



INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

HOUSE PRICE PREDICTION USING MACHINE LEARNING

Bharti Vidhury, Ansh Tyagi^{*1}, Jayant Kumar Jyoti^{*2}, Rajat Sharma^{*3}, Kaustubh Upadhyay^{*4}

1. Assistant Professor, Department of Computer Science & Engineering, SRM Institute of Science & Technology, Ghaziabad
2. Department of Computer Science & Engineering, SRM Institute of Science & Technology, Modinagar, Ghaziabad
3. Department of Computer Science & Engineering, SRM Institute of Science & Technology, Modinagar, Ghaziabad
4. Department of Computer Science & Engineering, SRM Institute of Science & Technology, Modinagar, Ghaziabad
5. Department of Computer Science & Engineering, SRM Institute of Science & Technology, Modinagar, Ghaziabad

ABSTRACT -

This project demonstrates the usage of machine learning algorithms in the prediction of House/Villa prices. House Price Index (HPI) is commonly used to estimate the changes in housing prices. Since housing price is strongly correlated to other factors such as location, area, and population, it requires other information apart from HPI to predict individual housing prices. This project will comprehensively validate multiple techniques in model implementation using AWS E2C (Amazon elastic compute cloud) and provide an optimistic result for housing price prediction.

Keywords: Dataset, Classifications, Machine Learning.

1. INTRODUCTION -

This section is about Machine learning problem and their solution methods. Generally, Machine Learning problems can be classified into classification or Binary and Distributive problems. Here we are dealing with the Distributive Problem in which we will have to process the data, categorize it, remove the null values, and then only we will be able to train the model for Distributive Problems we Mostly use LRA or Random Forest Algorithm (RFA) In forward problems, we have used Linear Regression for solving the Problem of Predicting the number of cases in coming future with help of real-time data collected from various sources here two parameters are Days and Confirmed Cases. In the area of Random forest, we just use various decision trees to train them with data collected then based on the output of all the decision trees.

JUPYTER NOTEBOOK

Jupyter Notebook is one of the best Python interpreters provided by Anaconda which is used for performing machine learning and data science processes. It is available free and comfortable to use.

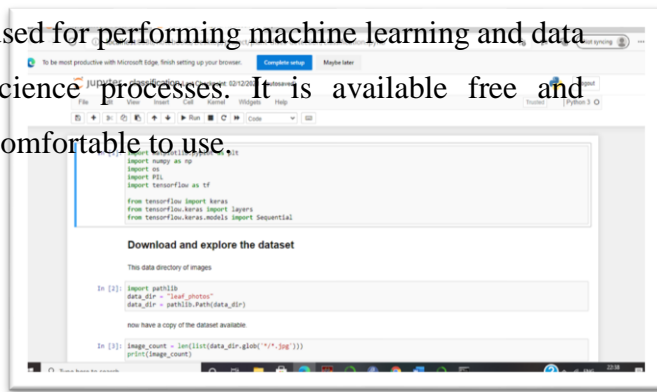


Figure 1: Jupyter Notebook interface

SYSTEM ANALYSIS.

We are going to discuss the various experiments done to find the most accurate model for calculating house prices. We will be discussing the problem and the system we are making to solve that problem. We will be dealing with various machine learning algorithms like decision trees, linear regression, random forest, etc. We will analyze the model to find how the price of a ride depends on different parameters like weather, time, destination, surge multiplier, icon, etc.

2. METHODOLOGY

2.1 PRE-PROCESSING

In this phase, we encode variables. As part of the clean-up, we do an imputation for missing values. Then, all attempts are made to remove disparity

within the set. The dataset is then portioned into a training and a test static. The involved steps are:

1. Transforming categorical features into numerical variables.
2. Replace the non-numeric or missing data with correct values without disturbing central tendency.
3. Data standardization or normalization.
4. Divide the dataset into train-test sections

The null values of the 'balcony' feature are imputed with mode. The null values of the 'bath' feature have been imputed with mode i.e, '2 BHK' in both sets. I observe that area values are in square meters. They are transformed into square feet as it is practically more relevant.

2.2 MODELING

This stage uses regression algorithms such as decision tree and lasso. These algorithms provide better results for regression problems.

2.3 PRICE PREDICTION

Following the classification results, we will forecast a property's price and discuss the findings.

