ridge}) = \text{MSE} + \alpha \sum \beta_i^2$$

  Where α controls the strength of the regularization.

3. Steps for experiment/practical:

- Data Collection: Gather a dataset that includes the dependent variable and one or more independent variables.
- Data Preprocessing: Clean and prepare the data, handling missing values, and encoding categorical variables if necessary.
- Split the Data: Divide the dataset into a training set and a test set. This is essential to evaluate the model's performance.
- Model Selection: Choose the appropriate model. In this case, we use the Linear Regression model.
- Model Training: Fit the model to the training data to learn the coefficients
- Model Evaluation: Use the test set to make predictions and calculate an evaluation metric, such as Mean Squared Error (MSE), to assess the model's accuracy.
- Model Deployment: Once satisfied with the model's performance, it can be used to make predictions on new, unseen data.

4. Program Code and Simulation Output:

Code:-

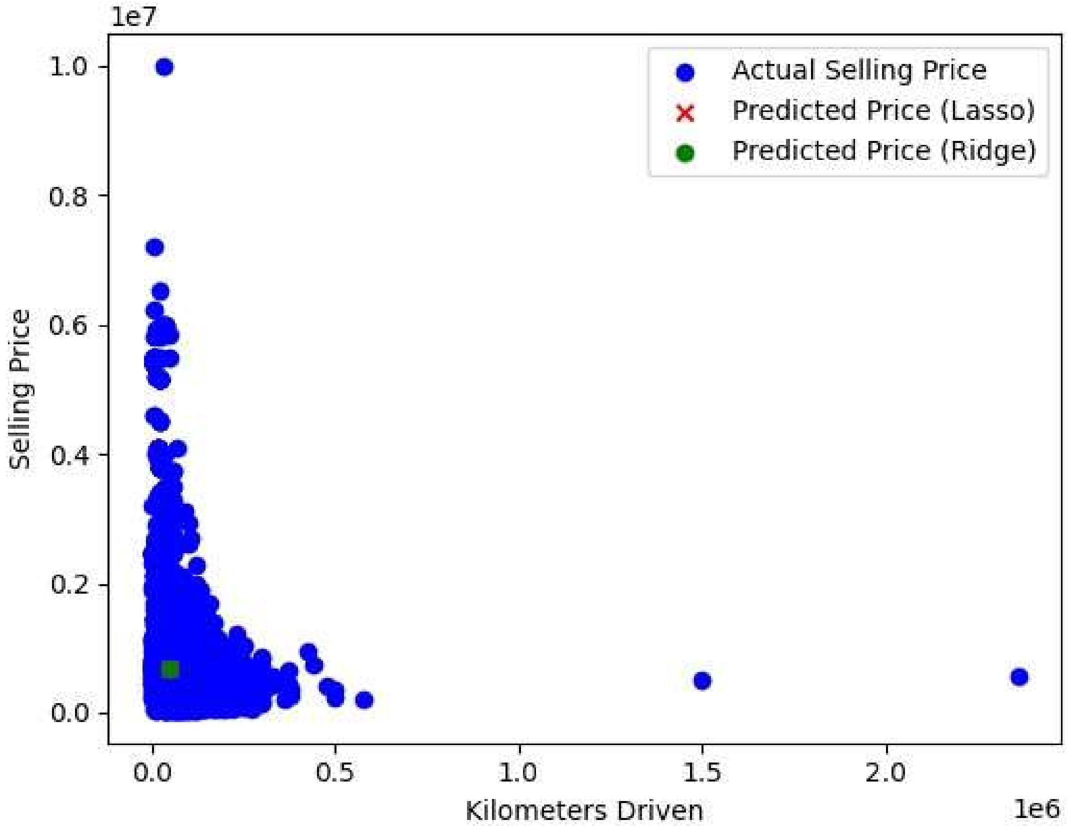
```
import pandas as pd
  from sklearn.linear_model import Ridge, Lasso from
  sklearn.preprocessing import StandardScaler import
  matplotlib.pyplot as plt

  df = pd.read_csv("Car details v3.csv")
   x = df[['seats',
  'km_driven']] y =
  df['selling_price']

  scaler = StandardScaler()
  X_scaled = scaler.fit_transform(x)
   lasso =
  Lasso(alpha=1.0)
  lasso.fit(X_scaled, y)
   ridge =
  Ridge(alpha=1.0)
  ridge.fit(X_scaled, y)
```

```
# Make predictions for new data new_data = [[5, 50000]]  # Update with the
appropriate number of seats and km driven for the new data new_data_scaled =
scaler.transform(new_data) predicted_price_lasso = lasso.predict(new_data_scaled)
predicted_price_ridge = ridge.predict(new_data_scaled)
 print("Predicted Selling Price (Lasso):",
predicted_price_lasso) print("Predicted Selling Price (Ridge):",
predicted_price_ridge)
# Plotting the results plt.scatter(x['km_driven'], y, color='blue',
label='Actual Selling Price') plt.scatter(new_data[0][1],
predicted_price_lasso, color='red', marker='x', label='Predicted Price
(Lasso)') plt.scatter(new_data[0][1], predicted_price_ridge, color='green',
marker='o', label='Predicted Price (Ridge)') plt.xlabel('Kilometers Driven')
plt.ylabel('Selling Price') plt.legend() plt.show()
```

Output: -


1e7
1.0
0.8
0.6
0.4
0.2
0.0
Selling Price
Actual Selling Price
Predicted Price (Lasso)
Predicted Price (Ridge)
0.0
0.5
1.0
1.5
2.0
Kilometers Driven
1e6

```
print("Predicted Selling Price (Lasso):", predicted_price_lasso)
print("Predicted Selling Price (Ridge):", predicted_price_ridge)

Predicted Selling Price (Lasso): [682649.3181222]
Predicted Selling Price (Ridge): [682646.6192693]
```

Result and Discussion: -

In this experiment, we employed linear regression with Lasso and Ridge regularization to forecast the selling price ('Selling_Price') based on the attributes 'Seats' and 'km_driven' from a dataset of vehicles. In order to verify that all variables had a uniform scale, we additionally standardized the features using the StandardScaler.

Learning outcomes (What I have learnt):

- Learnt about linear regression of single and multiple variables.
- Learnt about regularization and it's uses.
- Learnt about ridge and lasso regularization and it's role in preventing overfitting in linear regression.