ACKNOWLEDGMENT

With immense pleasure I would like to present this report on my topic **HOUSE PRICE PREDICTION.** I am thankful to all that have helped us a lot for successful completion of my project and providing us courage for completing the work.

I thankful to my Head of the Department **Dr. Sandip Modha**, my internal faculty guide **Prof. Dharmesh Tank**, for providing guidance throughout my work giving us their valuable time.

Finally, I would like to thank my parents and friends. who have directly or indirectly helped me in making the project work successfully.

DHRUVI DALAL [20BECE30023]
VANSHIL DETROJA[20BECE30031]
KHUSHI MODI [20BECE30034]
PARTH PANCHOLI [20BECE30068]

ABSTRACT

"House Price Predictor" is a system working on House Price Index (HPI) and is commonly used to estimate the changes in housing price. Since housing price is strongly correlated to other factors such as location, area, population, it requires other information apart from HPI to predict individual housing price. There has been a considerably large number of papers adopting traditional machine learning approaches to predict housing prices accurately, but they rarely concern about the performance of individual models and neglect the less popular yet complex models. As a result, to explore various impacts of features on prediction methods, this paper will apply both traditional and advanced machine learning approaches to investigate the difference among several advanced models. This paper will also comprehensively validate multiple techniques in model implementation on regression and provide an optimistic result for housing price prediction

TABLE OF CONTENTS:

TABLE OF FIGURES:

CHAPTER 1: INTRODUCTION

1.1 Introduction
1.2 Scope
1.3 Project Summary & purpose
1.4 Overview of project
1.5 Problem Definition
1.6 Plan of Work

1.1 INTRODUCTION:

As growth of innovations to business is going upward computer sciences tend to increase technological transformations. This can put out the vulnerability of security and increase protection of the data. By considering various machine learning models and using the data of real estate forms in Washington we predict the house prices in entire Washington. This project is all about predicting the house prices by considering the datasets of Washington real estate by using different class labels. As we need the data to predict house price, the supervised data is produced which plays key role in predicting the house price and help in dealing with the real estate entities. As we are using machine learning it is easier to achieve the target like higher intelligent predictions which are a benefit factor for futuristic projects and intelligent systems which are linked to robotics as well. Now a days, smartphones are super-advanced and handy devices which could be used for almost every daily tasks instead of laptops. Smartphones applications are widely available, popular and are easily adopted. Main methodology of machine learning is constructing the models using past data as a source to predict the new data. As population is increasing rapidly the market demand is also increasing at the same pace. Most of the public are vacating the rural areas because of scarcity of jobs and increment of unemployment. This ultimately results in increment of houses in cities. If they don't have enough idea about prices then it results in loss of moneys. Nowadays Machine Learning is a booming technology. Data is the heart of Machine Learning.

AI and Machine Learning holds the key position in the technological market. All industries are moving towards automation. So, we have considered ML as a main predicting subject in our project and worked using it. These days everything fluctuates. Starting with crypto and various business models varies day by day which includes real estate as well so in this project house prediction depends on real estate data and ML techniques. Many people want to buy a good house within the budget. But the disadvantage is that the present system doesn't calculate the house predictions so well and end up in loss of money. So, the goal of our project is to reduce money loss and buy good house. Many factors are there to be considered in order to predict the house price which includes budget factors and fewer house modifications according to the buyer. So, we are considering all of those factors and predicted using various machine learning techniques like SVR, KNN, SGB regression, CatBoost regression, Random Forest regression Keywords: Machine Learning, House prices, SVR, KNN, RFR, Decision-trees, CatBoost Regression, Power transformers, XGB Regression

1.2 Scope

Current Scope

Real estate appraisal is an integral part of the property buying process. Traditionally, the appraisal is performed by professional appraisers specially trained for real estate valuation. For the buyers of real estate properties, an automated price estimation system can be useful to estimate the prices of properties currently on the market. Such a system can be particularly helpful for novice buyers who are buying a property for the first time, with little to no experience.

Future Scope

There are quite a few things that can be polished or add in the future work. Though, we were able to identify most of the residential areas. There may be some more places that have housing complexes or multi-storey apartments which are located in commercial areas. Such apartments were not included in this paper and can be counted in future to give a more accurate result. With more and more demand for housing in metropolitan cities, there is a definite increase in the number private builders that provide real estate with additional amenities to attract more customers. There are several other models available that can be implemented for prediction. Data given as input to such model should be compatible with the tool used and the operators involved in the process. Also, more number of data sets can be used to increase the accuracy of the model. The main objective of using a different model should be to reduce the calculation time and carry out the whole process in ease.

1.3 Project Summary & Purpose

Project Summary

By analyzing historical data for house prices in Calgary along with various relevant features, we established some interesting patterns and trends. Using machine learning techniques, we were then able to identify a subset of the original features that are in a sense sufficient to describe our data. Having selected the most important features, we then trained an XGBoost model for change in house price prediction, which classified samples into one of four categories. This model gave an accuracy rate of 68.7 on a test set that we had kept separate during development. This model can therefore be used to predict, for example, which type of house within Calgary is likely to increase and decrease in price in the year 2021 based on various scenarios.

Purpose

Buying your own house is what every human wish for. Using this proposed model, we want people to buy houses and real estate at their rightful prices and want to ensure that they don't get tricked by sketchy agents who just are after their money. Additionally, this model will also help big companies by giving accurate predictions for them to set the pricing and save them from a lot of hassle and save a lot of precious time and money. Correct real estate prices are the essence of the market and we want to ensure that by using this model

1.4 Overview of the project

The paper studies the SVM algorithm in machine learning for house price prediction. It takes data from the user and process it and classify it using pre-available data and uses various classification algorithm and classifies data and predict the accurate price of the property. It then confirms that accurate prediction result also depends on the population and quality of the training dataset. Results obtained earlier through SVM vs optimized SVM were then be evaluated. From the comparative analysis done in the next section, SVM 10 shows comparable value over the other cryptocurrencies for this period. In the future, the model will be enhanced on the accuracy rate of the forecasted price. Future work will concentrate on the data preprocessing by including the sentiment data before the testing and training experiments.

1.5 Problem Definition

Let’s say we are a real estate agent, and we are in charge of selling a new house. We don’t know the price, and we want to infer it by comparing it with other houses. We look at features of the house which could influence the house, such as size, number of rooms, location, crime rate, school quality, distance to commerce, etc. At the end of the day, what we want is a formula on all these features which gives us the price of the house, or at least an estimate for it.

