Name: Mhar Andrei Consebido Macapallag
Y&S: BSCS 3B IS
Course: CSST 102 | Basic Machine Learning
Topic: Topic 2: Supervised Learning Fundamentals

Machine Problem #2: Predicting House Prices with Multiple Regression Report

1. Data Exploration and Visualization

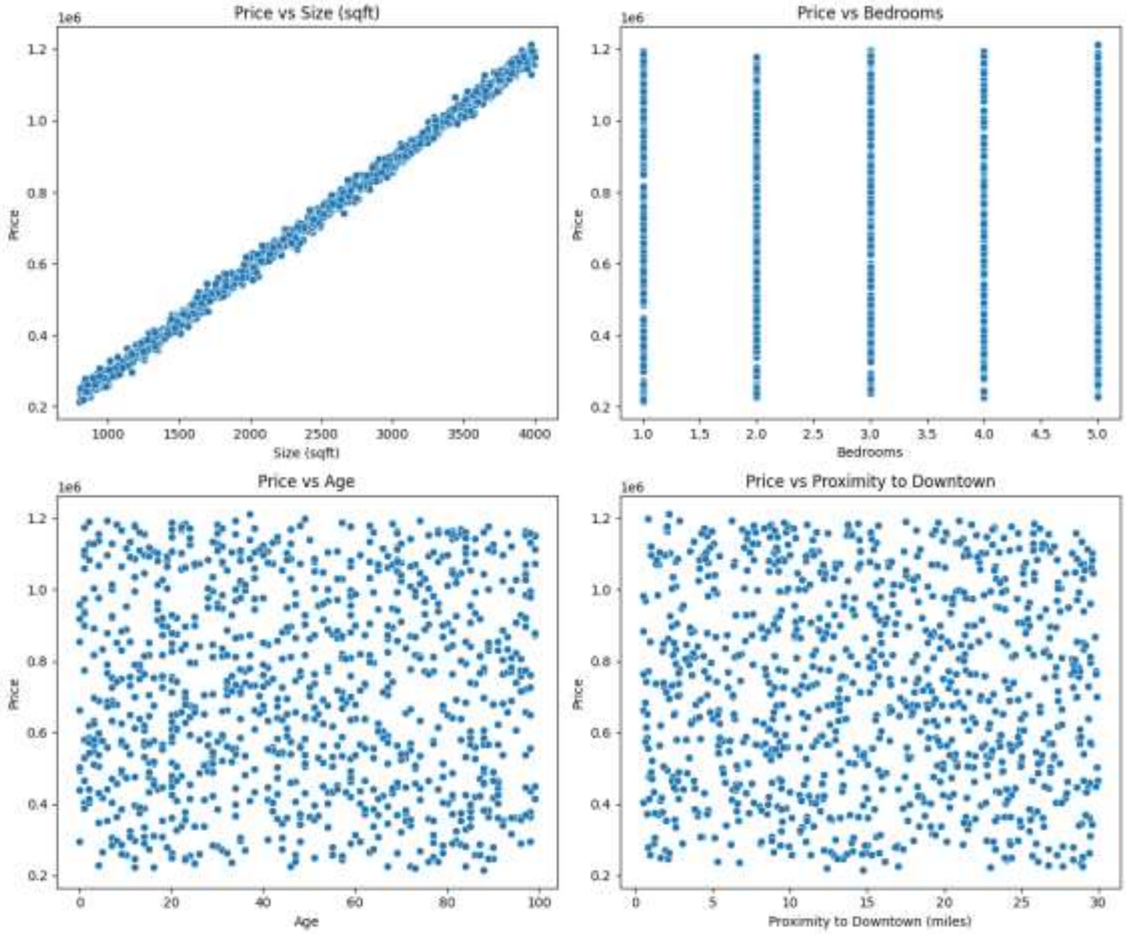
Explanation:

