📘 Task-1: Linear Regression – California Housing Dataset

📌 Cell 1: Import Required Libraries

```
import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns


from sklearn.datasets import fetch_california_housing

from sklearn.model_selection import train_test_split

from sklearn.linear_model import LinearRegression

from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score


import warnings

warnings.filterwarnings("ignore")
```

📌 Cell 2: Load California Housing Dataset

```
# Load dataset

data = fetch_california_housing(as_frame=True)


# Create DataFrame

df = pd.concat(

  [data.data, data.target.rename("MedHouseVal")],

  axis=1

)


# Display first rows

df.head()
```

📌 Cell 3: Basic Dataset Information

```
print("Dataset Shape:", df.shape)

print("\nDataset Info:")

df.info()
```

📌 Cell 4: Check Missing Values

```
df.isnull().sum()
```

✔ There are **no missing values** in this dataset.

📌 Cell 5: Statistical Summary

```
df.describe()
```

📌 Cell 6: Exploratory Data Analysis (EDA)

🔹 Distribution of Target Variable

```
plt.figure(figsize=(6,4))

sns.histplot(df["MedHouseVal"], bins=30, kde=True)

plt.title("Distribution of Median House Value")

plt.xlabel("Median House Value")

plt.ylabel("Frequency")

plt.show()
```

🔹 Correlation Heatmap

```
plt.figure(figsize=(10,8))

sns.heatmap(df.corr(), cmap="coolwarm", annot=False)

plt.title("Correlation Heatmap")

plt.show()
```

📌 **Cell 7: Feature Selection & Train-Test Split**

```
# Features and target
X = df.drop(columns="MedHouseVal")
y = df["MedHouseVal"]

# Train-test split
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)

print("Training set shape:", X_train.shape)
print("Testing set shape:", X_test.shape)
```

📌 **Cell 8: Train Linear Regression Model**

```
# Initialize model
model = LinearRegression()

# Train model
model.fit(X_train, y_train)

print("Model training completed.")
```

📌 **Cell 9: Make Predictions**

```
y_pred = model.predict(X_test)
```

📌 **Cell 10: Model Evaluation Metrics**

```
mae = mean_absolute_error(y_test, y_pred)
```

```
rmse = mean_squared_error(y_test, y_pred, squared=False)
r2 = r2_score(y_test, y_pred)

print(f"MAE  : {mae:.3f}")
print(f"RMSE : {rmse:.3f}")
print(f"R²   : {r2:.3f}")
```

📌 **Cell 11: Actual vs Predicted Values Plot**

```
plt.figure(figsize=(6,6))
plt.scatter(y_test, y_pred, alpha=0.4)
plt.plot([y_test.min(), y_test.max()],
         [y_test.min(), y_test.max()],
         color="red")
plt.xlabel("Actual House Value")
plt.ylabel("Predicted House Value")
plt.title("Actual vs Predicted House Prices")
plt.show()
```

📌 **Cell 12: Residual Plot**

```
residuals = y_test - y_pred

plt.figure(figsize=(6,4))
sns.histplot(residuals, bins=30, kde=True)
plt.title("Residual Distribution")
plt.xlabel("Residuals")
plt.show()
```

📌 **Cell 13 (Optional – Bonus): Save Model**

```
import pickle

with open("linear_regression_model.pkl", "wb") as file:
    pickle.dump(model, file)

print("Model saved successfully.")
```

Artificial Intelligence & Machine Learning – Task 1

House Price Prediction using Linear Regression

Name: Vijay Sharma
Internship Domain: Artificial Intelligence & Machine Learning
Organization: Maincrafts Technology
Task: Build & Evaluate a Linear Regression Model
Dataset: California Housing Dataset

---

1. Introduction

The objective of this task is to understand and implement the complete **Machine Learning workflow** by building a **Linear Regression model** to predict house prices. The project focuses on data loading, exploratory data analysis, preprocessing, model training, evaluation, and result interpretation.

Linear Regression is one of the fundamental supervised learning algorithms used for predicting continuous values. In this task, it is applied to predict the **median house value** based on various housing-related features.

---

2. Dataset Description

The **California Housing Dataset** is a real-world dataset provided by **scikit-learn**. Each row represents aggregated housing data from California districts.

Target Variable

- **MedHouseVal** – Median house value (in hundreds of thousands of dollars)

Input Features

- MedInc – Median income
- HouseAge – Median house age
- AveRooms – Average number of rooms
- AveBedrms – Average number of bedrooms
- Population – Population of the district
- AveOccup – Average occupancy
- Latitude – Latitude
- Longitude – Longitude

The dataset contains **20,640 rows and 9 columns** and does **not contain missing values**, making it suitable for regression modeling.

---

3. Exploratory Data Analysis (EDA)

Exploratory Data Analysis was performed to understand the structure and behavior of the dataset.

Steps Performed

- Checked dataset shape and data types
- Verified missing values (none found)
- Generated statistical summaries
- Visualized the distribution of the target variable
- Created a correlation heatmap

Key Observations

- Median house value shows a slightly right-skewed distribution.
- Median income has a strong positive correlation with house prices.
- Location-based features (latitude & longitude) also influence house values.

EDA helped in identifying important features and understanding data patterns before model training.

---

4. Data Preprocessing

The dataset was split into:

- **Features (X)** – All columns except MedHouseVal
- **Target (y)** – MedHouseVal

The data was divided into:

- **80% Training set**
- **20% Testing set**

This ensures that the model is evaluated on unseen data, improving reliability.

---

5. Model Training

A **Linear Regression** model from sklearn.linear_model was used.

Why Linear Regression?

- Simple and interpretable
- Suitable for continuous target prediction
- Efficient for baseline regression tasks

The model was trained using the training dataset and then used to predict house prices on the test dataset.

---

6. Model Evaluation

The model was evaluated using the following metrics:

- **MAE (Mean Absolute Error)**
- **RMSE (Root Mean Squared Error)**
- **R^2 Score (Coefficient of Determination)**

Results

- MAE indicates the average absolute prediction error.
- RMSE penalizes larger errors and reflects model accuracy.
- R^2 score shows how well the model explains variance in data.

The obtained results indicate that the model performs reasonably well for a baseline regression model.

---

7. Visualization & Interpretation

Actual vs Predicted Plot

- Shows how close predictions are to actual values.
- Points near the diagonal line indicate better predictions.

Residual Distribution

- Residuals are approximately normally distributed.
- Indicates no major bias in predictions.

These visualizations confirm that the linear regression model fits the data adequately.

---

8. Conclusion

In this task, a complete machine learning pipeline was implemented successfully:

- Data loading and analysis
- Feature selection
- Model training
- Evaluation and visualization

The Linear Regression model provided a good baseline performance for house price prediction.
This project helped in gaining hands-on experience with **real-world datasets**, **EDA**, and **model evaluation techniques**.

---

9. Future Improvements

- Apply feature scaling and normalization
- Use advanced models such as Ridge, Lasso, or Random Forest
- Perform hyperparameter tuning
- Add cross-validation
- Deploy the model using a simple web interface

---

10. Tools & Technologies Used

- Python
- Pandas & NumPy
- Scikit-learn
- Matplotlib & Seaborn
- Jupyter Notebook