predictive_models_report.md

Insights into Housing Prices: Neural Network vs. Random Forest Analysis

Abstract

This project employs the classic Ames, Iowa housing dataset to predict individual residential property prices using two distinct machine learning models. Experimentation involves enhancing a neural network through feature engineering and hyperparameter tuning, alongside the construction of a random forest model for comparative analysis. The subsequent discussion delves into the strengths and weaknesses of each model, offering insights into their unique attributes and trade-offs.

Table of Contents

Introduction

The Ames housing dataset, a comprehensive collection of local government data on housing prices spanning 2006-2010, offers a rich landscape of opportunities for predicting individual residential property prices leveraging machine learning. Curated by Dean De Cock and sourced from the Ames City Assessor's Office in 2011, the dataset comprises 80 variables capturing diverse attributes like size, room count, location, and age (see `/resources/data_description.txt`).

The primary objective was to come up with a linear regression machine learning model capable of predicting the residential property sale prices with at least a 0.80 R-squared, ensuring the model explains a substanstial proportion of the variance in price. Employing a dataset of 1460 samples provided by Kaggle, the project delves into the intricacies of neural network complexity and the robustness of a random forest model. This report encapsulates the methodologies employed and valuable insights gained from this comprehensive predictive modeling exercise.

Exploratory Data Analysis

Commencing our analysis, extensive research and consultations with real estate professionals guided our expectations, leading us to anticipate key features influencing property prices, including property condition, indoor square footage, room count, property age, and the timeless "location, location, location".

The intentionally blurred labels in this initial correlation heatmap underscore the myriad possibilities within this dataset.

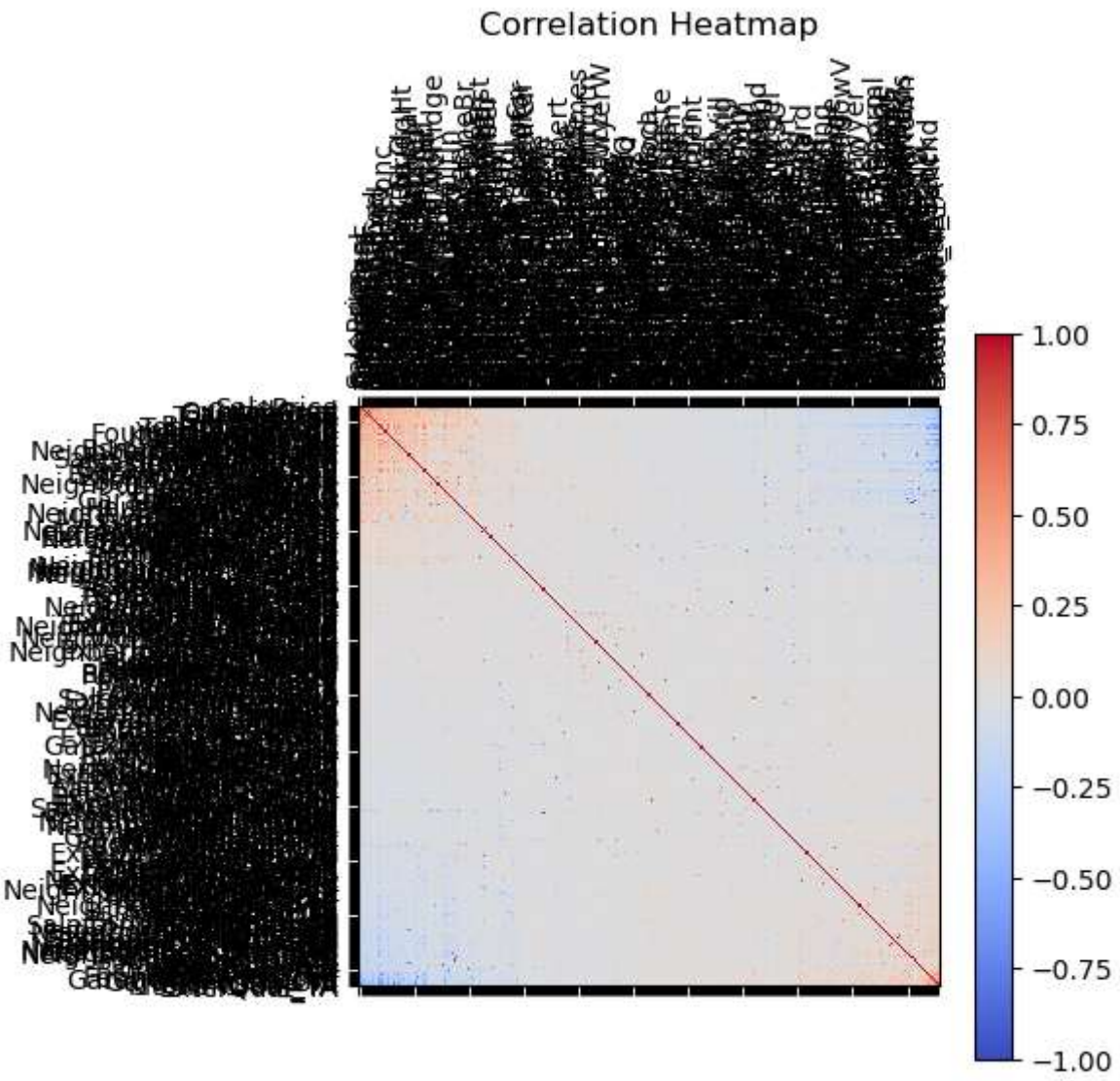
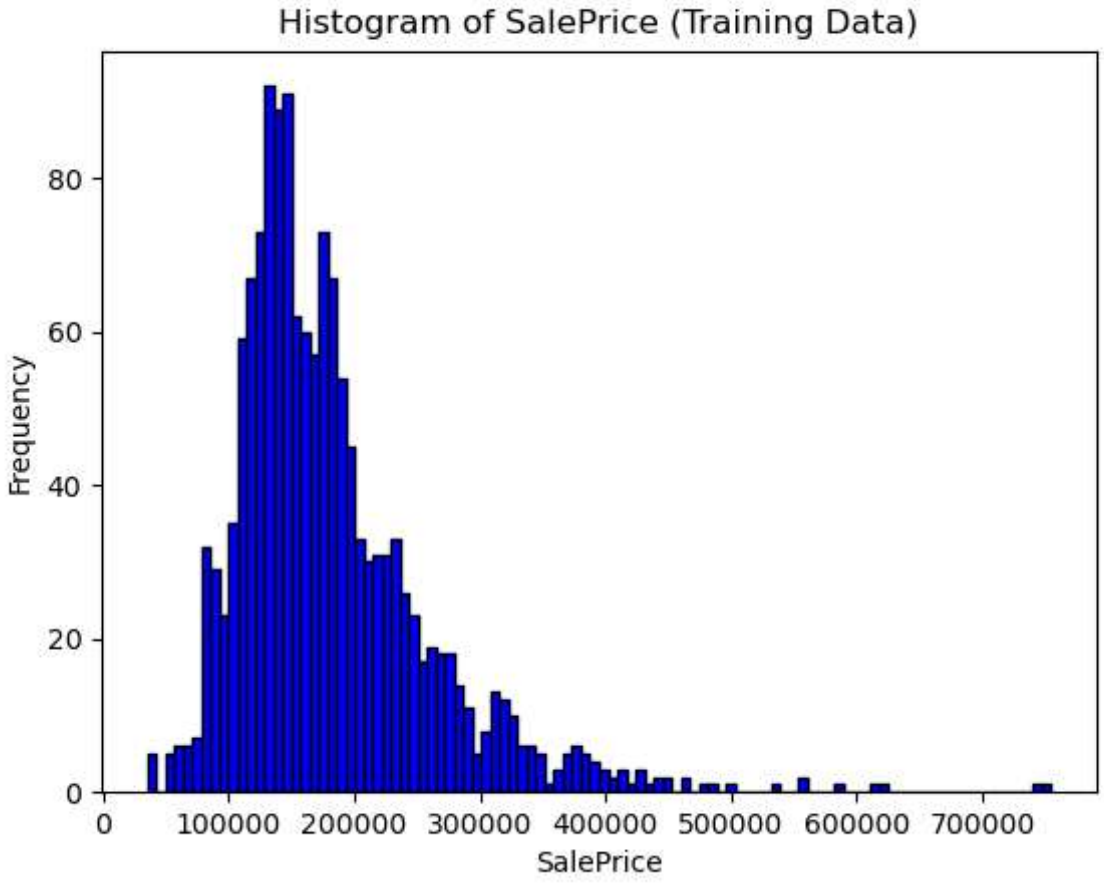


