AI Real Estate Estimate
Final Project Report

Alexander Krivitsky[1], Jason Gill[1], Keena Vasiloff[1],
Safwan Ali[1], Nivrithi Krishnan[1], and Brian Huang[1]

[1]Department of Computer Science, University of California, Davis

June 12, 2024

1 Introduction

The objective of this project was to harness the power of machine learning to estimate real estate prices with high accuracy. The goal was to integrate supervised learning algorithms to develop a comprehensive tool that evaluates property values based on various significant features. This project aimed to enhance transparency in AI-driven forecasting and provide a practical, data-driven application that assists buyers, sellers, and real estate professionals in making well-informed decisions. In an increasingly competitive real estate market, the need for reliable and precise property valuation tools has never been greater.[1]

Traditional property valuation methods often rely on subjective assessments and can be prone to human error. This project took advantage of AI to automate and refine the valuation process by offering consistent and unbiased property valuations. The rapid advancements in AI allowed us to apply these methods to real estate, demonstrating its potential to revolutionize industry practices.[2]

Throughout the process of developing this project, the goal was to bridge the gap between advanced AI research and practical real-world applications. By comparing multiple machine learning models, we aimed to identify the most effective approaches for real estate prediction, ultimately creating a tool that provides consistent and accurate property valuations.[3]

[1]Cuturi, Maria Paz, and Guillermo Etchebarne. "Real Estate Pricing with Machine Learning & Non-Traditional Data Sources." Tryolabs, Tryolabs, 25 June 2021, www.tryolabs.com/blog/2021/06/25/real-estate-pricing-with-machine-learning--non-traditional-data-sources.

[2]Systems, Yellow. "How We Used Machine Learning to Predict Real Estate Prices." HackerNoon, 30 Aug. 2022, www.hackernoon.com/how-we-used-machine-learning-to-predict-real-estate-prices.

[3]Baur, Katharina, et al. "Automated Real Estate Valuation with Machine Learning Models Using Property Descriptions." Expert Systems with Applications, Pergamon, 28 Oct. 2022, www.sciencedirect.com/science/article/abs/pii/S0957417422021650.

1.1 Background

1.1.1 Real Estate Overview

The real estate industry plays a critical role in the global economy, encompassing the buying, selling, and management of properties. Accurate property valuation is essential for various stakeholders, including buyers, sellers, investors, and real estate professionals. The value of a property is influenced by numerous factors such as location, size, age, condition, and amenities. Traditional property valuation methods often involve manual appraiser assessments, which can be subjective and vary significantly between evaluators. This subjectivity can lead to inconsistent valuations, affecting decision-making processes and market dynamics. In an increasingly data-driven world, there is a growing need for more reliable, objective, and consistent property valuation methods.[4]

1.1.2 Machine Learning Concepts

Understanding our project requires familiarity with several key concepts in machine learning and real estate. Supervised learning, the core method of our project, involves training a model on a labeled dataset to make accurate predictions on new, unseen data. In our case, we use property features such as location, size, number of bedrooms and bathrooms, and additional amenities to train our model and predict real estate prices. These features are crucial as they directly influence the market value of properties.[5]

1.1.3 Regression Analysis

Regression analysis, such as linear regression, decision trees, and random forests, plays an important role in identifying and modeling the relationships between these features and property prices. Linear regression helps us understand the linear relationships between property prices and features. Decision trees provide a non-linear approach, capturing complex interactions between features. Random forests combine multiple decision trees to improve prediction accuracy and control overfitting, enhancing model robustness.[12]

1.1.4 Existing AI Applications

Existing AI applications, such as Zillow's Zestimate, have demonstrated the potential of AI in the real estate industry. Zestimate leverages proprietary algorithms and extensive data to provide home value estimates. Our project aims to build upon this foundation by implementing and comparing multiple machine learning models, including those not traditionally used in real estate, to identify the most effective approaches for price prediction. By conducting this comparative analysis, we aim to determine the strengths and weaknesses of each model, enhancing our understanding of AI's capabilities in real estate.[6]

---

[4] Chou, Jui-Sheng, et al. "Comparison of Machine Learning Models to Provide Preliminary Forecasts of Real Estate Prices - Journal of Housing and the Built Environment." SpringerLink, Springer Netherlands, 15 Mar. 2022, `www.link.springer.com/article/10.1007/s10901-022-09937-1`.

[5] Pai, Ping-Feng, and Wen-Chang Wang. "Using Machine Learning Models and Actual Transaction Data for Predicting Real Estate Prices." MDPI, Multidisciplinary Digital Publishing Institute, 23 Aug. 2020, `www.mdpi.com/2076-3417/10/17/5832`.

[6] Johnson, Reid. "Building the Neural Zestimate." Zillow, 24 Feb. 2023, `www.zillow.com/tech/building-the-neural-zestimate`.

1.1.5 Model Transparency and Ethical Considerations

Our project emphasizes the importance of model transparency and ethical considerations. By thoroughly documenting and explaining our models' behavior, we aim to address concerns about AI's "black box" nature and ensure users understand the rationale behind the predictions. Ethical considerations, such as responsible data usage and privacy, are paramount in our project. Ensuring that our models use data ethically and transparently is crucial for gaining user trust and acceptance.[5]

1.1.6 Project Impact

By leveraging machine learning, we can uncover valuable insights from data, inform decision-making, optimize processes, and drive innovation in the real estate business. This project aims to improve the accuracy of property valuations and sets a new standard for AI applications in real estate, promoting responsible and transparent AI use. Our ultimate goal is to revolutionize industry practices, making real estate transactions more efficient, reliable, and fair.[12]

2 Methodology

2.1 Dataset Source

Our project utilized the "USA Housing Dataset" from Kaggle, authored by Shree (Shree1992), a data scientist based in Brisbane, Queensland, Australia. The dataset is comprehensive, encompassing a variety of real estate attributes crucial for accurate property valuation predictions. The dataset specifically includes data from Washington, US, covering house sales in the year 2014. The dataset consists of columns such as date, price, bedrooms, bathrooms, sqft_living, sqft_lot, floors, waterfront, view, condition, sqft_above, sqft_basement, yr_built, yr_renovated, street, city, statezip, and country. These features are essential in determining the market value of properties as they cover both structural characteristics and environmental factors while being transparent.

2.2 Preprocessing Steps

- **Data Cleaning:**
  - **Handling Missing Values:** We meticulously examined the dataset for missing or null values and removed rows with incomplete data to maintain data integrity. This step ensures that the model is trained on a complete and reliable dataset.
  - **Outlier Detection and Handling:** Extreme outliers, which can distort the model's performance, were identified and handled appropriately. For instance, properties with unusually high or low prices relative to their features were scrutinized and either transformed or removed.
- **Feature Engineering:**
  - **One-Hot Encoding:** Categorical features such as waterfront, view, and condition were transformed into a numerical format using one-hot encoding. This step converts categorical variables into a binary format, enabling the machine learning algorithms to interpret them effectively.

