Real Estate Valuation Models

Shani Shaham

Department of Atmospheric and Oceanic Sciences, University of California Los Angeles

AOS C111: Introduction to Machine Learning for Physical Sciences

Dr. Alexander Lozinski

December 6, 2024

1. Introduction

Real estate valuation, also referred to as property or land appraisal, is a critical process in the real estate industry. It involves estimating the value of properties based on various factors, including economic conditions, physical attributes, and location. Accurate valuation plays an essential role for stakeholders such as buyers, sellers, investors, lenders, and policymakers. For buyers and sellers, it ensures fair pricing in transactions, while for investors, it helps assess the profitability of real estate investments. Mortgage lenders rely on property valuations to determine loan amounts and evaluate risk, while governments and urban planners use them to guide taxation, zoning, and infrastructure development decisions.

However, real estate valuation is a challenging task due to the dynamic nature of the market and the unique nature of each listing. Market conditions, influenced by demand, supply, interest rates, and economic trends, can cause significant fluctuations in property values. Location is another critical factor, as the proximity to schools, public transport, shopping centers, and neighborhood safety all heavily impact valuation. Additionally, no two properties are identical, with variations in size, layout, construction quality, and features further complicating the process. The availability of reliable and up-to-date data is also a challenge, as external factors such as infrastructure projects, economic downturns, and zoning law changes can have unpredictable effects on property prices.

Machine learning techniques present powerful tools for addressing the challenges in real estate valuation. Arthur Samuels, widely considered the father of the field, first defined machine learning in 1959 as “the field of study that gives computers the ability to learn without being explicitly programmed.” It involves the development of algorithms and statistical models that enable computers to identify patterns, make decisions, or predict outcomes based on input data. Machine learning is categorized into several types, including supervised learning, where the model is trained on labeled data, and unsupervised learning, where the model analyzes unlabeled data. Machine learning has a wide range of applications across multiple domains, including but not limited to disease diagnostics, natural language processing, fraud detection, and autonomous vehicles.

2. Data

For the purposes of this project, I will utilize a dataset found on the UC Irvine Machine Learning Repository that contains market historical data of real estate valuation collection from Sindian Dist., New Taipei City, Taiwan between 2012 and 2013:
https://archive.ics.uci.edu/dataset/477/real+estate+valuation+data+set

There are 414 observations, putting this dataset on the smaller side. There are no missing values. There are six features, along with the target variable of house price per unit area.

Here is a description of the data in each column:

- `X1` : transaction date
- `X2` : house age in years
- `X3` : distance to the nearest MRT station in meters
- `X4` : number of convenience stores in the living circle on foot
- `X5` : geographic coordinate, latitude
- `X6` : geographic coordinate, longitude
- `Y` : house price of unit area in 10000 New Taiwan Dollar/Ping, where Ping is a local unit, 1 Ping = 3.3 meter squared

To prepare the dataset for modeling, I first installed the `ucimlrepo` Python package in a Google Colab notebook. This package is designed to easily download datasets from the UCI Machine Learning Repository. After downloading, I was able to clean and prepare the dataset, while also performing some initial exploratory analysis to better understand the nature of the data.

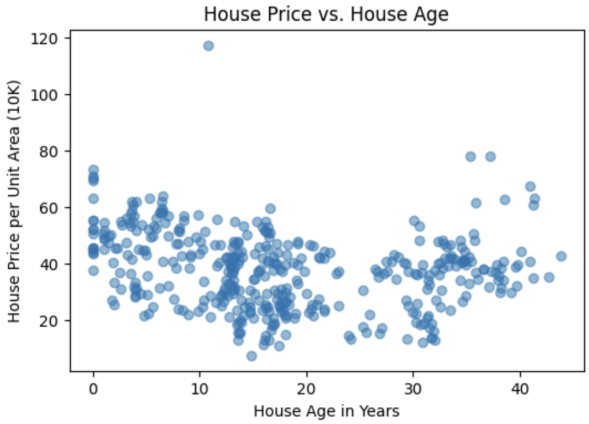

Figure 1: This scatter plot shows the non-linear relationship between features and target variables, indicating that linear regression type models may not perform strongly on this dataset.