Figure 1 - Correlation Heatmap: This heatmap provides an initial exploration of potential correlations between different features in the dataset.

A histogram of the target variable, SalePrice, revealed a fairly normal distribution with a long tail, suggesting potential outliers, later addressed in our data processing.



.

Figure 2 - Histogram of Sale Price: The histogram highlights the distribution of sale prices in the dataset.

Specific attention was given to indoor living square footage and indicated promising patterns.

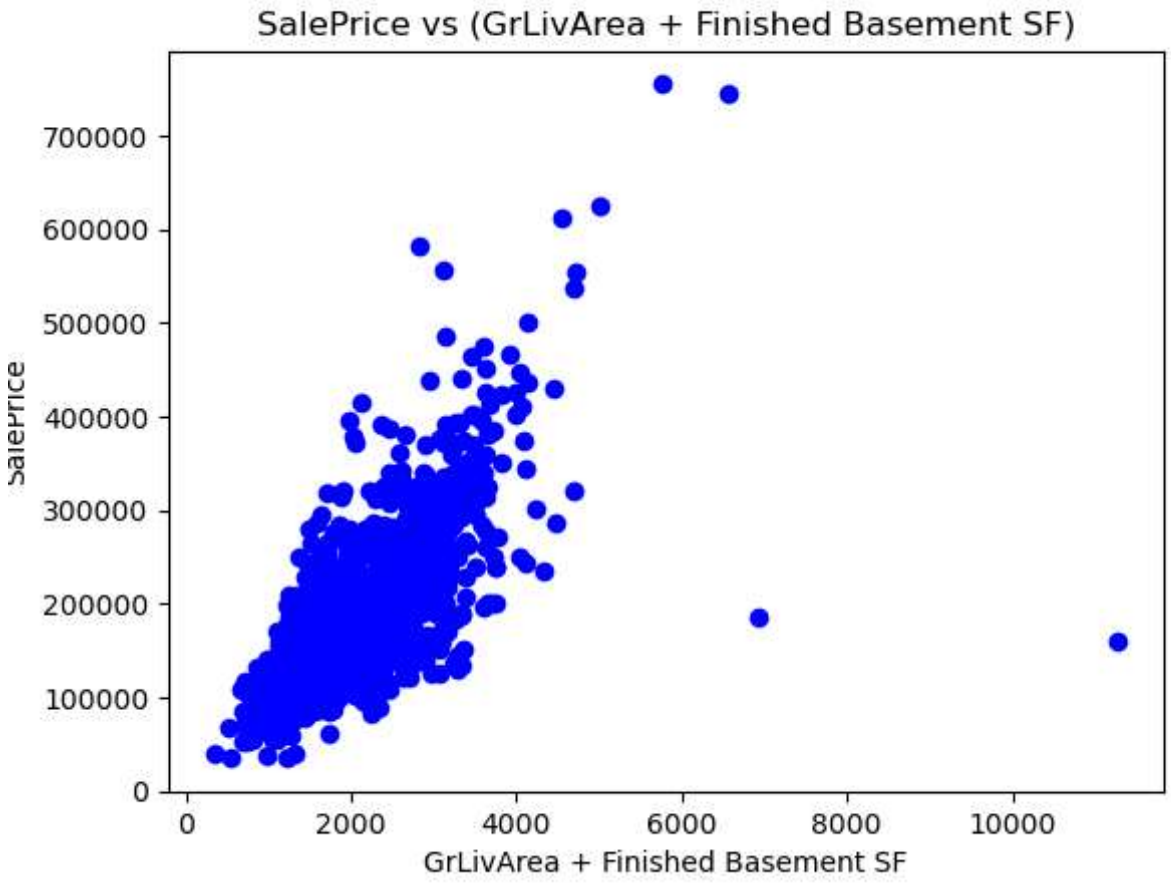


Figure 3 - Scatter Plot of Indoor Square Footage: This scatter plot illustrates the relationship between indoor square footage and sale prices.

Challenges emerged when using 'Neighborhood' as a proxy for location. Thirteen of the 28 categories had fewer than 50 values, including categories with 11, 9, and 2 instances.

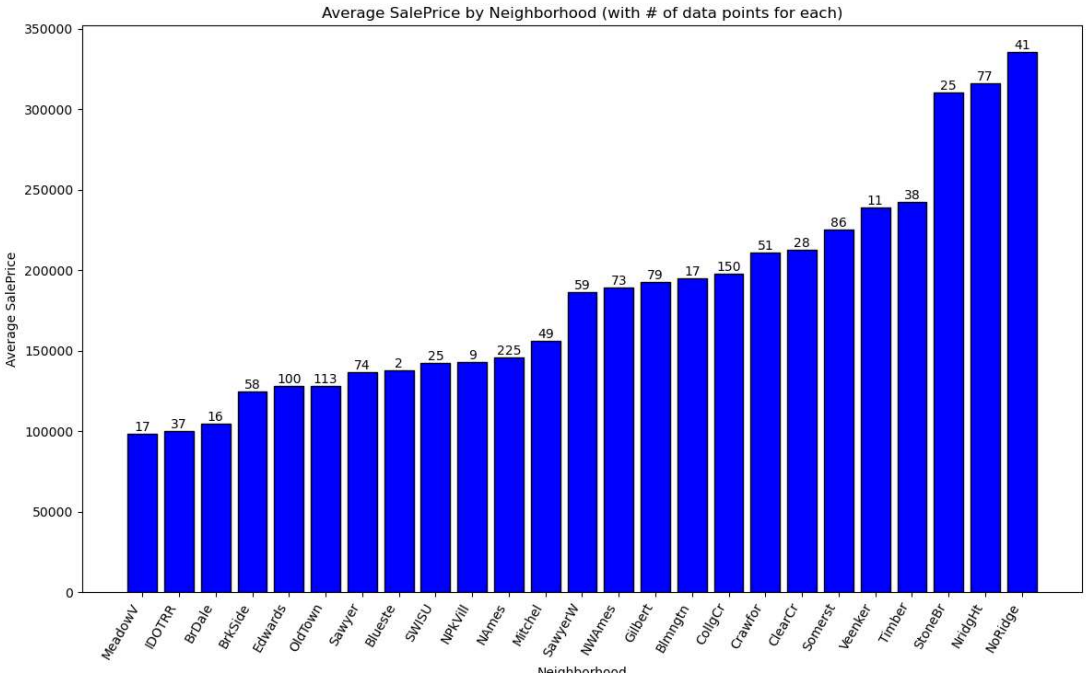


Figure 4 - Average Neighborhood Sale Price Analysis: This bar chart explores average sale prices across different neighborhoods, annotated with data point counts.

Investigation into sale conditions revealed 259 values not categorized as 'Normal'. Sale conditions covered a spectrum from trade and foreclosures to short sales, transactions between family members, and houses not completed when last assessed, typically associated with new constructions.

Visualizing both discrete and continuous data through histograms and scatter plots facilitated exploration of potential features influencing sale prices, narrowing our focus to overall quality and condition, aspects related to square footage, room count, and the year built or remodeled.

Nominal and ordinal data were examined using bar plots to guide future feature engineering and potential outlier identification, such as a single property lacking sewer and water utilities. Ordinal values related to quality exhibited promise for further analysis.