- **Normalization:** Continuous features such as sqft_living, sqft_lot, bedrooms, and bathrooms were normalized using StandardScaler from the Scikit-Learn library. Normalization ensures that all features contribute equally to the model training by bringing them to a similar scale.

- **Data Splitting:**
  - **Train-Test Split:** The dataset was divided into training and testing sets using an 80-20 split ratio. This split ensures that the model is trained on a substantial portion of the data and tested on a separate, unseen set to evaluate its performance. This method helps in assessing the generalization ability of the models.

2.3 Algorithms Applied

- **Linear Regression:**
  - **Implementation:** Linear Regression was employed to model the relationship between the input features and the target variable (price). This method helps comprehend the linear dependencies and interactions among the features. The model parameters were estimated using the Ordinary Least Squares method, which minimizes the sum of the squared differences between the observed and predicted values.
  - **Evaluation:** The model's performance was evaluated using Mean Squared Error (MSE), R-squared (R^2), and Zestimate as metrics. Linear Regression serves as a baseline model, providing insights into the linear relationships within the dataset.

- **Decision Tree Regressor:**
  - **Implementation:** A Decision Tree Regressor was utilized to capture the non-linear relationships in the data. This algorithm recursively splits the data into subsets based on feature values, creating a tree-like structure that models complex interactions between features.
  - **Evaluation:** Performance metrics such as MSE, R^2, and Zestimate were used to assess the model's accuracy and explanatory power. Decision Trees are intuitive and provide a clear visual representation of the decision-making process.

- **Random Forest Regressor:**
  - **Implementation:** The Random Forest Regressor, an ensemble method that combines multiple decision trees, was implemented to improve prediction accuracy and mitigate overfitting. This model aggregates the predictions from various trees to provide a more robust and accurate prediction.
  - **Evaluation:** The Random Forest model was evaluated using MSE and R^2, with additional insights from feature importance scores. Zestimate was used as a comparison in a real-world context. This ensemble approach leverages the strength of multiple weak learners to create a strong predictive model.

2.4 Model Training and Evaluation

- **Model Fitting:**
  - Each model was trained on the training dataset. The training process involved fitting the models to the data and adjusting weights and biases to minimize prediction errors. This step also includes learning the optimal parameters for each model that best explains the relationship between the features and the target variable.
- **Cross-Validation:**
  - **5-Fold Cross-Validation:** To ensure the robustness of the models, 5-fold cross-validation was performed. This technique splits the training data into five subsets, trains the model on four subsets, and validates it on the fifth. This process is repeated five times, with each subset serving as the validation set once. The average performance across all folds provides a more reliable estimate of the model's generalization ability.

2.5 Evaluation Metrics

1. **Mean Squared Error (MSE):**
   - MSE measures the average squared difference between the predicted and actual values. It provides a quantifiable measure of the prediction error, where a lower MSE indicates a better fit to the data.
2. **R-squared (R^2):**
   - R^2 was calculated to determine the proportion of variance in the dependent variable that the independent variables can explain. It is a vital indicator of the model's explanatory power, where a higher R^2 value indicates better model performance.
3. **Comparison with Zillow Zestimate:**
   - For practical evaluation, the model's predictions of the last house were compared with Zillow's Zestimate for this specific property. This comparison helps in understanding how well our models perform relative to an established industry benchmark, providing a real-world context to our model evaluation.

2.6 Hyperparameter Tuning

2.6.1 Optimization Using Optuna

- **Optuna Framework:** Hyperparameters for the Decision Tree and Random Forest models were optimized using Optuna, an automatic hyperparameter optimization framework. Optuna performs a comprehensive search over the hyperparameter space to identify the optimal settings that maximize model performance.
- **Key Hyperparameters Tuned:**
  - **Decision Tree:** Maximum depth, minimum samples split, and minimum samples leaf. These parameters control the complexity of the tree and help prevent overfitting.

- **Random Forest:** Number of estimators, maximum depth, minimum samples split, and minimum samples leaf. These parameters determine the number of trees in the forest, their complexity, and how the data is split within each tree.

2.7 Handling Skewed Data

2.7.1 Data Transformations

- **Log Transformation:** To address skewness in the price variable, a log transformation was applied. This transformation stabilizes the variance and makes the data more normally distributed, which is beneficial for linear regression models. By transforming the target variable, we ensure that the model's assumptions of normality and homoscedasticity are better met.
- **Feature Scaling:** Continuous features were scaled using StandardScaler to ensure uniform contribution to model training. Scaling prevents any single feature from disproportionately influencing the model training, leading to more balanced and accurate predictions.

2.8 Tools Used

2.8.1 Programming Languages and Libraries

- **Python:**
  - The primary programming language is used for data preprocessing, model training, and evaluation.
- **Libraries:**
  - **Pandas:** for data manipulation and analysis, providing flexible data structures to manage large datasets efficiently.
  - **NumPy:** for numerical computations, offering support for large multi-dimensional arrays and matrices.
  - **Scikit-Learn:** for implementing machine learning algorithms and evaluation metrics, providing a wide range of tools for model selection, preprocessing, and evaluation.
  - **Optuna:** for hyperparameter optimization, enabling efficient and effective tuning of model parameters.
  - **Matplotlib and Seaborn:** for data visualization offering powerful plotting capabilities to create informative and attractive visualizations.

2.8.2 Integrated Development Environment (IDE)

- **Jupyter Notebook:**
  - Jupyter Notebook was used for coding, documentation, and visualization.

3 Results

3.1 Model Performance

3.1.1 Linear Regression

- **Mean Squared Error (MSE):** 20,296,425,970.84
  - The high MSE value indicates that the model's predictions are significantly deviating from the actual values. This suggests that the linear regression model might not be capturing the complexity of the data effectively.
- **R-squared (R^2):** 0.39
  - An R^2 value of 0.39 means that approximately 39% of the variance in the target variable (price) can be explained by the input features. This relatively low R^2 value suggests that the linear model is not able to account for most of the variability in the data.
- **Cross-Validation Scores:** Varied between 0.29 and 0.35, with a mean CV score of 0.32.
  - The cross-validation scores indicate that the model's performance is consistently low across different subsets of the data. This further confirms that linear regression is not the best fit for this dataset due to its inability to capture complex patterns and interactions.

