Name: Khushi S Singh
D15A / 26

Experiment 2

Aim: Implement Multi Regression, Lasso, and Ridge Regression on real-world datasets

Theory:

1. Dataset Source

Dataset Name: **Insurance Premium Prediction Dataset**
Source: Kaggle
Link:
https://www.kaggle.com/datasets/noordeen/insurance-premium-prediction/data

This dataset is a real-world dataset used to predict **insurance charges** based on personal and medical attributes.

2. Dataset Description

The dataset contains information about individuals and their insurance charges. The objective is to predict the **insurance premium (charges)** using multiple input features.

Dataset Features

Feature	Description
age	Age of the person
sex	Gender (male/female)
bmi	Body Mass Index
children	Number of children
smoker	Smoking status
region	Residential region
charges	Insurance premium (Target Variable)

Dataset Characteristics

- Dataset Type: Regression Dataset
- Number of Features: 6 input features
- Target Variable: charges
- Dataset Size: ~1300+ records
- Contains numerical and categorical variables

Categorical features (sex, smoker, region) were converted into numerical form using encoding.

3. Mathematical Formulation of the Algorithm

a) Multiple Linear Regression

Multiple Linear Regression predicts output using a linear combination of input variables.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_n X_n + \epsilon$$

Where:

- Y = Predicted insurance charges
- $X_1, X_2 \ldots X_n$ = Input features
- β_0 = Intercept
- β_i = Regression coefficients
- ϵ = Error term

b) Ridge Regression

Ridge Regression adds **L2 regularization** to reduce overfitting.

$$Loss = \sum (y_i - \hat{y}_i)^2 + \lambda \sum \beta_i^2$$

Where:

- λ = Regularization parameter
- Reduces coefficient magnitude
- Prevents overfitting

c) Lasso Regression

Lasso Regression uses **L1 regularization**.

$$Loss = \sum (y_i - \hat{y}_i)^2 + \lambda \sum |\beta_i|$$

Where:

- Forces some coefficients to zero
- Performs feature selection
- Reduces model complexity

4. Algorithm Limitations

Multiple Linear Regression

- Sensitive to outliers
- Assumes linear relationship
- Can suffer from multicollinearity
- May overfit large datasets

Ridge Regression

- Does not eliminate features completely
- Requires feature scaling
- Choice of alpha affects performance

Lasso Regression

- May remove important features
- Sensitive to alpha parameter
- Requires feature scaling

5. Methodology / Workflow

Step 1: Dataset Loading

- The insurance dataset was loaded using Pandas.

Step 2: Data Preprocessing

The following preprocessing steps were performed:

- Dataset inspected using head()
- Categorical variables encoded
- Features and target separated

Input Features:

- age
- bmi
- children
- sex
- smoker
- region

Target Variable:

- charges

Step 3: Feature Scaling

StandardScaler was applied to normalize feature values.

Scaling is important because Ridge and Lasso regression depend on feature magnitude.

Step 4: Train-Test Split

Dataset was divided into:

- 80% Training Data
- 20% Testing Data

Step 5: Model Training

Three regression models were trained:

1 Multiple Linear Regression
2 Ridge Regression
3 Lasso Regression

Models were trained using a training dataset.

Step 6: Prediction

Each model predicted insurance charges on testing dataset.

Step 7: Visualization

Scatter plot was generated:

Actual Charges vs Predicted Charges

This plot shows the closeness between actual and predicted values.

6. Performance Analysis

Model Results

Multiple Linear Regression

- MSE = **33,600,065.35**
- R^2 Score = **0.7835**

Ridge Regression

- MSE = **33,608,105.85**
- R^2 Score = **0.7835**

Lasso Regression

- MSE = **33,600,486.25**
- R^2 Score = **0.7836**

Interpretation

- All three models produced similar results.
- R^2 score around **0.78** indicates good prediction accuracy.
- Lasso regression produced slightly better performance.
- Ridge regression reduced coefficient values.

Coefficient comparison showed that:

- **Smoker status** has the highest impact on insurance charges.
- Age and BMI also influence charges significantly.

7. Hyperparameter Tuning

Hyperparameter tuning was performed for Ridge and Lasso regression.

Ridge Regression

Alpha parameter tested:

- α = 0.1
- α = 1.0
- α = 10

Best performance obtained at:

- α = 1.0

Lasso Regression

Alpha parameter tested:

- α = 0.01
- α = 0.1
- α = 1.0

Best performance obtained at:

- α = 0.1

Impact of Tuning

- Proper alpha values improved model stability.
- Very large alpha reduced accuracy.
- Very small alpha increased overfitting.

CODE and OUTPUT:

STEP 1: Import Required Libraries

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LinearRegression, Ridge, Lasso
from sklearn.metrics import mean_squared_error, r2_score
```

STEP 2: Load Insurance Dataset

```
df = pd.read_csv("insurance.csv")
print(df.head())
```

```
   age     sex   bmi  children smoker     region  expenses
0   19  female  27.9         0    yes  southwest  16884.92
1   18    male  33.8         1     no  southeast   1725.55
2   28    male  33.0         3     no  southeast   4449.46
3   33    male  22.7         0     no  northwest  21984.47
4   32    male  28.9         0     no  northwest   3866.86
```

STEP 3: Data Preprocessing

```
df_encoded = pd.get_dummies(df, drop_first=True)
X = df_encoded.drop("expenses", axis=1)
y = df_encoded["expenses"]
```

STEP 4: Split Features and Target

