International Journal of Scientific Research in Science and Technology
Print ISSN: 2395-6011 | Online ISSN: 2395-602X (www.ijsrst.com)
doi : https://doi.org/10.32628/IJSRST523103134

A Novel Model for House Price Prediction with Machine Learning Techniques

Harshada S. Belsare*, Prof. Kanchan V. Warkar
Department of Computer Engineering, Bapurao Deshmukh College of Engineering Sevagram, Wardha, India

ARTICLEINFO

Article History:

Accepted: 01 June 2023
Published: 07 June 2023

Publication Issue
Volume 10, Issue 3
May-June-2023

Page Number
743-754

ABSTRACT

In this paper, we are going to use machine learning algorithms for house price prediction. House prices increases drastically every year, so we felt a need for a system that will predict house prices in the future. Due to a lack of knowledge of property assets people cannot guess the accurate price of houses. Therefore, we felt a need for a model that will predict an accurate house price. So, the main aim of our project is to predict the accurate price of the house without any loss. This survey also deals with a comparative analysis of the results of the algorithms used and the model with the highest accuracy and minimum error rate will be implemented. For the choice of prediction ways, we tend to compare and explore numerous prediction ways. We tend to utilize Linear and random forest regression as our model attributable to its liable and probabilistic methodology on model Choice. Our result exhibits that approach to the problem ought to achieve success and has the flexibility to predictions that will be compared to different house price prediction models. We have a proclivity to propose a house price prediction model to hold up a customer to estimate the proper valuation of a house.

Keywords :- Data Standardization, House Price Prediction, Machine Learning, Linear Regression, Random Forest, Machine Learning Algorithms.

I. INTRODUCTION

House price prediction using machine learning algorithms is an intriguing area of data science study and application. Predicting house prices is an important challenge in the real estate market, and machine learning approaches have shown considerable promise in handling this problem.

Machine Learning (ML) is an important part of modern industry and research. It gradually increases computer system performance by employing ML algorithms and neural network models (DL). With the growing availability of real estate data and improvements in machine learning algorithms, it is now feasible to reliably forecast the pricing of

properties in various places based on variables such as location, size, number of rooms, amenities, and more.

Machine learning models are capable of analysing massive volume of data, identifying patterns, and making accurate predictions based on data availability. The physical attributes of a house, such as its size, the year it was built, the number of bedrooms and bathrooms, location and other data that characterize the house's interior features, may influence its price.

The goal of house price prediction is to create a model that can accurately estimate a house's price based on its attributes. Many people buy houses to live in, whereas real estate brokers buy them to sell for a profit and consider it an income source.

The key point is that everyone should get the house for which they are paying, and no one should be tricked while buying a house. The home should be worth it based on their expectations and the price they are paying for it. Real estate agents, property buyers, and sellers can utilize this model to make informed judgements.

The main issue is that everyone should get the house for which they paid, and no one should be duped while purchasing a home. Based on their expectations and the money they are paying; the house should be worth it. This model can help real estate agents, buyers, and sellers make informed decisions.
The data will then be pre-processed, with missing values handled, duplicate values, category variables encoded, and numerical features scaled, mean or mode imputations. The scaling done based on the which kind of scaling technique we used while cleaning the dataset. Standard scaling, min-max scaling, robust scaling are the types of scaling that are used in general. The range of scaling is -1 to +1, 0 to 1, 0 to infinity respectively. Then, we'll choose a machine learning algorithm, such as regression or classification.

We will use techniques such as cross-validation to ensure that our model generalizes well and does not overfit the predicted house prices use a real estate dataset.

When a model is overfitted, we can claim that instead of learning, it memorizes the training dataset. In overfitting conditions, the model performs well on training data but fails on testing data.
The aim is to develop a model that can accurately predict the prices of houses and identify the key features that influence the prices.

Finally, we will assess our model's performance. If our model works successfully, we will be able to utilize it to make predictions on previously unseen data and provide important insights to real estate brokers, homeowners, and purchasers.

Supervised learning:

Supervised learning is a machine learning algorithm that trains the computer using labelled data. Labelled data is data that has already been categorized or sorted by humans, such as photos labelled with the things they contain or text labelled with the categories to which it belongs.

