Huber Loss based XGBoost For House Price Prediction

*Aadil Hussain Teeli*

[*hussainaadil378@gmail.com*](mailto:hussainaadil378@gmail.com)

*Department of Computer Science, University of Kashmir, Srinagar, India*

*M. Arif Wani*

[*awani@uok.edu.in*](mailto:awani@uok.edu.in)

*Department of Computer Science, University of Kashmir, Srinagar, India*

*Abstract*—The housing market is a crucial component of the global economy, with housing prices serving as a key indicator of economic health. Accurate prediction of house prices is essential for both homeowners and real estate professionals. This paper delves into the application of the Huber loss function in the context of XGBoost regression models. Our research explores the impact of the Huber loss on model performance, considering aspects such as dataset characteristics, distribution of residuals, and the presence of outliers. The paper provides insights into scenarios where the Huber loss demonstrates superiority over traditional loss functions, shedding light on its potential benefits and limitations. The study aims to guide researchers and practitioners in understanding the applicability of the Huber loss within the XGBoost framework, offering valuable insights into its effectiveness in improving model robustness and predictive accuracy. The paper evaluates the model's performance using several regression metrics, including Mean Absolute Error (MAE), Mean Squared Error (MSE), and R-squared (R^2) to assess prediction accuracy. The results of this study demonstrate that XGBoost Regression provides a robust and accurate framework for house price prediction. The optimized model outperforms traditional linear regression models in terms of both prediction accuracy and feature selection.

Keywords—Huber loss, XGBoost regression model, Mean Squared Error (MSE), R-squared (R^2), Linear regression model.

# **Introduction**

The world of house price prediction is not just a matter of luck or guesswork; it's a complex field that combines data analysis and mathematical modelling. Being able to accurately predict house prices isn't just about convenience for homebuyers and sellers; it has far-reaching implications for the broader economy.

The challenge here is that many factors contribute to fluctuations in house prices. It's not just about the number of bedrooms or square footage. It's about the essence of the neighbourhood, government regulations, economic policies, and other elements that are not as easily quantified. This makes predicting house prices a complex task.

Traditionally, the field of real estate has relied on linear regression models for price prediction. While these models have served their purpose, the complexity and multidimensionality of real estate data present challenges that often surpass the capabilities of linear regression. This is where XGBoost, an advanced machine learning algorithm, enters the scene as a game-changing solution.

XGBoost, short for "Extreme Gradient Boosting," has gained prominence for its prowess in handling structured data, making it particularly well-suited for real estate price prediction. This algorithm offers the ability to capture complex, nonlinear relationships within datasets, a crucial feature when dealing with the multitude of factors influencing property values.

In our research we present a robust approach to enhance the accuracy and efficiency of house price prediction by customised loss function known as Huber Loss function. The Huber loss function provides a mechanism to mitigate the influence of outliers by transitioning from quadratic to linear behavior beyond a certain threshold, as opposed to the unyielding quadratic behavior of MSE. This transition offers an adaptive compromise, allowing models to effectively handle both small and large residuals.

Our investigation considers critical factors such as the characteristics of the dataset, the distribution of residuals, and the impact of outliers. By doing so, we seek to elucidate scenarios where the Huber loss emerges as a robust alternative, offering improved model performance in the presence of outliers while maintaining computational efficiency.

As we delve into the empirical findings, we not only underscore the advantages of the Huber loss but also outline its practical implications for researchers and practitioners seeking to enhance the robustness and predictive accuracy of their regression models within the XGBoost paradigm. Through this exploration, we contribute insights that shed light on the efficacy of leveraging the Huber loss as a valuable tool in the repertoire of regression modeling techniques.

# **Related Work**

House price prediction has been an active area of research due to its importance in domains like real estate, banking, and urban planning. Various machine learning techniques have been applied for this task.

The value of a particular property depends on the infrastructure amenities surrounding the property. Recently, a few writers’ scopes for finding the best properties for the customers came along with various technologies.