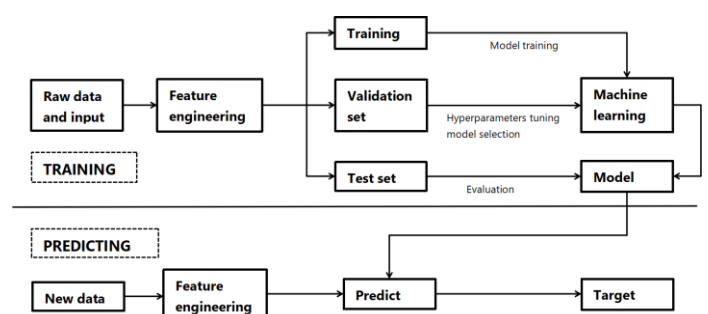


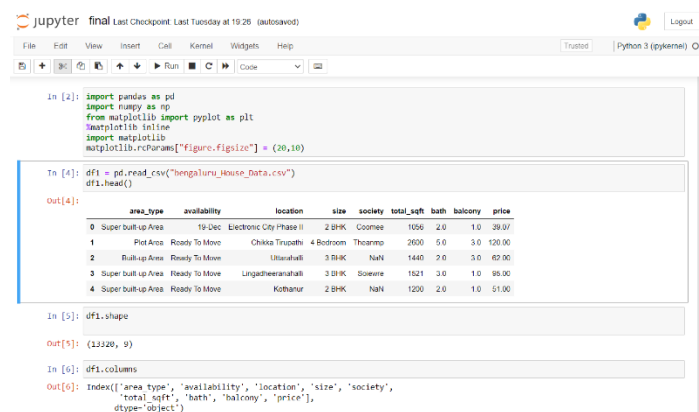
Figure 2: The proposed structure of the methodology

IMPLEMENTATION AND RESULT

IMPLEMENTATION

We first build a model using sklearn and linear regression using the Bangalore home prices dataset from kaggle.com. Then we write a Python flask server that uses the saved model to serve HTTP requests. The third component is the website built in HTML, CSS, and JavaScript that allows the user to enter the home square ft area, bedrooms, etc and it will call the python flask server to retrieve the predicted price

Data cleaning techniques for a house price prediction project, including downloading a dataset into pandas, examining the features of the dataset, dropping certain columns, handling null values, and creating a new column called BHK to account for inconsistencies in the size feature. The dataset has 13,000 rows, and the dependent variable is price. It covers installing Anaconda distribution and importing basic libraries such as Jupiter Notebook and Pandas. The data cleaning process involves dropping null values and applying a lambda function to transform the BHK column into an integer value.



```

In [2]: import pandas as pd
import numpy as np
from matplotlib import pyplot as plt
import matplotlib.pyplot as plt
import matplotlib
matplotlib.rcParams["figure.figsize"] = (20,10)

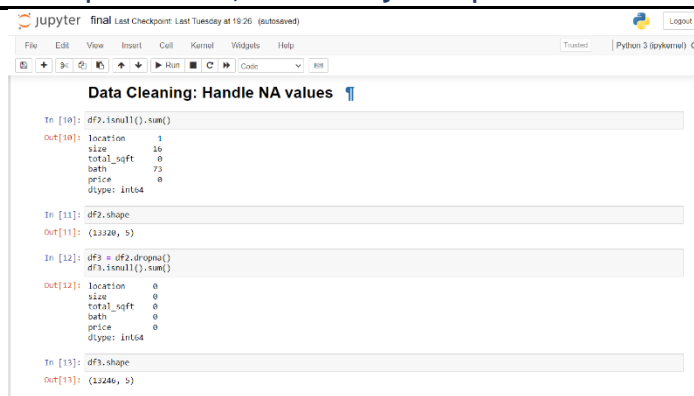
In [4]: df1 = pd.read_csv("bengaluru_house_data.csv")
df1.head()

Out[4]:
  area_type  availability  location  size  society  total_sqft  bath  balcony  price
0  Super built-up Area  19-Dec  Electronic City Phase II  2 BHK  Coomee  1056  2.0  1.0  39.0/
1  Plot Area  Ready To Move  Chikka Truspathi  4 Bedroom  Thearmp  2600  5.0  3.0  120.00
2  Built-up Area  Ready To Move  Ulhasahalli  3 BHK  NaN  1440  2.0  3.0  82.00
3  Super built-up Area  Ready To Move  Lingadheeranahalli  3 BHK  Soware  1821  3.0  1.0  95.00
4  Super built-up Area  Ready To Move  Kothanur  2 BHK  NaN  1200  2.0  1.0  51.00

In [5]: df1.shape
Out[5]: (13320, 9)

In [6]: df1.columns
Out[6]: Index(['area_type', 'availability', 'location', 'size', 'society',
            'total_sqft', 'bath', 'balcony', 'price'],
            dtype='object')
  