1.6 Plan of Work

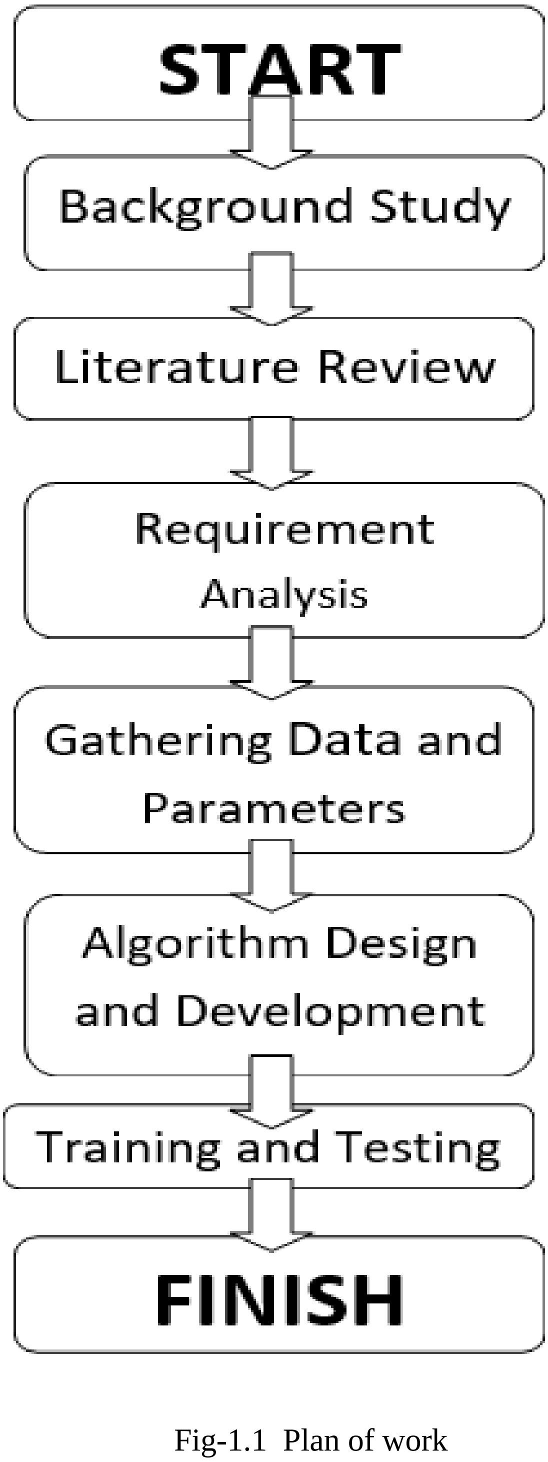


Fig-1.1 Plan of work

CHAPTER 2: LITERATURE REVIEW AND TECHNOLOGY

2.1 Literature Review
2.2 Technology Used
2.3 Software Used

2.1 Literature Review

House price prediction is a vast topic, which is implemented through a variety of Computer Science Methods. Like Machine Learning, Linear Regression, Decision Tree, Deep Learning, Fuzzy Logic, ANFIS (Adaptive-Neuro Fuzzy Inference System), and Linear performance pricing.

In proposed model of Machine Learning, the dataset is divided into two parts:

- Training and Testing.
- 80% of data is used for training purpose and 20% used for testing purpose.

The training set include target variable. The model is trained by using various machine learning algorithms, out of which SVR predict better results. For implementing the Algorithms, they have used Python Libraries NumPy and Pandas. SVR performs the best with 89.23% accuracy.

2.2 Technology Used

- **Python:** Python is a popular language for machine learning because it has a large and active community, a wealth of powerful libraries and frameworks, and a relatively simple and easy-to-learn syntax. Some of the most popular libraries and frameworks for machine learning in Python include Wrekzueg, and Flask. These libraries provide a wide range of tools and capabilities for tasks such as data preprocessing, model training and evaluation, and deployment. Additionally, Python has a wide range of other libraries and tools that are useful for machine learning, such as NumPy and pandas for data manipulation, and Matplotlib and Seaborn for data visualization.

- **Machine Learning:** Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy. Machine learning is a method of data analysis that automates analytical model building. It is a branch of artificial intelligence based on the idea that systems can learn from data, identify patterns and make decisions with minimal human intervention.

- **Transfer Learning:** Transfer learning is a technique in machine learning where a model trained on one task is reused as the starting point for a model on a second related task. The idea behind transfer learning is that many features learned by a model on one task can be reused on a second related task, allowing the second model to learn more quickly and perform better. Transfer learning is widely used in computer vision and natural language processing, where a model pre-trained on a large dataset can be fine-tuned on a smaller dataset. Transfer learning enables an efficient use of labeled data and time, allowing to solve problems with limited data, and achieve better performance in comparison to training a model from scratch.

2.3 Software Used

➢ **Google Colab:** Google Colab, or Google Colaboratory, is a cloud-based platform for machine learning and data science that allows users to write, run, and share code in Jupyter Notebooks. It is similar to Jupyter Notebook, but with the added benefit of running on Google's infrastructure, which provides users with access to powerful computing resources such as GPUs and TPUs. One of the main advantages of Google Colab is that it allows users to work with large datasets and perform computationally expensive tasks without the need for a powerful local machine. Additionally, Google Colab provides users with free access to GPUs, which can greatly speed up the training of machine learning models. Google Colab also integrates with other Google services, such as Google Drive, which allows users to easily store and share their notebooks. It also provides the option to connect to a local runtime, which e Google Colab is a great option for individuals, students, and researchers who want to work with machine learning and data science without the need to set up a local environment. It's free to use, and it's also a great option for collaboration and sharing work with others.nables the use of local hardware resources like a webcam or GPU.

CHAPTER 3: SYSTEM REQUIREMENT STUDY

3.1 User Characteristics
3.2 Hardware & Software Requirements
3.3 Constraints

3.1 User Characteristics

Educational Level:
At least users of the system should be comfortable with English language.
Technical Level:
User should be comfortable using general purpose applications on the system.
The user has to have at least Windows 7 and internet browsing skills to use the system.

3.2 Hardware & Software Requirements

Hardware Requirements

- Laptop with Basic Hardware
- Minimum 8 GB of RAM
- Free Space of 10 GB

Software Requirements

- Frontend: HTML, CSS
- Backend: Machine Learning & Python
- Operating System: Windows 7 or above.

3.3 Constraints

Reliability Requirements
The System database connectivity has been designed with a persistent connection to ensure system reliability. The system runs on a dedicated server to ensure that it is reliable at all the time.
Security considerations
The system has an authorization mechanism for users to identify their personal profiles. Therefore, different users will have different authorization levels to access the data. Password protection and simple procedures to prevent the unauthorized access are provided to the users. The system allows the user to enter the system only through proper user name and password.
Design Constraints
The system is an online web application which runs on any operating system platform. It is developed using HTML5, CSS, Python.

CHAPTER 4: SYSTEM DESIGN

4.1 Use Case Diagram
4.2 Class Diagram
4.3 Activity Diagram
4.4 Entity Relationship Diagram
4.5 Sequence Diagram
4.6 UML Diagram/ State Transition