Raghunandhan [1] mentioned the basic data mining concepts of how it works and supporting algorithms for the purpose of prediction. The most important part is which machine learning algorithm is best suited for predicting the house price.

Often the location's environmental conditions decide what kind of price we can expect for different types of houses, Manjula [2] presents various important features to use when forecasting property prices with good precision using a regression model.

A. Varma [3] designed a system that used real-time neighborhood data to get precise real-world valuations using Google maps.

Researchers also showed that there exist relationships between the visual appearance and non-visual attributes such as crime statistics, housing prices, population density, etc. of a city. For instance, “City Forensics: Using Visual Elements to Predict Non-Visual City Attributes” [4],

Using visual attributes to predict the sale price of the property. Hujia Yu, Jiafu Wu (2014) [5] used classification and regression algorithms. According to analysis, living area square feet, roof content, and neighborhood have the greatest statistical importance in estimating the selling price for a home. And prediction analysis can be improved by the PCA technique.

Li Li and kai-Hsuan Chu (2017) [6] studied various algorithms such as Backpropagation neural network (BPN) and Radial basis functional (RBF) neural networks. The use of RBF and BPN models is introduced to identify the difference between the house price index such as Cathy and sinny price index and complicated correlation function to detect the macroeconomic analysis.

Nihar Bhagat, Ankit Mohokar, Shreyash Mane (2016) [7] studied linear regression algorithms for prediction of the houses. The goal of the paper is to predict the efficient price of real estate for customers with respect to their budgets and priorities. Analysis of past market trends and price ranges will predict future house pricing.

# **Metheodology**

In this section, we will discuss how to improve house price prediction using the modification of loss function in XGBoost algorithm. .

## **XGBoost Regressoion:**

Extreme Gradient Boosting (XGBoost) [12] is an open-source library that provides an efficient and effective implementation of the gradient boosting algorithm. It can be used directly for regression predictive modeling. Therefore, we can perform XGBoost regression for advanced prediction modelling. The loss function of XGBoost is the sum evaluated over all the predictions and a regularization function for all predictors:

Loss=

where n is the sample size; yi, are the actual and predicted values of the dependent variable; t is the total number of predictors (trees); ft is the prediction coming from the tth tree.

The XGBoost loss function is composed of two parts:

1. **Training Loss(l(​-):**

This part measures the difference between the predicted  *and actual yi* values for each training instance.

## **Regularization Term (ft​):**

This term includes regularization functions on the individual trees in the ensemble, preventing over-fitting.

## **Evaluation:**

We will be measuring the accuracy of our model by using the following matrices: Mean Absolute Error (MAE), Mean Square Error (MSE), Root Mean Square Error (RMSE). Their Formulas are as follows:

### **Mean Absolute Error (MAE):**

Represents the average absolute difference between the predicted and actual values. A lower MAE indicates better accuracy. The formulae is given below:

MAE=

### **2) Mean Square Error (MSE):**

Represents the average of the squared differences between predicted and actual values. A lower MSE also indicates better accuracy.

MSE=

### **3) RMean Square Error (MSE):**

Represents the square root of the average squared differences between predicted and actual values. RMSE is in the same unit as the target variable. A lower RMSE indicates better accuracy.

RMSE=

## **Proposed Model:**

In this section, we present the layered architecture of our proposed model for house price prediction. Our model is based on the XGBoost with critical modifications including the modification in loss function with hyper-parameter tuning.

### **Data:**

1. **Raw Data:**

This involves the raw data related to house features (e.g., square footage, number of bedrooms, location) and their corresponding prices.

1. **Data Preprocessing:**

Cleaning and pre-processing steps are performed here, which may include handling missing values, encoding categorical variables, scaling numerical features, and other data transformations.

### **Feature Engineering:**

1. **Feature Selection:**

Choose relevant features for the model. This involves analyzing correlations, feature importance, or domain knowledge.