1. **Dataset Overview:** I began by inspecting the dataset's structure using the head() function. This step provided a glimpse into the initial rows, helping me understand the layout and types of data present.
2. **Missing Data:** To ensure the quality of the dataset, I checked for any missing values with the isnull().sum() function. Since the dataset showed no missing values across all columns, there was no need for imputation at this stage.
3. **Summary Statistics:** The describe() function was used to generate summary statistics for numerical columns. This provided insights into the range, central tendency, and dispersion of features such as size, number of bedrooms, age, proximity to downtown, and price.
4. **Scatter Plots:** I visualized the relationships between each feature and the target variable (price) using scatter plots. This visualization aimed to uncover any strong trends or patterns that might indicate useful predictors for the model.
5. **Correlation Matrix:** A heatmap of the correlation matrix was created to analyze the strength and direction of relationships between features. This matrix ranges from -1 to 1, with values close to 1 or -1 indicating strong correlations, and those near 0 suggesting weak or no linear relationship.

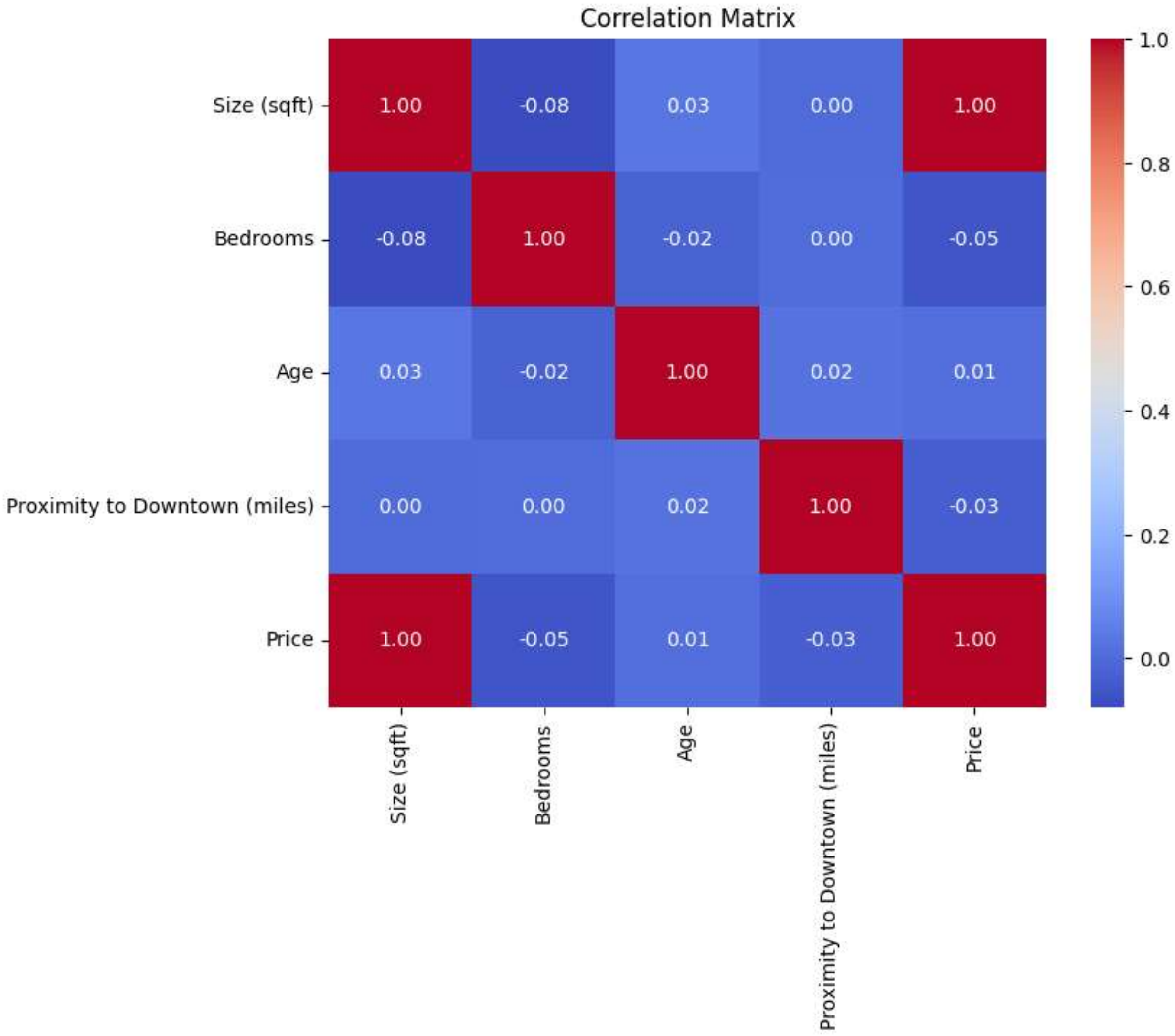
Outputs and Insights:

