# TEAM 06

# PROJECT REPORT

# USA Housing Listings

Homes for sale within the United States

**Report by**:

Chiranjeet Banerjee

Riya Singh

**Appendix**

Introduction………………………………………………………………………………....3

Description of Dataset………………………………………………….…………………...4

Methodology…………………………………………………………...…………………...5

Data cleaning and Pre-processing……………………………………...…………………...5

Exploratory Data Analysis……………………………………………..…………………...8

Feature Engineering………………………………………………………………………..12

Model Selection and Evaluation…..……………………………………………………….13

Result and analysis…………………………………………………………………………16

Conclusion……………………………………………………………..…………………...17

Summary and Future Work…………………………………………………………………18

References……………………………………………………………..……………………19

**Introduction**

**Background:**

The United States housing market is a complex ecosystem influenced by various factors such as economic conditions, demographics, and regional trends. While there is abundant data available on housing prices and characteristics in specific urban areas, there is a need for a comprehensive dataset that captures the housing landscape across the entire country. This dataset serves as a valuable resource for understanding housing dynamics and trends at a granular level.

**Motivation:**

This project is driven by the imperative to harness the vast and varied housing data available on Craigslist to address critical challenges in the real estate market. By consolidating disparate housing listings into a comprehensive dataset, we can overcome the limitations of fragmented data sources and unlock deeper insights into housing dynamics. The rich information provided by Craigslist offers a unique opportunity to gain a holistic understanding of housing trends.

Overall, we aim to make a meaningful contribution to provide better insights for the public to be able to find house listings as a whole instead of isolated urban housing markets. We also aim to find the broader trends and patterns in the housing market which can provide valuable insights for policymakers, real estate professionals, and investors.

**Goal:**

The goal of this project is to leverage the extensive dataset from Craigslist to develop predictive models for housing prices. By applying machine learning algorithms to this dataset, we aim to extract valuable insights that can enhance our understanding of housing market dynamics across different states. Ultimately, the predictive models derived from this analysis will provide stakeholders with powerful tools to forecast housing prices accurately, enabling them to make more informed decisions in buying, selling, and investing in real estate properties and help the public in making right decisions on buying a desired house for themselves for the right price.

### **Description of the dataset**

### The dataset consists of data that is scraped and contains most of the relevant information that Craigslist provides on retail sales. This dataset includes different attributes which have been categorized in a way which defines the house listings based on region, sq feet , parking options etc.

The size of our dataset is 558.44 MB

**Columns:**

1. Id: listing id
2. URL: listing URL
3. region: craigslist region
4. region\_url: region URL
5. price: rent per month (Target Column)
6. type: housing type
7. sqfeet: total square footage
8. beds: number of beds
9. baths: number of bathrooms
10. cats\_allowed: cats allowed boolean (1 = yes, 0 = no)
11. dogs\_allowed: dogs allowed boolean
12. smoking\_allowed: smoking allowed boolean
13. wheelchair\_access: has wheelchair access boolean
14. electric\_vehicle\_charge: has electric vehicle charger boolean
15. comes\_furnished: comes with furniture boolean
16. laundry\_options: laundry options available
17. parking\_options: parking options available
18. image\_url: image URL
19. description: description by poster
20. lat: latitude
21. long: longitude
22. state: state of listing

Datasource:https://www.kaggle.com/datasets/austinreese/usa-housing-listings/data

**Methodology**

**Data Preprocessing and Cleaning**

Data preprocessing and cleaning are essential steps in data analysis and machine learning workflows, aiming to ensure data accuracy, completeness, and consistency. Through the use of Pandas and NumPy functions, we calculate summary statistics and address missing values, outliers, and inconsistencies to enhance data quality. This process involves transforming raw data into a refined and standardized format suitable for subsequent tasks such as modeling, visualization, or exploration. Key steps in data preprocessing and cleaning include handling missing values, detecting and treating outliers, performing data transformation, feature engineering, integrating data from multiple sources, normalization and standardization, data reduction, and data splitting for training and validation purposes. These steps collectively enable analysts and machine learning practitioners to generate reliable insights and build robust models.

**Why is Preprocessing Important?**

Imagine feeding a messy pile of ingredients to a chef. It would be difficult, if not impossible, to create a delicious meal. Similarly, feeding an uncleaned dataset to a machine learning model leads to unreliable results. Here's how preprocessing helps:

* **Improves Accuracy:** By handling missing values and inconsistencies, you remove noise that could mislead your model.
* **Boosts Performance:** Cleaning outliers and scaling features creates a more consistent training environment, leading to better model performance.
* **Enables Efficient Modeling:** Preprocessed data allows algorithms to run smoother and converge faster.