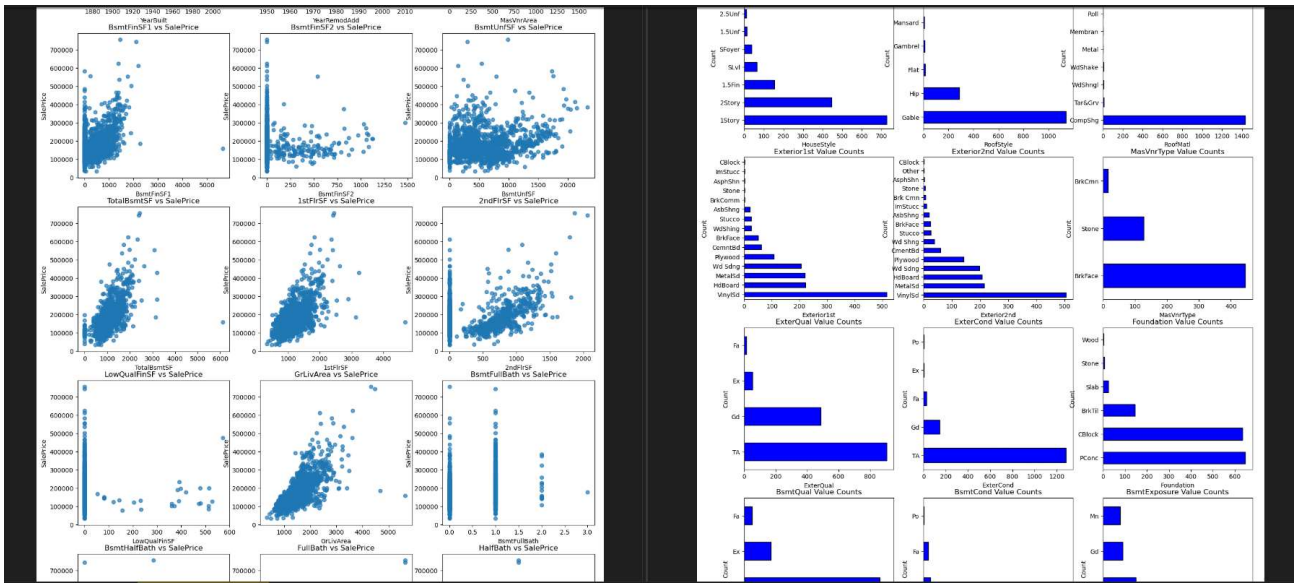


Figure 5 - Sample Visualization of Available Data: This visualization provides a snapshot of the dataset, displaying both numerical and categorical data.

This meticulous data analysis provided crucial insights into feature selection and potential relationships between variables and sale prices, laying a solid foundation for subsequent modeling.

Neural Network Creation

Creating a neural network, `nn_template.ipynb` was specifically designed for developing trial models.

Leveraging data retrieved from a SQLite database, the template Jupyter Notebook contains a section dedicated to data processing and feature engineering. It extracts the target feature, SalePrice, and partitions the data into training, testing, and validation sets before normalizing the data.

Implementation of a Keras Tuner automates exploration of hyperparameter space, facilitating optimization of the configuration of hidden layers, neurons, and activation functions. Keras Tuner does not inherently support optimization of R-squared, a metric quantifying variance in predicted values. The Notebook utilizes mean absolute error as a surrogate metric, to mitigate the influence of outliers.

Post-tuning, the model is constructed selecting the best hyperparameters, incorporating early stopping to counteract potential overfitting.

Comprehensive evaluation metrics include R-squared, mean squared error, mean absolute error, and mean percentage error. Visualization of results centers on a scatter plot comparing actual and predicted sales. Key indicators of potential overfitting are presented through the depiction of training and validation loss. A detailed portrayal of model residuals concludes the evaluation.

This structured approach allows for an immense range of data preprocessing and feature engineering, automates hyperparameter optimization, and provides readily interpretable metrics for evaluating model performance before designing subsequent trials.

Trial Models

The initial model successfully met our objective by attaining an R-squared above 0.80. In search of improvement, multiple iterations were conducted, documented in `/trial_models/` and detailed in `House Prices Spreadsheet.xlsx`. Anomalies were identified and removed to enhance model robustness, such as the neighborhood Bluestem, with only 2 values, and properties with more than 4000 square feet of indoor space (see Exploratory Data Analysis).

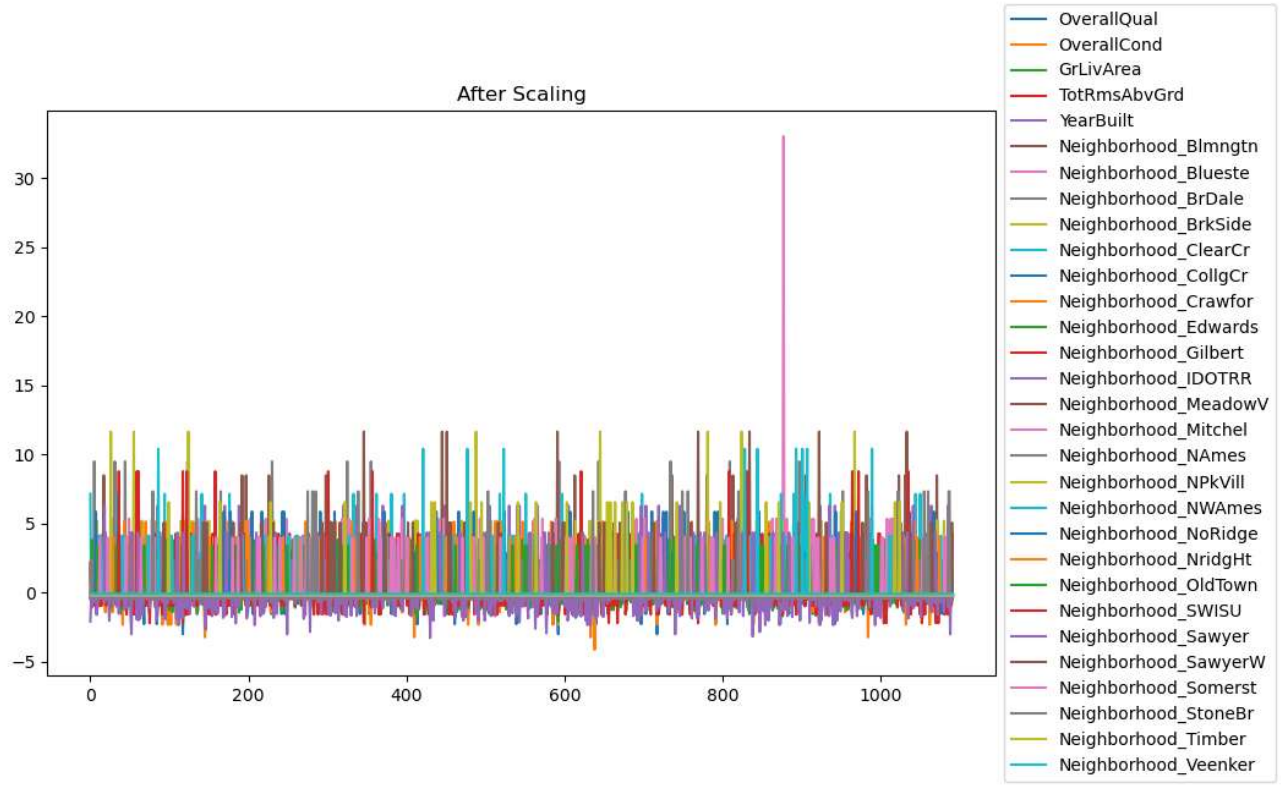


Figure 6 - Post-Normalization BlueStem Neighborhood Outlier: This image illustrates the impact of normalization in revealing a neighborhood outlier.

