

# Real Estate Price Prediction

Adithya R K Nambiar  
School of Computing  
Amrita Vishwa Vidyapeetham  
Amritapuri, India  
[amenu4aie21105@am.students  
.amrita.edu](mailto:amenu4aie21105@am.students.amrita.edu)

Advaith P R  
School of Computing  
Amrita Vishwa Vidyapeetham  
Amritapuri, India  
[amenu4aie21106@am.students  
.amrita.edu](mailto:amenu4aie21106@am.students.amrita.edu)

Dhruv Dinesh  
School of Computing  
Amrita Vishwa Vidyapeetham  
Amritapuri, India  
[amenu4aie21124@am.students  
.amrita.edu](mailto:amenu4aie21124@am.students.amrita.edu)

Gokul Krishna B R  
School of Computing  
Amrita Vishwa Vidyapeetham  
Amritapuri, India  
[amenu4aie21131@am.students  
.amrita.edu](mailto:amenu4aie21131@am.students.amrita.edu)

Nandana Ajoy  
School of Computing  
Amrita Vishwa Vidyapeetham  
Amritapuri, India  
[amenu4aie21145@am.students  
.amrita.edu](mailto:amenu4aie21145@am.students.amrita.edu)

S Anand  
School of Computing  
Amrita Vishwa Vidyapeetham  
Amritapuri, India  
[amenu4aie21155@am.students  
.amrita.edu](mailto:amenu4aie21155@am.students.amrita.edu)

***Abstract – Real estate price prediction involves using statistical and machine learning methods to predict future real estate prices based on historical data and current market trends. This is a crucial aspect of real estate investment decision-making and helps investors determine the potential return on investment. The prediction models take into account various factors such as location, property type, economic indicators, and demographic trends.***

***Index Terms*** – Training, Multiple Linear regression, Random forest Prediction, Accuracy, Ridge regression, LASSO.

## I. INTRODUCTION

Real estate price prediction is the process of forecasting future real estate prices based on past trends, current market conditions, and other relevant factors. With the increasing demand for real estate investment, accurate price predictions have become increasingly important.

Real estate price prediction models take into account various factors such as the location, type of property, demographic trends, and economic indicators. The goal of these models is to provide a comprehensive understanding of the real estate market, which can help investors make informed investment decisions. Additionally, real estate price predictions can also assist policymakers in formulating effective housing policies. Overall, real estate price prediction has become an essential tool for anyone involved in the real estate industry,

from investors and developers to policymakers and market analysts.

Overall, real estate price prediction has become an essential tool for anyone involved in the real estate industry, from investors and developers to policymakers and market analysts.

## II. OBJECTIVE

This project is done in an effort to predict the real estate prices of a particular region taking several attributes into account. The current framework includes estimating the real estate prices without any expectations of market prices and cost increment. By using data values given in the datasets, price will be predicted by different models and the most best result will be deducted.

## III. PROBLEM DEFINITION

The problem definition for real estate price prediction using machine learning algorithms is to predict the value or price of a real estate property, such as a house or an apartment, based on various factors such as location, size, number of rooms, age of the property, etc. The goal is to use historical data and machine learning algorithms to build a model that can accurately predict the prices of properties based on their features and then use that model to make predictions for new properties.

## IV. DATASET AND RELATED WORKS

We have used 3 datasets in our project.

### Dataset 1

The datasets that we are using for the project has been taken from Kaggle ('Real estate.csv')  
Real estate dataset with 414 inputs and 7 attributes including the actual price of the plot.

#### Attribute Information:

There are 6 main attributes:

X1=the transaction date (for example,  
2013.250=2013 March, 2013.500=2013 June, etc.)

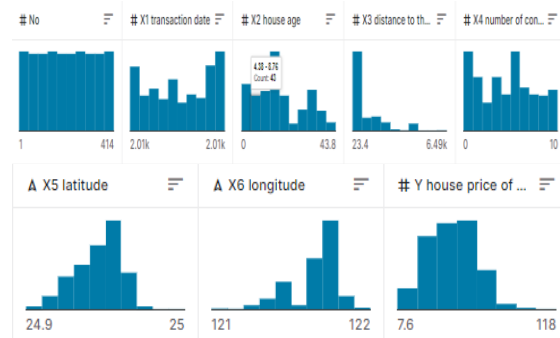
X2=the house age (unit: year)

X3=the distance to the nearest MRT station (unit:  
meter)

X4=the number of convenience stores in the living  
circle on foot (integer)

X5=the geographic coordinate, latitude. (unit:  
degree)

X6=the geographic coordinate, longitude. (unit:  
degree)



### Dataset 2

The datasets that we are using for the project has been taken from Kaggle ('riga\_re.csv')  
Riga real dataset has 4689 inputs and 13 columns.