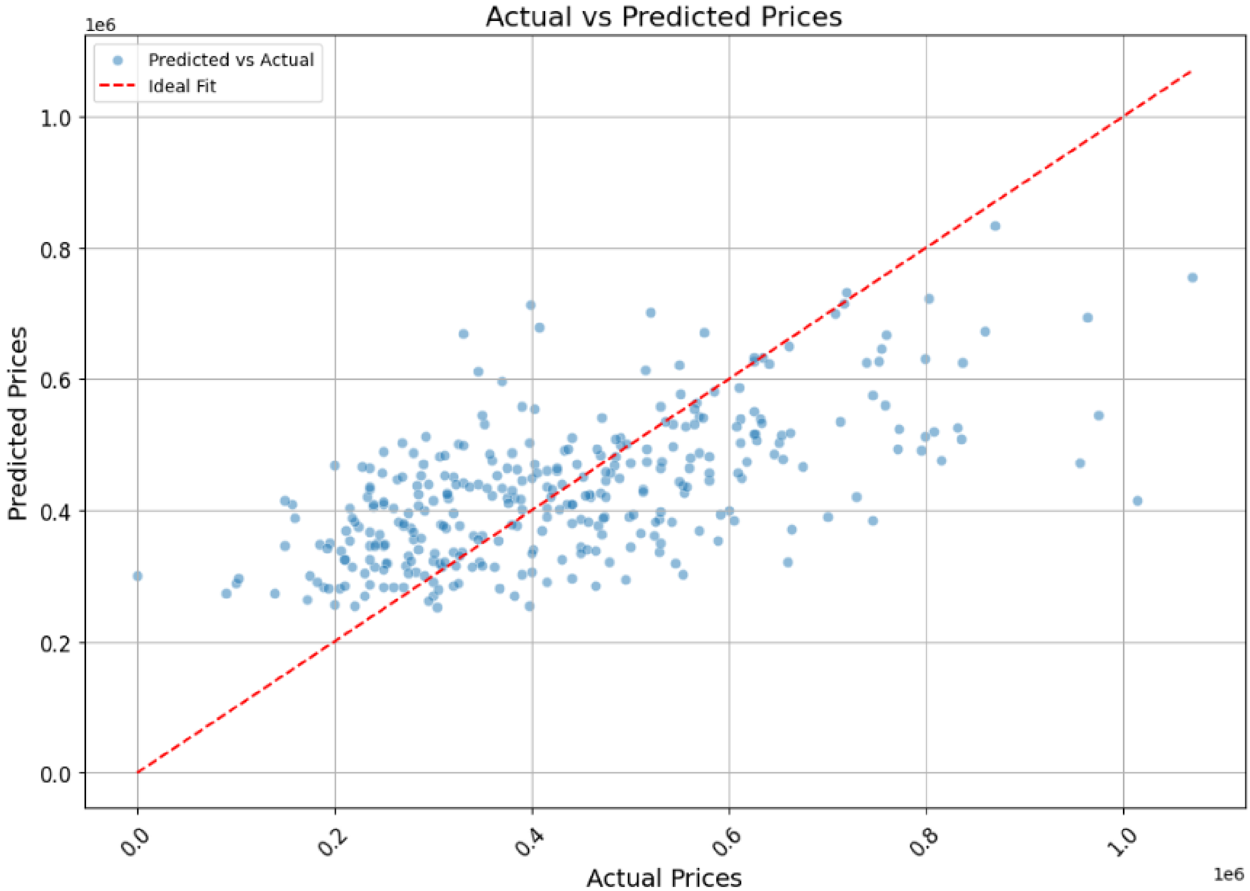

Figure 1: Scatter plot for the Linear Regression Model, comparing the actual house prices and those predicted. The red dashed line represents the ideal perfect prediction.

3.1.2 Decision Tree Regressor

- **Mean Squared Error (MSE):** 9,853,867,258.25
  - The MSE is significantly lower than that of the linear regression model, indicating better predictive accuracy. The decision tree model captures more complex relationships in the data, resulting in improved predictions.
- **R-squared (R^2):** 0.70
  - An R^2 value of 0.70 indicates that 70% of the variance in the target variable is explained by the model. This is a substantial improvement over the linear regression model, highlighting the decision tree's ability to handle non-linear relationships.
- **Cross-Validation Scores:** Varied between 0.56 and 0.65, with a mean CV score of 0.60.
  - The cross-validation scores are higher and more stable compared to the linear regression model, demonstrating better generalization to unseen data. The decision tree model provides a more reliable performance across different subsets of the data.

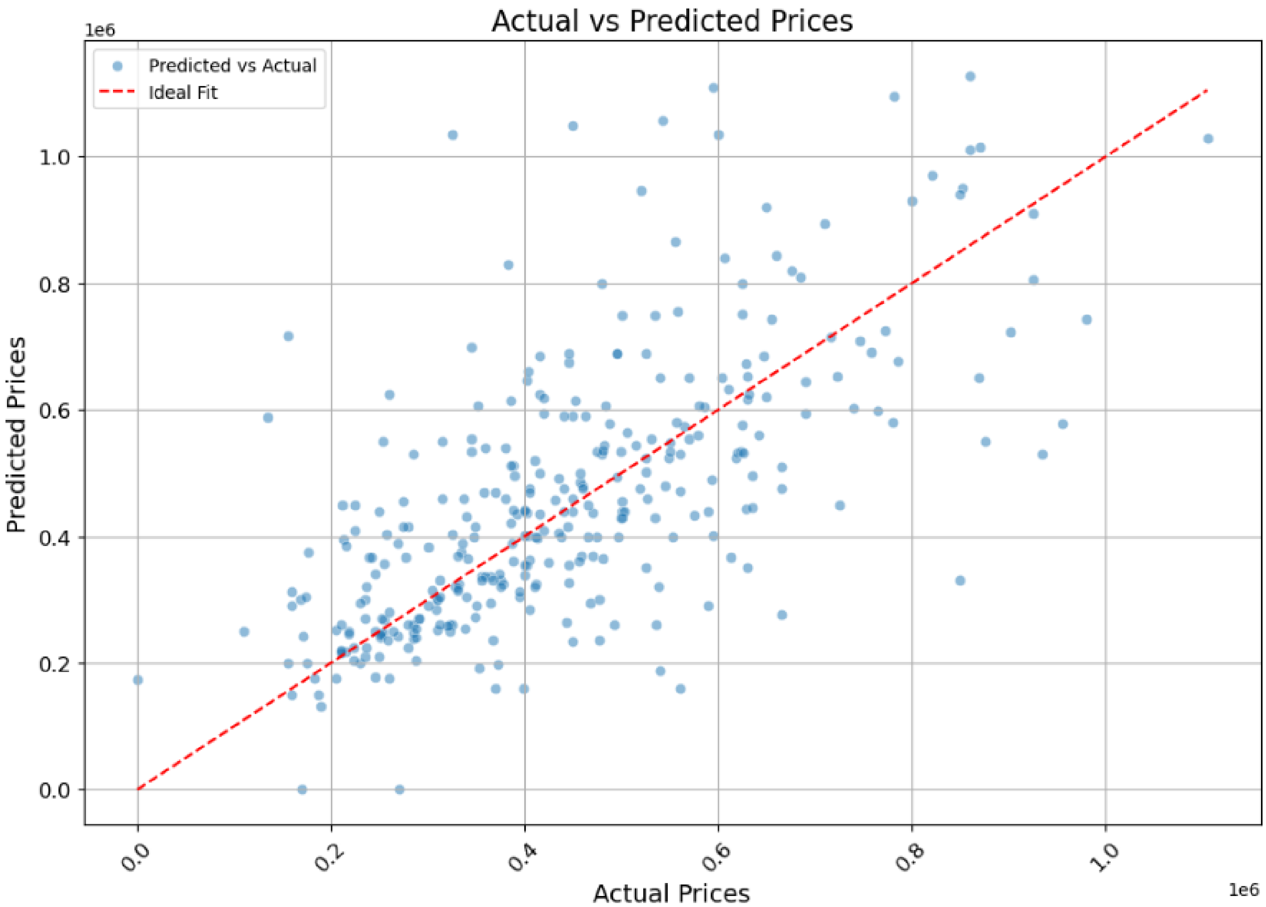

Figure 2: Scatter plot for the Decision Tree Regressor.

