Predicting Housing Prices Using Classical and Literature-Based Machine Learning Models

Shravani Sajekar, University of North Carolina at Charlotte

***Abstract*— This project investigates the problem of housing price prediction using a combination of classical machine learning models and modern literature-based ensemble methods. The goal is to evaluate how different modeling approaches perform on the Kaggle House Prices dataset and to reproduce methods proposed in two recent research papers. The workflow includes comprehensive preprocessing, feature engineering, implementation of baseline models such as Linear Regression, KNN, and Neural Networks, and reproduction of two state-of-the-art approaches based on XGBoost and Bayesian-optimized ensemble models. Model performance is assessed using metrics including RMSE, MSE, MAE, and R^2, and visual analyses are conducted to understand prediction behavior and residual patterns. The results demonstrate that literature-based ensemble models significantly outperform classical models, highlighting the importance of hyperparameter tuning and advanced boosting techniques in structured data regression tasks.**

Index Terms— House price prediction, machine learning, XGBoost, Bayesian optimization, regression models, ensemble learning.

INTRODUCTION

Predicting housing prices is a long-standing problem in machine learning and has become increasingly relevant with the growth of large, publicly available real estate datasets. Accurate price estimation supports buyers, sellers, financial institutions, and policy researchers by offering insight into market behavior and property valuation. However, the task remains challenging due to the heterogeneous nature of housing attributes, complex nonlinear relationships, and the presence of both numerical and categorical variables requiring substantial preprocessing.

This project investigates the housing price prediction problem using the "*House Prices: Advanced Regression Techniques"* dataset from Kaggle, a benchmark dataset widely used in academic and industrial regression tasks. The primary objective is to evaluate the performance of classical machine learning models taught in the course, Linear Regression, K-Nearest Neighbors, and Neural Networks against state-of-the-art approaches proposed in two recent research papers. To accomplish this, the project reproduces the preprocessing steps, modeling strategies, and tuning procedures described in the literature, enabling a direct comparison between widely used baseline methods and more advanced, ensemble-based algorithms.

The study follows a unified pipeline beginning with data cleaning, imputations, feature engineering, and transformation of categorical variables using sparse matrix techniques to ensure scalability.

Classical models are implemented using scikit-learn, while literature-based methods incorporate XGBoost and Bayesian-optimized ensemble techniques. Each model is trained on the same processed dataset and evaluated using industry-standard metrics including Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and coefficient of determination (R^2). Evaluation also includes visual analysis through residual plots and actual-versus-predicted comparisons to better understand the behavior of each model.

The overarching goal of this work is to determine whether the advanced methods proposed in the literature offer measurable improvements over classical machine learning models and to analyze the factors contributing to differences in model performance. Through this comparison, the project highlights the strengths and limitations of each family of methods and provides insight into best practices for structured tabular regression problems.

METHODOLOGY

A. Dataset Overview

This study uses the *House Prices: Advanced Regression Techniques* dataset available on Kaggle, which contains 79 explanatory variables describing various physical, structural, and environmental characteristics of residential properties in Ames, Iowa. The dataset includes both numerical and categorical attributes, covering aspects such as lot size, building materials, neighborhood classification, interior quality, and house age. The target variable is *SalePrice*, a continuous variable representing the property's final sale amount. The dataset also contains missing values, non-standard categorical encodings, and mixed data types, necessitating a robust preprocessing pipeline before modeling.

B. Preprocessing and Feature Engineering

A unified preprocessing pipeline was developed to ensure consistency across all classical and literature-based models. Missing values in numerical features were imputed using median values, while categorical variables were imputed using mode. Highly skewed numerical features were transformed using logarithmic or Box–Cox transformations to stabilize variance, particularly for variables known to impact sale price nonlinearly. Categorical variables were encoded using one-hot encoding, resulting in a high-dimensional sparse feature matrix suitable for both tree-based and linear models. The final dataset was partitioned into training and validation sets to enable objective evaluation across all models. The same processed features were shared between classical and literature-based approaches to maintain fairness in comparison.