4.1 Use Case Diagram

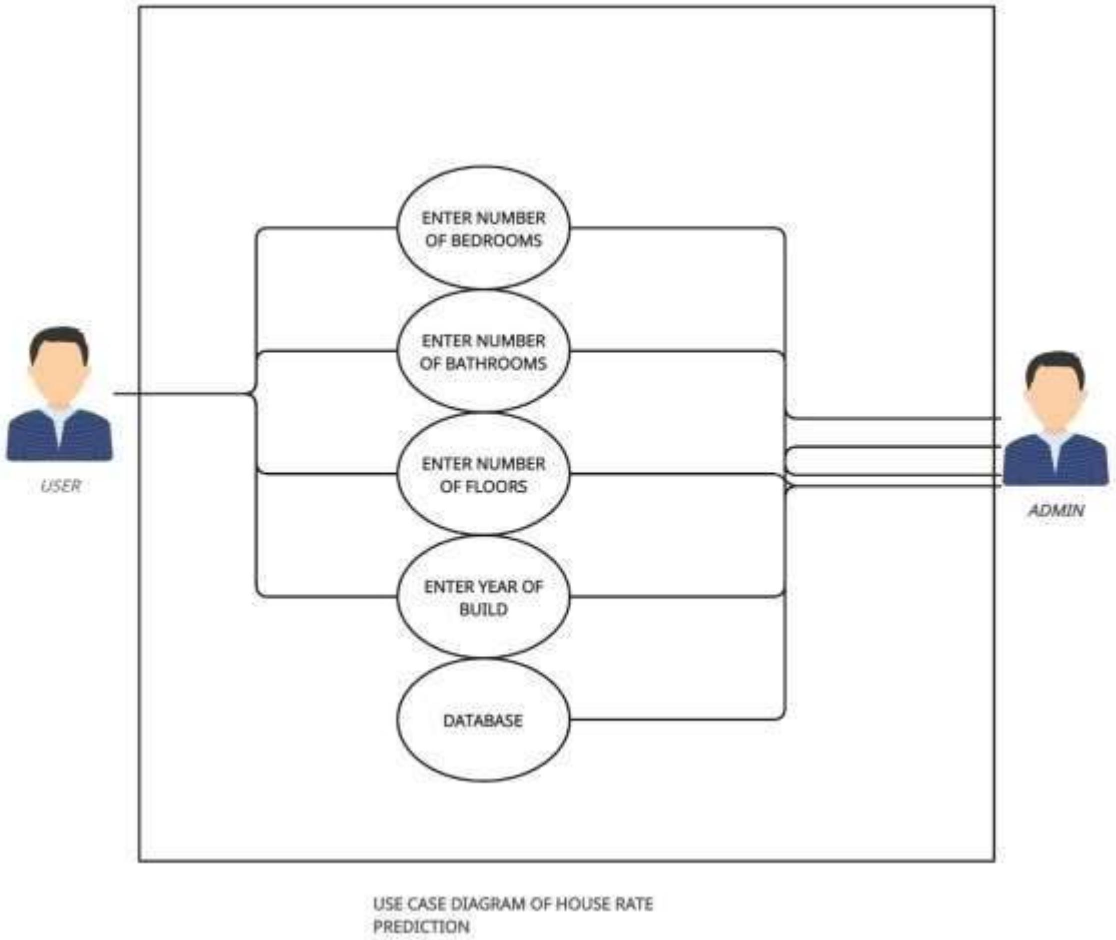


Fig-4.1 Use Case Diagram

4.2 Class Diagram

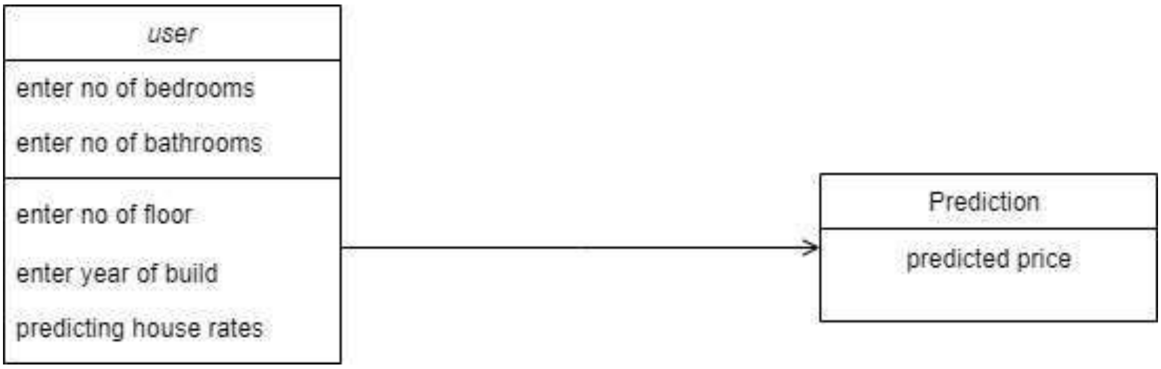


Fig-4.2 Class Diagram

4.3 Activity Diagram

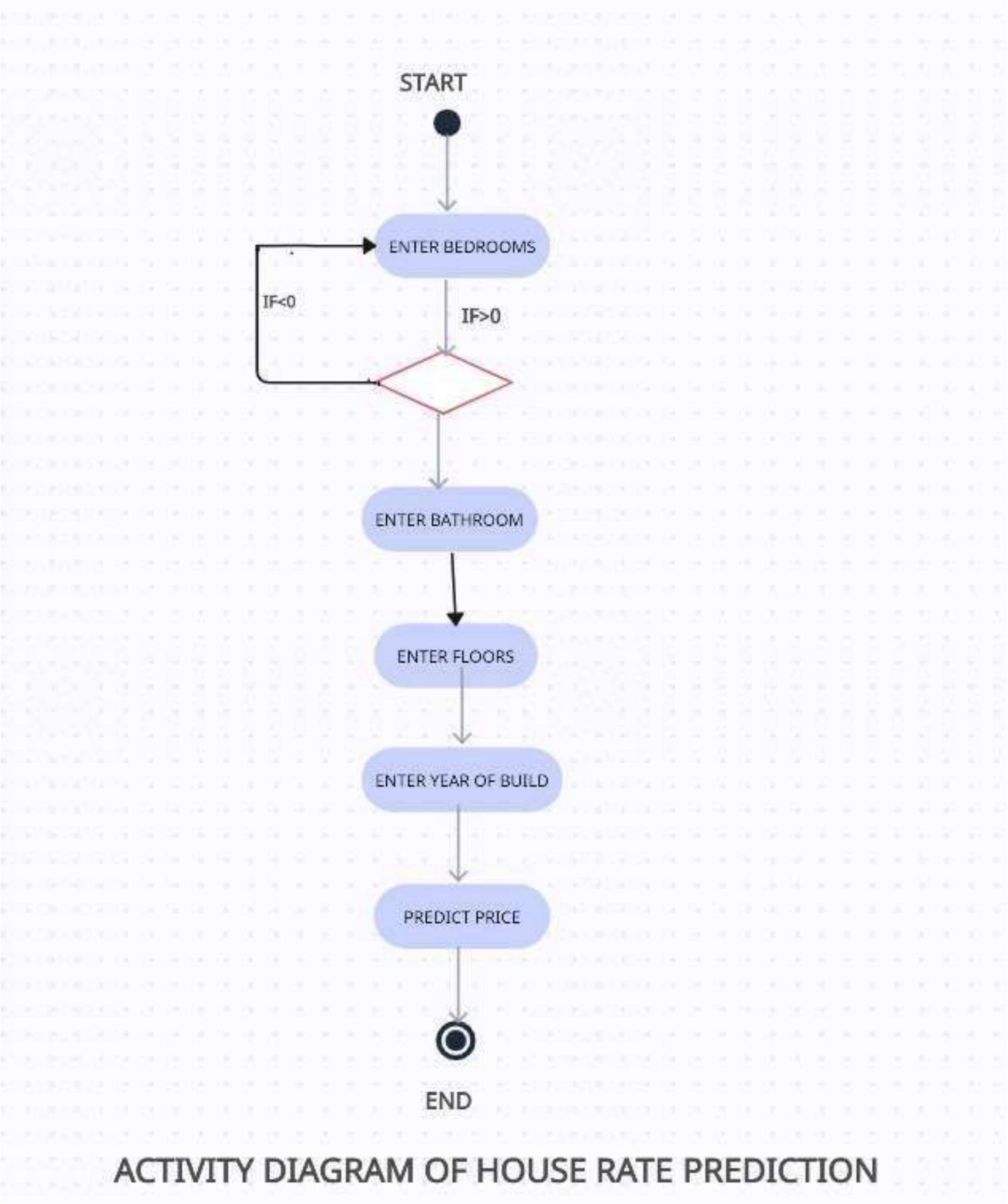


Fig-4.3 Activity Diagram

4.4 Entity Relationship Diagram

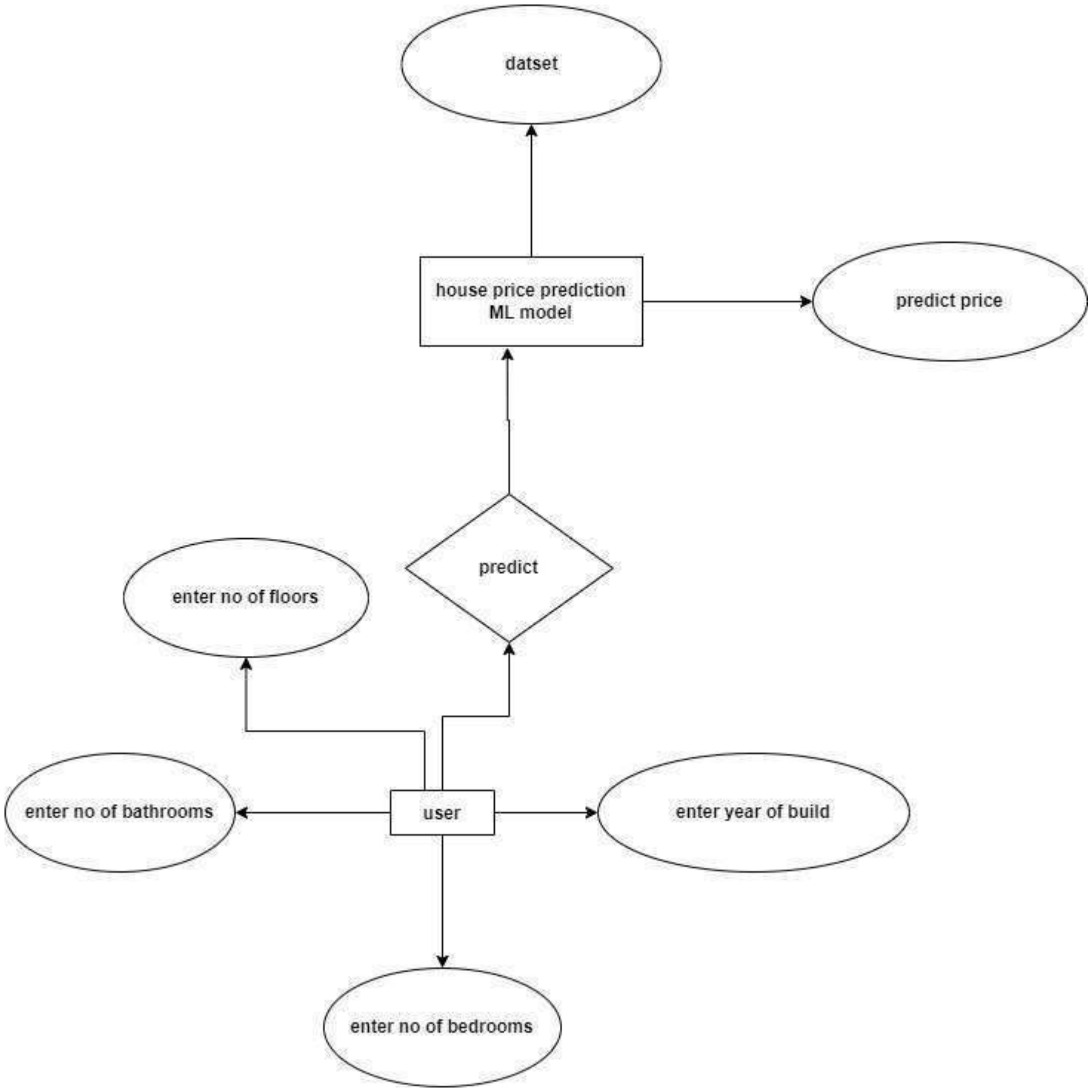