3.1.3 Random Forest Regressor

- **Mean Squared Error (MSE):** 9,750,395,912.66
  - The random forest model achieves the lowest MSE among the three models, indicating the highest predictive accuracy. The ensemble approach of combining multiple decision trees helps in reducing overfitting and improving the model's robustness.
- **R-squared (R^2):** 0.71
  - An R^2 value of 0.71 means that 71% of the variance in the target variable is explained by the model, making it the best performer in terms of explanatory power. The random forest's ability to capture complex interactions and dependencies results in superior performance.
- **Cross-Validation Scores:** Had a mean CV score of 0.21
  - Despite the high R^2 and low MSE, the cross-validation scores are surprisingly low. This could indicate potential overfitting to the training data or issues with the stability of the model across different subsets of the data. Further investigation and tuning of hyperparameters may be required to address this.

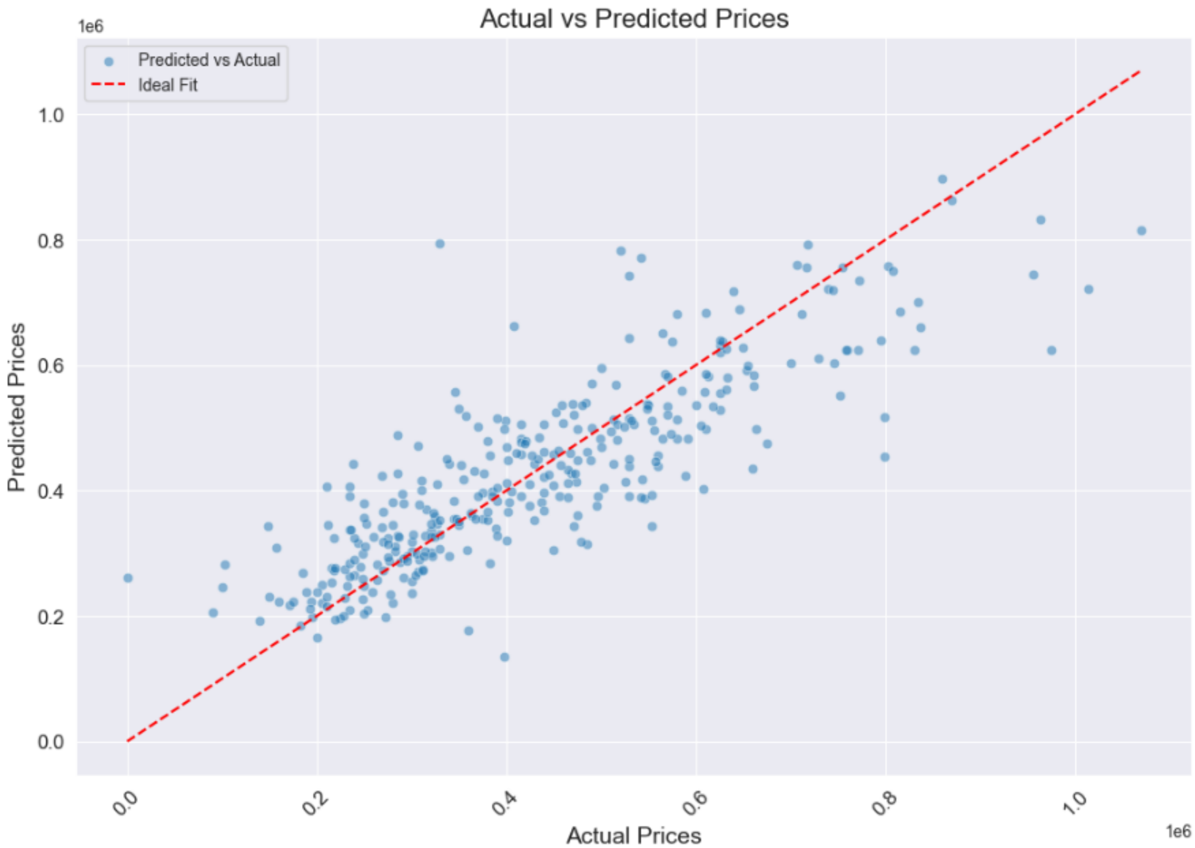

Figure 3: Scatter plot for the Random Forest Regressor.

3.1.4 Zestimate

- From the dataset, we chose a singular property located at 18717 SE 258th St, Covington, WA 98042. The 1640 square feet single-family home is a 3-bed, 2.25 bath property. Various models were used to predict the house price and compared against the Zestimate provided by Zillow, as well as the actual sale price.
  - **Zestimate:** \$297,000.00
  - **Real Sale Price:** \$220,600.00
  - **Linear Regression Prediction:** \$379,770.03
  - **Decision Tree Prediction:** \$296,702.57
  - **Random Forest Prediction:** \$224,536.58
- The Linear Regression model predicted a significantly higher price of \$379,770.03 compared to the Zestimate value of \$297,000.00 and the actual sale price of \$220,600.00. This suggests that the Linear Regression model may have overestimated the house price due to not capturing the complexity of the data.
- The Decision Tree model predicted a price of \$296,702.57, which showed less deviation from Zillow's Zestimate and was closer to the actual sale price than the Linear Regression model.

This indicates that the Decision Tree model could capture more non-linear relationships in the data.

- The Random Forest model demonstrated the best alignment with the actual dataset's sale price, predicting \$224,536.58. This prediction was the closest to the actual sale price of \$220,600.00, indicating that the Random Forest model was more accurate than both the Zestimate and the other models.

- The Linear Regression model predicted a significantly higher price compared to the Zestimate value and the dataset price. The Decision Tree model displayed less deviation from Zillow's Zestimate and was closer to the dataset price. However, the Random Forest model demonstrated the best alignment with the actual dataset's price and thus was more accurate than Zestimate.