The primary idea behind supervised learning is to utilize labelled data to educate the computer (or algorithm) how to find patterns or relationships in the data, so that when provided with new, unlabelled data, it can make accurate predictions or classifications.

The computer is given a set of input features and a corresponding output label in supervised learning, and its purpose is to learn a function that maps the inputs to the proper outputs. The computer is trained by modifying the parameters of the function based on how well it performs on labelled data and then assessing its performance on new, unseen data.

Linear regression, logistic regression, decision trees, and support vector machines are all examples of supervised learning techniques. Image classification, natural language processing, deep learning, computer vision, and predictive modelling are all examples of how these algorithms are employed.

Unsupervised learning:

Unsupervised learning is a machine learning approach that trains the computer using unlabelled data. In contrast to supervised learning, there are no predefined labels or categories from which the computer can learn. Instead, the computer is given the duty of discovering patterns or links in the data on its own.

Unsupervised learning aims to find hidden structure in data, such as clusters or groups of similar data points, without knowing what those clusters might represent. Unsupervised learning algorithms accomplish this by analysing data and detecting patterns that can be used to categories the data.
In this the algorithm uses the distance between the datapoints. Closer the datapoints same the characteristics of the datapoint.

Clustering is a typical sort of unsupervised learning process in which the computer clusters similar data points together based on their proximity to one another. Dimensionality reduction is another sort of unsupervised learning procedure in which the computer minimizes the number of characteristics in the data while retaining the most significant information.

Unsupervised learning techniques are utilized in many applications, including customer segmentation, anomaly detection, picture and text analysis. Unsupervised learning algorithms that are commonly used include k-means clustering, hierarchical clustering, principal component analysis (PCA), and t-SNE.

In clustering the datapoints are grouped into the different discrete groups. The distance between the datapoints can be calculated by L1 or L2 i.e., Manhattan and Euclidean distance. Distance between the cluster and a datapoint is also called as inertia.

II. PROBLEM DEFINITION

To develop a predictive model that can accurately estimate the prices of houses based on various features such as location, size, number of rooms, amenities, and more. The goal of this prediction is to help real estate agents, buyers, and sellers make informed decisions about buying, selling, and pricing their properties

The main challenge in this problem is to develop a model that can accurately capture the complex relationships between the features and the target variable (house price) in a given real estate dataset. This requires careful selection and pre-processing of the features, choosing an appropriate machine learning algorithm, and tuning the model's hyperparameters to optimize its performance

In addition to accuracy, the model's interpretability is also important as to which features played a significant role in the predictions. This is important for real estate agents, buyers, and sellers to understand and make informed decisions based on the predictions. It involves developing a model that is accurate, interpretable, and can provide insights into the factors that influence house price prediction techniques.

III.PROPOSED METHODOLOGY

Collection: The first stage is to gather information on the factors that influence house pricing, such as location, size, number of bedrooms and baths, amenities, location and so on. This information is

available from a variety of sources, including internet real estate portals, government documents, and real estate agents.

The accuracy and dependability of your machine-learning model can be greatly influenced by the quality and quantity of data you collect. Here are some data collection tips for predicting property prices:

Determine the following variables: Location, size, number of bedrooms and bathrooms, age of the property, condition of the property, amenities, and surrounding attractions are the most important criteria in predicting house prices. Choose which variables to include in your study and then collect data for those variables.

Data sources: You can acquire data for house price forecast from a variety of sources. Real estate websites are examples of this. You can also obtain information from local real estate brokers and property management firms.

Data cleaning: After collecting your data, you must clean it to remove any missing or incorrect numbers. To make the data more suited for analysis, you may need to convert or normalize it.

Data labelling: In order to train a supervised machine learning model, you must label your data. In the case of house price prediction, each data point must be labelled with the actual price of the property.

Data augmentation: By enhancing your data, you can improve its quality and quantity.

After collecting and cleaning your data, you must divide it into training, validation, and testing sets. This allows you to train your machine learning model on a portion of data and then test its accuracy on another subset of data.