**Data Exploration and Understanding:**

* + We used Pandas functions like info() and head() to get an initial overview of the data (data types, missing values etc.).

Understanding the data

A screenshot of a computer

Description automatically generated

When we execute df.info() on a DataFrame df, it prints out information about the DataFrame, including:

**Observation:**

1. **Range Index**: The range of row labels i.e. 384977.
2. **Data columns**: The column names and data types of each column.
3. **Memory usage**: The total memory usage of the DataFrame i.e. 64.6+ MB.
4. **Non-null counts**: The count of non-null values for each column.

This method is particularly useful for getting a quick overview of the DataFrame's structure, data types, and memory usage, especially when working with large datasets.

After getting an overview of the dataset we can now deep dive into understanding the data in order to get our target variable estimation which is our ultimate goal.

We will now proceed with the methodologies.

In order to preprocess and clean the raw data we do the following:

1. **Missing Values**:

The output we got shows that some columns have fewer non-null values than others. For example, the laundry\_options column has 305,951 non-null values, and the parking\_options column has 244,290 non-null values out of 384,977 total rows. Similarly, the latitude and longitude also have 1918 missing values respectively so we removed the null values similarly.

These columns likely have missing values that need to be addressed through imputation, or other techniques depending on the analysis goals and missing data patterns.

A graph of a bar chart

Description automatically generated with medium confidence

The result is a pandas Series, where each value represents the percentage of null values in the corresponding column of the original DataFrame df.

**Observation**: From this we can deduce that laundry\_options has ~20% null values followed by parking\_options which has ~37% of null values.

1. **Outlier Detection**

In a house price prediction regression analysis, detecting and handling outliers in features such as the number of bedrooms, bathrooms, and square footage is crucial. Outliers can significantly impact the accuracy and reliability of the regression model, as they can distort the true relationship between the predictor variables and the target variable (house price).

It's essential to carefully examine and understand the context of the outliers before deciding on a course of action. In some cases, outliers may represent legitimate data points, and removing or adjusting them could lead to information loss or biased results. Therefore, a thorough understanding of the domain knowledge and the impact of outliers on the regression model's performance is crucial for making informed decisions.

**Observation:** In our analysis for regression, we've identified outliers in the continuous variables like beds, baths, and square feet. These outliers have the potential to skew our model's performance and compromise the accuracy of our predictions. To mitigate this, we'll implement outlier detection techniques tailored for regression analysis. By identifying and removing these outliers, we aim to refine our dataset and improve the robustness of our regression model.

1. **Removing Duplicates and irrelevant columns**:

Duplicate entries in the dataset can arise from various sources, including data entry mistakes, merging datasets, or multiple listings of the same property. These duplicates can introduce bias into the model training process and inflate the importance of certain observations.

To address duplicates in the dataset, comparisons can be made based on unique identifiers such as property addresses or listing IDs. Records sharing identical identifiers are considered duplicates and can be safely removed from the dataset, ensuring each property is represented only once.

**Observation:** Since this is a regression analysis we have observed that some columns which have text data do not have a significant role to play for our target variable prediction so would be dropping these columns "id", "url", "region\_url", "image\_url","description".

**Now that we've addressed missing values, eliminated duplicates, and removed irrelevant columns, our dataset is considered clean. This sets a solid foundation for us to delve into data visualization, allowing us to gain a deeper understanding of the data's underlying patterns and relationships**.

**Exploratory Data Analysis (EDA)**

This step would involve visually and statistically exploring the dataset to gain insights into its structure, patterns, and relationships between variables. Techniques include summary statistics, data visualization (e.g., histograms, scatter plots, box plots), and correlation analysis.

For a house price regression prediction task, Exploratory Data Analysis (EDA) is a crucial step to understand the characteristics of the data and identify potential patterns, relationships, and issues that may impact the model's performance. Based on the provided columns, here's a short paragraph on EDA for this dataset:

EDA should focus on examining the distribution and relationships among numerical features like 'price', 'sqfeet', 'beds', 'baths', 'lat', and 'long'. Visualizations such as histograms, box plots, and scatter plots can reveal skewness, outliers, and correlations. For categorical features like 'type', 'region', 'laundry\_options', 'parking\_options', and 'state', bar plots and countplots can explore their distributions and potential impact on house prices. Additionally, analyzing the presence of missing values in columns like 'laundry\_options', 'parking\_options', 'lat', and 'long' is essential to determine appropriate handling techniques. Pairwise scatter plots and correlation matrices can uncover multicollinearity among numerical features. EDA should also investigate potential interactions between categorical and numerical variables through techniques like grouped box plots or violin plots.