Fig-4.4 ER Diagram

4.5 Sequence Diagram

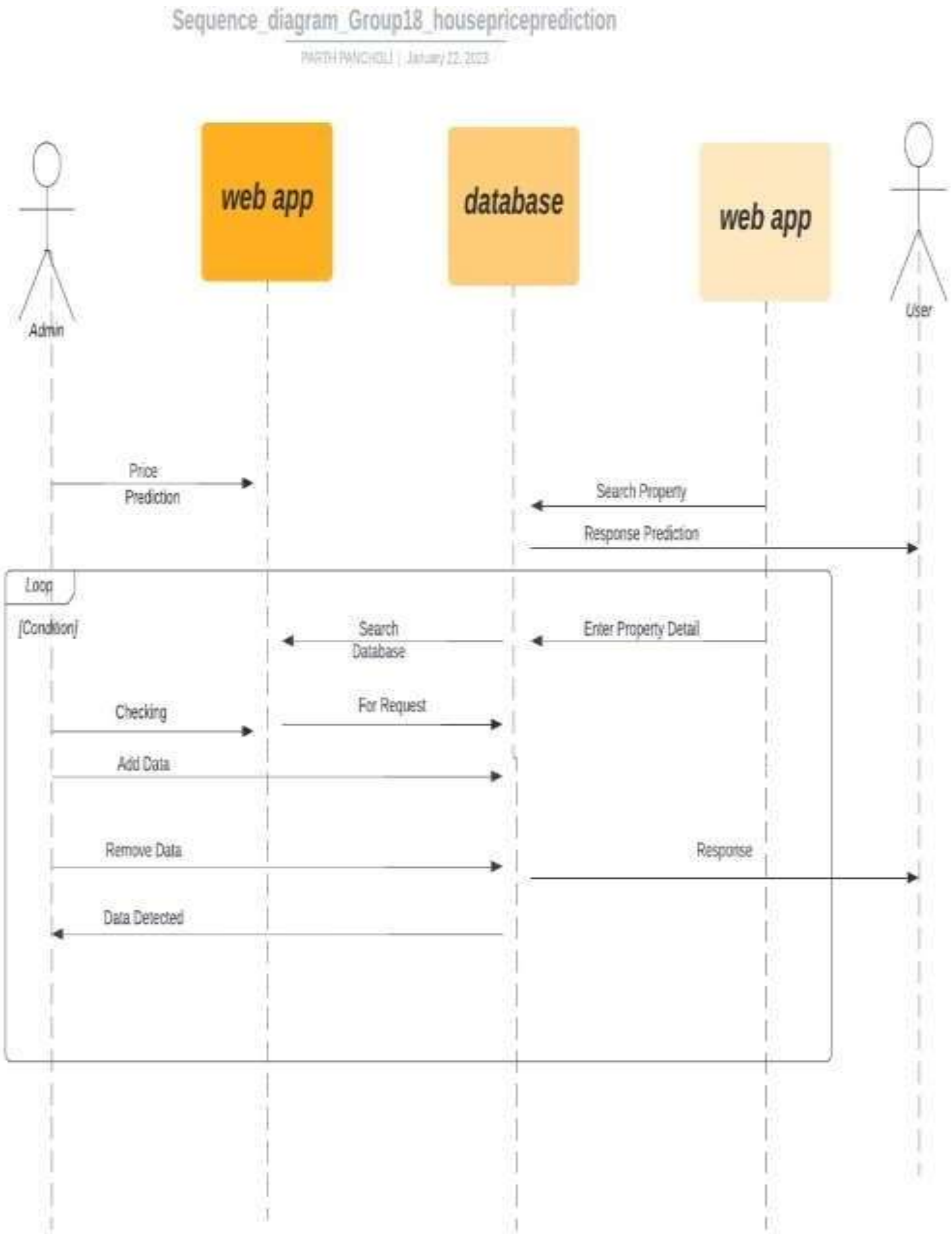


Fig-4.5 Sequence Diagram

4.6 UML Diagram/ State Transition

Context Level DFD

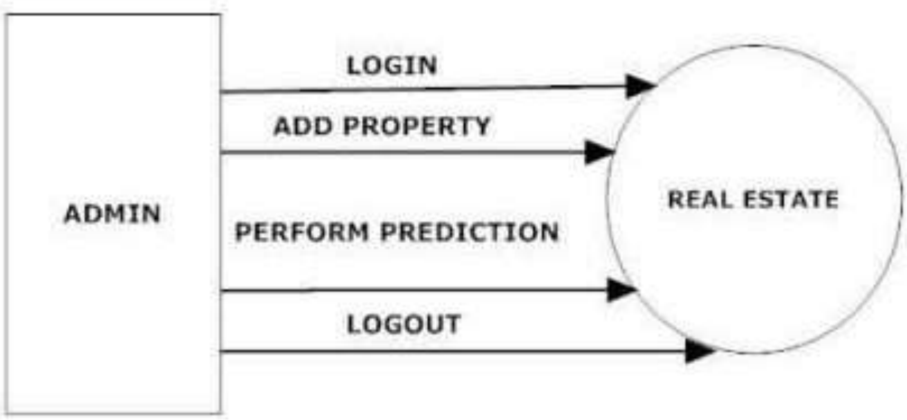
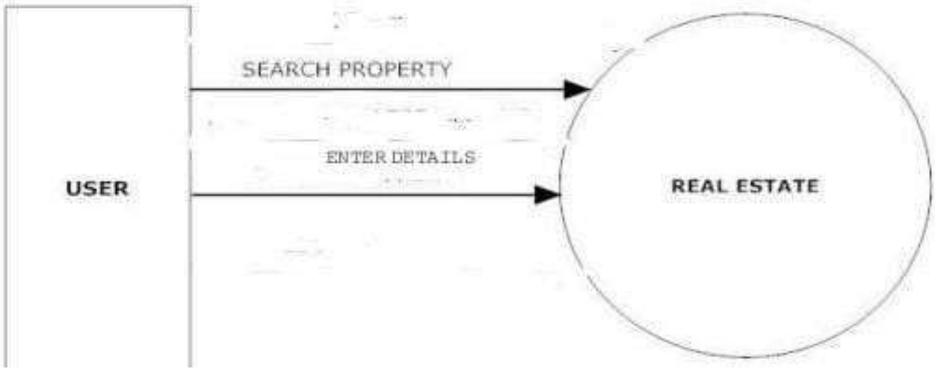


Fig-4.6 Context Level DFD Admin Module

Fig-4.7 Context Level DFD User Module



First Level DFD