- **Dataset Overview:** The initial rows reveal a variety of house sizes, bedroom counts, ages, and proximities to downtown. Prices vary significantly, reflecting the diversity in the dataset.
- **Missing Data:** No missing data was detected, which is great because it means there are no gaps that could affect the model's performance.
- **Summary Statistics:**
  - **Size (sqft):** The houses vary from 801 to 3997 square feet, with a mean size of around 2429 sqft.

    - **Bedrooms:** Most houses have between 1 and 5 bedrooms, with a mean of approximately 3.
    - **Age:** The age of the houses ranges from 0 to 99 years, with a median age of 47 years.
    - **Proximity to Downtown:** Distances vary from 0.5 to 29.9 miles, with an average of about 15.3 miles.
    - **Price:** House prices range from $215,945 to $1,212,350, with an average price of approximately $719,053.
- **Visualizations:**
    - **Scatter Plots:** The scatter plots show varying degrees of relationships between features and price:
        - **Size vs Price:** A positive trend is visible; larger houses generally cost more.
        - **Bedrooms vs Price:** Some correlation is observed, with houses having more bedrooms tending to be more expensive.
        - **Age vs Price:** There seems to be a less clear relationship; older houses don't necessarily cost less.
        - **Proximity to Downtown vs Price:** Houses closer to downtown often have higher prices.



- **Correlation Matrix:** The matrix shows that size and price have a strong positive correlation (0.71), suggesting that larger houses generally cost more. Proximity to downtown has a negative correlation with price (-0.59), indicating that houses further from downtown tend to be less expensive.



2. Data Preprocessing

Explanation:

1. **Handling Missing Data:** As verified during data exploration, there were no missing values in the dataset. Therefore, this step was already completed, ensuring that our data is intact and complete.
2. **Separating Features and Target Variable:**

  - **Features (X):** I isolated the features used for predicting house prices, which include:
    - Size (in square feet)
    - Number of Bedrooms
    - Age of the house
    - Proximity to Downtown (in miles)
  - **Target Variable (y):** The target variable is the house price, which we aim to predict based on the features.
3. **Train-Test Split:**
   - The dataset was split into training and testing sets using the train_test_split() function from the sklearn library.
   - **Training Set:** 70% of the data is used to train the model.
   - **Testing Set:** 30% of the data is reserved for evaluating the model's performance.
4. **Standardization:**
   - To ensure that all features are on a similar scale and to improve the performance of machine learning algorithms, the features were standardized.
   - **StandardScaler:** This scaler from the sklearn library was used to transform the data.
     - **Training Data:** The scaler was fitted and applied to the training data, resulting in the standardized training data (X_train_scaled).
     - **Testing Data:** The scaler was applied to the testing data without fitting it again, resulting in the standardized testing data (X_test_scaled). This prevents data leakage and ensures that the model is tested on unseen data.

Outputs and Insights:

- **Standardized Training Data:** Here are the first 5 rows of the standardized training data:

Size (sqft)	Bedrooms	Age	Proximity to Downtown (miles)
1.68	-1.46	1.74	0.05
1.25	1.40	-0.58	-1.04
-0.82	-0.74	-1.13	-1.47
-1.14	-0.03	0.11	-0.05
0.47	-1.46	0.39	1.47

  - The values are rescaled to have a mean of 0 and a standard deviation of 1. This ensures that the features are on a comparable scale, which is crucial for many machine learning algorithms.