```

Figure 3: Model using sklearn



```

In [10]: df2.isnull().sum()
Out[10]: location      1
        size         16
        total_sqft    0
        bath         73
        price         0
        dtype: int64

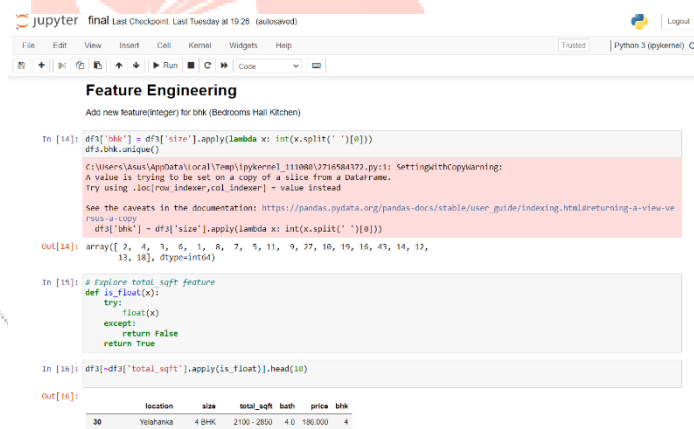
In [11]: df2.shape
Out[11]: (13320, 5)

In [12]: df3 = df2.dropna()
df3.isnull().sum()
Out[12]: location      0
        size         0
        total_sqft    0
        bath         0
        price         0
        dtype: int64

In [13]: df3.shape
Out[13]: (12246, 5)
  
```

Figure 4: Data handling

Feature engineering involves creating a new “price per square feet” feature that can help in outlier detection and removal. Dimensionality reduction techniques are used to handle high dimensionality problems like too many locations in the categorical feature "location," which is reduced using the "other" category.



```

In [14]: df3['bkh'] = df3['size'].apply(lambda x: int(x.split(' ')[0]))
df3.bkh.unique()

C:\Users\Ajay\AppData\Local\Temp\ipykernel_11088\2734581372.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-vs-copy
df3['bkh'] = df3['size'].apply(lambda x: int(x.split(' ')[0]))

Out[14]: array([ 2,  4,  3,  6,  1,  8,  7,  5, 11,  9, 27, 10, 19, 16, 43, 14, 12,
        13, 18], dtype=int64)

In [15]: # Explore total_sqft feature
def is_float(x):
    try:
        float(x)
    except:
        return False
    return True

In [16]: df3[~df3['total_sqft'].apply(is_float)].head(10)

Out[16]:
   location  size  total_sqft  bath  price  bkh
30  Yelahanka  4 BHK  2100-2850  4.0  186.000  4
  
```

Figure 5: Feature engineering

As a data scientist when we have a conversation with a business manager (who is an expert in real estate), he told us that normally square ft per bedroom is 300 (i.e 2 bhk apartment is a minimum of 600 sqft. If you have for example 400 sqft apartment with 2 bhk then that seems suspicious and can be removed such outliers by keeping our minimum threshold per BHK to 300 sqft.

Outlier detection and removal in a real estate price prediction project. Outliers are data points that represent extreme variations in a dataset and can create issues. Techniques for detecting and removing outliers include using the standard deviation or domain knowledge. An example of using domain knowledge is removing data points where the square footage per bedroom is less than a typical threshold of 300.

Scatter plot where green stars are 2bhk and blue dots are 3bhk, where the x-axis has the price of that area and y axis has the total square feet area.