Data cleaning and preprocessing: After collecting the data, it must be cleaned and preprocessed to eliminate any missing or irrelevant information, perform feature engineering, and put the data into a format suitable for machine learning algorithms. The obtained data will almost certainly contain missing values, outliers, and other mistakes that must be addressed before modelling. As a result, the data must be preprocessed to remove any flaws or inconsistencies.

Feature engineering: This phase entails converting the data into a format that the machine learning system can understand. Selecting relevant features, scaling and normalizing the data, and encoding categorical variables are all part of this process.

Split data into training and testing sets: The data is divided into two sets: training and testing. The training set is used to train the machine learning algorithm, and the testing set is used to assess the algorithm's performance.

Choose a machine learning algorithm: Machine learning algorithms such as linear regression, decision trees, and random forests can be used to predict housing prices. The algorithm used is determined by the nature of the problem and the amount of the dataset. Linear regression will be used in this case.

Linear Regression:

Linear regression is a widely used machine learning technique for modelling the relationship between a dependent variable (commonly referred to as the target or response variable) and one or more independent variables (often referred to as predictors or features). The dependent variable in the case of house price prediction would be the price of a property, and the independent factors may include things like the number of bedrooms, the size of the house, the neighborhood, and so on. Linear regression

is used in machine learning for both regression and classification applications.

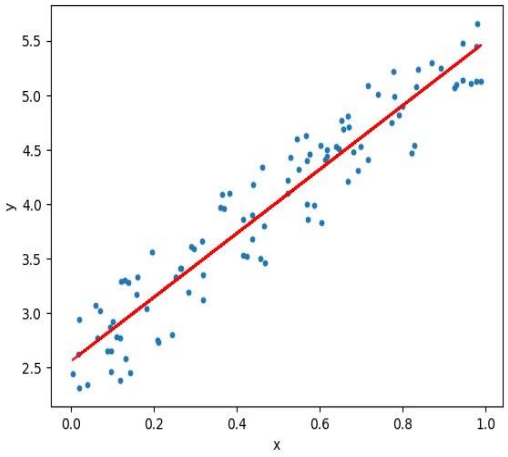


Fig3.1: linear regression

The goal is to select the best-fit line that minimizes the total of the squared discrepancies between predicted and actual dependent variable values.

When there is only one independent variable, the equation for simple linear regression is:

$$Y = \beta 0 + \beta 1X + \varepsilon$$

where Y is the dependent variable, X is the independent variable, β0 is the intercept, β1 is the slope, and ε is the error term. The objective of the linear regression is to estimate the values of β0 and β1 that minimize the sum of squared errors.

Multiple Linear Regression:

Multiple linear regression is a statistical approach for modelling the connection between two or more independent variables and a dependent variable. It is an extension of simple linear regression, in which just one independent variable is used.

Multiple linear regression, where there are multiple independent features. There are some common acceptances while using the multilinear regression.

Error should come from standard normal distribution – normal distribution curve is the curve having reverse U like structure. From this curve we can say that the maximum amount of data is at center of the curve and the datapoints are equally distributed from each other on both sides. The mean of the standard normal distribution curve is at zero and having variance (sigma).

All features should be independent of each other –

If features of the model or dataset having any relation with each other or they are having any kind of correlation between them then the adjusting the one variable will cause the changes in the state of other dependent variable. We can inspect the feature dependencies from pair plot and from VIF (Variance Inflation Factor)

Error should be independent of the feature - if the error is dependent on the features, then the change in feature state will cause change in the error. There must be no or zero correlation between the error and features and this can be validated by plotting scatter plot between features and error.

Homoscedasticity - This says that variance must be constant through the model. Also, the error should not change with respect to variance. To validate this, we can plot graph between the error and predicted value.

Linearity in parameters – as this is multilinear model so the target variable must have the linear relation with individual features.

The mathematical equation for multiple linear regression is:

$$y = b0 + b1x1 + b2x2 + ... + bn*xn + e$$

where y is the dependent variable, x1, x2, ..., xn are the independent variables, b0 is the intercept, b1, b2, ..., bn are the coefficients of the independent variables, and e is the error term.

This equation denotes a linear relationship between a dependent variable and several independent variables. The coefficients b1, b2,..., bn denote the change in the dependent variable caused by a unit change in the corresponding independent variable while all other independent variables remain constant.