All non-'Normal' Sale Condition values were excluded from analysis.

The hyperparameter space for the neural network, including layers, neurons, and activation features, underwent systematic manipulation. Various columns, including promising correlated features such as garage area and the presence of a fireplace, were added and dropped in pursuit of optimal model configuration. Diverse feature engineering attempts were made, generating new values for total living square footage, room count, and even kitchen quality. To evaluate the impact of location, rare neighborhood values were binned.

The incorporation of these adjustments aimed to fine-tune the models and explore the nuanced relationships between features and sale prices. Each trial contributed to the comprehensive understanding of the predictive capabilities and limitations of the neural network model.

Random Forest

The incorporation of a random forest model added a comparative dimension to the analysis. Despite the simplicity of the random forest, its ability to handle diverse feature types and potential to capture complex interactions was immediately evident.

Results

The predictive models underwent a rigorous evaluation process. The best-performing neural network was crafted through meticulous feature engineering and hyperparameter tuning, showcasing considerable predictive prowess.

Sample Neural Network Evaluation Metrics:

```
R-squared: 0.8781233961368512
Mean Squared Error: 446280152.9154423
Mean Absolute Error: 15541.401302083334
Mean Percentage Error: -12.6439235740575
```

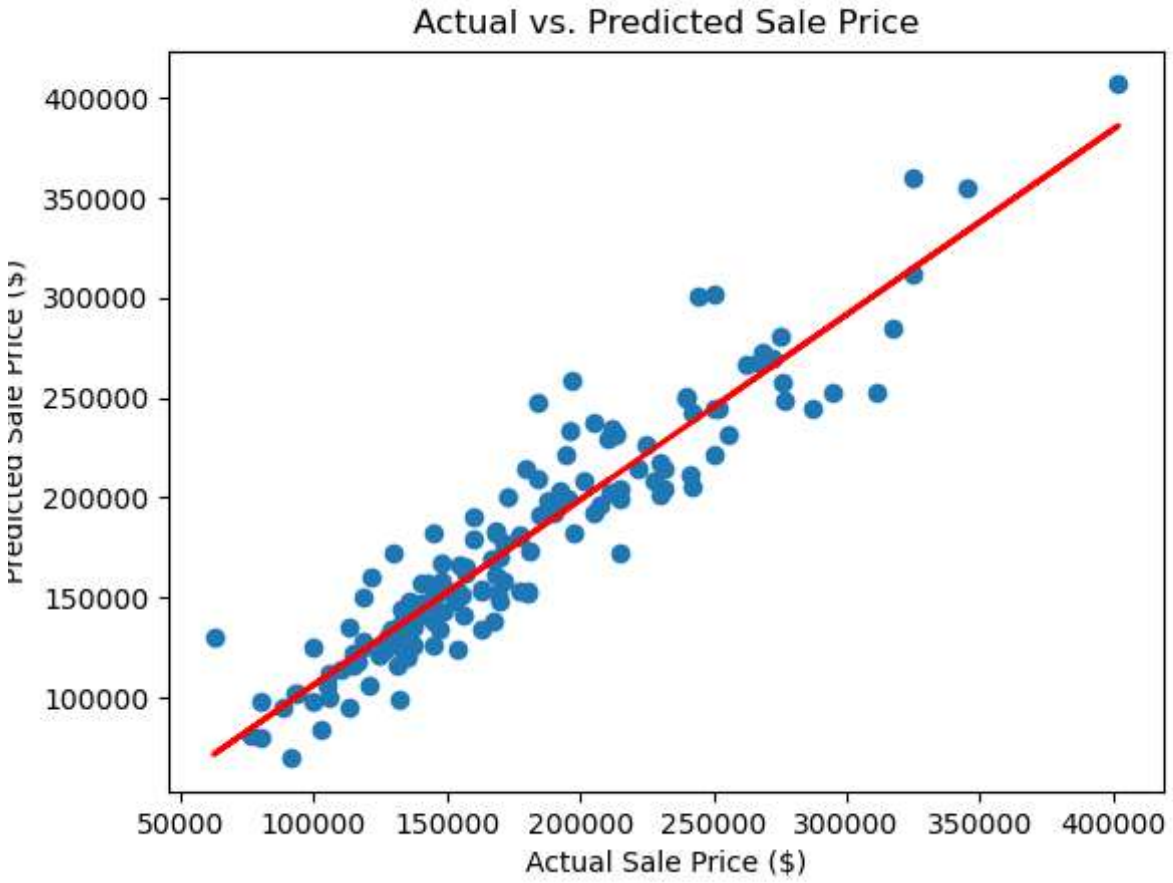


Figure 7 - Scatter Plot of Neural Network Actual vs Predicted Sale Price: The diagram shows that actual and predicted prices are positively correlated or better still have a direct relationship. This implies that any discrepancies between the prices under measure might imply market inefficiencies or could indicate potential errors in our model.

