Ridge and lasso

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import Ridge, Lasso, ElasticNet
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import r2_score,mean_squared_error

#Load the dataset
df = pd.read_csv('BostonHousing (1).xls')
print(df.head(10))

# Prepare features and target
X = df['rm'].values
y = df['medv'].values

df.isnull().sum()

# Split data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=0)

scaler= StandardScaler()
X_train_scaled = scaler.fit_transform(X_train.reshape(-1, 1))
X_test_scaled = scaler.transform(X_test.reshape(-1, 1))

# Initialize models
ridge = Ridge(alpha=1.0)
lasso = Lasso(alpha=0.1)

#fit models
ridge.fit(X_train_scaled, y_train)
lasso.fit(X_train_scaled, y_train)



# Make predictions
y_pred_ridge = ridge.predict(X_test_scaled)
```

```
y_pred_lasso = lasso.predict(X_test_scaled)

# Evaluate models
mse_ridge = mean_squared_error(y_test, y_pred_ridge)
mse_lasso = mean_squared_error(y_test, y_pred_lasso)

print("Ridge Regression MSE:", round(mse_ridge, 2))
print("Lasso Regression MSE:", round(mse_lasso, 2))

# Train and predict with Ridge Regression
print("Prediction for 5 rooms is:",ridge.predict([[5]])[0])

# Train and predict with Lasso Regression
print("Prediction for 5 rooms is:",lasso.predict([[5]])[0])

plt.scatter(X, y, color='red', label="Original Data")
plt.plot(X_test, y_pred_ridge, color='blue', label="Ridge Regression")
plt.plot(X_test, y_pred_lasso, color='green', label="Lasso Regression")
plt.xlabel("Number of Rooms (RM)")
plt.ylabel(" Price")
plt.title("Ridge vs Lasso Regression (Boston Housing)")
plt.legend()
plt.show()
```