When all independent variables are equal to zero, the intercept term b0 indicates the value of the dependent variable.

The purpose of multiple linear regression is to discover the coefficient values b0, b1, b2,..., bn that best fit the data. This is commonly done using the least squares approach, which entails determining the coefficient values that minimize the sum of the squared differences between the predicted and actual values of y(target variable).

Once the coefficients have been estimated, the multiple linear regression equation can be used to predict the value of the dependent variable for a given set of values of the independent variables.

There are several advantages of using multiple linear regression.

Flexibility: numerous linear regressions analyze the relationship between numerous independent factors and a single dependent variable, which can be beneficial in a variety of sectors such as social sciences, engineering, and finance.

Prediction: Multiple linear regression can be used to make predictions about the dependent variable based on the values of the independent variables. This can be helpful in many contexts, such as predicting sales based on advertising expenditure or predicting stock prices based on economic indicators.

Ensemble Learning:

Ensemble learning is a machine learning technique that combines the predictions of numerous independent models to increase the total prediction's accuracy and robustness. Ensemble learning is based on the premise that many models have different strengths and weaknesses, and that by combining their predictions, we can achieve higher performance than any single model alone.

There are several types of ensembles learning techniques, including:

Boosting:

Boosting is a machine learning technique in which numerous weak models are combined to build a strong model. The goal behind boosting is to train a sequence of models progressively, with each successive model attempting to fix the errors generated by the preceding models.

In boosting, each weak model is typically a decision tree, and the ensemble of models is referred to as a "boosted tree." Boosting is often used in supervised learning tasks, such as classification and regression.

Bagging:

Bagging (Bootstrap Aggregating) is a popular ensemble learning technique in which multiple models are trained on different random subsets of the training data, with replacement. The basic idea of bagging is to reduce the variance of the individual models by averaging their predictions.

The steps involved in bagging are as follows:

Randomly select a subset of the training data with replacement.
Train a model on the subset of the data.
Repeat the above steps for a fixed number of times, each time selecting a different random subset of the training data.
Predict the output using each of the trained models.

Combine the predictions of all the models, usually by taking the average.

The fundamental benefit of bagging is that it reduces the variance of individual models, resulting in more accurate predictions. This is especially important when the individual models are prone to overfitting, as the ensemble can help to reduce overfitting by averaging out the individual models' predictions.

Random Forest and Extra Trees are two prominent bagging algorithms. To boost the variety of the different models, these methods use decision trees as the basic models and randomly select subsets of features as well as subsets of training data during the training process.

Overall, bagging is a powerful and extensively used ensemble learning strategy that can increase machine learning model performance.

Random Forest:

Random Forest is a well-known ensemble learning technique for classification, regression, and other tasks in machine learning. It is part of the decision tree-based model family, in which many decision trees are trained on subsets of the training data and their predictions are aggregated to generate the final prediction.

The technique operates by building numerous decision trees, each with a randomly chosen portion of the input features and a randomly chosen subset of the training data. The ultimate forecast is formed by aggregating all of the decision trees' predictions.

The main advantages of using Random Forest are:

It can handle a large number of input features and noisy data.

It can perform well on both classification and regression tasks.

It is less prone to overfitting compared to other decision tree-based algorithms.

It provides feature importance measures, which can be useful in feature selection.

To build a Random Forest model, the following steps are typically taken:

Data preprocessing: The input data is preprocessed by handling missing values, encoding category variables, and, if necessary, scaling the data.

Splitting the data: The input data is split into training and testing sets. The training set is used to train the model, and the testing set is used to evaluate its performance.

Creating decision trees: A fixed number of decision trees are created, each trained on a randomly selected subset of the input data.

Growing decision trees: Each decision tree is grown by recursively splitting the data into smaller subsets, based on the most informative features, until a stopping criterion is met.

Making predictions: The final prediction is made by aggregating the predictions of all the decision trees. For classification tasks, the most common prediction is selected, and for regression tasks, the average prediction is used.

Evaluating the model: The performance of the model is evaluated using metrics such as accuracy, precision, recall, and F1 score.