**Categorical Variables:**

In categorical variable EDA, we analyze distribution and relationships. Using visualizations like bar plots, we explore frequency distribution, identify outliers, and understand patterns. Cross-tabulation and heatmaps reveal interactions between categories. Bar plots and box plots help assess the relationship with the target variable. We address missing values, outliers, and consider variable transformations. Statistical tests determine significance, guiding subsequent analysis. These insights inform feature engineering and model building.

Top of Form

**Relation between Relatype and price?**

According to the trends and knowledge we have we know that for a house to be predicted the type of the house is very crucial so we tried to see the relationship between price and type and get a general idea about it.

Since house prices depend on what type of house it is we visualize it using a count plot.

A graph with blue squares

Description automatically generated

**Analysis**: This bar plot provides a visual representation of how the property type influences the price in the given dataset. It allows for easy comparison of average prices across different property types and can help identify potential trends or patterns in the data.

**Effect of parking options on house price prediction?**

For predicting a house we need to have a general idea about how are the parking options as per analysis we know that almost every household in the US have a transportation and while choosing a house parking option is equally important.

**A graph of parking options

Description automatically generated**

**Analysis:** Properties with "valet parking" tend to have higher median prices and wider price ranges compared to those with "no parking" or "street parking." Similarly, properties with "attached garage" or "detached garage" also exhibit higher median prices and wider price ranges compared to those with "off-street parking" or "carport." These observations suggest that parking amenities such as valet parking or garage types are associated with higher property prices. Box plots visually compare price distributions across different parking options, with higher plots indicating higher prices and wider ranges.

We can deduce from the analysis that parking options has a role to play when it comes to predicting the house prices.

How are cats allowed and dogs allowed values effecting the price?

Cats Allowed Dogs Allowed

**A graph with blue squares

Description automatically generated**

**A graph with blue squares

Description automatically generated**

**Analysis:** We can observe from the above figure that the prices of the house are getting higher if the dogs and cats are allowed and tend to both tend to show a linear relationship with the prices.

**Effect of laundry options on price?**

Here we are trying to understand the text data laundry options and trying to understand how the categories in laundry options influence the price.

**A graph of a laundry

Description automatically generated with medium confidence**

**Analysis:** We can observe w/d hookups has a little effect on price and and the houses that contain them have more price compared to other values in the data.

**Continuous variables**

**Effects of beds on house price**

**A diagram of a graph

Description automatically generated with medium confidence**

**Observation:** As the number of bedrooms increases, the median price also rises, indicating a trend of higher prices for properties with more bedrooms. Additionally, the interquartile range (IQR), represented by the height of the boxes, widens with an increasing number of bedrooms, suggesting a broader spread of prices. Furthermore, the presence of outliers beyond the whiskers highlights a significant range of prices, particularly in the 3- and 4-bedroom categories, including some very high-priced properties

**Relation between sqfeet and price**

**A blue and white dotted diagram

Description automatically generatedObservation**: In the scatter plot, we observe a positive trend between square footage (sqfeet) and price, indicating that as square footage increases, prices tend to rise. However, the spread of data points around the trendline suggests variability in prices for properties with similar square footage, indicating that the correlation may not be strong enough for accurate price predictions based solely on square footage. Other factors such as location, amenities, and property condition likely also influence pricing decisions. Therefore, while square footage plays a role in determining price, it is not the sole determinant, and considering additional variables may be necessary for accurate predictions.

**Understanding the dataset using heatmap.**

**Observation:Top of Form**

A screenshot of a graph

Description automatically generatedlksdcnlkdnottom of Form

No strong correlations: There are no dark red or dark blue squares in the off-diagonal elements, which would indicate strong positive or strong negative correlations, respectively. This suggests that there are no extremely high correlations among the variables in the dataset.

Based on the lack of dark red or dark blue squares in the off-diagonal elements, we can infer that there is no evidence of severe **multicollinearity** in this dataset. The absence of strong correlations between pairs of variables suggests that the independent variables are relatively uncorrelated with each other, which is a desirable condition for regression analysis and other statistical modeling techniques.