#### Attribute Information:

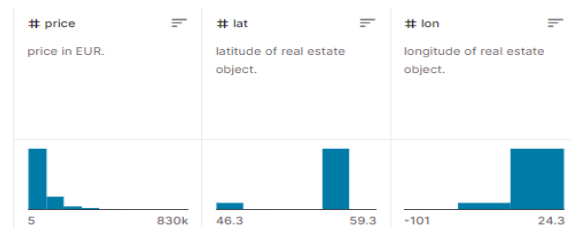
There are 13 attributes:

- op\_type - offer type
- district - where real estate is located.
- street - address of real estate object.
- rooms - number of rooms.
- area - living area of real estate object.
- floor - floors of real estate objects.
- total\_floors - total number of floors in a building.

- house\_seria - house design
- house\_type - type of building
- condition - stuffing premises
- lat / lon - latitude and longitude of real estate.

▲ op_type	▲ district	▲ street	▲ rooms	▲ area
offer type ('For rent', 'For sale', 'Buying', 'Renting', 'Change', 'Other')	district, where real estate object located.	address of real estate object.	number of rooms.	living area of real estate object.
For sale	57%	centrs	31%	[null]
For rent	33%	Purvcieks	8%	Kungu 25
Other (484)	10%	Other (2894)	62%	Other (4212)

# floor	# total_floors	▲ house_seria	▲ house_type	▲ condition
floor of rel estate object.	total amount of floors in building.	house design ('LT proj.', '602', 'P. kara', 'Jaun.', 'Specpr', 'Hrušć', '119', 'M. ģim.', 'Renov.', '103', 'nan', 'Priv. m.', '467', 'Stajlna',	type of building ('Brick-Panel', 'Panel', 'Wood', 'Masonry', 'Brick', 'Panel-Brick').	stuffing premises ('All amenities', 'Partial amenities', 'Without amenities').
		P. kara 19%	Masonry 28%	All amenities 87%
		Jaun. 17%	Panel 26%	<b>null</b> 9%
		Other (2993) 64%	Other (2157) 46%	Other (189) 4%



### Dataset 3

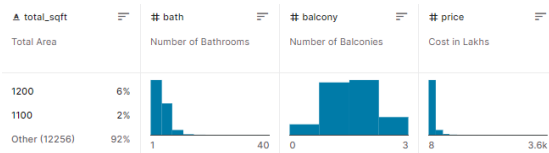
The datasets that we are using for the project has been taken from Kaggle ('BHP.csv').It has 7496 rows and 9 columns.

#### Attribute Information:

The main attributes are:

- Area Type: Type of Plot
- Availability: Ready to Move or Not
- Location: Region of Bangalore
- Size: BHK
- Society: Colony in which the House is Present in
- Total Sq. Ft: Total Area
- Bath: Number of Bathrooms
- Balcony: Number of Balconies
- Price: Cost in Lakhs

▲ area_type	▲ availability	▲ location	▲ size	▲ society					
Type of Plot	Ready to Move or Not	Region of Bangalore	BHK	Colony in which the House is Present in					
Super built-up Area	66%	Ready To Move	79%	Whitefield	4%	2 BHK	39%	[null]	41%
Built-up Area	18%	18-Dec	2%	Serjapur Road	3%	3 BHK	32%	OrvaGr	5%
Other (2112)	16%	Other (2432)	18%	Other (12381)	93%	Other (3811)	29%	Other (7738)	16%



## V. METHODS

The working of the system involves steps:

- Data Analysing
- Data Pre-processing
- Data Visualisation
- Data Encoding
- Validation and Prediction

In this study, we evaluate and analyse dataset characteristics using different machine learning algorithms like Multiple Linear Regression, Lasso Regression, Random forest regression and Ridge regression.

### a. Multiple Linear Regression

Multiple linear regression is a statistical method used to model the linear relationship between a dependent variable and several independent variables. It's called "multiple" because it deals with more than one independent variable. The objective of multiple linear regression is to find the best line of fit that minimizes the sum of squared differences between the actual and predicted values of the dependent variable. The line of fit is represented by an equation that contains the estimated coefficients for each independent variable, which are calculated using an optimization algorithm. The model can be used to make predictions about the dependent variable given new values of the independent variables.

