

Linear Regression

1. Objective

Build a Linear Regression model to predict house prices based on house size.

2. Install Required Libraries (One Time Only)

```
pip install numpy pandas matplotlib scikit-learn
```

3. Import Libraries

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

4. Create Sample Dataset

```
X = np.array([300, 700, 950, 1300, 1500, 1900, 2000, 2500, 2800, 3200])
y = np.array([15, 35, 50, 57, 80, 95, 100, 125, 150, 170])
```

```
mydata = pd.DataFrame({
    "Size": X,
    "Price": y
})
```

```
print(mydata)
```

5. Separate Feature and Target

```
X = mydata[["Size"]]
y = mydata["Price"]
```

6. Split Train and Test Data

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

7. Create Model

```
model = LinearRegression()
```

8. Train Model

```
model.fit(X_train, y_train)  
print("Model Trained Successfully")
```

9. View Equation Values

```
print("Slope:", model.coef_[0])  
print("Intercept:", model.intercept_)
```

Equation:

Price = m * Size + b

10. Make Predictions

```
y_pred = model.predict(X_test)  
print("Actual:", list(y_test))  
print("Predicted:", list(y_pred))
```

11. Evaluate Model

```
mse = mean_squared_error(y_test, y_pred)  
r2 = r2_score(y_test, y_pred)  
  
print("MSE:", mse)  
print("R2 Score:", r2)
```

12. Predict New House Price

```
new_size = [[2200]]  
predicted_price = model.predict(new_size)  
print("Predicted Price:", predicted_price[0])
```

13. Visualization

```
plt.scatter(X, y)  
plt.plot(X, model.predict(X), linewidth=2)  
plt.xlabel("House Size")  
plt.ylabel("House Price")
```

```
plt.title("Linear Regression ")
plt.show()
```

14. Key Learning

- Linear Regression finds best fit line
- Model learns relationship from data
- Model can predict unseen values

Linear Regression predicts continuous values by fitting a straight line that minimizes error.