**Hence now that we have analyzed the dataset, understood their relationships and correlations with target variable, how these values are affecting the house predictions and their relationship among themselves we can now further proceed to do feature engineering of the dataset.**

**Feature Engineering**

Now we would be transforming existing features to improve the performance of machine learning models. We will compare many different columns like sq feet, price, region, and state to see which when combined will help us predict the house listings more accurately. The goal is to extract relevant information from the raw data and create input features that enhance the predictive power of the models.

A screenshot of a computer

Description automatically generatedAfter cleaning the dataset and understanding the data we will now prepare the dataset in a way so that when we run the dataset under a model we would able to get the best results.

* **Combining columns**

In the EDA step we analyzed that the cats allowed and dogs allowed have somewhat similar effect on the house price and since they look similar we can combine the cats\_allowed and dogs\_allowed and drop the existing columns.

* **Removing noisy data and outliers**

By restricting the number of bedrooms and bathrooms to reasonable values (less than or equal to 4 and less than 7.5, respectively), as well as limiting the price and square footage to sensible ranges (less than $5000 and 4000 square feet, respectively), the code aims to eliminate extreme observations that could distort the predictive model. This meticulous filtering process not only enhances the dataset's accuracy but also facilitates the creation of a more robust and dependable regression model for house price prediction.

* **Dropping similar columns**

In the context of feature engineering for house price prediction, the removal of the "state" column from the dataset can be framed as a process aimed at optimizing feature selection. By dropping the "state" variable, which is similar to the "region" feature, we aim to reduce redundancy and multicollinearity within the dataset. This step ensures that we retain only the most informative and relevant predictors while minimizing the risk of overfitting and model complexity. Ultimately, by streamlining the feature space and focusing on essential predictors such as location-related features, we enhance the predictive power and interpretability of the regression model, thereby improving its performance in accurately predicting house prices.

* **Label encoding**

Label encoding is a pivotal aspect of feature engineering, particularly for preparing categorical variables for machine learning models. By converting categorical data, such as "state" and "type", into numerical format, label encoding enables seamless integration of these features into the model's training process. It assigns unique numerical labels to each category, preserving their ordinality and facilitating interpretation. This transformation enhances model performance by allowing it to effectively learn from categorical information and capture underlying patterns in the data. Overall, label encoding plays a crucial role in enriching the feature space and optimizing the model's predictive capabilities.

**After thoroughly understanding the dataset and optimizing it based on our insights, we have refined the data to ensure its relevance and effectiveness for further analysis. This involved eliminating redundant information, removing duplicate columns, and converting categorical variables into numerical representations to facilitate efficient modeling. With these preprocessing steps completed, we are now prepared to apply machine learning algorithms to the dataset, leveraging its enhanced quality to derive meaningful insights and make accurate predictions.**

**Model Selection and Evaluation**

Finally, we would be choosing appropriate machine learning algorithms for the predictive task based on the dataset characteristics and problem requirements. Techniques include evaluating various algorithms and selecting the best-performing ones based on performance metrics and find which algorithm would give the best scores that would help us predict the house prices that are listed in the dataset.

Each of these methodologies plays a crucial role in the predictive modeling process, from data preparation and exploration to feature engineering and model selection. By systematically applying these techniques, we can develop accurate and reliable predictive models for housing prices using the provided dataset.

**Linear Regression**

Linear regression is a fundamental statistical technique widely employed in modeling datasets to predict house prices. Leveraging the linear relationship between predictor variables and the target variable, linear regression aims to estimate the coefficients of these variables to form a linear equation that best fits the data. In the context of house price prediction, predictor variables typically include features like square footage, number of bedrooms and bathrooms, location factors, and amenities. By fitting a linear regression model to historical housing data, the algorithm learns the relationships between these features and house prices, enabling it to make predictions for new instances. Additionally, techniques such as feature scaling, regularization, and feature selection are often applied to enhance model performance and mitigate overfitting. Overall, linear regression serves as a foundational tool in the realm of real estate analytics, providing valuable insights into the factors influencing house prices and aiding in informed decision-making processes.

|  |  |
| --- | --- |
| R^2 Score | 0.24269092005494675 |
| MSE Score | 225320.22902871412 |
| RMSE Score | 474.67908004115174 |

**Model Analysis:**

The linear regression model achieves an R-squared score of 0.24, indicating that it explains approximately 24% of the variance in house prices. While this demonstrates some predictive capability, there's room for improvement. The Mean Squared Error (MSE) is 225320.23, suggesting the model's predictions deviate from actual prices. The Root Mean Squared Error (RMSE) of 474.68 quantifies this deviation. Despite these metrics, further refinement and feature exploration may enhance predictive accuracy.