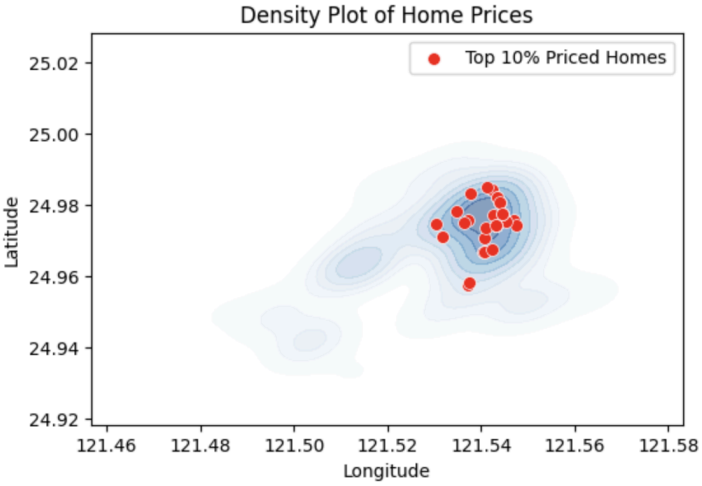

Figure 2: This visualization explores the relationship between price and location. It plots home prices based on their geographic location, highlighting the areas where the top 10% most expensive homes are located.

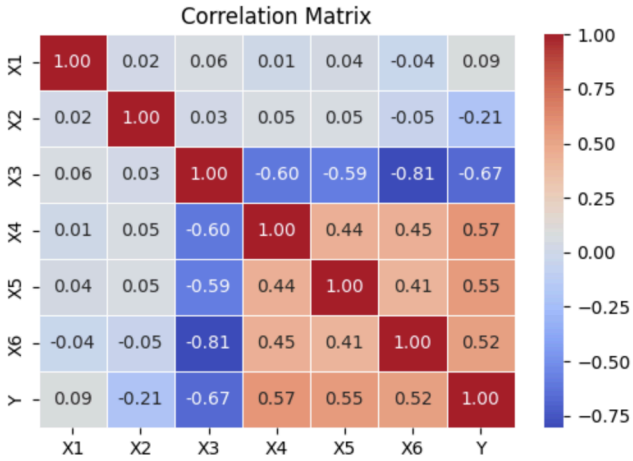

Figure 3: This heatmap analyzes the correlation between different features in the dataset. It can be interpreted that X4, X5, and X6 are strong predictors of our target variable.

3. Modeling

Based on the interpretation of the first figure, the dataset exhibits a seemingly nonlinear relationship between features and targets, so I decided to compare various tree based regression models, which typically better capture relationships between nonlinear variables. Tree based models are a class of supervised learning algorithms that construct decision trees. Decision trees are flow-chart like structures that offer decision-making processes similar to human reasoning. In a decision tree, each branch represents a decision based on features, and leaves represent the final prediction. These algorithms begin by splitting the dataset by important features based on a specific measure. Typically in classification models, decision trees use Gini impurity (measure of the probability of misclassification in random instances) or entropy (measure of information uncertainty in the dataset) as the splitting criterion. For the purposes of a regression task, the splitting criteria is MSE, where the algorithm selects the decision that minimizes the mean squared error in the child nodes (ie. resulting subsets of the data after a split). The splitting process is repeated for each resulting subset until a predefined stopping criteria is met.

As the tree grows, each leaf is given a predicted outcome based on the mean of the target values in the resulting dataset. This allows the algorithm to approximate relationships between features and capture nonlinear patterns in the dataset. The strengths of decision trees in modeling include interpretability, efficiency, and non-parametric nature. Some disadvantages of this approach include high variance and unstableness, greediness, and proneness to overfitting. Ensemble learning methods combine predictions from multiple trained decision trees to create a more accurate and stable model. These methods leverage the strengths of single decision trees while reducing overfitting, increasing generalizability, and improving overall performance.

For the purpose of this project, I decided to focus on four different tree based regression models, Decision Tree Regression, Random Forest Regression, Gradient Boosting Regression, and Extreme Gradient Boosting Regression. Decision Tree Regressions follows the outline for decision tree models described above. The remaining model types are ensemble methods. In Random Forest Regression, each individual decision tree is trained independently on random subsets of the data, which helps to reduce overfitting. In both the Gradient Boosting type algorithms, each tree is trained sequentially to improve the performance of the previous tree. Gradient Boosting Regression uses gradient descent to minimize the mean squared error. Extreme Gradient Boosting ("XGBoost") enhances traditional Gradient Boosting to create a faster and more robust model. XGBoost includes built-in regularization techniques (Lasso and Ridge), something that the traditional Gradient Boosting model lacks.

