1. Problem Description

This project focuses on developing a machine learning model to predict house prices using different property attributes. I selected this topic because predicting real estate prices is a common and practical example of regression in real-world applications. While the concept is straightforward, the task is made complex by the many factors that influence pricing, including location, size, and property condition. This made it a suitable problem for applying the complete machine learning workflow, from data preparation and cleaning to model training and performance evaluation.

2. Dataset

I chose the dataset from the Kaggle competition: House Prices – Advanced Regression Techniques. It contains 1,460 rows and 81 columns, including one target variable (SalePrice) and 80 features describing each house (e.g., LotArea, YearBuilt, Neighborhood, GarageType, etc.).

I started the data preparation process by handling missing values. Some columns, including PoolQC, MiscFeature, and Alley, had too many missing entries and were excluded from the dataset. For other columns with fewer missing values, like LotFrontage, I replaced the missing data with the column's average. Any remaining rows with missing entries were removed to keep the data consistent. To make the dataset compatible with the linear regression model, I used one-hot encoding to transform categorical features into numeric format.

3. Modeling Approach

I chose to use Linear Regression as the primary model for this project. Linear regression is simple, interpretable, and efficient for a continuous target variable like house price. After separating the features (X) and the target (y), I used a standard train-test split (80% training, 20% testing) to evaluate the model's performance.

Since one-hot encoding was applied to categorical columns, and the numeric features were already on relatively compatible scales, no further scaling or normalization was necessary. I trained the model using scikit-learn's LinearRegression class and fit it to the training data.

4. Evaluation

To evaluate the model's performance, I used three common regression metrics: R^2 Score, Mean Absolute Error (MAE), and Mean Squared Error (MSE). The R^2 Score was 0.851, indicating that the model explains approximately 85% of the variation in house prices. The MAE was $17,204, which means that, on average, the model's predictions differ from the actual prices by around $17,000. The MSE was 635,828,801, a value that emphasizes larger prediction errors due to squaring. Overall, these results suggest that the model performs well on unseen data, especially

considering the relatively simple preprocessing steps and the use of a basic linear regression algorithm.

5. Reflection

One of the main challenges I faced in this project was handling missing data and deciding which features were worth retaining. I also had to learn how to properly apply one-hot encoding to convert categorical variables into a suitable format for modeling. This process showed me that data cleaning and feature transformation can often have a greater impact on model performance than the choice of algorithm. In the future, I plan to experiment with more advanced techniques such as Ridge Regression, Lasso, Random Forests, and ensemble methods to compare their results. I'm also interested in exploring feature engineering by creating new variables or transforming skewed features like 'SalePrice' to enhance accuracy. Overall, this project gave me valuable hands-on experience with the complete regression workflow and a practical understanding of how machine learning is used in real-world applications.