### b. Lasso Regression

Lasso Regression is a type of linear regression that adds a regularization term to the objective function to reduce overfitting and promote sparsity in the feature coefficients. The regularization term is a parameter multiplied by the absolute value of the coefficients, which discourages the model from assigning too much weight to any single feature. As a result, some of the feature coefficients may be exactly zero, effectively removing them from the model. The Lasso Regression optimization problem is solved using a technique called sub-gradient

descent. The regularization parameter, often denoted by  $\lambda$ , is a hyperparameter that determines the strength of the regularization. The value of  $\lambda$  is set by the user based on the specific problem and data.

### c. Random Forest Regression

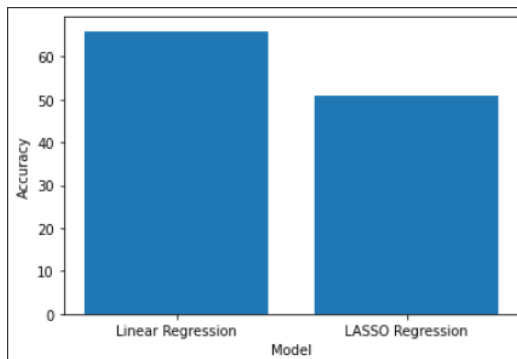
Random Forest is an ensemble learning method for classification and regression that operates by constructing a large number of decision trees and combining their predictions. Each decision tree is trained on a random subset of the data and a random subset of the features, and makes predictions by majority voting among the terminal nodes (leaves) of the tree. The use of multiple trees reduces the overfitting that can occur with individual trees and results in a more robust model. The prediction of the Random Forest model is based on the average or weighted average of the predictions of individual trees, depending on the problem. In classification, each tree votes for the class it thinks is most likely, and the class with the most votes wins. In regression, the prediction is the average of the predictions of the trees.

### d. Ridge Regression

Ridge Regression is a type of linear regression that adds a regularization term to the objective function to reduce overfitting and encourage small coefficients. The regularization term is a parameter multiplied by the sum of the squares of the coefficients, which discourages the model from assigning too much weight to any single feature. The Ridge Regression optimization problem is solved using an optimization algorithm, such as gradient descent. The regularization parameter, often denoted by  $\lambda$ , determines the strength of the regularization.

## VI. EVALUATION AND DISCUSSION

### Dataset 1 (Real estate.csv)



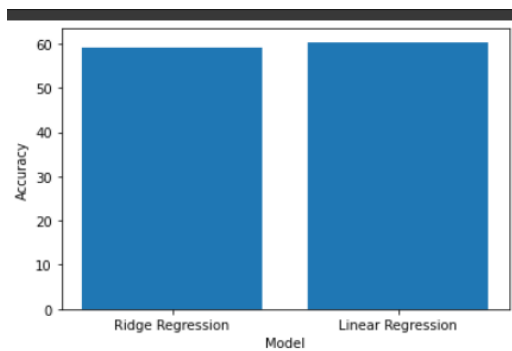
```
accuracy=lin_model.score(X_test,y_test)
print('Accuracy of the model: ',accuracy)

Accuracy of the model: 0.6591473968309041

accuracy2=lasso_model.score(X_test,y_test)
print('Accuracy of the model: ',accuracy2)

Accuracy of the model: 0.5097943850508748
```

Fig 1: Accuracy of ML Algorithms in Real estate.csv dataset (LL)



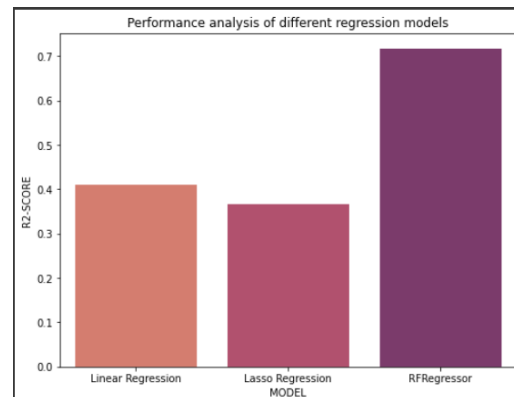
```
accuracy=lin_model.score(X_test,y_test)
print('Accuracy of the model: ',accuracy)

Accuracy of the model: 0.6303589468255337

accuracy2=ridge_model.score(X_test,y_test)
print('Accuracy of the model: ',accuracy2)

Accuracy of the model: 0.6037032938359059
```