1. **Feature Transformation:**

Apply transformations to features, such as polynomial features, logarithmic transformations, or other engineering techniques

### **Model Selection:**

1. **XGBoost-Huber Loss:**

The core of the architecture involves the XGBoost model. Huber loss is used as the objective function, which is a combination of the squared error and absolute error, providing a balance between robustness and efficiency.

Huber loss, also known as smooth L1 loss, is a loss function commonly used in regression problems, particularly in machine learning tasks involving regression tasks. It is a modified version of the Mean Absolute Error (MAE) and Mean Squared Error (MSE) loss functions, which combines the best properties of both. It can be defined by following formulae:

=

Here,

y is observed value, f(x) is the predicted values and *δ* is the hyper-parameter that determines the threshold beyond which the loss becomes linear.

In our model we created customized loss function that uses the combination of Huber loss and MSE. This modified loss function is a variation of the original XGBoost loss function, and the regularization term (*ft*​)) is implicitly handled by the default regularization parameters in XGBoost.

1. **Hyper-parameter Tuning:**

We utilized the grid search to find the optimal set of hyper-parameters for the XGBoost model. This involves trying different combinations and selecting the ones that result in the best performance.

1. **Training:**

Train the XGBoost model on the preprocessed and engineered data. The model learns to map input features to house price

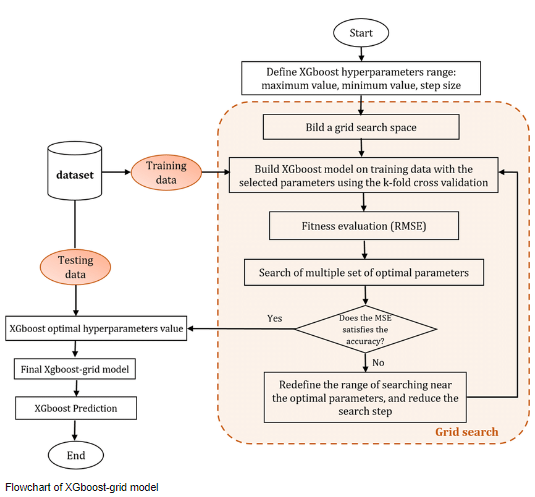
### **Evaluation Metrices:**

1. **Performance Metrics:**

Evaluate the model's performance using relevant metrics such as mean absolute error (MAE), mean squared error (MSE), or Huber loss on a validation set.

1. **Cross-Validation:**

Perform cross-validation to ensure the model's generalization performance. This involves splitting the dataset into multiple folds, training the model on different folds, and evaluating on the remaining folds.



# **model fine-tuning**

In our model, we did hyper-parameter tuning. This experiment aimed to adapt modified loss function to our specific task and data set. This was done to achieve better performance.

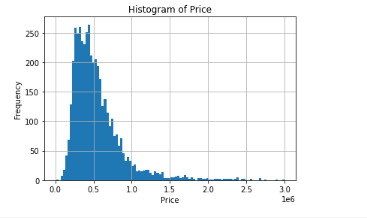
We conducted a systematic approach for finding the best hyper-parameters for an XGBoost model. We obtained best hyper-parameters by using the Grid-Search. It explores a grid of hyper-parameters, evaluates the model's performance using a specified scoring metric, and employs cross-validation to ensure reliable performance assessment. The result is an optimized XGBoost model.

# **Implementation**

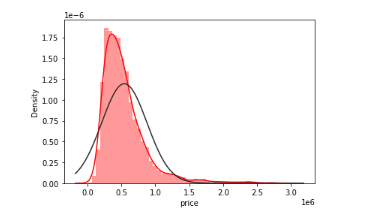
**A. Data set Pre-processing**

Checking the distribution of the target variable ‘price’ this will be the starting point for making meaningful decisions in our data analysis and modelling journey. It's like looking at the map before embarking on a road trip; it helps you plan your route effectively.