Tuning the model: The model hyperparameters can be tuned to improve its performance on the validation set. Common hyperparameters to tune include the number of decision trees, the depth of each tree, and the size of the subset of features to consider at each split.

Random Forest models are commonly used for a variety of machine learning tasks such as classification, regression, and feature importance analysis. They are known for their ability to handle noisy data and prevent overfitting. However, they can be computationally expensive, and their performance can be affected by the choice of hyperparameters.

Grid Search:

Grid search is a machine learning hyperparameter optimization approach used to determine the optimal combination of hyperparameters for a given model. Hyperparameters are model parameters that are established before training rather than learned from data. The number of hidden layers in a neural network, the learning rate of a gradient descent algorithm, and the depth of a decision tree are all examples of hyperparameters. Grid search entails describing a grid of possible hyperparameter values and then evaluating the model's performance for each hyperparameter combination. The grid search algorithm explores all potential hyperparameter combinations exhaustively and delivers the set of hyperparameters that produce the highest performance according to a specified evaluation criterion, such as accuracy.

Evaluation: After the model is trained, we need to evaluate its performance using various metrics such as mean squared error, R-squared, and accuracy. The model can be further fine-tuned using hyper parameter tuning techniques to improve its performance.

IV. SYSTEM REQUIREMENTS

The software requirements include the programming languages and libraries needed to implement machine learning algorithms. Python is a popular language used for machine learning, and libraries such as scikit-learn, pandas, and numpy are commonly used for data preprocessing, feature selection, and model training

Hardware Requirements:

Computer:

The hardware requirements depend on the size of the dataset and the complexity of the machine learning algorithm used. A powerful computer with a multi-core processor and a dedicated GPU is recommended for training large datasets and complex models. For example, a computer with at least 16GB RAM and an Nvidia GTX 1080 GPU or higher would be suitable.

Software Requirements:

Python: Python is a high-level, interpreted programming language. It is widely used in web development, scientific computing, data analysis, artificial intelligence, and many other areas. One of the main advantages of Python is its large and active community of developers, who contribute to the development of libraries and tools that make it easier to work with Python. Most popular libraries include NumPy, Pandas, SciPy, Matplotlib, and Tensor Flow.

Google colaboratory: Google Colaboratory, also known as Colab, is a cloud-based Jupyter notebook environment that is provided by Google. It is a free platform that allows users to write and run Python code in a web browser without the need to install any software on their local machine

Numpy :

NumPy is a Python library that stands for "Numerical Python". It is a fundamental package for scientific computing with Python, providing support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays. NumPy is widely used in data science, machine learning, scientific research, and other fields that require efficient numerical operations on large datasets. Mathematical functions: NumPy includes a large collection of mathematical functions that can be applied to arrays, including basic operations like addition and subtraction, as well as

more complex functions like matrix multiplication and trigonometric functions.

Linear algebra: NumPy has a set of functions for performing linear algebra operations such as matrix multiplication, matrix inversion, and linear equation solving. Overall, NumPy is a robust Python library that is required for many scientific computing and data analysis activities. Its array processing skills are efficient, and it has a wide range of mathematical functions.

Pandas:

pandas is a popular Python data manipulation and analysis toolkit. It has strong data structures for processing structured data, such as data frames, which are analogous to database tables.

It is an effective tool for data pretreatment and feature engineering due to its capacity to handle structured data and execute complicated data transformations. It is a popular choice for machine learning projects due to its ability to interact with various data analysis and visualization packages. It is used in machine learning for data preparation, feature engineering, data visualization, and data analysis, among other things.

Seaborn :

Seaborn is a popular Python library for data visualization, built on top of Matplotlib. It provides a high-level interface for creating statistical graphics, including heatmaps,

scatterplots, line charts, bar charts, and more. Seaborn is particularly useful for creating complex visualizations with minimal code. overall, Seaborn is a powerful library for creating complex visualizations with minimal code. Its integration with Pandas and built-in statistical visualizations make it a popular choice for data visualization in Python.

Matplotlib :

Matplotlib is a well-known Python data visualization package that offers a variety of tools for building static, animated, and interactive visualizations. Multiple plot types, interactive plots, publication-quality plots, and interaction with pandas are some of Matplotlib's important features.