```
X = df_encoded.drop("expenses", axis=1)
y = df_encoded["expenses"]
```

STEP 5: Train–Test Split

```
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)
```

STEP 6: Feature Scaling

```
scaler = StandardScaler()

X_train_scaled = scaler.fit_transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

STEP 7: Multiple Linear Regression

```
    lr = LinearRegression()
    lr.fit(X_train_scaled, y_train)

    y_pred_lr = lr.predict(X_test_scaled)

    print("Multiple Linear Regression")
    print("MSE:", mean_squared_error(y_test, y_pred_lr))
    print("R2 Score:", r2_score(y_test, y_pred_lr))
```

```
Multiple Linear Regression
MSE: 33600065.35507785
R2 Score: 0.7835726930039904
```

STEP 8: Ridge Regression (L2 Regularization)

```
    ridge = Ridge(alpha=1.0)
    ridge.fit(X_train_scaled, y_train)

    y_pred_ridge = ridge.predict(X_test_scaled)

    print("\nRidge Regression")
    print("MSE:", mean_squared_error(y_test, y_pred_ridge))
    print("R2 Score:", r2_score(y_test, y_pred_ridge))
```

```
Ridge Regression
MSE: 33608105.85844779
R2 Score: 0.7835209018996321
```

STEP 9: Lasso Regression (L1 Regularization)

```
    lasso = Lasso(alpha=0.1)
    lasso.fit(X_train_scaled, y_train)

    y_pred_lasso = lasso.predict(X_test_scaled)

    print("\nLasso Regression")
    print("MSE:", mean_squared_error(y_test, y_pred_lasso))
    print("R2 Score:", r2_score(y_test, y_pred_lasso))
```

```
Lasso Regression
MSE: 33600486.2517238
R2 Score: 0.7835699818923707
```

STEP 10: Compare Coefficients (Feature Importance)

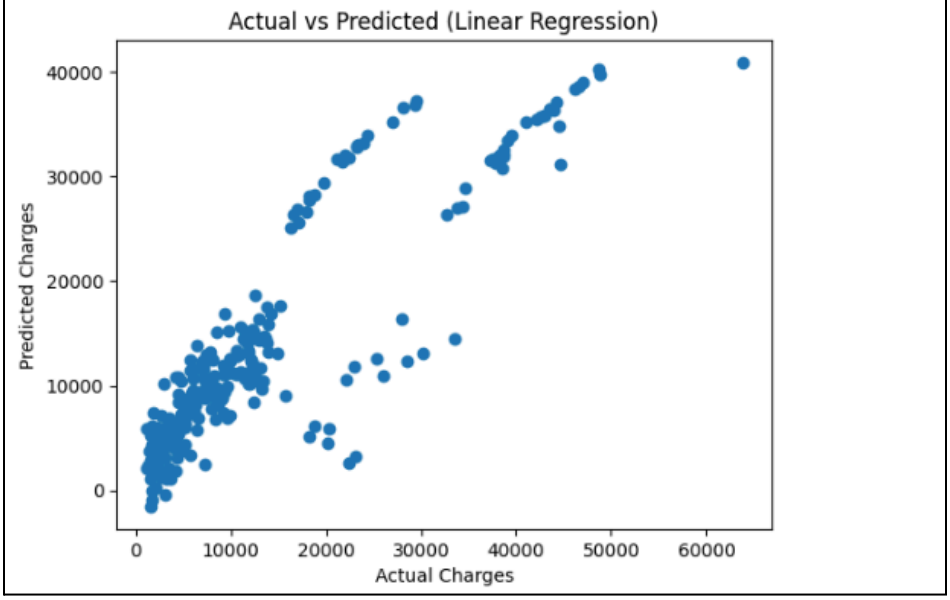
```python
coef_df = pd.DataFrame({
    "Feature": X.columns,
    "Linear": lr.coef_,
    "Ridge": ridge.coef_,
    "Lasso": lasso.coef_
})

print(coef_df)
```

	Feature	Linear	Ridge	Lasso
0	age	3614.697633	3611.077119	3614.610608
1	bmi	2037.268555	2035.403365	2037.119826
2	children	517.330947	517.201538	517.234528
3	sex_male	-9.257136	-8.580119	-9.144034
4	smoker_yes	9558.151403	9548.947305	9558.041695
5	region_northwest	-157.985768	-157.478589	-157.680644
6	region_southeast	-290.531103	-288.919475	-290.178359
7	region_southwest	-348.865173	-348.025435	-348.545100

STEP 11: Visual Comparison (Predicted vs Actual)

```python
plt.figure()
plt.scatter(y_test, y_pred_lr)
plt.xlabel("Actual Charges")
plt.ylabel("Predicted Charges")
plt.title("Actual vs Predicted (Linear Regression)")
plt.show()
```



Result:

Multiple Linear Regression, Ridge Regression, and Lasso Regression models were successfully implemented on the Insurance Premium Prediction dataset. The models achieved an average **R^2 score of about 0.78**, indicating good prediction accuracy. Among the three models, **Lasso Regression showed slightly better performance**, while Ridge Regression produced more stable coefficient values.

Conclusion:

The experiment demonstrated the use of Multiple Linear Regression, Ridge, and Lasso Regression for predicting insurance charges. All models performed similarly, but regularization methods helped improve model stability. The experiment shows that regression techniques are effective for real-world prediction problems.