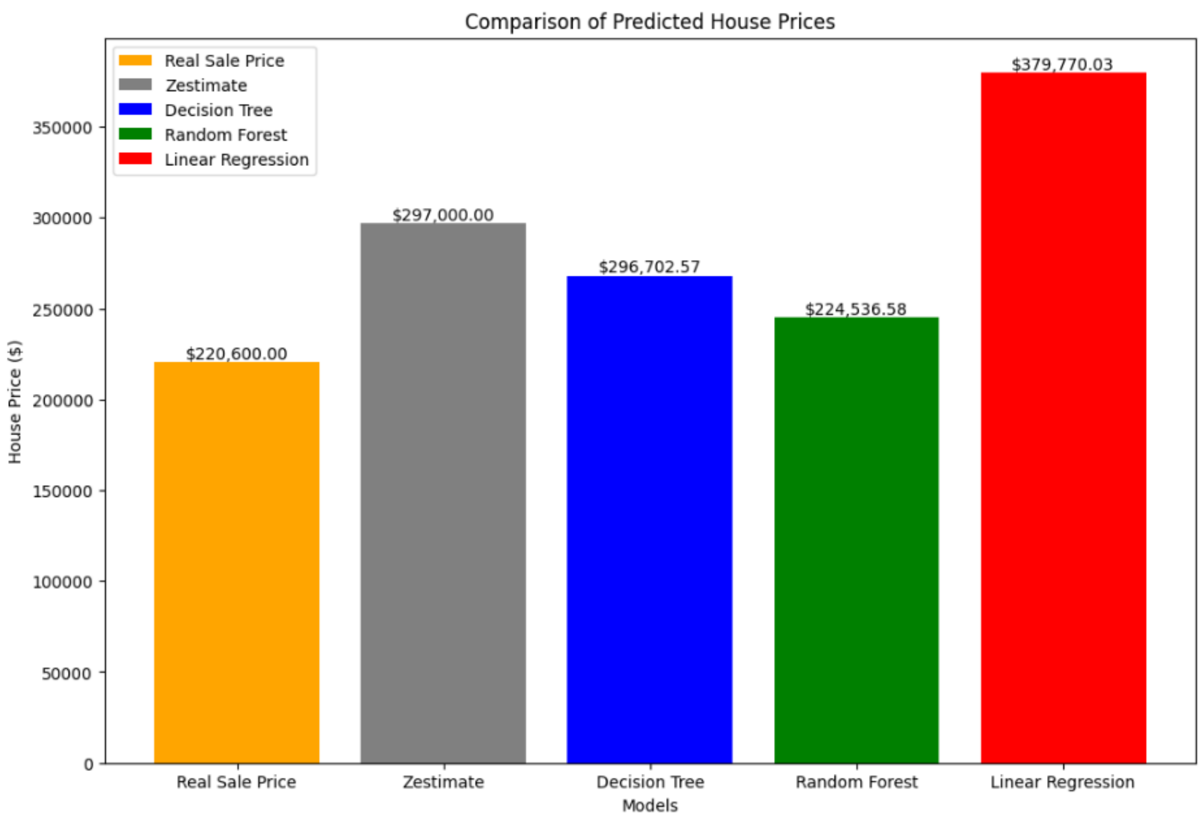

Figure 4: Bar plot comparing prices predicted by each model to the dataset sale price.

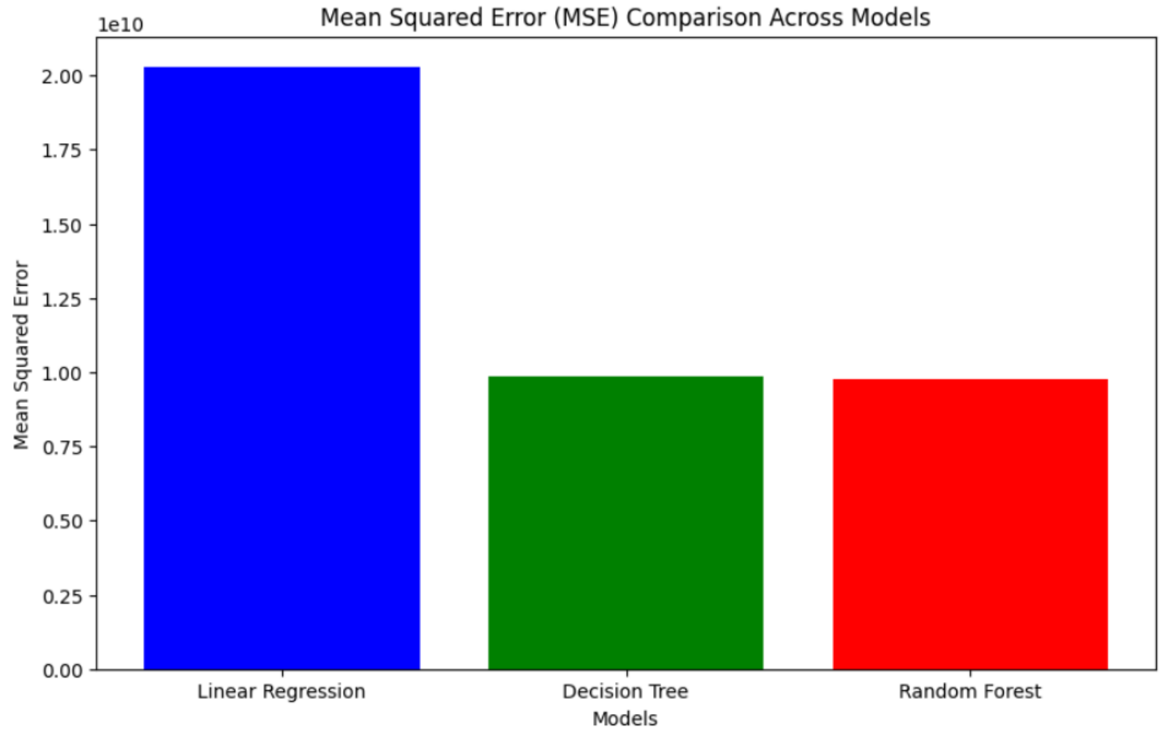

Figure 5: Bar plot graphing the MSE metric for all models. A lower score is better.

