DATA SCIENCE PROJECT

Here you can understand linear regresion problem in easyway

Non-Critical & easy to implement

we'll use a simple dataset containing information about house prices based on their areas.

Step -1 : Dataset

Area (sqft)	Price (USD)
1000	250000
1200	280000
1500	320000
1800	370000
2000	400000
2500	480000
3000	550000

Step -2 : Import Libraries

```
import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

from sklearn.linear_model import LinearRegression
```

we'll use a simple dataset containing information about house prices based on their areas.

Step -3 : Load & Explore the data

```
# Load the dataset
data = {
    'Area': [1000, 1200, 1500, 1800, 2000, 2500, 3000],
    'Price': [250000, 280000, 320000, 370000, 400000, 480000, 550000]
}
df = pd.DataFrame(data)

# Explore the data
print(df.head())
print(df.describe())
```

we'll use a simple dataset containing information about house prices based on their areas.

Step -4 : Data Visualization

```
plt.scatter(df['Area'], df['Price'])

plt.xlabel('Area (sqft)')

plt.ylabel('Price (USD)')

plt.title('House Price vs. Area')

plt.show()
```

we'll use a simple dataset containing information about house prices based on their areas.

Step -5 : Prepare Data for Training

```
X = df[['Area']]

y = df['Price']

# Split the data into training and testing sets (80% train, 20% test)

from sklearn.model_selection import train_test_split

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

we'll use a simple dataset containing information about house prices based on their areas.

Step -6 : Train Linear Regression Model

```
# Create the linear regression model

model = LinearRegression()

# Train the model on the training data

model.fit(X_train, y_train)
```

we'll use a simple dataset containing information about house prices based on their areas.

Step -7 : Evaluate the Model

```
# Make predictions on the test set
y_pred = model.predict(X_test)

# Evaluate the model
from sklearn.metrics import mean_squared_error, r2_score
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)

print("Mean Squared Error:", mse)
print("R-squared:", r2)
```

we'll use a simple dataset containing information about house prices based on their areas.

Step -8 : Visualize the Regression Line

```
plt.scatter(df['Area'], df['Price'])

plt.plot(X_test, y_pred, color='red', linewidth=2)

plt.xlabel('Area (sqft)')

plt.ylabel('Price (USD)')

plt.title('House Price vs. Area with Regression Line')

plt.show()
```

we'll use a simple dataset containing information about house prices based on their areas.

Step -10 : Make Predictions

```
# Let's say we want to predict the price for an area of 2200 sqft

new_area = np.array([[2200]])

predicted_price = model.predict(new_area)

print("Predicted Price for 2200 sqft:", predicted_price[0])
```

Thank you

Follow me on LinkedIn & GitHub