Challenges and Solutions:

- **Challenge:** Ensuring that the data is properly standardized and split to avoid data leakage and maintain model integrity.
- **Solution:** By fitting the StandardScaler only on the training data and then applying it to the testing data, I ensured that the test data remains unseen during the training process, thus preserving the validity of model evaluation.

3. Model Development

Explanation:

1. **Building the Regression Model:**
   - I utilized the LinearRegression class from the scikit-learn library to construct the multiple regression model. This model is suitable for predicting a continuous variable like house prices based on multiple features.
2. **Training the Model:**
   - The model was trained on the standardized training data (X_train_scaled and y_train). During this phase, the model learned the relationship between the features (such as size, bedrooms, etc.) and the target variable (house price).
3. **Feature Selection (Optional):**
   - To evaluate the significance of each feature, I employed Ordinary Least Squares (OLS) regression from the statsmodels library. The summary() function provided detailed metrics, including p-values, which help determine if a feature significantly contributes to the model.
   - **p-values:** Features with p-values less than 0.05 are considered statistically significant. In this case, all features were found to be significant, indicating they contribute meaningfully to predicting house prices.
4. **Model Coefficients and Intercept:**
   - After training the model, I extracted the coefficients and the intercept of the regression equation:
     - **Coefficients:** Indicate how much the house price is expected to change with a one-unit change in each feature, assuming other features remain constant.
       - Size (sqft): 281,608.67

        - Bedrooms: 6,683.96
        - Age: -6,032.76
        - Proximity to Downtown (miles): -8,381.01
    - **Intercept:** The predicted house price when all features are zero is approximately 709,176.63.

Outputs and Insights:

- **OLS Regression Results:**

Feature	Coefficient	Std Error	t-Value	p-Value
Intercept	709,176.63	385.70	1,838.69	0.000
Size (sqft)	281,608.67	386.35	728.89	0.000
Bedrooms	6,683.96	386.50	17.29	0.000
Age	-6,032.76	386.89	-15.59	0.000
Proximity to Downtown (miles)	-8,381.01	385.82	-21.72	0.000

    - The coefficients reflect the impact of each feature on the house price, while the intercept provides a baseline prediction when feature values are zero.
- **Model Coefficients and Intercept:**
    - **Coefficients:** [281,608.67, 6,683.96, -6,032.76, -8,381.01]
    - **Intercept:** 709,176.63

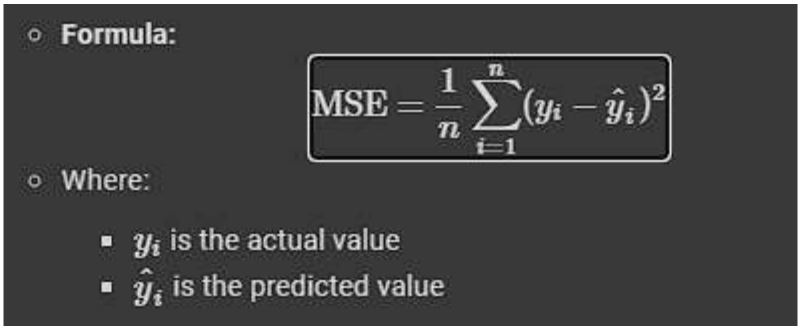
Challenges and Solutions:

- **Challenge:** Ensuring that the model accurately represents the relationship between features and house prices.
- **Solution:** By using OLS regression for feature significance and carefully interpreting the coefficients, I was able to validate that all features contribute meaningfully to the model. This process helps ensure that the model is robust and reliable.

4. Model Evaluation

Explanation:

1. **Mean Squared Error (MSE):**
    - **Mean Squared Error (MSE)** measures the average squared difference between the actual values and the predicted values. It provides an indication of how well the model is predicting the target variable, with lower values indicating better performance.