Overall, it is a strong Python package for producing static, animated, and interactive visualizations. Its customizability and support for different plot styles make it a useful tool for data visualization and exploration. It is a popular choice for data analysis and machine learning projects due to its connection with Pandas and support for numerous backends.

V. RESULTS

We performed the feature engineering on the dataset to get insights from the data. From feature engineering on the data, the used data is continuous in nature and we have to use regression models to analyze and prediction. We have python inbuilt libraries to analyze. Scikit-Learn, also known as sklearn is a python library to implement machine learning models and statistical modelling. Through scikitlearn, we can implement various machine learning models for regression, classification, clustering, and statistical tools for analyzing these models. From sklearn we have import train test split module to split the data into training and testing. Train data – it is selected random. Generally, it is 75%,80%,85%,90% of the total data Test data – it is the remaining data from the dataset. The validation dataset is different from the test dataset that is also held back from the training of the model, but is instead used to give an unbiased estimate of the skill of the final tuned model when comparing or selecting between final models. Using linear regression, we trained our model on 80% of the data and rest 20% is used to test the accuracy of the trained model using regression algorithm.

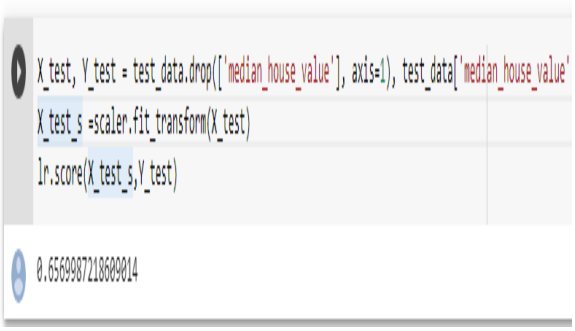


Fig 5.1 Linear Regression Result

From the above fig we get the accuracy of 65% using regression model. In this we have used the fit transform function.

In the fit () method, where we use the required formula and perform the calculation on the feature values of input data and fit this calculation to the transformer.

For applying the fit() method (fit transform in python), we have to use fit() in front of the transformer object.

For changing the data, we probably do transform in the transform () method, where we apply the calculations that we have calculated in fit () to every data point in feature F. We have to use. transform () in front of a fit object because we transform the fit calculations.

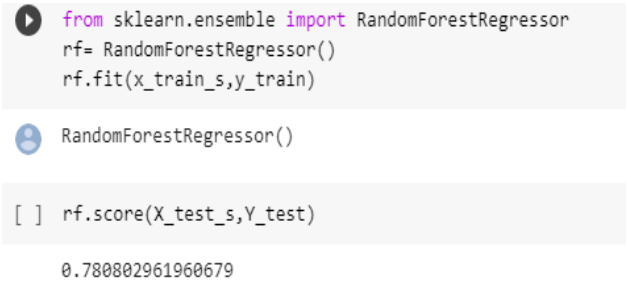
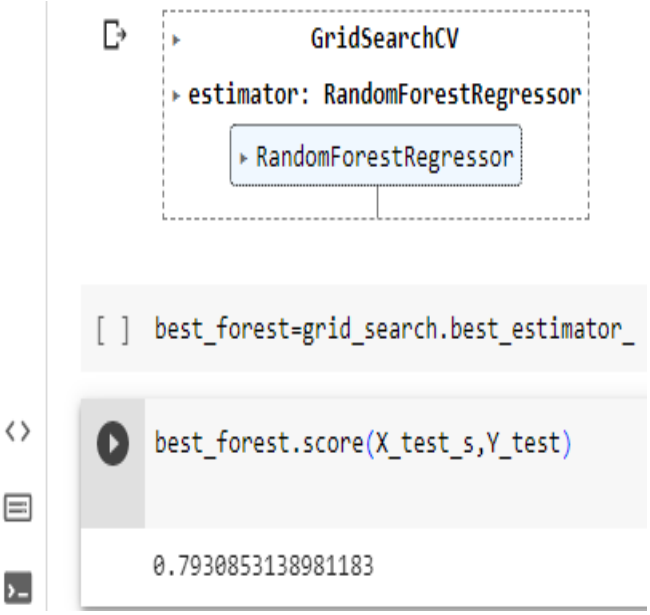


Fig 5.2 Random Forest Result

The fit_transform() method is basically the combination of the fit method and the transform method. This method simultaneously performs fit and transform operations on the input data and converts the data points. Using fit and transform separately when we need them both decreases the efficiency of the model. Instead, fit_transform() is used to get both works done.

