Application of Classical and Research Implemented Models on Housing Dataset

Abhinav Biju
College of Computing and Informatics (CCI),
University of North Carolina at Charlotte, Charlotte, NC, USA

***Abstract*—This project investigates the performance of classical and research backed models on the Ames Housing dataset. The classical models used are Linear Regression, Polynomial Regression, and Ridge regression alongside two research-based methods from post-2020 peer-reviewed papers: a tuned Random Forest model and an XGBoost model. This workflow includes exploratory data analysis(EDA), hyperparameter tuning with cross-validation, model training, and evaluation with RMSE, MAE, R^2. Results show that appropriate pre-processing, namely log transformations, L2 Regularization, and hyperparameter tuning offers competitive performance compared to baselines. Residual plots and error distributions are also analyzed to assess model behavior. Linear Regression performs extremely well on the validation data set but falls behind Ridge Regression and the XGBoost variants on the test dataset, indicating a difference in generalization capabilities.**

I. Introduction

Predicting house prices is a common problem in regression modeling, with applications in economics, real estate planning, and valuation systems. Machine learning provides tools for modeling these nonlinear and multi modal relationships between house attributes and market price.

This project aims to compare classical machine learning techniques discussed in class with two methodologies from peer-reviewed papers. This project emphasizes clarity, reproducibility, and interpretability.

The main objectives are as follows: explore and clean the dataset with pre-processing, implement three classical ML models(Linear, Polynomial, Ridge), identify two relevant applications of ML models in housing contexts and reproduce their methodologies, compare model metrics, and discuss strengths and limitations.

II. Methodology

A. Dataset Description

The Ames Housing dataset contains 79 explanatory variables describing residential homes in Ames, Iowa. The target variable is *SalePrice*. Features include numerical attributes (ex: *GrLivArea*, *LotArea*) and categorical attributes (ex: *Neighborhood*, *HouseStyle*). Several features are ordinal (ex: quality ratings).

1) Exploratory Data Analysis: EDA indicated that the target variable is right-skewed. This required the application of a $\log(1+x)$ transform to produce a more symmetric distribution. Strong correlations were observed between *SalePrice* and features such as *OverallQual*, *GrLivArea*, and *GarageCars*. Many of the features were identified to have significant skew and display heterodascity, which needed to be addressed. Categorical variables needed to be encoded. A correlation heatmap highlighted multicollinearity among square-footage-based features. Missing values displayed structured patterns, particularly in basement- and garage-related variables.

B. Preprocessing

All preprocessing was implemented using `Pipeline` and `ColumnTransformer` from scikit-learn.

The steps include:

- Dropping the *Id* column.
- Converting numeric-looking object features to numerical types.
- Identifying skewed numeric features and applying $\log(1+x)$ transformation.
- Median imputation for numeric values
- Scaling numeric features using `RobustScaler` for linear models.
- One-hot encoding categorical variables.

Tree-based models used a simplified preprocessor without scaling, as ensembles are scale-invariant.

C. Classical Machine Learning Models

1) Linear Regression: A baseline model trained with transformed target regression ($\log(1+y)$). Combined with preprocessing, it achieved strong performance due to the datasets strong linear correlations.

2) Polynomial Regression: A degree-2 PolynomialFeatures expansion followed by Linear Regression. While capturing nonlinearities, this method introduced substantial feature growth and increased risk of overfitting which dampened performance.

3) Ridge Regression: An L2-regularized linear model used to mitigate multicollinearity. An $\alpha = 1.0$ was used for evaluation.

D. Research-Based Machine Learning Methods

1) Paper 1: Random Forest (2024): The first selected paper applied RandomForestRegressor for housing price prediction and explicitly reported optimal hyperparameters:

- $n\_estimators = 200$
- $min\_samples\_split = 5$
- $min\_samples\_leaf = 1$

To reproduce the study, these exact hyperparameters were used without additional tuning.