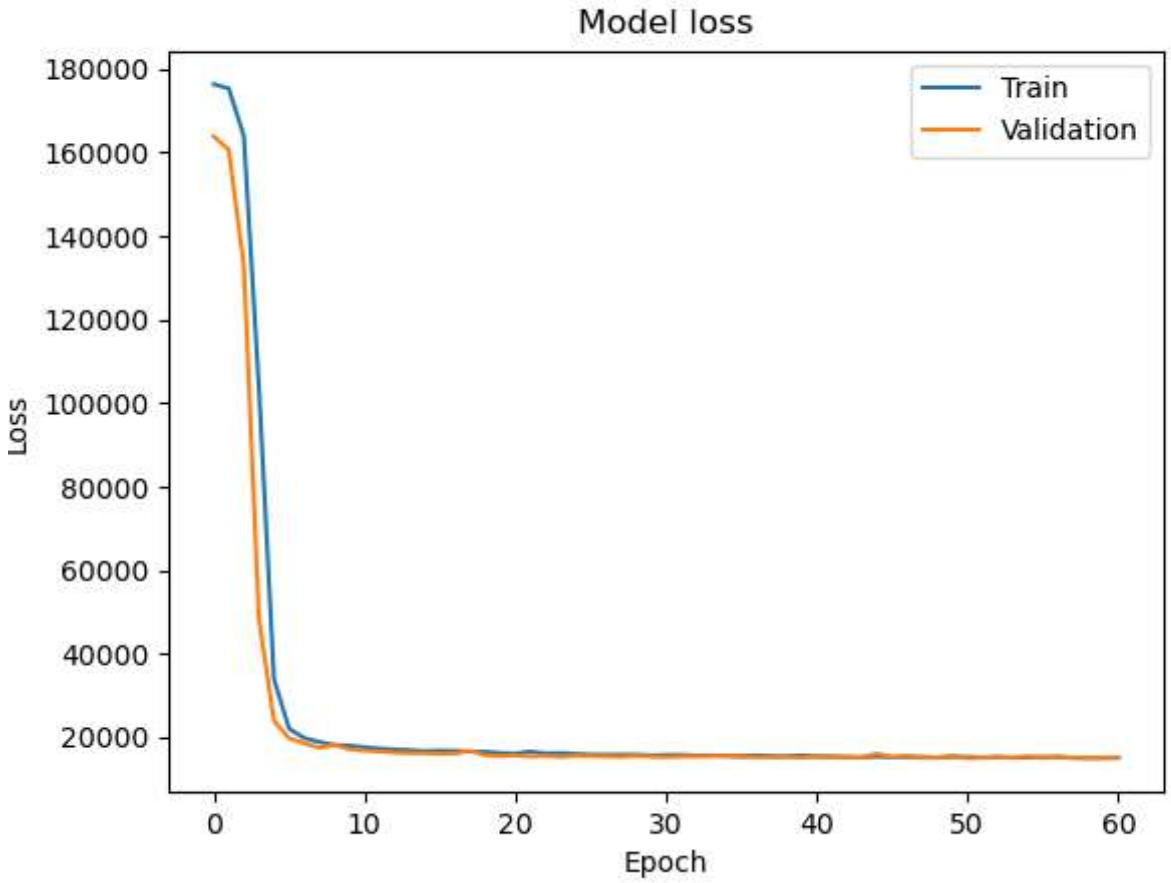


Figure 8 - Neural Network Training and Validation Loss: A decreasing trend in both training and validation loss indicates the model learns and generalizes well.

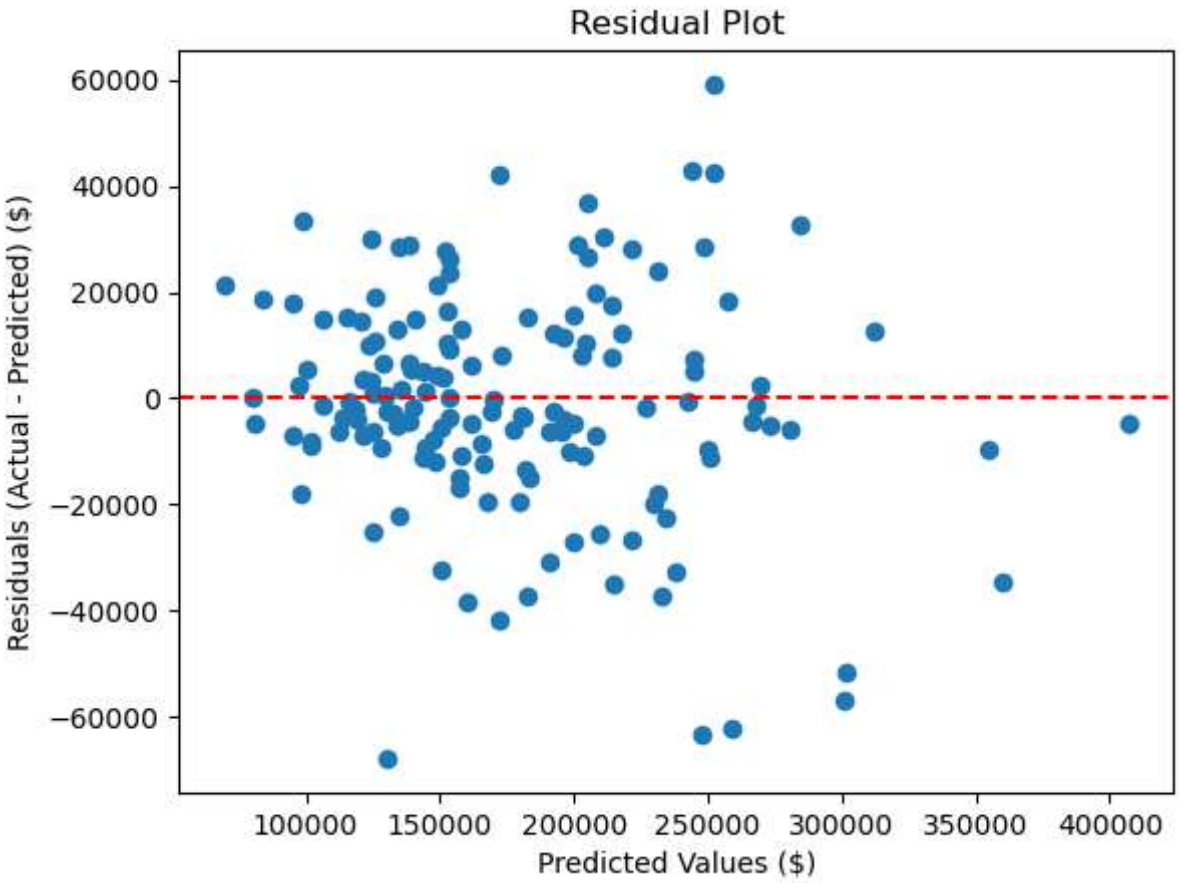


Figure 9 - Neural Network Scatter Plot of Residuals: If the model did not provide a particular pattern, this implies our model did a great job accounting for systematic errors, which gives our results some validity. Random distribution around zero indicates the models capture most economic factors impacting sale prices.

