import pandas as pd

# Load the dataset

real\_estate\_data = pd.read\_csv("/Real\_Estate.csv")

# Display the first few rows of the dataset and the info about the dataset

real\_estate\_data\_head = real\_estate\_data.head(6)

data\_info = real\_estate\_data.info()

print(real\_estate\_data\_head)

print(data\_info)

import matplotlib.pyplot as plt

import seaborn as sns

# Set the aesthetic style of the plots

sns.set\_style("whitegrid")

# Create histograms for the numerical columns

fig, axes = plt.subplots(nrows=3, ncols=2, figsize=(12, 12))

fig.suptitle('Histograms of Real Estate Data', fontsize=16)

cols = ['House age', 'Distance to the nearest MRT station', 'Number of convenience stores',

        'Latitude', 'Longitude', 'House price of unit area']

for i, col in enumerate(cols):

    sns.histplot(real\_estate\_data[col], kde=True, ax=axes[i//2, i%2])

    axes[i//2, i%2].set\_title(col)

    axes[i//2, i%2].set\_xlabel('')

    axes[i//2, i%2].set\_ylabel('')

plt.tight\_layout(rect=[0, 0.03, 1, 0.95])

plt.show()

# Scatter plots to observe the relationship with house price

fig, axes = plt.subplots(nrows=2, ncols=2, figsize=(12, 10))

fig.suptitle('Scatter Plots with House Price of Unit Area', fontsize=16)

# Scatter plot for each variable against the house price

sns.scatterplot(data=real\_estate\_data, x='House age', y='House price of unit area', ax=axes[0, 0])

sns.scatterplot(data=real\_estate\_data, x='Distance to the nearest MRT station', y='House price of unit area', ax=axes[0, 1])

sns.scatterplot(data=real\_estate\_data, x='Number of convenience stores', y='House price of unit area', ax=axes[1, 0])

sns.scatterplot(data=real\_estate\_data, x='Latitude', y='House price of unit area', ax=axes[1, 1])

plt.tight\_layout(rect=[0, 0.03, 1, 0.95])

plt.show()

# Correlation matrix

# Excluding the 'Transaction date' column which is of type object (string)

correlation\_matrix = real\_estate\_data.select\_dtypes(include=['number']).corr()

# Plotting the correlation matrix

plt.figure(figsize=(10, 6))

sns.heatmap(correlation\_matrix, annot=True, cmap='coolwarm', fmt=".2f", linewidths=.5)

plt.title('Correlation Matrix')

plt.show()

print(correlation\_matrix)

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LinearRegression

from sklearn.metrics import mean\_squared\_error, r2\_score

# Selecting features and target variable

features = ['Distance to the nearest MRT station', 'Number of convenience stores', 'Latitude', 'Longitude']

target = 'House price of unit area'

X = real\_estate\_data[features]

y = real\_estate\_data[target]

# Splitting the dataset into training and testing sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Model initialization

model = LinearRegression()

# Training the model

model.fit(X\_train, y\_train)

# Making predictions using the linear regression model

y\_pred\_lr = model.predict(X\_test)

# Visualization: Actual vs. Predicted values

plt.figure(figsize=(10, 6))

plt.scatter(y\_test, y\_pred\_lr, alpha=0.5)

plt.plot([y\_test.min(), y\_test.max()], [y\_test.min(), y\_test.max()], 'k--', lw=2)

plt.xlabel('Actual')

plt.ylabel('Predicted')

plt.title('Actual vs. Predicted House Prices')

plt.show()