```
1 from google.colab import files
2 uploaded = files.upload()
3 import pandas as pd
4 import numpy as np
5 from sklearn.model_selection import train_test_split
6 from sklearn.linear_model import LinearRegression
7 from sklearn.metrics import mean_squared_error, mean_absolute_error
8
```

Choose files boston_cleaned.csv

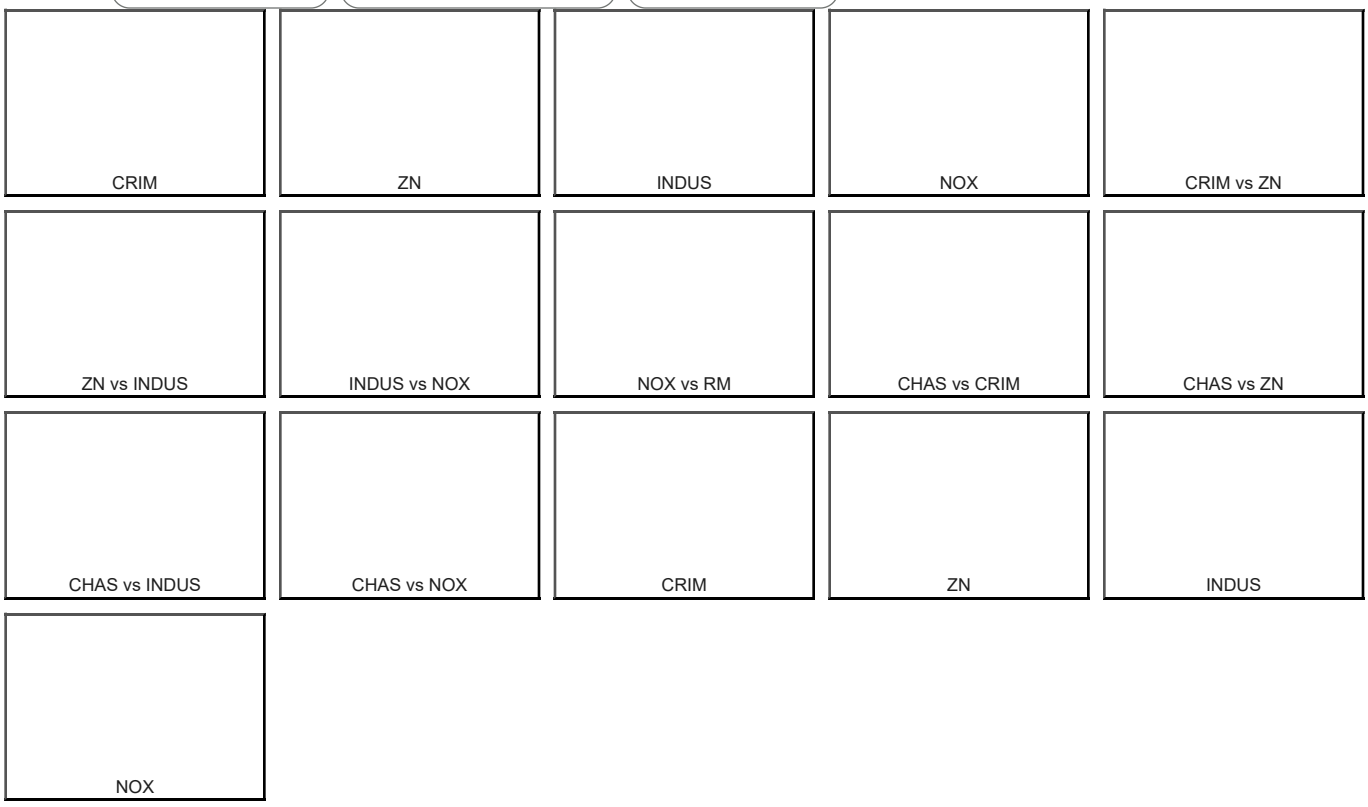
- **boston_cleaned.csv**(text/csv) - 41685 bytes, last modified: 30/08/2025 - 100% done

```
Saving boston_cleaned.csv to boston_cleaned (3).csv
```

```
1 # Step 2: Load dataset
2 df = pd.read_csv("boston_cleaned.csv")
3 # Preview dataset
4 df.head()
5
```

	CRIM	ZN	INDUS	CHAS	NOX	RM	AGE	DIS	RAD	TAX	PTRATIO	B	LSTAT	MEDV
0	0.00632	18.0	2.31	0.0	0.538	6.575	65.2	4.0900	1.0	296.0	15.3	396.90	4.98	24.0
1	0.02731	0.0	7.07	0.0	0.469	6.421	78.9	4.9671	2.0	242.0	17.8	396.90	9.14	21.6
2	0.02729	0.0	7.07	0.0	0.469	7.185	61.1	4.9671	2.0	242.0	17.8	392.83	4.03	34.7
3	0.03237	0.0	2.18	0.0	0.458	6.998	45.8	6.0622	3.0	222.0	18.7	394.63	2.94	33.4
4	0.06905	0.0	2.18	0.0	0.458	7.147	54.2	6.0622	3.0	222.0	18.7	396.90	5.33	36.2

Next steps: Generate code with df | View recommended plots | New interactive sheet




```
1 # Step 3: Split into features (X) and target (y)
2 X = df.drop("MEDV", axis=1)   # Features (all columns except target)
3 y = df["MEDV"]                # Target column
4
```

```
1 # Step 4: Train/Test split (80% train, 20% test)
2 X_train, X_test, y_train, y_test = train_test_split(
3     X, y, test_size=0.2, random_state=42
```

```
)
```

```
# Step 5: Train Linear Regression model
model = LinearRegression()
model.fit(X_train, y_train)
```

```
LinearRegression()
```

```
# Step 6: Make predictions
y_pred = model.predict(X_test)
```

```
# Step 7: Evaluate model
mse = mean_squared_error(y_test, y_pred)
mae = mean_absolute_error(y_test, y_pred)
rmse = np.sqrt(mse)
mape = np.mean(np.abs((y_test - y_pred) / y_test)) * 100

print("Model Evaluation Metrics:")
print(f"Mean Squared Error (MSE): {mse:.2f}")
print(f"Mean Absolute Error (MAE): {mae:.2f}")
print(f"Root Mean Squared Error (RMSE): {rmse:.2f}")
print(f"Mean Absolute Percentage Error (MAPE): {mape:.2f}%")
```

```
Model Evaluation Metrics:
Mean Squared Error (MSE): 12.01
Mean Absolute Error (MAE): 2.40
Root Mean Squared Error (RMSE): 3.47
Mean Absolute Percentage Error (MAPE): 13.29%
```