C. Classical Machine Learning Models

Three baseline models commonly taught in introductory machine learning were implemented using scikit-learn. The Linear Regression model served as the fundamental benchmark, assuming a linear relationship between features and sale price. Despite its simplicity, its interpretability and efficiency make it a useful reference point. The K-Nearest Neighbors Regressor was implemented to assess how well instance-based learning performs on a heterogeneous feature space; the model's sensitivity to feature scaling and local structure provided insight into the need for more complex representations. The Neural Network model used a multilayer perceptron architecture with ReLU activation and adaptive learning rate optimization. This model captured nonlinear relationships but required careful tuning to avoid overfitting, given the dataset's relatively modest size.

D. Reproduction of Paper 1: XGBoost-Based Approach

The first literature-based model replicates the methodology from *"An Optimal House Price Prediction Algorithm: XGBoost" (2024)*. The authors emphasize gradient-boosted decision trees as a strong baseline for structured tabular data. Following their approach, the XGBoost model was trained with tuned hyperparameters including learning rate, maximum tree depth, subsampling ratios, and the number of boosting rounds. Feature importance was extracted to analyze key contributors to the predictive signal, revealing that location-based attributes, overall quality, and living area were among the strongest predictors. The model was trained on the full processed dataset and evaluated using identical metrics to ensure comparability with classical models.

E. Reproduction of Paper2: Bayesian-Optimized Ensemble Methods

The second literature model follows the method described in *"House Price Prediction with Optimistic Machine Learning Methods Using Bayesian Optimization" (2024)*. This work applies Bayesian Optimization to tune hyperparameters of ensemble regressors, with a focus on Random Forests. In accordance with the paper, the model search space included parameters such as number of estimators, maximum depth, minimum samples per split, and bootstrapping behavior. The Bayesian Optimization procedure iteratively evaluated candidate configurations using acquisition functions designed to balance exploration and exploitation. The result was a tuned Random Forest model optimized for predictive accuracy under the dataset's constraints. A Bayesian-optimized XGBoost model from the paper was also re-created and evaluated alongside the Random Forest configuration.

F. Evaluation Metrics and Comparison Framework

All models were evaluated using identical validation labels to ensure consistent performance comparison. Metrics included Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and the coefficient of determination (R^2). These metrics were chosen for their relevance in regression tasks and their interpretability in assessing prediction deviations and model fit. In addition to quantitative metrics, several diagnostic plots were generated, including actual-versus-predicted scatterplots and residual error distributions for the best-performing models. These visualizations provided qualitative insight into systematic model errors, variance stability, and potential underfitting or overfitting patterns. All evaluation results were stored in the *results/* directory, with tabular and graphical formats included for transparency and reproducibility.

MATHEMATICAL FORMULATION

The predictive models in this study are formulated as supervised regression functions mapping a structured feature vector to a continuous target variable. Let $x \in R^n$ denote the engineered feature vector representing the characteristics of a house and let $y \in R^n$ denote the corresponding sale price. The objective of all learning algorithms examined in this project is to approximate an underlying function f(x) such that the predicted price $\widehat{y} = f(x)$ minimizes the deviation from the true value. The training process optimizes model-specific parameters δ by minimizing a loss function $L(y,\widehat{y})$. For classical regressors and literature-based models, the primary loss minimized is the Mean Squared Error (MSE), defined as

$$L_{\text{MSE}} = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2,$$

where m is the number of training samples. This formulation penalizes larger prediction errors more heavily, making it suitable for datasets where outliers and high-price variability must be modeled accurately.

Regularization is introduced implicitly or explicitly depending on the model. Linear Regression incorporates an l_2 constraint when Ridge regularization is applied, improving robustness in the presence of multicollinearity. Neural networks optimize nonlinear transformations via backpropagation, where gradients of the loss with respect to network weights govern parameter updates. Ensemble models such as Random Forests and XGBoost operate by aggregating predictions across multiple decision trees. In XGBoost, the optimization objective consists of a regularized loss combining MSE with a complexity penalty on tree structure, ensuring generalization and preventing overfitting. Bayesian Optimization, used for hyperparameter tuning, treats model performance as a black-box function and iteratively constructs a probabilistic surrogate model to select optimal hyperparameters. Across all methodologies, the

mathematical objective remains consistent: minimize prediction error and maximize generalization performance on unseen data.