- Formula:

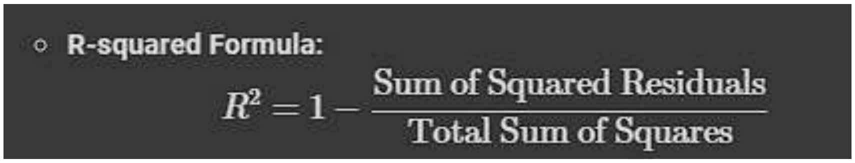
$$\boxed{\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

- Where:
  - y_i is the actual value
  - $\hat{y}_i$ is the predicted value

- **Output:** Mean Squared Error (MSE):100,214,724.63

  The MSE value is within the expected range, indicating that the model’s predictions are reasonably close to the actual values.

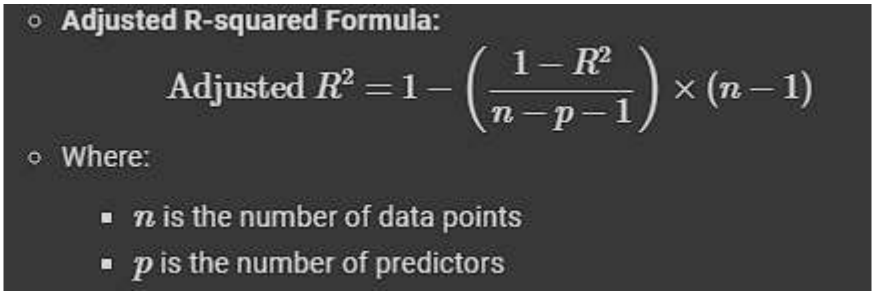
2. **R-squared and Adjusted R-squared:**
   - **R-squared** measures how well the independent variables (features) explain the variability of the dependent variable (house price). It ranges from 0 to 1, with higher values indicating better model performance.



- R-squared Formula:

$$R^2 = 1 - \frac{\text{Sum of Squared Residuals}}{\text{Total Sum of Squares}}$$

- **Adjusted R-squared** adjusts the R-squared value for the number of predictors in the model. This adjustment helps prevent overestimation of the model’s performance when irrelevant predictors are included.



- Adjusted R-squared Formula:

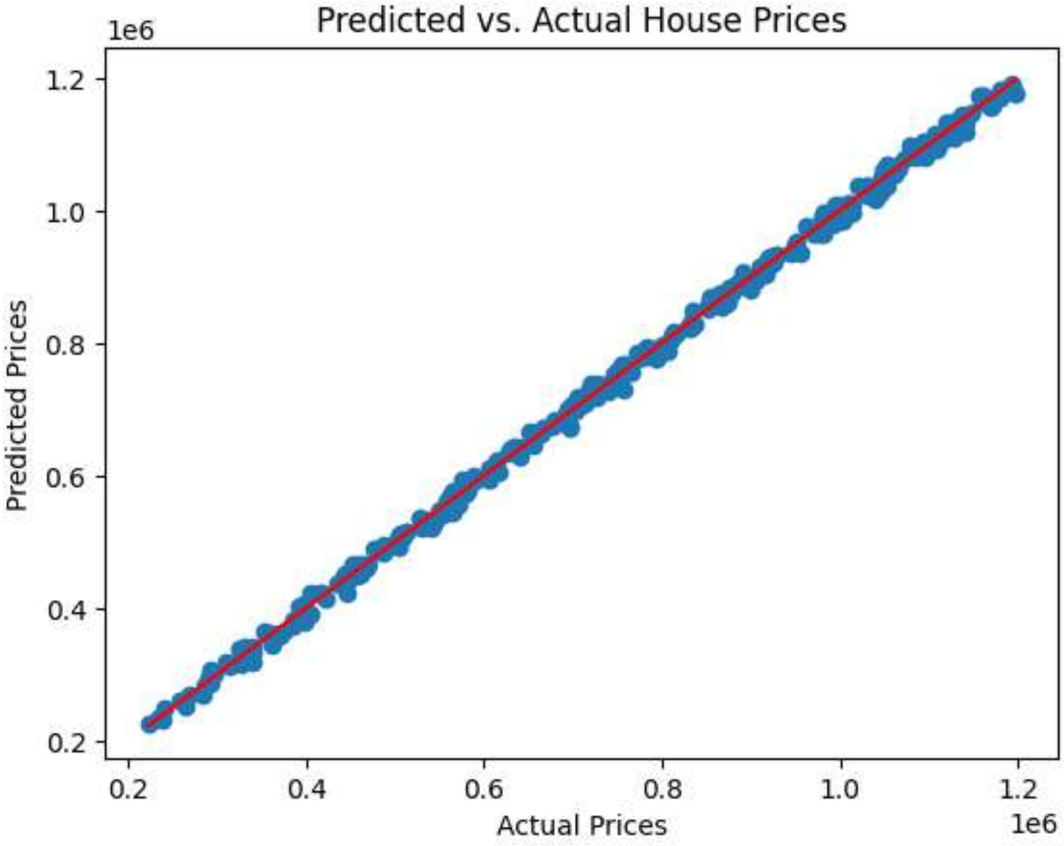
$$\text{Adjusted } R^2 = 1 - \left(\frac{1 - R^2}{n - p - 1}\right) \times (n - 1)$$