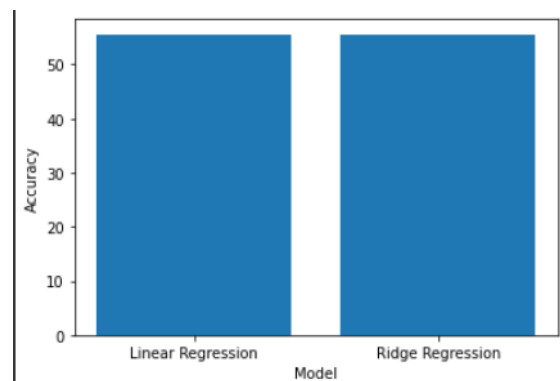
Fig 2: Accuracy of ML Algorithms in Real estate.csv dataset (LR)



```
print('Accuracy of lasso regression model: ',lin_r2)
print('Accuracy of linear regression model: ',lasso_r2)
print('Accuracy of the randomo forestregression model: ',score)

Accuracy of lasso regression model: 0.41091904331672546
Accuracy of linear regression model: 0.3666731143025953
Accuracy of the randomo forestregression model: 0.7183715677082574
```

Fig 3: Accuracy of ML Algorithms in blr\_real\_estate\_prices.csv dataset (LLRF)



```
score=lin_model.score(X_test,y_test)
print('Score of the model: ',score)

Score of the model: 0.5553362967937576

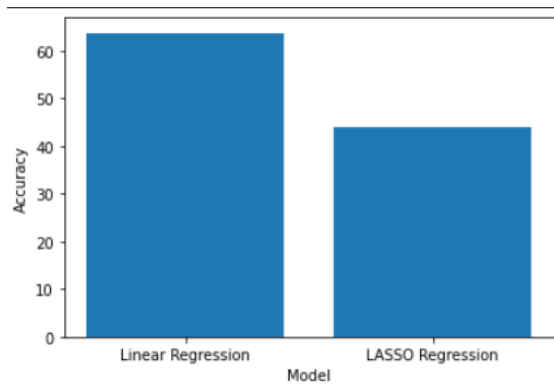
score=ridge_model.score(X_test,y_test)
print('Score of the model: ',score)

Score of the model: 0.5554674061376306
```

Fig 4: Accuracy of ML Algorithms in blr\_real\_estate\_prices.csv (LR)

Dataset 2 (blr\_real\_estate\_prices.csv)

Dataset 3 (riga\_re.csv)



```
accuracy=model1.score(x_test,y_test)
print('accuracy of linear regression model- ',accuracy)

accuracy of linear regression model- 0.636876830916275

accuracy1=model2.score(x_test,y_test)
print('accuracy of LASSO regression model- ',accuracy1)

accuracy of LASSO regression model- 0.43895038204021686
```

Fig 5: Accuracy of ML Algorithms in riga\_re.csv dataset (LL)

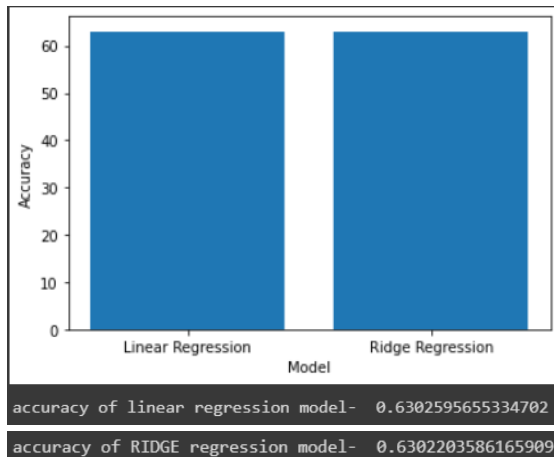


Fig 6: Accuracy of ML Algorithms in riga\_re.csv dataset (LR)

## FINDINGS AND FEATURE DIRECTIONS

The experiment compared the performance of for real estate price prediction on 3 datasets .The results of the experiment showed that the best performing model was Random Forest regression, which outperformed Multiple linear regression, LASSO regression and Ridge regression. Additionally, it's also important to consider interpretability of the model, as it may be helpful to understand the factors that are most important in determining house prices and how they influence the predicted prices.