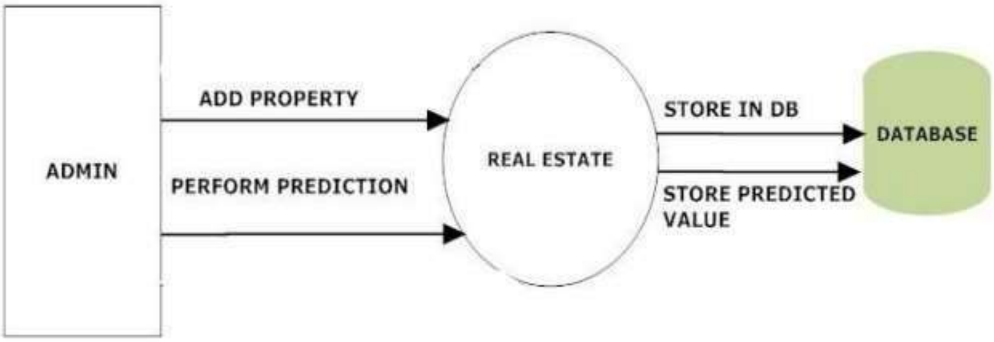
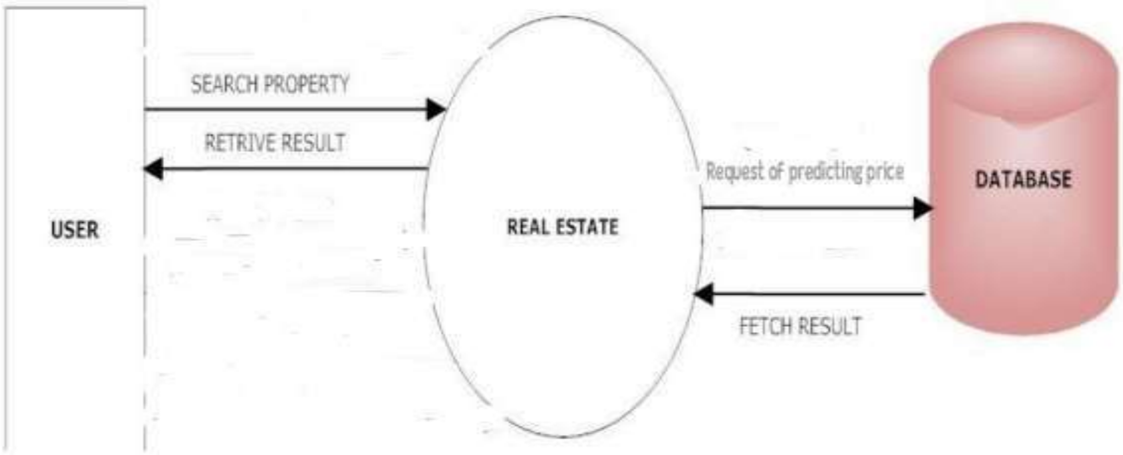


Fig-4.8 First Level DFD Admin Module

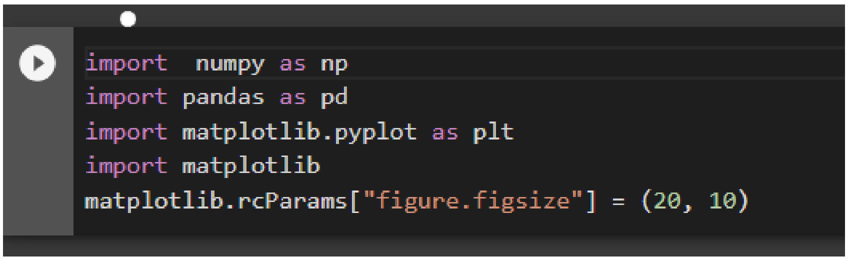
Fig-4.9 First Level DFD User Module



CHAPTER 5: IMPLEMENTATION

5.1 Install Packages
5.2 Data Cleaning & Handling Missing Data
5.3 Data Distribution
5.4 Data Frame creation
5.5 Predictions
5.6 Result