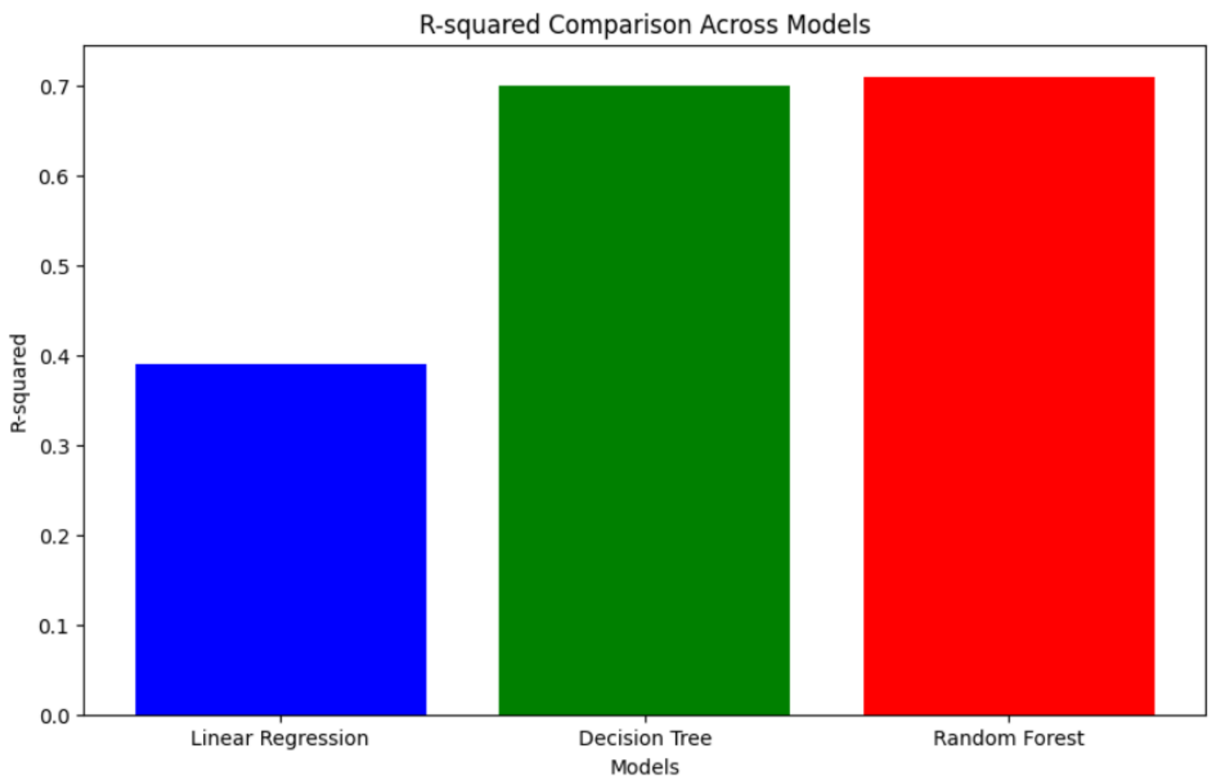

Figure 6: Bar plot graphing the R^2 metric for all models. A higher score is better.

3.2 Feature Importance

3.2.1 Linear Regression

- Feature importance is determined by the model coefficients, which indicate the strength and direction of the relationship between each feature and the target variable. For example:
  - **Feature 1:** A positive coefficient indicates a positive relationship with price.
  - **Feature 2:** Negative coefficient indicates a negative relationship with price.
- The model obtained two positive features:
  - **sqft_living:** This feature has the highest importance, making it the most crucial predictor of price.
  - **sqft_above:** This feature also plays a significant role in predicting price.

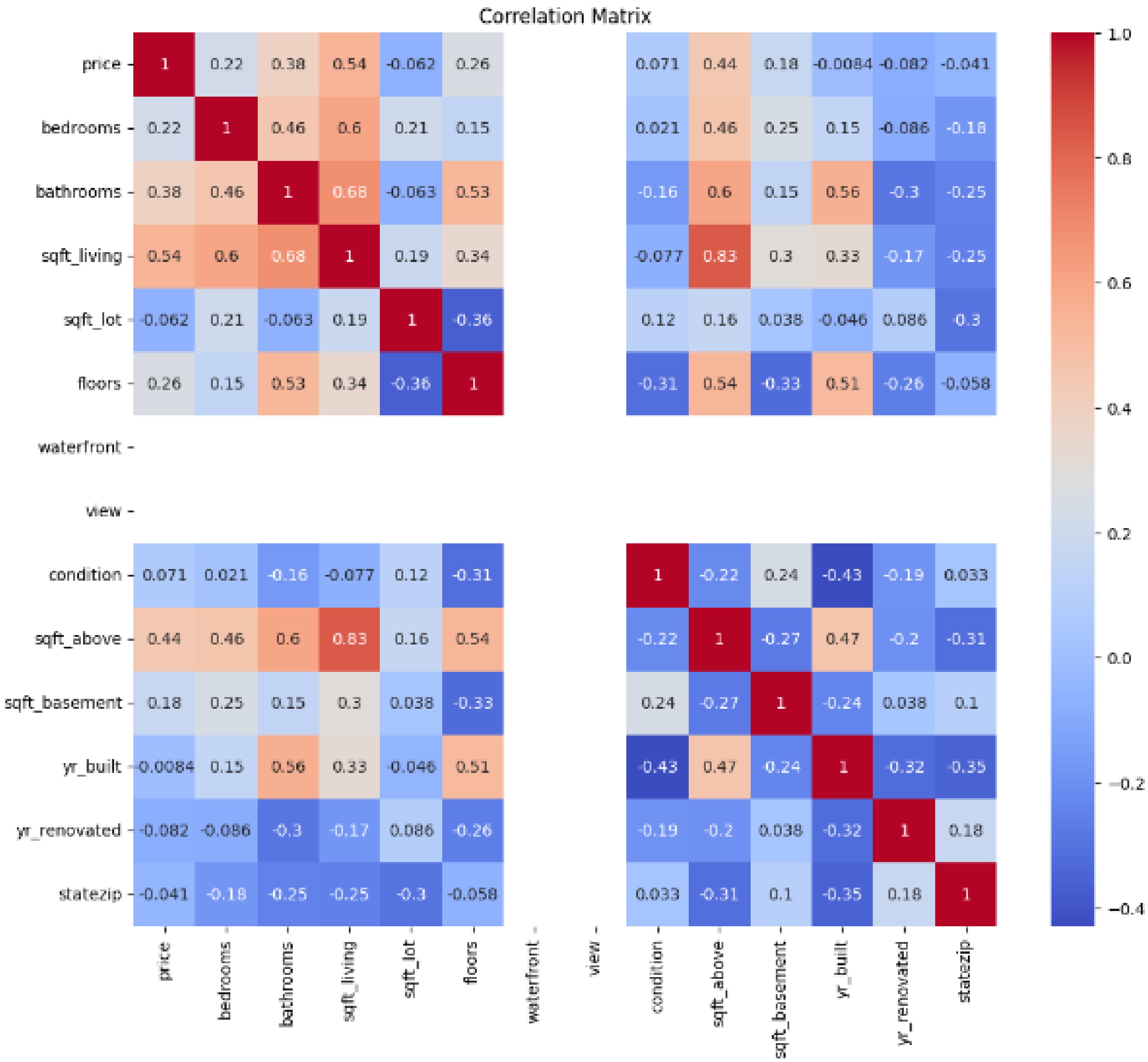

Figure 7: Correlation matrix plotting feature importance in the Linear Regression Model. This heat map illustrates the correlation between various features in the dataset. Strong positive correlations are shown in red, and strong negative correlations are in blue.