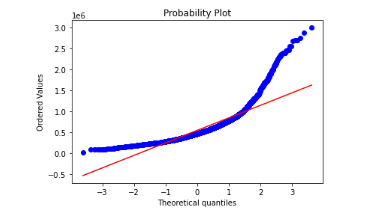
**Figures showing** **Distribution of price before removing outliers:**



**Fig No1**

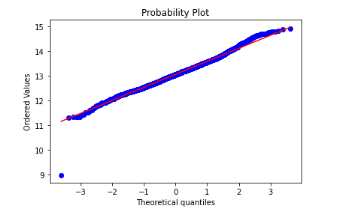
**

**Fig No2 Showing right skew-ness**

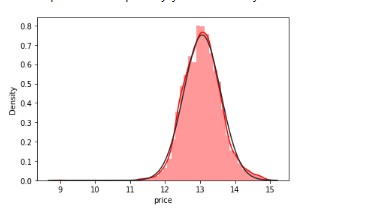


**Fig No3 QQ-plot**

**Figures showing** **Distribution of price after removing outliers:**



**Fig No4 QQ-plot after outlier removal**



**Fig No 5 Normally distributed price**

# **Results**

In our data set we split the data set and then use 80% of the data as the training set and the remaining 20% as the testing set. The results obtained from the XGBoost regression were as follow:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data** | **Method** | **MSE** | **RMSE** | **MAE** |
| Training set | XGBoost Regression | 2.113 | 1.02 | 2.31 |
| Testing set | XGBoost  Regression | 2.110 | 1.03 | 2.31 |

Then we used the huber loss based XGBoost algorithm for real estate prediction. After training, we obtain a trained regression model. Then we visualize the results as shown below. The model achieved an impressive results of MAE, MSE and RMSE for both training and testing data. Table below gives the summary of results:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Data** | **Method** | **MAE** | **MSE** | **RMSE** |
| Training set | XGBoost using Huber Loss | 0.3141 | 0.1788 | 0.4228 |
| Testing set | XGBoost using Huber Loss | 0.3239 | 0.1850 | 0.4301 |

# **Conclusion**

In our research we present an approach to detect the Price of the House. Our model has got the remarkable performance on the evaluation matrices. This demonstrates its capacity to provide highly accurate predictions of house prices.

The key to our success lies in the sophistication of the XGBoost algorithm, which excels in handling complex relationships within the data. By leveraging the strengths of XGBoost, we've achieved predictive accuracy that can significantly benefit various industries, including real estate, finance, and urban planning.

XGBoost's ensemble of decision trees is particularly effective in identifying the most impactful predictor variables for house price prediction. Its ability to handle feature selection not only enhances the transparency of the model but also plays a pivotal role in guarding against over-fitting, a crucial consideration in any regression task.

While our findings are indeed promising, the world of machine learning is ever-evolving, offering a multitude of opportunities for further exploration. Future research might delve deeper into the effects of different hyper-parameter settings specific to XGBoost, or even consider the inclusion of additional domain-specific features to further enhance predictive accuracy.

As we embark on this research journey, we invite fellow researchers and practitioners to collaborate, explore the proposed model, and contribute to its future deployment.

***References***

[1] Lakshmi, B. N., and G. H. Raghunandhan. "A conceptual overview of data mining." 2011 National Conference on Innovations in Emerging Technology. IEEE, 2011.

[2] Manjula, R., et al. "Real estate value prediction using multivariate regression models." Materials Science and Engineering Conference Series. Vol. 263. No. 4. 2017.

[3] A. Varma et al., “House Price Prediction Using Machine Learning And Neural Networks,” 2018 Second International Conference on Inventive Communication and Computational Technologies, pp. 1936–1939, 1936.

[4] Arietta, Sean M., et al. "City forensics: Using visual elements to predict non-visual city attributes." IEEE transactions on visualization and computer graphics 20.12 (2014): 2624-2633.

[5] Yu, H., and J. Wu. "Real estate price prediction with regression and classification CS 229 Autumn 2016 Project Final Report 1–5." (2016).