For each of these models, I split the dataset into train and test sets using sci-kit learn's `train_test_split` function from the `model_selection` module. In order to optimize each model, I performed a grid search to find the best combination of hypermaters that would yield the highest accuracy score. I utilized the `GridSearchCV` tool from the `sklearn` library to automate this process. This way, I was able to find the 'best' version of each of these models for my dataset. Additionally, I performed 5-fold cross validation on each of these 'best' model versions to analyze the models generalizability on unseen data.

4. Results

Figure 4: Table displaying results of grid search and 5-fold cross validation

Model	Best Hyperparameters	Accuracy Score	Cross-Validation Mean RMSE
Decision Tree	`max_depth` = 5 `max_features` = None `max_leaf_nodes` = 50 `min_samples_leaf` = 5 `min_samples_split` = 2	69.10%	8.29
Random Forest	`max_depth` = 20 `max_features` = 'sqrt' `n_estimators` = 200 `min_samples_leaf` = 2 `min_samples_split` = 2	83.17%	5.31
Gradient Boosting	`learning_rate` = 0.01 `max_depth` = 7 `max_features` = 'sqrt' `n_estimators` = 300 `min_samples_leaf` = 4 `min_samples_split` = 10 `subsample` = 0.8	83.15%	7.08
XGB	`learning_rate` = 0.1 `max_depth` = 2 `min_child_weight` = 5 `n_estimators` = 75 `reg_alpha` = 0.1 `reg_lambda` = 2	79.29%	7.32

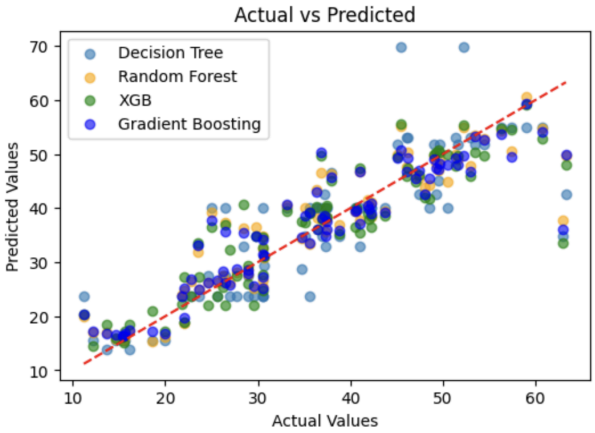

Figure 5: Comparing actual vs predicted values.

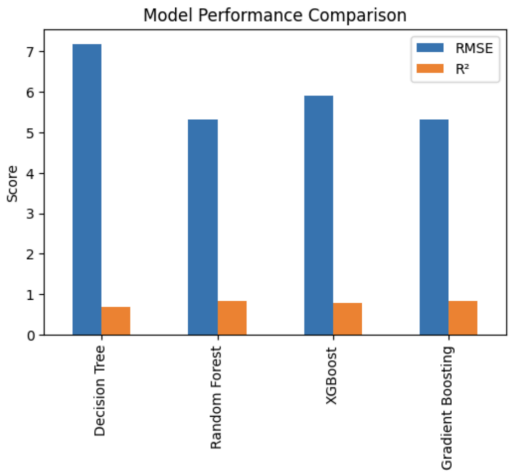

Figure 6: Comparing RMSE and R^2 scores.

Points closer to the red line represent better predictions.

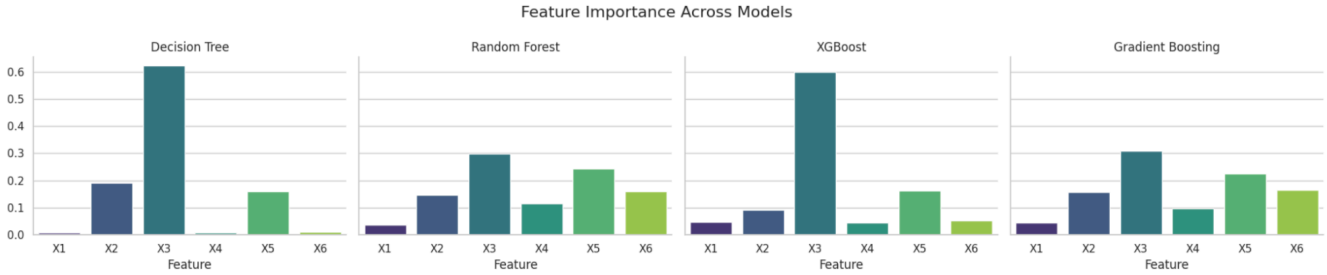

Figure 7: Comparing feature importance of each model.