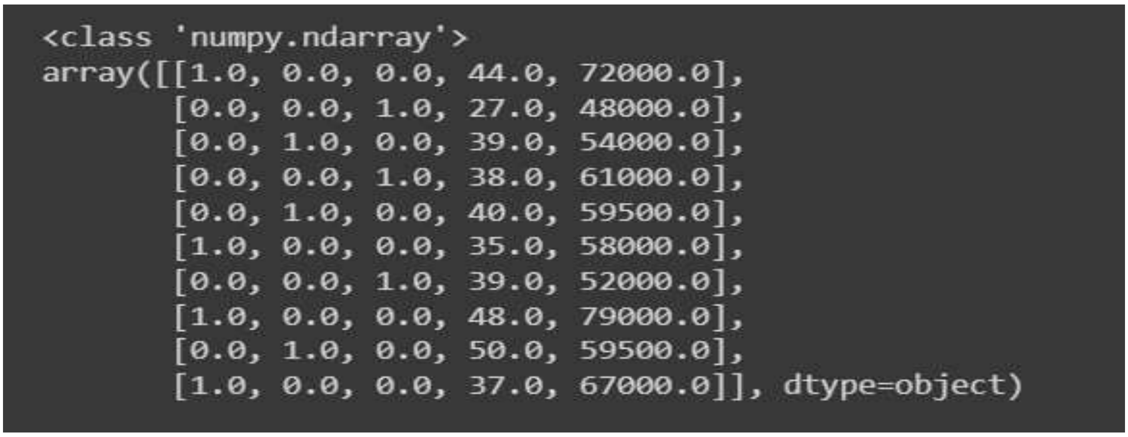
5.1 Install Packages



```
import  numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import matplotlib
matplotlib.rcParams["figure.figsize"] = (20, 10)
```

Fig- 5.1 Installation of packages

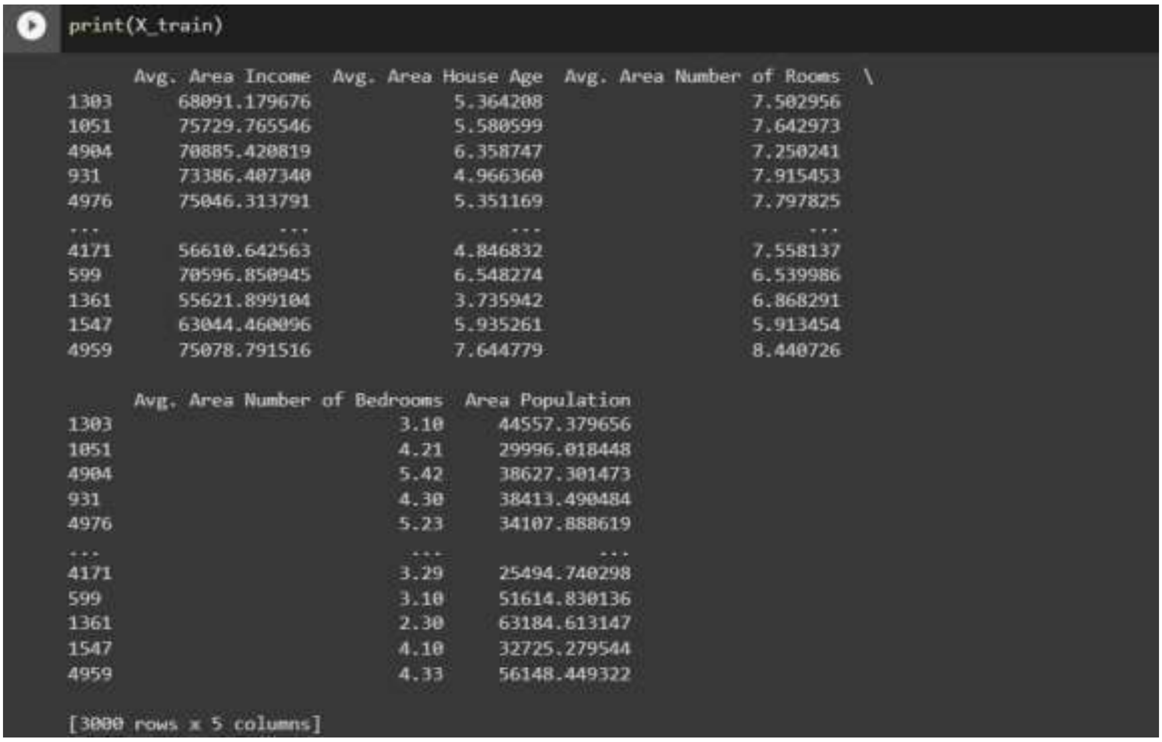
5.2 Data Cleaning & Handling Missing Data



```
<class 'numpy.ndarray'>
array([[1.0, 0.0, 0.0, 44.0, 72000.0],
       [0.0, 0.0, 1.0, 27.0, 48000.0],
       [0.0, 1.0, 0.0, 39.0, 54000.0],
       [0.0, 0.0, 1.0, 38.0, 61000.0],
       [0.0, 1.0, 0.0, 40.0, 59500.0],
       [1.0, 0.0, 0.0, 35.0, 58000.0],
       [0.0, 0.0, 1.0, 39.0, 52000.0],
       [1.0, 0.0, 0.0, 48.0, 79000.0],
       [0.0, 1.0, 0.0, 50.0, 59500.0],
       [1.0, 0.0, 0.0, 37.0, 67000.0]], dtype=object)
```