[6] Li, Li, and Kai-Hsuan Chu. "Prediction of real estate price variation based on economic parameters." 2017 International Conference on Applied System Innovation (ICASI). IEEE, 2017.

[7] Nihar Bhagat, Ankit Mohokar, Shreyash Mane "House Price Forecasting using Data Mining" International Journal of Computer Applications,2016.

[8] N. N. Ghosalkar and S. N. Dhage, "Real Estate Value Prediction Using Linear Regression," 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India, 2018, pp.

[9] Pow, Nissan, Emil Janulewicz, and Liu Dave Liu. "Applied Machine Learning Project 4 Prediction of real estate property prices in Montréal." Course project, COMP-598, Fall/2014, McGill University (2014).

[10] Sampathkumar, V., Santhi, M. H., & Vanjinathan, J. (2015). Forecasting the land price using statistical and neural network software. Procedia Computer Science, 57, 112-121.

[11] Banerjee, Debanjan, and Suchibrota Dutta. "Predicting the housing price direction using machine learning techniques." 2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI). IEEE, 2017

[12] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD ’16, pages 785–794, New York, NY, USA, 2016. ACM.

[13] Mak S, Choy L & Ho W, “Quantile Regression Estimates of Hong Kong Real Estate Prices”, Urban Studies, 47(11), (2010), pp.2461– 2472.

[14] Limsombunchai V, Gan C, & Lee M, “House Price Prediction : Hedonic Price Model vs Artificial Neural Network”, American Journal of Applied Sciences, 1(3), (2004), pp.193–201.

[15] Graham MH, “Confronting multicollinearity in ecological multiple regression”, Ecology, 84(11), (2003), pp.2809-2815.

[16] Kraha A, Turner H, Nimon K, Zientek & Henson RK, “Interpreting multiple regression in the face of multicollinearity”, Frontiers in Psychology, 3, (2012), pp.1–10.

[17] Bin Shafi MA, Bin Rusiman MS and Che Yusof NSH, “Determinants Status of Patient After Receiving Treatment at Intensiv.

[18] L. Xu, A. Krzyzak, and C. Y. Suen, “Methods of combining multiple classifiers and their applications to handwriting recognition,” IEEE Trans. Syst., Man, Cybern., vol. 22, no. 3, pp. 418–435, May/Jun. 1992.

[19] .Woods,W. P. J. Kegelmeyer, and K. Bowyer, “Combination of multiple classifiers using local accuracy estimates,” IEEE Trans. Pattern Anal.Mach. Intell., vol. 19, no. 4, pp. 405–410, Apr. 1997.

[20] L. I. Kuncheva, J. C. Bezdek, and R. P. W. Duin, “Decision templates for multiple classifier fusion: An experimental comparison,” Pattern Recognit.,vol. 34, no. 2, pp. 299–314, 2001.

[21] Pattern Anal. Mach. Intell, “A theoretical study on six classifier fusion strategies,” IEEETrans.vol. 24, no. 2, pp. 281–286, Feb. 2002.

[22] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, “Adaptive mixtures of local experts,” Neural Comput., vol. 3, no. 1, pp. 79–87, 1991.

[23] M. J. Jordan and R. A. Jacobs, “Hierarchical mixtures of experts and the EM algorithm,” Neural Comput., vol. 6, no. 2, pp. 181–214, 1994.

[24] H. Drucker, C. Cortes, L. D. Jackel, Y. LeCun, and V. Vapnik, “Boosting and other ensemble methods,” Neural Comput., vol. 6, no. 6 pp. 1289–1301, 1994.

[25] D. H. Wolpert, “Stacked generalization,” Neural Netw., vol. 5, no. 2,pp. 241–259, 1992

[26] B. V. Dasarathy and B. V. Sheela, “Composite classifier system design: Concepts and methodology,” Proc. IEEE, vol. 67, no. 5, pp. 708–713,May 1979.