The random forest model, while less intricate, presented a robust alternative, showcasing notable performance without any hyperparameter tuning, providing a valuable benchmark for comparison.

```
Random Forest R-squared: 0.8969215420814829
Random Forest Mean Squared Error: 447478229.20507044
Random Forest Mean Absolute Error: 14698.426263736264
Random Forest Mean Percentage Error: -3.8002864470701163
```

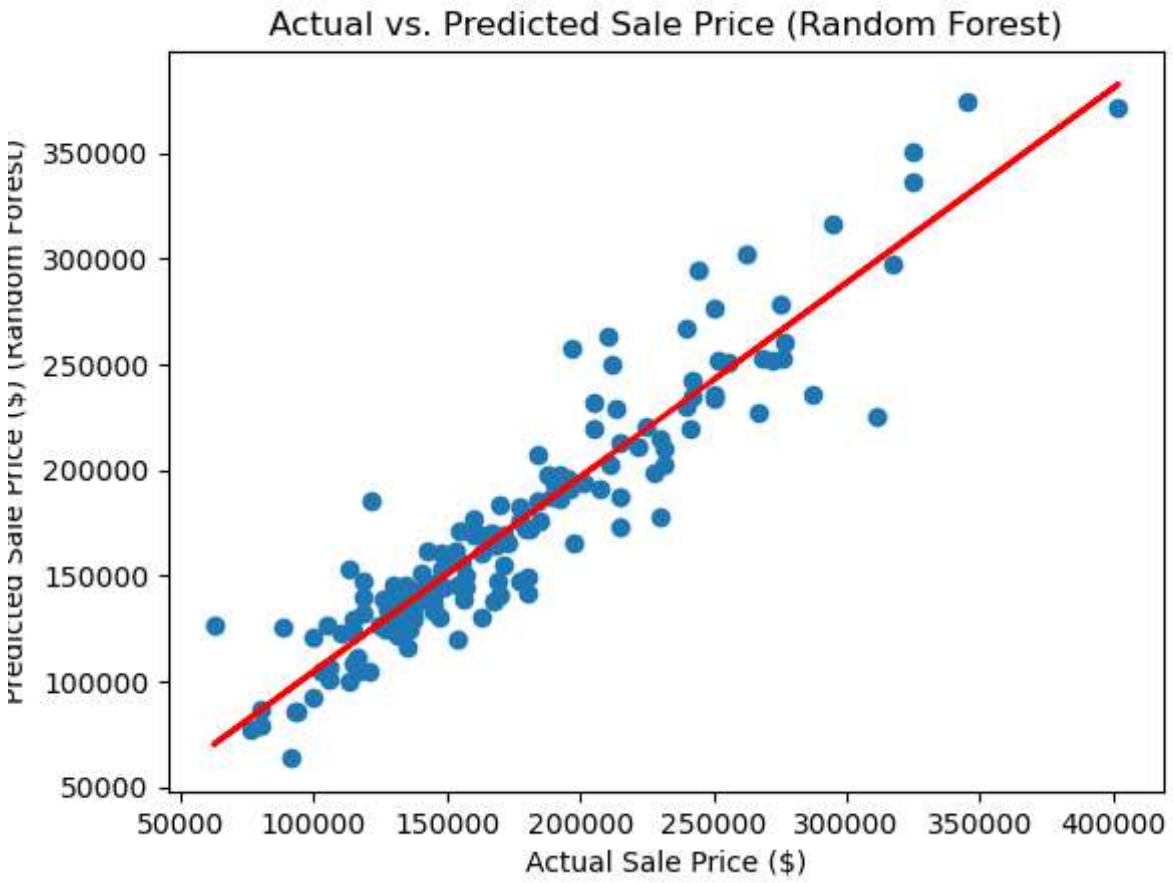


Figure 10 - Scatter Plot of Random Forest Actual vs Predicted Sale Price: The diagram shows that actual and predicted prices are positively correlated or better still have a direct relationship. This implies that any discrepancies between the prices under measure might imply market inefficiencies or could indicate potential errors in our model.

Moreover, the random forest model sheds light on the significance of various features, thereby amplifying our understanding of how these features influence the sale price.