**XGBoost (Extreme Gradient Boosting)**

XGBoost (Extreme Gradient Boosting) regression is a powerful machine learning technique widely used in house price regression analysis. By leveraging the ensemble learning approach, XGBoost combines multiple weak learners to create a robust predictive model. In the context of house price prediction, XGBoost regression can effectively capture complex relationships between various features such as square footage, number of bedrooms, location, and amenities, leading to accurate price estimates. Its ability to handle large datasets efficiently and its flexibility in tuning parameters make XGBoost regression a popular choice for real estate analysis. With its superior performance and interpretability, XGBoost regression is well-suited for identifying key factors influencing house prices and providing valuable insights for property valuation and investment decisions.

|  |  |
| --- | --- |
| R^2 Score | 0.8591917250216776 |
| MSE Score | 41894.32506151345 |
| RMSE Score | 204.68103249083305 |

**Model analysis:**

The XGBoost model achieves exceptional performance in house price prediction, boasting an impressive R-squared score of 0.86. With only a small Mean Squared Error (MSE) of 41894.33 and Root Mean Squared Error (RMSE) of 204.68, it provides highly accurate predictions, outperforming linear regression. These results signify the model's ability to effectively capture underlying patterns in the data, offering valuable insights for real estate valuation and investment decisions.

**Gradient Boosting Regressor**

GradientBoostingRegressor is a robust machine learning algorithm frequently employed in house price prediction regression analysis. It operates by iteratively training weak learners, typically decision trees, to correct errors made by preceding models. This iterative process focuses on minimizing the residual errors, gradually improving the predictive accuracy. In the realm of house price prediction, GradientBoostingRegressor excels in capturing intricate relationships among various features such as property size, location, amenities, and market trends. By effectively handling nonlinear relationships and complex interactions between predictors, GradientBoostingRegressor produces accurate predictions and delivers valuable insights into the factors influencing house prices. Its adaptability to different types of data and its capability to handle large datasets make GradientBoostingRegressor a favored choice for real estate professionals and data scientists alike, facilitating informed decision-making in property valuation and investment strategies.

|  |  |
| --- | --- |
| R^2 Score | 0.8641147179785857 |
| MSE Score | 40429.59958821296 |
| RMSE Score | 201.07113066826116 |

**Model analysis:**

The Gradient Boosting Regressor excels in predicting house prices, boasting an impressive R-squared score of 0.86. With a low Mean Squared Error (MSE) of 40429.60 and Root Mean Squared Error (RMSE) of 201.07, it provides precise predictions, making it valuable for real estate valuation. These results underscore its effectiveness in accurately estimating property values, empowering informed investment decisions in the real estate industry.

**Top of FormRandomForestRegressor**

RandomForestRegressor is a versatile machine learning algorithm widely used in house price prediction regression analysis. It operates by building an ensemble of decision trees, where each tree learns from a random subset of the data and features. This randomness helps reduce overfitting and improves the model's generalization performance. In the context of house price prediction, RandomForestRegressor excels in capturing complex interactions between various predictors such as property characteristics, location, amenities, and economic factors. By aggregating predictions from multiple trees, RandomForestRegressor delivers robust and accurate price estimates, making it well-suited for real estate valuation and investment decisions. Its ability to handle both numerical and categorical data, as well as its resistance to outliers, contributes to its popularity among data scientists and practitioners in the real estate industry.

|  |  |
| --- | --- |
| R^2 Score | 0.8870564493859879 |
| MSE Score | 33603.805058711005 |
| RMSE Score | 183.31340665295326 |

**Model Analysis:**

The Random Forest Regressor performs admirably in predicting house prices, boasting an impressive R-squared score of 0.89. With a low Mean Squared Error (MSE) of 33603.81 and Root Mean Squared Error (RMSE) of 183.31, it delivers precise predictions, making it invaluable for real estate valuation. These results underscore its accuracy and effectiveness in estimating property values, empowering informed decision-making in the real estate industry.

**Result and final analysis:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Metric** | **LR** | **XGB** | **GBR** | **RAN** |
| R^2 | 0.242691 | 0.799807 | 0.864784 | 0.886449 |
| MSE | 225320.229029 | 59562.970408 | 40230.434983 | 33784.583432 |
| RMSE | 474.679080 | 244.055261 | 200.575260 | 183.805831 |

The table presents the evaluation metrics for four different models (LR, XGB, GBR, and RAN) on a particular dataset or task. The metrics shown are R^2 (coefficient of determination), MSE (Mean Squared Error), and RMSE (Root Mean Squared Error).