Fig- 5.2 Data Cleaning

5.3 Data Distribution



```
print(X_train)
```

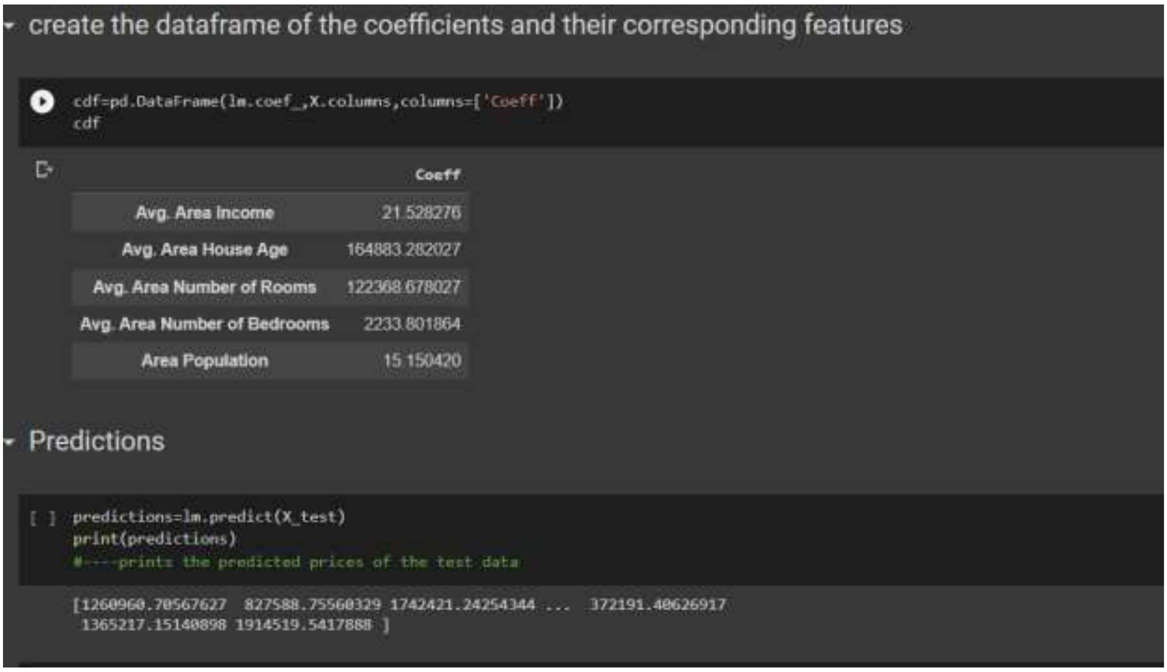
```
      Avg. Area Income  Avg. Area House Age  Avg. Area Number of Rooms  \
1303      68091.179676             5.364208                   7.502956
1051      75729.765546             5.580599                   7.642973
4904      70885.420819             6.358747                   7.250241
931       73386.407340             4.966360                   7.915453
4976      75046.313791             5.351169                   7.797825
...                ...                  ...                        ...
4171      56610.642563             4.846832                   7.558137
599       70596.850945             6.548274                   6.539986
1361      55621.899104             3.735942                   6.868291
1547      63044.460096             5.935261                   5.913454
4959      75078.791516             7.644779                   8.440726

      Avg. Area Number of Bedrooms  Area Population
1303                          3.10     44557.379656
1051                          4.21     29996.018448
4904                          5.42     38627.301473
931                           4.30     38413.490484
4976                          5.23     34107.888619
...                            ...              ...
4171                          3.29     25494.740298
599                           3.10     51614.830136
1361                          2.30     63184.613147
1547                          4.10     32725.279544
4959                          4.33     56148.449322

[3000 rows x 5 columns]
```

Fig- 5.3 Data Distribution

5.4 Data Frame creation



create the dataframe of the coefficients and their corresponding features

```
cdf=pd.DataFrame(lm.coef_,X.columns,columns=['Coeff'])
cdf
```

	Coeff
Avg. Area Income	21.528276
Avg. Area House Age	164883.282027
Avg. Area Number of Rooms	122368.678027
Avg. Area Number of Bedrooms	2233.801864
Area Population	15.150420

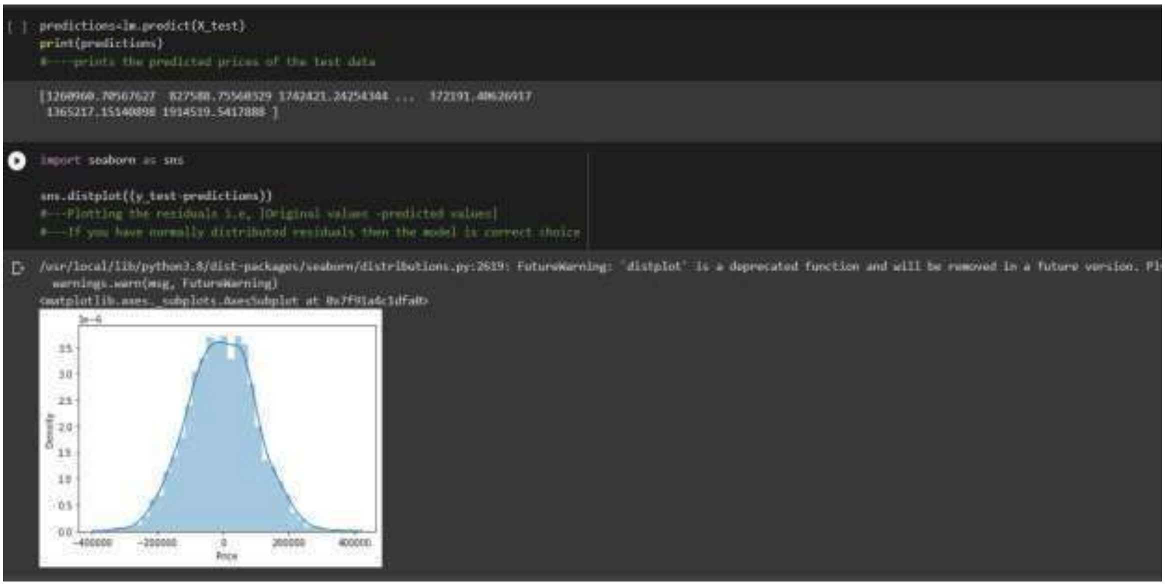
Predictions

```
predictions=lm.predict(X_test)
print(predictions)
#----prints the predicted prices of the test data
```

```
[1260960.70567627  827588.75560329 1742421.24254344 ...  372191.40626917
 1365217.15140898 1914519.5417888 ]
```

Fig- 5.4 Dataframe

5.5 Predictions



```
predictions=lm.predict(X_test)
print(predictions)
#----prints the predicted prices of the test data
```

```
[1260960.70567627  827588.75560329 1742421.24254344 ...  372191.40626917
 1365217.15140898 1914519.5417888 ]
```

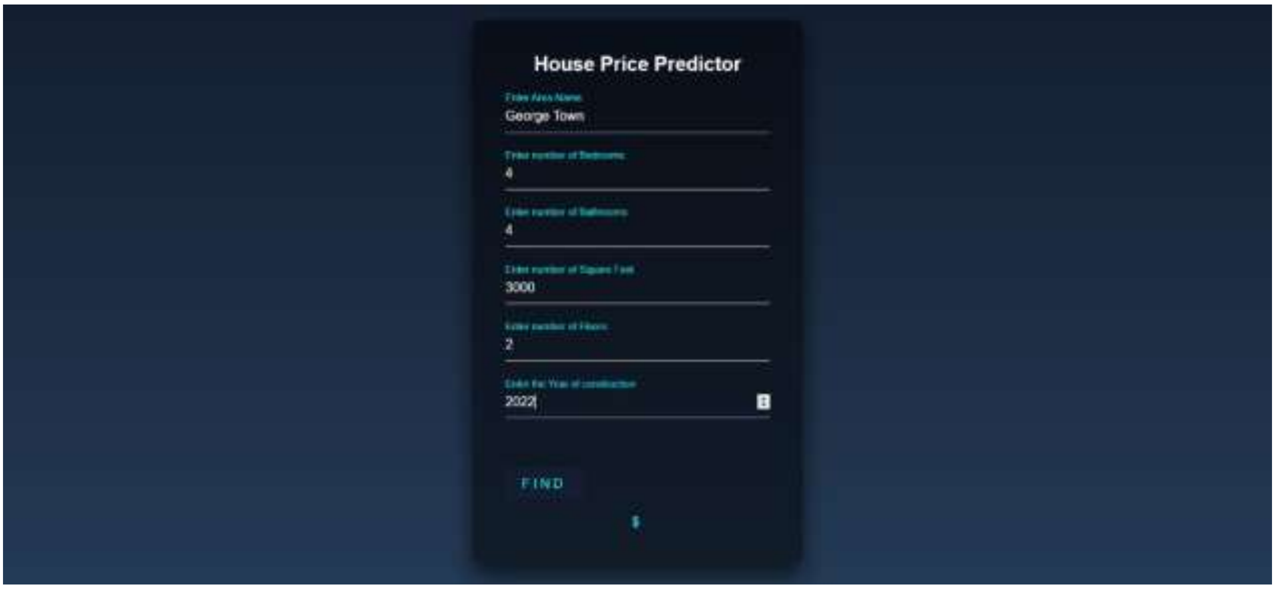
```
import seaborn as sns

sns.distplot((y_test-predictions))
#---Plotting the residuals i.e, ]Original values -predicted values]
#---If you have normally distributed residuals then the model is correct choice
```

```
/usr/local/lib/python3.8/dist-packages/seaborn/distributions.py:2619: FutureWarning: `distplot` is a deprecated function and will be removed in a future version. Pl
  warnings.warn(msg, FutureWarning)
<matplotlib.axes._subplots.AxesSubplot at 0x7f91a4c1dfa0>
```