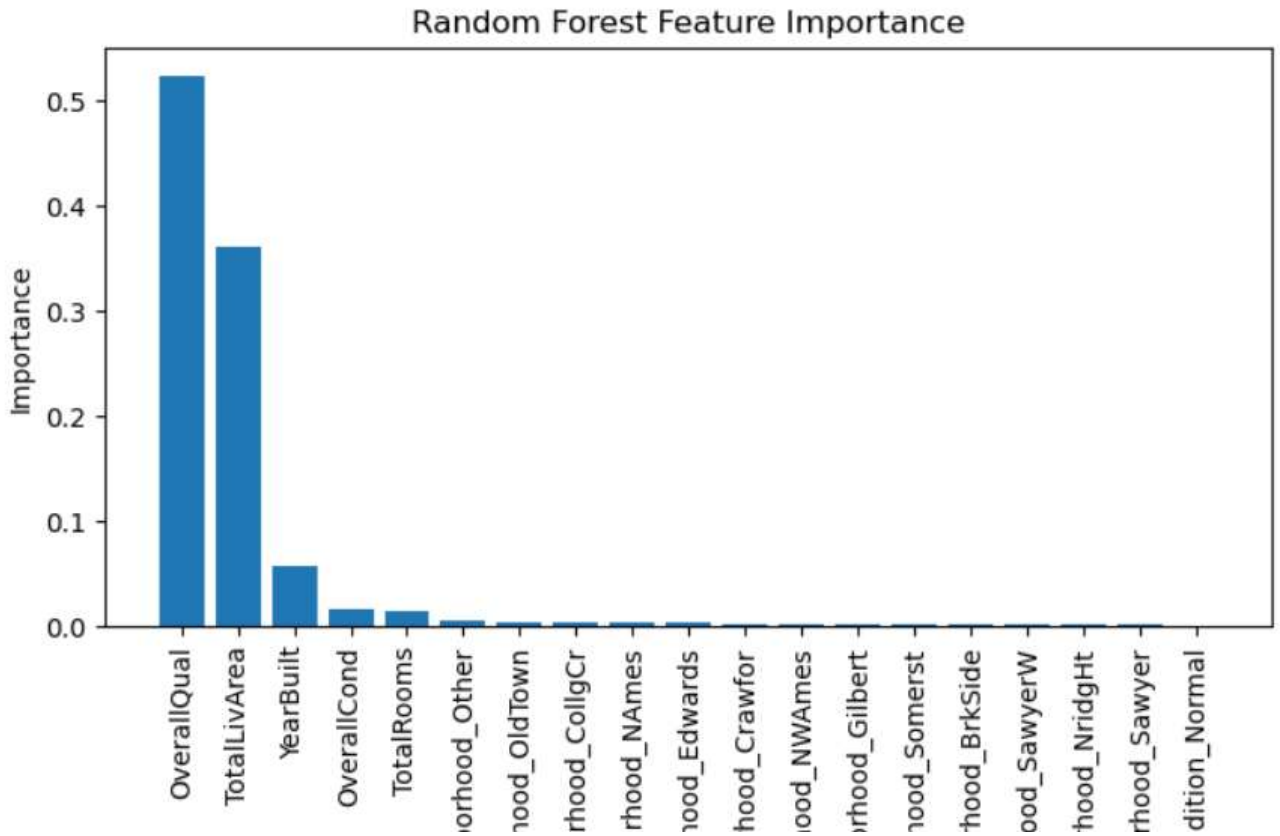


Figure 11 - Random Forest Feature Importance: This plot provides a visual representation of the significance of different features in the final random forest model. This analysis reveals the importance of each feature in influencing the model's predictions, highlighting a distinctive advantage not present in the neural network model.

Discussion

This predictive modeling exercise unveiled insights into the intricate relationship between various residential real estate property features and sale prices. Extensive exploratory data analysis was pivotal in shaping the modeling approach, setting a solid foundation by initially identifying potential features, like overall quality, square footage, room count, and property age. Accomplishing our target R-squared of 0.80 on the initial attempt underscores the efficacy of exploratory data analysis.

The neural network, implemented with meticulous adjustments to hyperparameters and feature engineering, demonstrated commendable performance. The evaluation metrics, including R-squared, mean squared error, mean absolute error, and mean percentage error, provided a comprehensive understanding of the model's predictive capabilities. Visualization tools, such as scatter plots for actual versus predicted sale prices, training and validation loss plots, and residual analyses, enriched the interpretability of the model's behavior.

To complement the neural network trials, a random forest regression model was introduced. The random forest, known for its simplicity and ensemble learning approach, offered a different lens through which to analyze the data. Implemented with the default parameters and a lack of feature engineering, the random forest model's performance highlighted the advantages of simplicity.

The trade-offs between the two models are noteworthy. The neural network, with its capacity for learning intricate patterns, excels in capturing nuanced relationships in the data. However, its susceptibility to overfitting and the need for extensive tuning pose challenges. On the other hand, the random forest, while less prone to overfitting and requiring minimal hyperparameter tuning, may not capture subtle patterns as effectively as the neural network.

In terms of interpretability, the random forest model tends to provide more transparency in understanding feature importance. Random forests offer insights into which features significantly contribute to the model's predictions, making it easier to comprehend the factors influencing the outcome. This interpretability can be crucial, especially in scenarios where understanding the driving forces behind predictions is as important as the predictions themselves.

On the contrary, neural networks, with their intricate architectures and hidden layers, often act as "black boxes," making it challenging to decipher how individual features contribute to the final output. While neural networks can uncover complex relationships, the lack of interpretability may limit their application in contexts where understanding the reasoning behind predictions is essential.

These considerations highlight the importance of balancing predictive power with interpretability. Neural networks shine in big data scenarios, offering exceptional predictive capabilities by capturing intricate patterns. However, their lack of transparency and substantial demands on computational and human resources raise challenges. On the other hand, random forests provide transparency, making them interpretable, and demand less time and computational resources. Yet, they may struggle with extremely large datasets and might miss nuanced relationships present in the data. The choice between these models hinges on navigating this trade-off based on specific project requirements and available resources.

Looking ahead, further avenues for exploration include a more in-depth analysis of categorical features, additional feature engineering experiments, and robustness testing against external factors like the 2000s subprime real estate bubble. Utilizing cross-validation could enhance model generalization, and ongoing refinement is warranted for achieving optimal predictive accuracy.

In summary, the neural network and random forest models offer distinct advantages and trade-offs. The neural network excels in capturing intricate relationships but demands meticulous tuning, while the random forest provides robustness with less complexity. The choice between them hinges on the specific goals of the modeling task and the inherent characteristics of the dataset. In this context, a deeper exploration of the random forest is recommended, given its simplicity and robust performance. This exploration could uncover additional insights and potentially enhance predictive capabilities for this particular project.