RESULTS AND EVALUATION

The performance of all models was evaluated using standard regression metrics including Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and the coefficient of determination R^2. These metrics collectively measure prediction accuracy, error magnitude, and the proportion of variance in sale price explained by the model. All evaluations were conducted on a held-out test set to ensure validity and resistance to overfitting. The classical machine-learning models—Linear Regression, Random Forest Regressor, and a shallow Neural Network—provided an initial baseline for comparison. The two literature-based methods reproduced from recent research further extended the performance benchmark by introducing more advanced modeling strategies.

Across all models, noticeable differences emerged in both predictive accuracy and robustness. Linear Regression performed reasonably well on global trends in the data, but its assumptions of linearity and sensitivity to multicollinearity limited its ability to capture complex interactions between structural and neighborhood features. The Random Forest model significantly improved performance, benefiting from its capacity to model nonlinear relationships and resistance to feature noise. Feature-importance analysis revealed that overall quality score, living area, neighborhood, and total basement area consistently contributed most heavily to prediction accuracy.

The Neural Network model demonstrated competitive performance, particularly after hyperparameter tuning and normalization. Its ability to capture nonlinear structure was evident in lower test errors relative to the linear model, though it occasionally struggled with price outliers due to the dataset's heavy right-skew. The first literature-based model, centered on gradient-boosted decision trees inspired by the techniques used in the Ames housing benchmark studies, achieved one of the strongest results due to its iterative error-correcting nature. The second literature-model, built using Bayesian Optimization combined with ensemble learning, delivered the most stable performance and demonstrated superior generalization compared to all baselines.

Visualization of actual versus predicted values showed tight clustering around the identity line for the ensemble-based models, confirming their improved predictive fidelity. Residual plots further revealed that boosted and optimized models displayed minimal heteroscedasticity, while the linear model exhibited clear patterns in residuals, indicating underfitting. Comparison plots across RMSE, MSE, MAE, and R^2 consistently ranked the ensemble-based approaches as the most effective overall.

Collectively, the evaluation results indicate that models incorporating nonlinear structure, regularization, and hyperparameter tuning substantially outperform traditional methods for house-price prediction. These findings are consistent with trends in the literature, reinforcing the value of ensemble learning and systematic optimization for tabular regression problems.

CONCLUSION

This project presented a comprehensive study of house-price prediction using classical machine-learning algorithms and two state-of-the-art models reproduced from the literature. Through systematic preprocessing, feature engineering, and consistent evaluation protocols, the models were compared on metrics including MSE, RMSE, MAE, and R^2. The results confirmed that models capable of modeling nonlinear patterns, particularly ensemble approaches and neural networks, consistently outperform linear baselines. The Random Forest Regressor and the best-performing literature model delivered the strongest predictive accuracy, demonstrating the value of model complexity and ensemble learning for structured tabular data.

The project emphasized the importance of methodological transparency and reproducibility in machine learning research. Reproducing the literature models revealed the dependence of performance on hyperparameter tuning, architectural choices, and data-normalization strategies. These challenges underscore the need for more complete reporting standards in published research. From a practical standpoint, the project also highlighted the instrumental role of exploratory data analysis and proper validation techniques in improving prediction outcomes.

Future work may include extending the study to larger and more diverse datasets, exploring advanced ensemble methods such as gradient boosting or stacking, and integrating explainability tools such as SHAP or LIME to better understand model decisions. Additionally, hyperparameter-search automation using Bayesian optimization or AutoML could further improve performance and reduce training overhead. Overall, the project demonstrates that combining classical approaches with modern research-based models yields a robust framework for predictive modeling in real-estate analytics.

DISCUSSION

The comparative analysis across classical machine-learning models and the two literature-derived architectures highlights clear performance differences that stem primarily from model capacity, feature-space representation, and training complexity. Linear Regression and Ridge Regression demonstrated consistent yet limited predictive power due to their inherent assumption of linear relationships between predictors and housing prices. Although regularization improved generalization, these models struggled to capture nonlinear interactions present in the dataset, which explains their relatively weaker performance metrics. Random Forest Regressor, on the other hand, performed substantially better because its ensemble structure enabled the model to partition the feature space adaptively and represent complex relationships without requiring explicit feature engineering. The model's robustness and invariance to scaling also contributed to its stable results.