R^2: This metric measures the proportion of the variance in the dependent variable that can be explained by the independent variables in the model. A higher value of R^2, closer to 1, indicates a better fit of the model to the data.

MSE: This metric measures the average squared difference between the predicted values and the actual values. A lower value of MSE indicates a better model performance, as it means the predictions are closer to the true values.

RMSE: This metric is the square root of the MSE, and it represents the standard deviation of the residuals (prediction errors). A lower value of RMSE is desirable, as it indicates smaller prediction errors.

From the table, we can make the following observations:

1. The RAN model has the highest R^2 value of 0.886449, indicating that it explains approximately 88.8% of the variance in the dependent variable, which is the best among the four models.

2. The RAN model also has the lowest MSE of 33784.583432 and the lowest RMSE of 183.805831, suggesting that it has the smallest prediction errors and the best overall performance.

3. The GBR model has the second-best performance, with an R^2 of 0.864784, MSE of 40230.434983, and RMSE of 200.575260.

4. The XGB model performs better than the LR model, with an R^2 of 0.799807, MSE of 59562.970408, and RMSE of 244.055261.

5. The LR model has the poorest performance, with the lowest R^2 of 0.242691, the highest MSE of 225320.229029, and the highest RMSE of 474.679080.

**Conclusion:**

Based on the evaluation metrics presented in the table, the RAN (Random Forest or Random Forest) model appears to be the best-performing model for this dataset or task, as it has the highest R^2, the lowest MSE, and the lowest RMSE. The GBR (Gradient Boosting Regression) model also performs reasonably well, followed by the XGB (Extreme Gradient Boosting) model. The LR (Linear Regression) model seems to be the least suitable for this particular dataset or task, as it has the worst performance across all the metrics. Therefore, for this specific problem, it is recommended to use the RAN model or, alternatively, the GBR model, as they demonstrate superior predictive performance compared to the other models evaluated.

**Summary**

In this project, we conducted a thorough regression analysis to predict house prices using Linear Regression (LR), Extreme Gradient Boosting (XGB), Gradient Boosting Regression (GBR), and Random Forest (RAN) algorithms. Prior to modeling, we performed extensive data preprocessing and feature engineering to enhance data quality and relevance.

After preparing the data, we trained and evaluated the models using R-squared (R^2), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE) metrics. Results indicated that the Random Forest (RAN) model outperformed others, achieving the highest R^2 of 0.888 and lowest MSE and RMSE.

Overall, our study underscores the importance of data preprocessing, feature engineering, and algorithm selection in building accurate regression models for house price prediction.

**Future Work**

In addition to the regression analysis conducted in this study, there are several avenues for future research and improvement. Firstly, exploring more advanced feature engineering techniques could potentially enhance model performance by capturing additional nuances in the data. Techniques such as interaction features, polynomial features, and dimensionality reduction methods like Principal Component Analysis (PCA) could be investigated.

We also tried running the SVM model but were not able to succeed in getting apt results.

We could focus on tuning the parameters and apply the SVM model as well.

Techniques such as grid search, random search, or Bayesian optimization could be employed to fine-tune model parameters and improve predictive accuracy.

Moreover, exploring ensemble methods that combine the predictions of multiple models, such as stacking or blending, could potentially yield even better results by leveraging the strengths of different algorithms.

Lastly, conducting robustness tests and sensitivity analyses to assess the models' performance under different scenarios and data conditions would provide valuable insights into their reliability and generalizability.

By pursuing these avenues for future work, we can continue to refine and improve the predictive models for house price prediction, ultimately providing valuable tools for stakeholders in the real estate industry.

Top of Form

Bottom of Form

**References:**

[**https://www.kaggle.com/code/fikretulus/usa-housing-regression**](https://www.kaggle.com/code/fikretulus/usa-housing-regression)

[**https://www.kaggle.com/code/vpkprasanna/house-price-prediction**](https://www.kaggle.com/code/vpkprasanna/house-price-prediction)

[**https://medium.com/sfu-cspmp/xgboost-a-deep-dive-into-boosting-f06c9c41349**](https://medium.com/sfu-cspmp/xgboost-a-deep-dive-into-boosting-f06c9c41349)

Top of Form

Bottom of Form

Bottom of Form