In random forest we have used the fit () function and we get the accuracy i.e. rf score as 78% which is improved from the regression model.



` Fig 5.3 Grid Search Result

The majority of machine learning models contain parameters that can be adjusted to vary how the model learns. For example, the logistic regression model, from sklearn, has a parameter C that controls regularization, which affects the complexity of the model. This technique is known as a grid search. If we had to select the values for two or more parameters, we would evaluate all combinations of the sets of values thus forming a grid of values. In machine learning, the grid search approach is used to identify the ideal values for a model's hyperparameters. Building a grid of hyperparameter values, training a model for each set of values, and assessing the model's performance are the steps to make this model function. From gris search we got accuracy of 79%

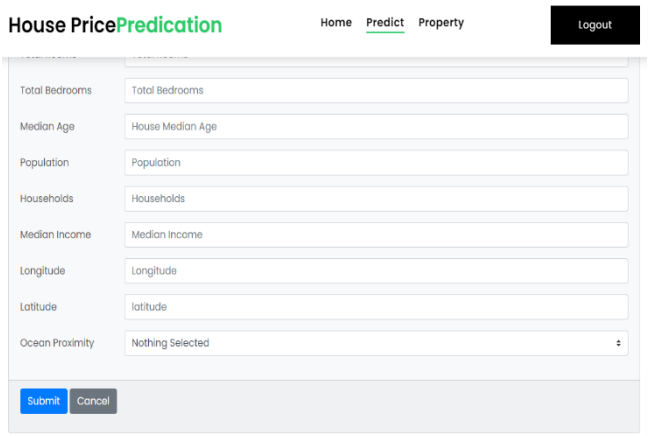


Fig 5.4: house price prediction user interface

The above screen shot shows the various attributes that user needs to fill as an input and after filling the form user have to click on the submit button and then the data will go to the server and price of the house will get predicted. The predicted house value will appear on the screen.

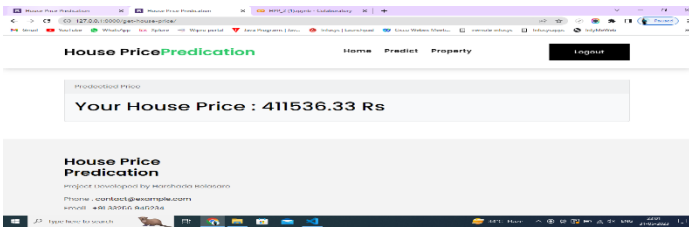


Fig 5.5: house price predicted value

From the above screenshot we can see the final value of the house is predicted by the model trained on the server. In this way the price appears on the user screen.

VI. CONCLUSION

This paper examined and analyzed the current research on the significant attributes of house price and analyzed the machine learning techniques used to predict house price. Technically, houses with a strategic location such as the accessibility to shopping mall or other facilities tend to be more expensive than houses in rural areas with limited numbers of facilities. The accurate prediction model would allow investors or house buyers to determine the realistic price of a house as well as the house developers to decide the affordable house price. This paper addressed the attributes used by previous researchers to forecast a house price using various prediction models. Taken together, the results of the survey have shown the potential of Linear Regression and Random Forest in predicting house prices. These models were developed based on several input attributes and they work significantly positive with house price.

VII. REFERENCES

[1]. Keren He, Cuiwei He, "Housing price analysis using linear regression and logistic regression: A comprehensive explanation using Melbourne real estate data" 2021 IEEE International conference on computing (ICOCO)

[2]. Mansi Jain, Himani Rajput, Neha Garg, Pronika Chawla ,Manav Rachna, (2020) "Prediction of House Pricing Using Machine Learning with Python" International Conference on Electronics and Sustainable Communication Systems (ICESC 2020)

[3]. Data Imran, Umar Zaman, Muhammad Waqar and Atif Zaman "Using Machine Learning Algorithms for Housing Price Prediction: The Case of Islamabad Housing" Soft computing and machine intelligence journal (2021)

[4]. Yong Piao, Ansheng Chen, Zhendong Shang, "Housing Price Prediction Based on CNN" 2019 9th International Conference on Information Science and Technology (ICIST).

