**Project Overview:**

The House Price Prediction project aims to create a predictive model that accurately estimates house prices based on several key features. By leveraging machine learning techniques, this project seeks to provide stakeholders in the real estate market with actionable insights into pricing dynamics, helping them make informed decisions.

**Project Goal:**

The goal is to create a model capable of accurately predicting house prices based on various features, such as house age, distance to the nearest MRT station, and the number of convenience stores.  
 **Project Method:**

The project implements and compares different regression models, including:

* Linear Regression
* Decision Tree Regression
* Random Forest Regression
* Gradient Boosting Regression

The models are evaluated using the Mean Absolute Percentage Error (MAPE) metric.

**Project Outcome:**The project aims to provide actionable insights into house pricing dynamics, assisting stakeholders in making informed decisions.

**Project Structure**

**Project Steps:**

1. Dataset Loading and Exploration: Initial loading of the dataset, followed by exploration to understand its structure and key features.
2. Data Preprocessing: Handling missing values, encoding categorical variables, and feature scaling to prepare the data for modeling.
3. Exploratory Data Analysis (EDA): Visualizing the relationships between features and the target variable to identify trends and correlations.
4. Model Training: Training the selected regression models using the processed dataset.

**Dataset:**The dataset consists of various features influencing house prices, including:

* House age
* Distance to the nearest MRT station
* Number of convenience stores
* Latitude and longitude coordinates

**Technologies Used:**

* Python
* Pandas
* Scikit-learn
* Matplotlib
* Seaborn

**Importance of Feature Scaling in Machine Learning:**Feature scaling is a crucial preprocessing step in many machine learning algorithms, especially for regression models. Here’s why it is important:

* Equal Contribution: Features in a dataset can have varying units and scales. For example, in your dataset, the Distance to the nearest MRT station is measured in meters, while House age is measured in years. If these features are not scaled, the model may give undue weight to the feature with larger numerical values. Feature scaling ensures that each feature contributes equally to the model's learning process.
* Improved Convergence: Many optimization algorithms used in machine learning (like gradient descent) converge faster when the features are on a similar scale. If features vary widely, the optimization process may take longer and struggle to converge, leading to inefficient training.
* Enhanced Model Performance: Some algorithms are sensitive to the scale of the input features. For instance, models like Support Vector Machines (SVM), K-Nearest Neighbors (KNN), and Neural Networks rely on distance calculations. If the features are not scaled, the distance calculations can become biased toward features with larger scales, resulting in poor model performance.
* Better Interpretability: Scaling features can help with the interpretability of the model coefficients, especially in linear models. When features are on the same scale, it becomes easier to understand the relationship between each feature and the target variable.

**Common Scaling Techniques:**There are several techniques for feature scaling, with the most common being:

* **Standardization (Z-score normalization):** This technique transforms features to have a mean of 0 and a standard deviation of 1. It is useful when the data follows a Gaussian distribution.  
  z=(x−μ)σz = \frac{(x - \mu)}{\sigma}z=σ(x−μ)​
* **Min-Max Scaling**: This technique scales the features to a specific range, usually [0, 1]. It is useful when the distribution of the data is not Gaussian.  
  x′=(x−min(x))(max(x)−min(x))x' = \frac{(x - \text{min}(x))}{(\text{max}(x) - \text{min}(x))}x′=(max(x)−min(x))(x−min(x))​

In your project, you used StandardScaler to standardize the features. This step helps ensure that all features contribute equally to the model training and leads to more reliable and faster convergence during optimization, ultimately improving model performance

**Acknowledgments**I would like to express my gratitude to **Aman Kharwal** for the invaluable learning material that guided me throughout this project. Your insights and teachings greatly enhanced my understanding of machine learning and statistics.

For further insights and the complete project, please check the project repository here.

**References**

* **Compare Multiple Machine Learning Models**: Aman Kharwal. [The Clever Programmer](https://thecleverprogrammer.com/2024/02/19/compare-multiple-machine-learning-models/)
* **Machine Learning - Standard Deviation**: [TutorialsPoint](https://www.tutorialspoint.com/machine_learning/machine_learning_standard_deviation.htm)
* **Understanding Standard Deviation, Confidence Intervals, and Variance in Machine Learning**: Ravindran. [Medium](https://medium.com/javarevisited/understanding-standard-deviation-confidence-intervals-and-variance-in-machine-learning-97fae2a9dd21)