

Task 3: Linear Regression - House Price Prediction

Objective

Implement and understand simple & multiple linear regression.

Tools Used

Scikit-learn, Pandas, Matplotlib, Seaborn

Step 1: Import Libraries & Load Dataset

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.linear_model import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

# Load dataset
df = pd.read_csv("https://raw.githubusercontent.com/selva86/datasets/master/BostonHousing.csv")
df.head()
```

Step 2: Preprocessing

```
print(df.isnull().sum())
print(df.describe())
```

Step 3: Simple Linear Regression

```
X = df[['rm']]
y = df['medv']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
model = LinearRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)

print('MAE:', mean_absolute_error(y_test, y_pred))
print('MSE:', mean_squared_error(y_test, y_pred))
```

```
print('R² Score:', r2_score(y_test, y_pred))
```

Step 4: Plot Regression Line

```
plt.scatter(X_test, y_test, color='blue', label='Actual')
plt.plot(X_test, y_pred, color='red', linewidth=2, label='Predicted Line')
plt.xlabel('Average number of rooms (RM)')
plt.ylabel('Median value of homes (MEDV)')
plt.title('Simple Linear Regression')
plt.legend()
plt.show()
```

Step 5: Interpret Coefficients

```
print('Intercept (b0):', model.intercept_)
print('Slope (b1):', model.coef_[0])
```

Step 6: Multiple Linear Regression

```
X = df[['rm', 'lstat', 'ptratio']]
y = df['medv']
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
multi_model = LinearRegression()
multi_model.fit(X_train, y_train)
y_pred_multi = multi_model.predict(X_test)

print('MAE (Multiple):', mean_absolute_error(y_test, y_pred_multi))
print('MSE (Multiple):', mean_squared_error(y_test, y_pred_multi))
print('R² Score (Multiple):', r2_score(y_test, y_pred_multi))

coefficients = pd.DataFrame(multi_model.coef_, X.columns, columns=['Coefficient'])
print(coefficients)
```

Summary

Model Type	Features Used	R² Score	Interpretation
Simple Linear	rm	~0.6-0.7	Positive impact of rooms
Multiple Linear	rm, lstat, ptratio	~0.7-0.8	More accurate