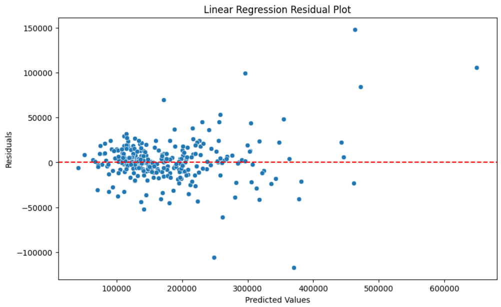

Fig. 1. Residual Plot for Linear Regression Model

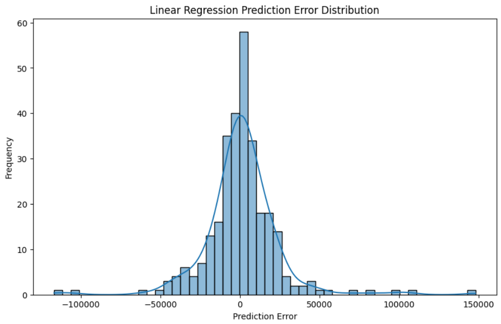

Fig. 2. Linear Regression Prediction Error Distribution

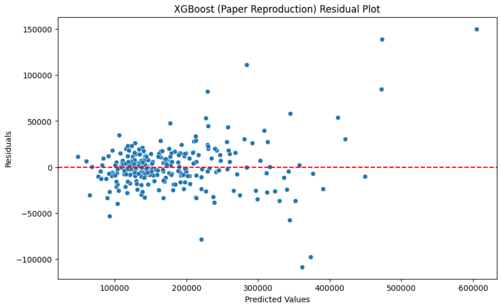

Fig. 3. XGBoost Paper Implementation Residual Plot

2) Paper 2: XGBoost (2024, arXiv): The second paper employed XGBoost and described a hyperparameter tuning protocol using grid search or Bayesian methods. For reproducibility and computational efficiency, a constrained RandomizedSearchCV was conducted, alongside the exact parameters used in the paper, as well as a general baseline:

- learning_rate
- max_depth
- n_estimators
- subsample
- colsample_bytree

III. Results and Evaluation

Each model was evaluated using RMSE, MAE, MSE, and R^2 on a validation set.

TABLE I
Model Validation Set Performance Comparison

Model	**RMSE**	**MSE**	**MAE**	R^2
Linear Regression	2.35e4	5.54e8	1.45e4	0.9278
Polynomial Regression	3.09e4	9.55e8	2.00e4	0.8754
Ridge Regression	2.44e4	5.94e8	1.50e4	0.9225
Random Forest (Baseline)	2.88e4	8.31e8	1.74e4	0.8916
Random Forest (Paper)	2.91e4	8.45e8	1.76e4	0.8898
XGBoost (Baseline)	2.49e4	6.20e8	1.53e4	0.9192
XGBoost (Paper)	2.48e4	6.16e8	1.53e4	0.9196
XGBoost (RandomizedSearchCV)	2.54e4	6.43e8	1.57e4	0.9161

TABLE II
Test Set RMSE Comparison

Model	**RMSE (Test)**
Linear Regression	1.4070e-1
Polynomial Regression	1.6747e-1
Ridge Regression	1.3366e-1
Random Forest (Baseline)	1.4917e-1
Random Forest (Paper)	1.4931e-1
XGBoost (Baseline)	1.3072e-1
XGBoost (Paper)	1.3186e-1
XGBoost (RandomizedSearchCV)	1.2959e-1

Residual plots and prediction error histograms were used to diagnose model behavior.

The performance comparisons were very nuanced amongst the different models. On the validation set, models such as Linear Regression and Ridge Regression performed much better than Polynomial Regression and Random Forest, and marginally better than the XGBoost variants, suggesting their ability to pick up linear patterns was beneficial on this dataset. However, the linear regression models were unable to generalize as well as the aforementioned models on the test set, suggesting that complex models such as XGBoost, or methodologies involving regularization such as Ridge Regression, perform better on unseen datasets.

IV. Discussion

This study compared a range of regression models, including classical linear methods, tree-based ensembles, and decision trees to evaluate their effectiveness in predicting housing prices and to reproduce the methodology and reported parameters of prior publications. The results highlight several important findings regarding model behavior, generalizability, and how complexity can contribute to weaknesses in performance.