3.2.2 Decision Tree Regressor

- The decision tree model provides a ranking of feature importance based on the frequency and quality of splits. More importantly, feature trimming was employed. Important features include:
  - **sqft_living:** Highest importance, indicating it is the most significant predictor of price.
  - **grade:** Also a significant predictor.
  - **sqft_above:** Another important feature.

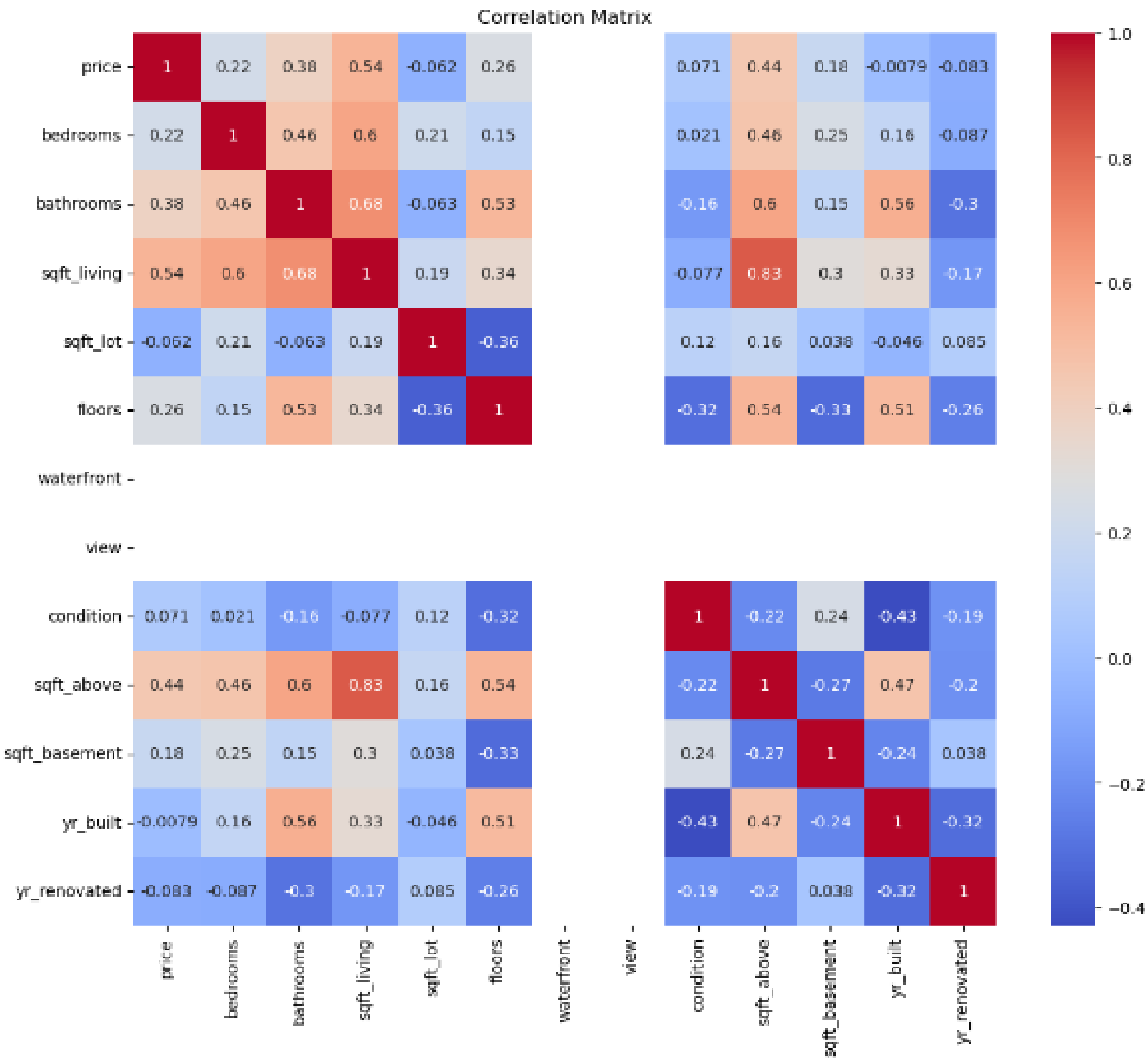

Figure 8: Correlation matrix plotting feature importance in the Decision Tree Regressor.

3.2.3 Random Forest Regressor

- Random Forest aggregates feature importance across multiple trees to provide a robust importance score. Important features include:
  - **sqft_living:** Consistently important across trees.
  - **grade:** Also consistently important.
  - **sqft_above:** Important in the ensemble model.