Fig- 5.5 Predictions

5.6 Result



House Price Predictor

Enter Area Name
George Town

Enter number of Bedrooms
4

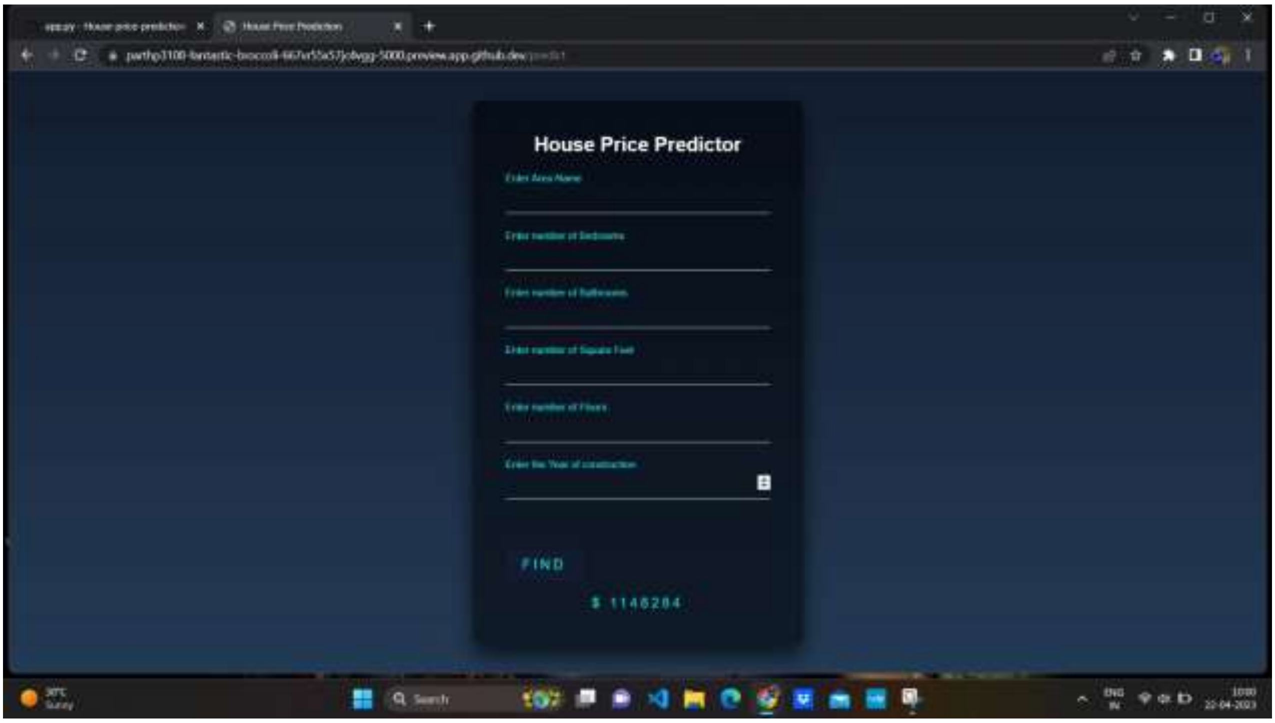
Enter number of Bathrooms
4

Enter number of Square Feet
3000

Enter number of Floors
2

Enter the Year of construction
2022

Fig- 5.6 Output



app.py - House price prediction
House Price Prediction
parthp3100-fantastic-broccoli-667vr55x57jc4vgg-5000.preview.app.github.dev/predict

House Price Predictor

Enter Area Name

Enter number of Bedrooms

Enter number of Bathrooms

Enter number of Square Feet

Enter number of Floors

Enter the Year of construction

$ 1148284

CHAPTER 6: CONCLUSION

6.1 Conclusion

- We have built several machine learning regression models from scratch and we gained complete knowledge and several insights were obtained about regression models and power transformers and how they are developed.
- We have explored many algorithms in search of better accuracy in predicting the house prices such as support vector regressor, linear regression, k nearest neighbors, random forest regressor, AdaBoost regressor, CAT Boost regressor, XG Boost regressor, etc.
- We have compared all the algorithms which are mentioned in the earlier statement and came to a conclusion that the Cat Boost Regressor and SVR are giving the highest accuracy about to 90%. We have improved the prediction accuracy by up to 15% to the existing models

CHAPTER-7 REFERENCES

- www.google.com
- https://ijarcce.com/upload/2017/december-17/IJARCCE%2016.pdf
- www.github.com
- www.Kaggle.com
- www.draw.io
- Garriga C., Hedlund A., Tang Y., Wang P, "Regional Science and Urban Economics Ruralurban migration and house prices in China"
- Regional Science and Urban Economics (2020), p. 103613, March 2020.
- Wang X., Li K., Wu J. "House price index based on online listing information : The case of China" Journal of Housing Economics, 50 (2020), p. 101715 May 2018.
- G.Naga Satish, Ch.V.Raghavendran, M.D.Sugnana Rao, Ch.Srinivasulu "House Price Prediction Using Machine Learning". IJITEE, 2019. [4] Bharatiya, Dinesh, et al. "Stock market prediction using linear regression." Electronics, Communication, and Aerospace Technology (ICECA), 2017 International conference of. Vol. 2. IEEE, 2017.
- Anand G. Rawool1 , Dattatray V. Rogye , Sainath G. Rane , Dr. Vinayk A. Bharadi, "House price predition using Machine Learning, IRE Journals, May 2021.
- E.Laxmi Lydia, Gogineni Hima Bindu, Aswadhati Sirisham, Pasam Prudhvi Kiran, "Electronic Governance of Housing Price using Boston Dataset Implementing through Deep Learning Mechanism", IJRTE, Volume-7 Issue-682, April-2019.
- Li Yu, Chenlu Jiao, Hongrun Xin, Yan Wang, Kaiyang Wang, " Prediction on Housing
- Price Based on Deep Learning", World Academy of Science, Engineering and Technology International Journal of Computer and Information Engineering Vol.12, No.:2, 2018