## VII. CONCLUSION

In conclusion, predicting real estate prices is a complex task that requires a combination of domain knowledge, data preprocessing, and machine learning techniques. The process begins with understanding the problem and the data that is available. This includes understanding the factors that influence real estate prices and identifying any missing or irrelevant data. Next, the data needs to be cleaned, transformed, and prepared for modeling. Once the data is prepared, various machine learning algorithms can be applied to the data to build a model. We used models like Linear Regression, Lasso, Ridge and Random Forest, then we calculated R2 score for comparing the accuracy among the different algorithms. Additionally, it's also important to consider interpretability of the model, as it may be helpful to understand the factors that are most important in determining real estate prices and how they influence the predicted prices.

## VIII. IMPLEMENTATION

- [https://colab.research.google.com/drive/1Jl\\_vFBDJWdU\\_OisbKqTAC6cfyfRyzV3Ti?usp=sharing](https://colab.research.google.com/drive/1Jl_vFBDJWdU_OisbKqTAC6cfyfRyzV3Ti?usp=sharing)
- <https://colab.research.google.com/drive/1-WTTcYMS52Kz-VQslD4ceN67RIpL1ntk?usp=sharing>
- <https://colab.research.google.com/drive/1WoEF6q2w8FtUWvzHZvUZSgKYoaztsKlj#scrollTo=Qc85cnWi4Q-a>
- [https://colab.research.google.com/drive/1r6B8JBeGqUYhmv\\_dprdg8jtFvaCiZRNp?usp=sharing](https://colab.research.google.com/drive/1r6B8JBeGqUYhmv_dprdg8jtFvaCiZRNp?usp=sharing)
- [https://colab.research.google.com/drive/1Z0le3nmSP9BEIyq\\_uiJXPk73yoeGpSky?usp=sharing](https://colab.research.google.com/drive/1Z0le3nmSP9BEIyq_uiJXPk73yoeGpSky?usp=sharing)
- <https://colab.research.google.com/drive/1jyP6UvmUR7TR4wmGOfqqvM7uzTXAX2DE?usp=sharing>

## IX. APPENDIX

The python packages used include pandas, sklearn, matplotlib and seaborn.

### **Pandas**

It is a Python library for data analysis and manipulation that includes functions for reading and writing data in formats including CSV, Excel, and SQL as well as 1-D (Series) and 2-D (DataFrame) data structures. Powerful data cleaning and manipulation capabilities are available.

### **Scikit-learn (sklearn)**

Scikit-learn, or simply sklearn, is a Python library for machine learning that offers tools for tasks like classification, regression, clustering, dimensionality reduction, and more. It also provides support for model selection, pre-processing, and evaluation. The library is built on top of NumPy and SciPy and integrates well with the wider scientific Python ecosystem. Sklearn is known for its straightforward, efficient, and user-friendly API and is widely utilized in the fields of data science, machine learning, and academia.

### **Matplotlib**

It is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK. It also supports other backends such as WebAgg, pdf and svg. It provides a large number of customizable options and visualization types, such as line plots, scatter plots, histograms, bar plots, and more. It's often used in data visualization and scientific visualization. It's also a powerful library that allows you to make

basic to complex plots, charts, and figures. It's considered one of the most widely used libraries for data visualization in Python.

### **Seaborn**

Seaborn is a Python data visualization library is a Python data visualization library based on Matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics. Seaborn is particularly well suited for visualizing complex, multi-dimensional data, and it is often used in conjunction with Pandas data frames.

## X. ACKNOWLEDGEMENT

We would like to express our sincere gratitude towards Dr. Aiswarya S. Kumar for her guidance throughout this project. Our appreciation to our friends and family for their input and thoughts on this study. Furthermore, we would like to express our sincere gratitude to Amrita University for allowing us to perform this study and Chancellor Amma for her presence and grace.

## XI. REFERENCE

- <https://www.geeksforgeeks.org/house-price-prediction-using-machine-learning-in-python/>
- <https://www.analyticsvidhya.com/blog/2022/01/using-sequential-model-to-predict-prices-of-real-estate/>
- <https://towardsdatascience.com/tagged/house-price-prediction>
- <https://medium.com/@manilwagle/predicting-house-prices-using-machine-learning-cab0b82cd3f>