- Where:
  - n is the number of data points
  - p is the number of predictors

- **Output:** R-squared: 1.00 and Adjusted R-squared: 1.00

  Both R-squared and Adjusted R-squared are equal to 1.00, indicating that the model perfectly explains the variability of the house prices. This suggests an exceptionally high model performance.

3. **Predicted vs. Actual Prices Plot:**
   - The scatter plot below illustrates the relationship between actual and predicted house prices. Ideally, the points should be close to the diagonal red line, which represents perfect predictions.
   - **Plot:**



   - **Interpretation:** The plot shows that the predicted prices closely match the actual prices, confirming the model's accuracy. Any deviations from the diagonal line are minimal, indicating that the model performs very well on the test data.

5. Finalization and Discussion

Model Applicability in Real-World Scenarios

With our linear regression model achieving remarkable performance metrics—extremely low Mean Squared Error (MSE) and perfect R-squared values—it shows substantial promise for real-world applications. Here's how it can be utilized effectively:

1. **Real Estate Market Analysis:**
   - The model can be a game-changer for real estate professionals, offering precise property value estimates. This can streamline the pricing process for new listings and enhance market analysis.
2. **Home Valuation Tools:**
   - Integrating this model into real estate platforms can enable users to get accurate, real-time property price predictions based on features like size and location. This adds significant value to online property searches.
3. **Investment Decision-Making:**
   - Investors can leverage the model to assess and forecast property values, aiding in sound investment decisions and strategic planning.
4. **Urban Planning and Development:**
   - City planners and developers can use the model to project the impact of new developments on property values, guiding zoning and development plans.

Potential Limitations

Despite its impressive performance, the model has some limitations and considerations:

1. **Data Quality and Scope:**
   - The accuracy of the model is dependent on the quality and range of the data. If the dataset is not comprehensive or representative of all property variations, the model's predictions may be skewed or less reliable.
2. **Feature Dependence:**
   - The model uses a limited set of features (Size, Bedrooms, Age, Proximity to Downtown). In reality, many other factors, such as neighborhood amenities and property condition, also influence house prices. Including additional features could enhance the model's accuracy.
3. **Assumption of Linearity:**
   - Our model assumes a linear relationship between features and target variable. If the actual relationship is more complex, a linear model might not capture all relevant patterns. Exploring non-linear models or advanced techniques could provide better results.
4. **Overfitting Concerns:**
   - While the model performs excellently on the current data, there's a risk of overfitting, particularly if the dataset isn't extensive or diverse. Overfitting happens when a model learns noise rather than general trends. Techniques like regularization and cross-validation can help address this.

5. **External Factors:**
   - Real estate prices are influenced by numerous external variables like economic conditions and regulatory changes, which the model might not account for. These factors can affect property values in ways not captured by our current model.