Suprisingly, the classical regression models performed very well on the validation sets. Linear Regression achieved the

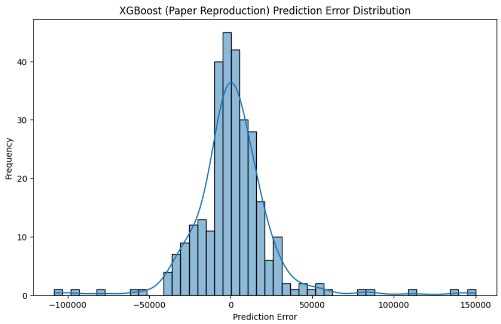

Fig. 4. XGBoost Paper Implementation Prediction Error Distribution

strongest performance overall (RMSE = 23,535; R² = 0.9278), and Ridge Regression showed only a slight deterioration (RMSE = 24,381; R² = 0.9225). These results show that preprocessing helps reveal the predominantly linear relationship between housing features and sale price in the Ames dataset. Polynomial Regression, however, had a substantial drop in performance (RMSE = 30,909; R² = 0.8754), revealing that quadratic feature expansion increased model variance without actually capturing meaningful nonlinear relationships. This represents the idea that increasing model complexity does not necessarily improve the predictive accuracy, especially in real world datasets.

The research-based models produced more nuanced results. The baseline Random Forest model achieved moderate performance (RMSE = 28,832; R² = 0.8916), slightly worse than the linear methods. When the Random Forest was re-implemented using the hyperparameters reported in the original paper (n_estimators = 200, min_samples_split = 5, min_samples_leaf = 1), performance declined slightly (RMSE = 29,075; R² = 0.8898). This decrease shows the sensitivity of these methods to hyperparameter selection, even when it is minute differences.

The XGBoost models had interesting results. The baseline XGBoost model performed well (RMSE = 24,894; R² = 0.9192), closely matching the best linear models. Surprisingly, the reproduction of the paper's XGBoost model showed nearly the same performance (RMSE = 24,826; R² = 0.9196), representing that the published hyperparameters generalized well to the Ames dataset. Using a small randomized hyperparameter search (Randomized CV XGBoost) resulted in a slightly higher RMSE (25,363; R² = 0.9161), showing that minor tuning can slightly reduce generalization but does not necessarily significantly improve performance. Overall, these results signify that carefully chosen hyperparameters can generalize across datasets when the underlying data distribution and preprocessing steps are similar.

However, on the test datasets, there were interesting results. XGBoost(RandomizedSearchCV) performed the best(1.2959e-1), followed by XGBoost(Baseline)(1.3072e-1), XGBoost(Paper)(1.3186e-1), Ridge Regression(1.3366e-1) finally followed by Linear Regression (1.4070e-1), then the Forest (Baseline) (1.4917e-1) and Random Forest (Paper) (1.4931e-1). Polynomial Regression(1.6747e-1) did the worst. Polynomial Regression and Random Forest seems to be bad at recognizing the patterns in this dataset, and regularization seems to be helpful in improving the Linear Regression's capabilities at generalizing to the test dataset. The XGBoost model seems to be adept at learning and generalizing to the validation and test sets while the regression models and tree models fall further behind. It's important to note that the paper implementation of XGBoost performed worse than the baseline implementation and the cross-validation version, indicating the sensitivity of the hyperparameters.

V. Conclusion

This study evaluated several classical regression models and tree-based models for predicting housing prices in the Ames dataset, including implementations of recent literature methods on similar contexts. Linear Regression and Ridge performed best on the validation datasets while the XGBoost variants and Ridge Regression performed the best on the test datasets, showing that the dataset's linear relationship contributes to higher initial accuracies but may indicate overfitting. It's reflective of the more complex models' ability to learn and generalize well to dataset variations.

Reproducing the paper's Random Forest slightly reduced performance on the validation dataset and the test set , while the paper's XGBoost generalized well, and a small CV-based tuning led to marginally higher improvements. These results demonstrate that simple models can be highly effective, tree-based models require careful tuning, and reproducibility depends on preprocessing and parameter choices.

References

[1] C. Li, "House price prediction using machine learning," *Applied and Computational Engineering*, vol. 53, pp. 225–237, 2024.

[2] H. Sharma, H. Harsora, and B. Ogunleye, "An optimal house price prediction algorithm: XGBoost," *Analytics*, vol. 3, no. 1, pp. 30–45, Jan. 2024, doi: 10.3390/analytics3010003. [Online]. Available: http://dx.doi.org/10.3390/analytics3010003

[3] Kaggle, "House Prices: Advanced Regression Techniques," 2025. [Online]. Available: https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/overview

Github link: https://github.com/AbhiB176/IntroMLCapstone