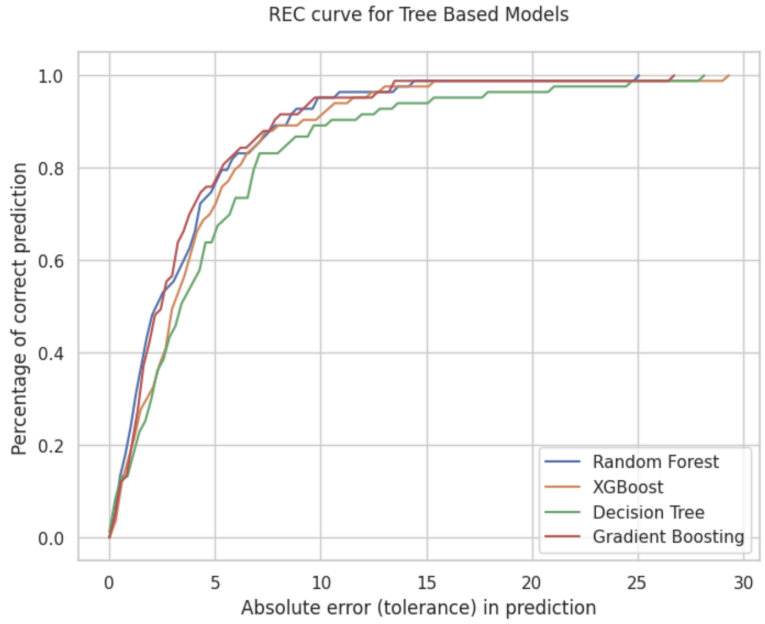

Figure 8: Comparing REC curves of each model.

5. Discussion

As presented in figures 4, 5, and 6 above, the Random Forest model flaunts the strongest performance against the test data, with the highest accuracy score of 83.17% and the lowest mean root mean square error ("RMSE") of 5.31. Gradient Boosting and XGBoost display similar accuracy scores of 83.15% and 79.29%, but fall short with a higher RMSE of 7.08 and 7.32, meaning these models are not as generalizable to unseen data when compared to the Random Forest model. Lastly, the Decision Tree regression model performed the worst, with the highest RMSE and the lowest accuracy score, indicating lack of generalization and overfitting. Thus, it is clear that predictions on this dataset were enhanced with ensemble techniques rather than individual trees. It may even be inferred that the models benefited from bagging ensemble methods (individually trained trees) instead of boosting ensemble methods (sequentially trained trees).

The Regression Error Characteristic ("REC") Curve are visualizations that aid in model interpretation and performance evaluation. The curve represents how quickly each model achieves higher accuracy scores as the error tolerance increases. An ideal REC curve displays a steep initial rise at very low error tolerance (ϵ), that plateaus into a flat line at $F(\epsilon)=1$. This ideal curve is rare, typically only seen for nearly perfect models. A more common REC curve is one that rises steeply and flattens early, indicating that the model has low errors for most predictions. In figure 8, Random Forest and Gradient Boosting REC curves outperform XGBoost and Decision Tree Regression curves across varying error tolerances, further suggesting that Random Forest and Gradient Boosting Regression are the strongest models for our dataset.

Feature importance is a metric that quantifies the relative importance of input features in determining the model's output, helping to identify the most influential variables in the model. In figure 7, it is understood that feature importance varies across each model. Decision trees and XGBoost rely predominantly on X3, distance to the nearest public transportation station, in determining home price. This may indicate that these models are overfitting to this feature, without strong enough consideration to the contributions of other input variables. Random Forest and Gradient Boosting balance importance across X3, X5 (latitude), and X6 (longitude) more evenly, indicating robust feature interaction which helps to prevent overfitting.

A larger dataset may aid the ensemble method models in better pattern recognition within the data, leading to stronger model performance. More diverse data would also help to reduce overfitting and generalizability of the models. It is worth noting that the features in this dataset are rather simple, and do not fully capture the nuance that goes into proper real estate valuation. Some metrics that may aid in better predictive power for a regression model include, but are not limited to: home size, historical sale data, neighborhood safety levels, most recent renovations, schools in vicinity, and any important recent housing market events. That being said, these four tree-based regression models were still able to adequately predict house price in a supervised learning environment.

6. Conclusion

From this study, the following conclusions can be drawn:

a. This dataset benefitted from ensemble learning methods.
b. Random Forests worked best to implement a predictive model.
c. Distance to the nearest transportation station, latitude, and longitude were the most important features.
d. Gradient Boosting and Extreme Gradient Boosting models performed well, but may have over complicated the model when sequentially training each tree, leading to higher error metrics.
e. The Decision Tree model was prone to overfitting and lack of generalization.