[5]. Pei-Ying Wang, Chiao-Ting Chen, Jain-Wun Su, Ting- Yun Wang, and Szu-Hao Huang, (2021) "Deep Learning Model for House Price Prediction Using Heterogeneous Data Analysis Along with Joint Self-Attention Mechanism".

[6]. Smith Dabreo, Shaleel Rodrigues, Valiant Rodrigues, Parshvi Shah, "Real Estate Price Prediction" International Journal of Engineering Research and Technology (2021)..

[7]. Abigail Bola Adetunjia , Oluwatobi Noah Akande , Funmilola Alaba Ajala , Ololade Oyewo , Yetunde

Faith Akande , Gbenle Oluwadara, ‘‘House Price Prediction using Random Forest Machine Learning Technique ”The 8th International Conference on Information Technology and Quantitative Management Procedia Computer Science 2022.

[8]. The Danh Phan “Housing Price Prediction using Machine Learning Algorithms: The Case of Melbourne City Australia” 2018 International Conference on Machine Learning and Data Engineering (iCMLDE)

[9]. Quang Truong, Minh Nguyen, Hy Dang, Bo Mei “Housing Price Prediction via Improved Machine Learning Techniques” 2019 International Conference on Identification Information and Knowledge in the Internet of Things (IIKI2019)

[10]. Imran, Umar Zaman, Muhammad Waqar and Atif Zawan “Using Machine Learning Algorithms for Housing Price Prediction: The Case of Islamabad Housing Data” (Soft Computing and Machine Intelligence Journal, Vol (1), Issue (1), 2021

[11]. Anand G. Rawool , Dattatray V. Rogye , Sainath G. Rane , DR. Vinayk A. haradi (2021),”House Price Prediction Using Machine Learning” Finolex Academy of Management and Technology, Mumbai University.

[12]. Bandar Almaslukh, “A gradient boosting method for effective prediction of housing price in complex real estate systems” 2020 International conference on technologies and applications of artificial intelligence (TAAI)

[13]. Maryam Heidari, Samira Zad, Setareh Rafatirad, “Ensemble of supervised and unsupervised learning models to predict a profitable business decision”2021 IEEE

[14]. Karshiev asanjar, Olimov Bekhzod, Jaesoo Kim, Anand Paul and Jeonghang Kim ”Missing data imputation of geolocation-based price prediction using KNN-MCF Method”, international journal of geo information (2020).

[15]. Xiangqin Cheri, “Optimizations of training dataset on house price estimation”2021 2nd International Conference on big data economy and international management (BDEIM).

[16]. Ping-Feng Pai and Wen-Chang Wang “Using Machine Learning Models and Actual Transaction Data for Predicting Real Estate Prices” (2020) Department of Information Management, National Chi Nan University, 1 University Rd., Puli, Nantou 54561, Taiwan

[17]. Debanjan Banerjee, Suchibrota Dutta “Predicting the Housing Price Direction using Machine Learning Techniques” IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI-2017)

[18]. Marjan Ceh, Milkan Kilibarda, Anka Lisec and Branislav Bajat “Estimating the Performance of Random Forest versus Multiple Regression for Printing Prices of the Apartments” (2018). International journal of Geo-Information

[19]. Susmita Ray “A Quick Review of Machine Learning Algorithms” (2019) IEEE, International Conference of On Machine Learning, Big Data, Cloud and Parallel Computing (Com-IT-Con), India

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

Cite this article as :

Harshada S. Belsare, Prof. Kanchan V. Warkar " A Novel Model for House Price Prediction with Machine Learning Techniques", International Journal of Scientific Research in Science and Technology(IJSRST), Print ISSN : 2395-6011, Online ISSN : 2395-602X, Volume 10, Issue 3, pp.743-754, May-June-2023. Available at doi : https://doi.org/10.32628/IJSRST523103134

Journal URL : https://ijsrst.com/IJSRST523103134