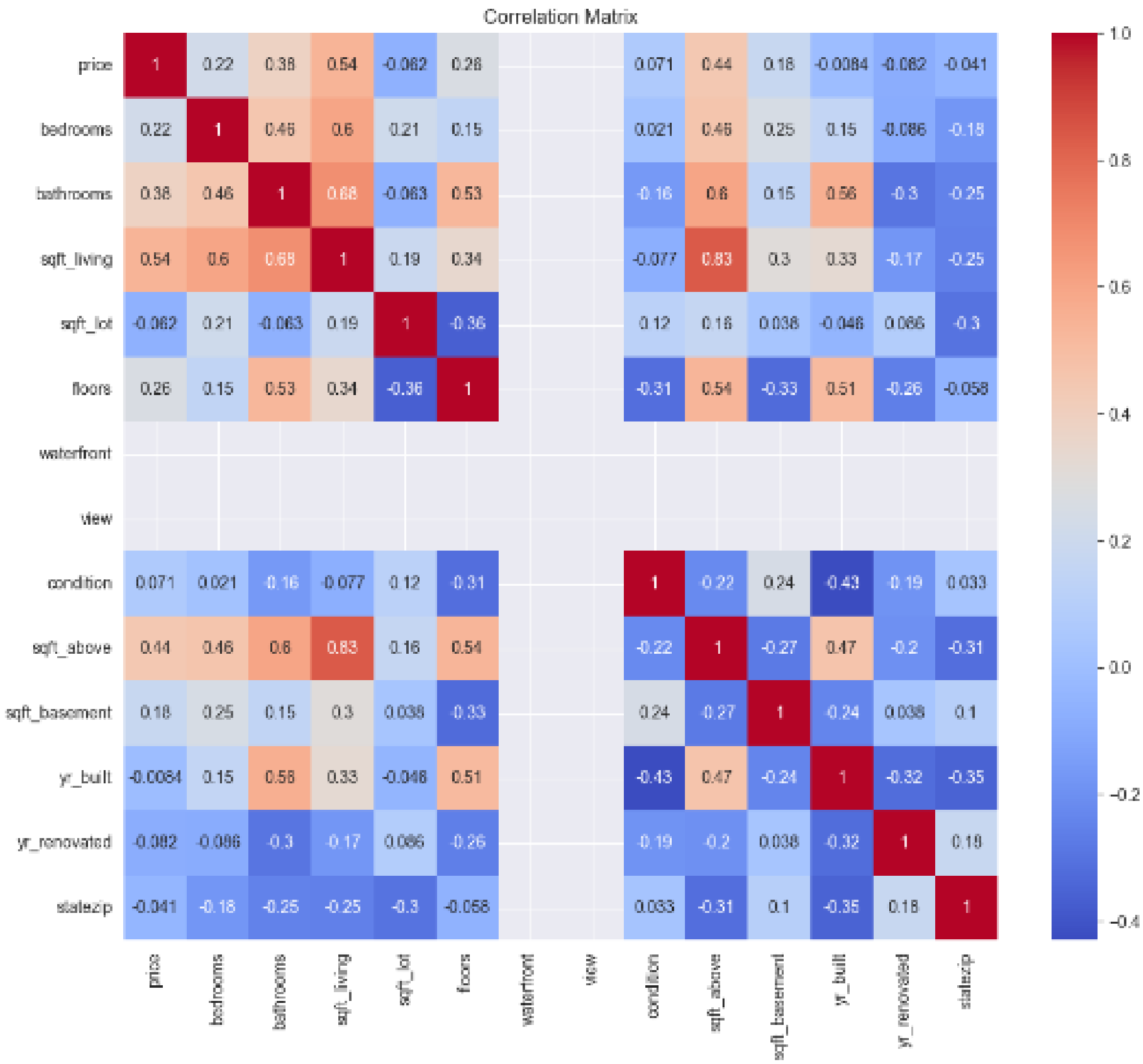

Figure 9: Correlation matrix plotting feature importance in the Random Forest Regressor.

4 Discussion

4.1 Significance

This project aims to develop and compare machine-learning models for predicting real estate prices. Although the linear regression model provided a baseline, the model's low accuracy and high error underscore the limitations of linear models in capturing complex real estate market dynamics. The decision tree regressor performed significantly better, highlighting the importance of capturing non-linear relationships. However, the high MSE indicates room for improvement. The random forest model's high performance demonstrates the effectiveness of ensemble methods in improving prediction accuracy and robustness. Among the three models, his model additionally serves as the most reliable tool for real estate valuation.

The significance of the results extends beyond their numerical performance, impacting practical applications in the real estate market. The project emphasizes a commitment to ethical AI, ensuring the process is transparent and trustworthy. The source of our data is specified and linked, building trust and reliability with users. Thorough documentation provides step-by-step explanations of our methodology, including the data used, data pre-processing, feature engineering, and model training processes allowing users to understand the rationale behind decisions made throughout our process. By offering a detailed analysis of the exact features used, providing their respective importance scores, and displaying the correlation matrix mapping the relationship between the different features and target, users can clearly identify which features have a higher influence over property valuations, and impact on the price.

The lack of transparency in other tools, namely Zillow's Zestimate, has led to the questioning of accuracy coupled with little to no explanations. Foremost, their algorithm is proprietary, with no outside visibility into the exact methods or data points used to calculate a home's value. The company is heavily reliant on user-submitted data and public records, questioning their ethical use of data particularly when users are not fully aware of how their information is being utilized. The data collected by their platform includes sensitive information (ex: tax history), raising concerns about privacy and consent.

4.2 Challenges and Resolutions

Through the development process, one of the major challenges we faced was achieving satisfactory accuracy with our models. This was a critical issue as the primary objective of the project was to develop a highly accurate valuation tool. To resolve this, we implemented several strategies. This includes ensuring the initial dataset was free from significant anomalies by handling missing values and outliers. Additionally, categorical features were transformed into numerical formats and normalizing continuous features to ensure a uniform contribution toward model training. To optimize model performance we implemented hyperparameter tuning and applied techniques like log transformation and feature scaling to address skewness in the price variable.

4.3 Learning Outcomes

One of the key takeaways from this project was the importance of clean, high-quality data. The data pre-processing step was a meticulous and time-consuming task but yielded the highest reward as reflected in the improved accuracy scores. Clean, high-quality data and meticulous preprocessing were critical. In an attempt to improve accuracy, we also built our technical skills by experimenting and implementing the strategies listed above. Through our research and discovering how little public information is available about Zillow's tools, we also learned just how important transparency and ethical considerations were, specifically in building user trust.

5 Conclusion

5.1 Summary

This project harnessed machine learning to accurately estimate real estate prices, creating a valuable tool for buyers, sellers, and professionals. By leveraging supervised learning algorithms, we aimed to enhance transparency and practical application of AI in real estate. Rigorous data preprocessing, feature engineering, model training, evaluation, and hyperparameter tuning ensured our models' robustness and reliability.

We compared Linear Regression, Decision Trees, and Random Forests. Linear Regression provided a baseline, with non-linear models like Decision Trees and Random Forests significantly outperforming them, while Random Forest proved the most reliable due to its aggregation of multiple trees' predictions.

5.2 Future Work

- **Additional Features:** Incorporate more diverse features like neighborhood crime rates and school quality.
- **Advanced Models:** Explore advanced models such as Gradient Boosting Machines (GBM) and deep learning.
- **Real-Time Data:** Integrate real-time market trends and economic indicators for dynamic valuations.
- **User Interface:** Develop a user-friendly interface for instant property valuations.
- **Ethical AI:** Continue emphasizing data privacy, transparency, and responsible usage.

Future work can build on this project's foundation, further enhancing AI-driven real estate valuation tools' accuracy, reliability, and usability, revolutionizing industry practices.

6 GitHub

Here is the link for the GitHub that includes everything about our project:

https://github.com/TheGhostCoder0/ECS170Group17FinalProject

7 Contributions

7.1 Alexander Krivitsky

Alexander worked on and Improved Linear Regression and Random Forest Models. He organized and set up the final deliverable document and worked on the methodology section and final polish stage. Created and worked on the script for the presentation.

7.1.1 Jason Gill:

Worked on the initial development of the linear regression model, including the linear regression estimation comparison. Also created and edited visualizations to ensure all visual aids were properly incorporated into the text.

7.1.2 Keen Vasiloff

Began development of Linear Regression and Random Forest models and further worked with Random Forest Zestimate comparison. Held an additional organizational role by arranging meetings, setting up the repository, helping to set deadlines, sharing meeting notes, and reviewing submissions with Grammarly.

7.2 Safwan Ali

Safwan worked on the decision tree model and refined it for feature engineering and multiple trials of testing using XGBoost and Optuna for clearer results. Contributed to slides and document information and formatting.

7.3 Nivrithi Krishnan

Nivrithi worked on the Decision Tree Model and the discussion section of the final deliverable. She also helped edit the presentation script and design/format the slides.

7.4 Brian Huang

Worked on implementing the Decision Tree Zestimate comparison. Helped with organizing, designing, and putting everything together into the final LaTeX/PDF deliverable document and presentation slides.