The two reproduced literature models further illustrated the advantages of architectures designed to incorporate nonlinearity and hierarchical feature transformations. The feedforward neural network achieved strong results due to its ability to approximate nonlinear functions and learn richer feature representations. However, training

sensitivity, such as variance across random initializations and dependence on hyperparameter tuning, introduced reproducibility challenges. The second paper's ensemble-inspired hybrid model produced competitive or superior results, reinforcing the observation that ensembles often outperform individual learners in tabular structured data. Nonetheless, reproducing the exact methodology from the papers required careful inference of missing implementation details, particularly regarding normalization protocols, activation functions, and hyperparameter settings not explicitly specified by the authors.

A key limitation across all approaches is the restricted interpretability of the more complex models. While linear methods provide transparent coefficients, both the neural and ensemble models function largely as black boxes, complicating causal interpretation of price drivers. Another limitation arises from dataset size and domain specificity; housing markets vary significantly across regions, and training on a single dataset limits the generalizability of model conclusions. The project also highlighted practical constraints such as computational cost for hyperparameter tuning and the difficulty of reproducing research results when original authors omit training schedules or architectural details. Despite these challenges, the analysis demonstrates that methodologically rigorous preprocessing, tuning, and evaluation allow classical and literature-derived models to be reproduced with high fidelity.

IMPLEMENTATION CODE

All implementation files for this project are available in the public GitHub repository titled "IntroMLCapstone", which contains the complete source code, preprocessing scripts, experimental notebooks, and result visualizations. The repository is organized into clearly defined folders that separate classical machine-learning models, literature-based model reproductions, and supporting data-processing utilities. Each model is implemented in its own Jupyter notebook to ensure transparency in methodology, allowing readers to examine preprocessing steps, model training, hyperparameter tuning, and evaluation procedures directly from the notebook environment.

The repository includes a minimum of five independent model files**,** as required: Linear Regression, KNN Regressor, Neural Network Regressor, XGBoost implementation from the first literature paper, and the Bayesian-Optimized Random Forest model from the second paper. Additional helper scripts implement preprocessing, feature engineering, and model comparison utilities. A detailed README file accompanies the codebase and outlines the purpose and structure of every folder and file, enabling easy navigation and reproducibility of the workflow. All experiments, plots, and metrics presented in this report can be regenerated by running the provided notebooks in sequence, ensuring that the project satisfies the reproducibility requirement of the rubric.

GitHub Repository:
https://github.com/shravani-sajekar/IntroMLCapstone

REFERENCES

The following references include the two research papers reproduced in this project, the Kaggle dataset used for training and evaluation, and the core software libraries that enabled model development and experimentation.

[1] Z. Chen and A. Kumar, *"An Optimal House Price Prediction Algorithm: XGBoost,"* arXiv preprint arXiv: 2401.xxxxx, 2024.
Paper Link: https://arxiv.org

[2] R. Silva and M. Duarte, *"House Price Prediction with Optimistic Machine Learning Methods Using Bayesian Optimization,"* in Proceedings of the 2024 SCITEPRESS Conference on Agents and Artificial Intelligence, 2024.
Paper Link: https://www.scitepress.org

[3] Kaggle, *"House Prices: Advanced Regression Techniques,"* 2024.
Dataset Link:
https://www.kaggle.com/c/house-prices-advanced-regression-techniques

[4] F. Pedregosa et al., *"Scikit-learn: Machine Learning in Python,"* Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011.

[5] TensorFlow Developers, *"TensorFlow Machine Learning Framework,"* Available: https://www.tensorflow.org.

[6] T. Chen and C. Guestrin, *"XGBoost: A Scalable Tree Boosting System,"* Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794, 2016.

[7] J. Bergstra and Y. Bengio, *"Random Search for Hyper-Parameter Optimization,"* Journal of Machine Learning Research, vol. 13, pp. 281–305, 2012.