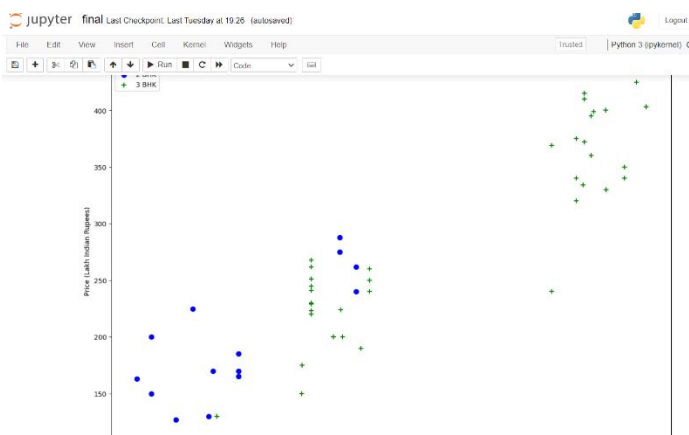


Figure 5: Scatter plot

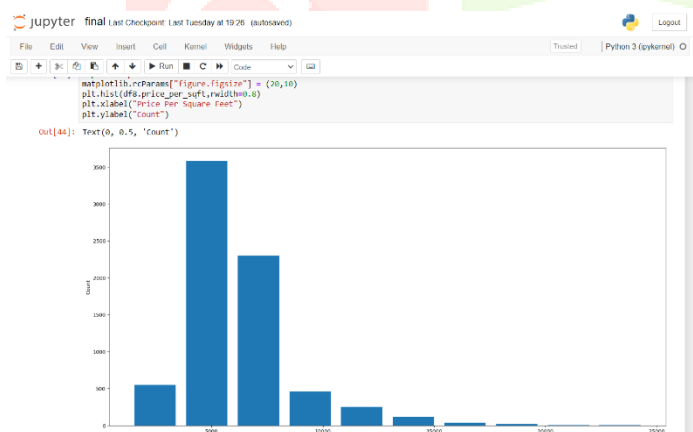


Figure 6: Histogram

Using k-fold cross-validation and grid cv to find the best algorithm and parameters. One hot encoding is used to convert the location column from categorical to numerical data. The panda's dummies method is used for one hot encoding. A separate data frame is created for the dummy columns and appended to the main data frame. To avoid a dummy variable trap, one less dummy column is used. The data frame is

prepared for model building by dropping the unnecessary columns and creating X and Y variables. The data set is split into a training and test data set. A linear regression model is created and trained on the X and Y training data. The model score is evaluated to be 84%, which is considered decent. K-fold cross-validation and shuffle split are used to evaluate the model's performance with different samples.

Now that our model is built and the artifacts are exported the next step would be to write a Python flask server that can sell HTTP requests made from the UI and it can predict the house prices. we are going to write that Python flask server which will be used as a back end for our UI application. The first step is to download PyCharm Community Edition. The project directory has three subfolders - client, server, and model - and two artifacts - a saved model and a columns JSON file. In the server folder, create a file named server.py and import the Flask module. Configure the interpreter as anaconda in the File Settings. Define the main function to run the Flask app on a specific port. Define a simple 'hello' function to return "Hi" using app.route() method. Run the Flask server on a specific URL by using the 'python server.py' command in the terminal. Create a subdirectory within the server directory named 'artifacts' and copy the exported model and columns JSON file into it. Define a function named 'get_location_names' in util.py to read the column JSON file and return a list of all the locations. Import util.py in server.py and call the get_location_names() function to return a JSON response containing all the location names. Load the saved artifacts into global variables using a function named 'load_saved_artifacts' in util.py. Define a function named 'get_estimated_price' in util.py to return the predicted price of a given location, total square feet, number of bedrooms, and number of bathrooms.

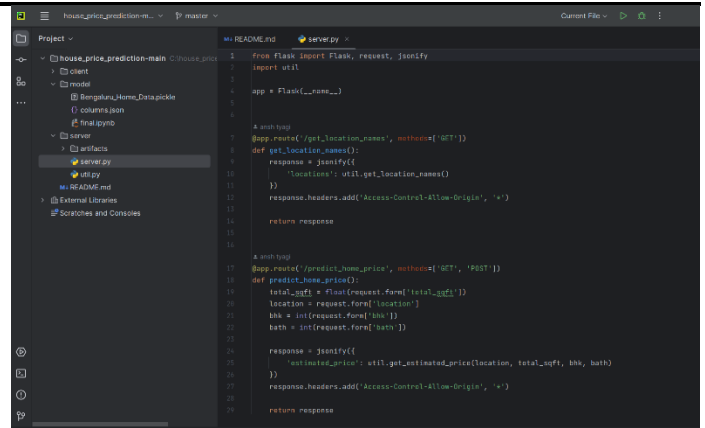
Import `util.py` in `server.py` and call the `get_estimated_price()` function to return a JSON response containing the predicted price. Test the Flask server by running the `'python server.py'` command in the terminal, opening the URL in the browser, and testing the `'get_location_names'` and `'get_estimated_price'` functions.

DEPLOY

Deploy a machine learning model to production on Amazon EC2 instance using Flask and Nginx. This project will be deployed is a Bangalore home price prediction website. The website and Flask server will be running on the same Amazon EC2 instance. The architecture of the application involves using Nginx as the web server and Flask as the Python server. Nginx will handle two HTTP requests, one for the website and the other for the Flask server. The website files will be returned by Nginx, while the Flask server will handle the prediction request using a saved ML model.

RESULT

We build a website using HTML, CSS, and JavaScript to serve as the front end of the house price prediction project. The website communicates with a back-end server using jQuery to retrieve data and estimate prices. We used Visual Studio Code as the code editor for the website. The HTML code contains two sections, head, and body, and the body includes various UI elements such as input fields, dropdowns, and buttons. The JavaScript code communicates with the back-end server to retrieve data and dynamically populate the dropdowns. Finally, we implement a function for the "estimate price" button to provide the estimated price to the user. Python flask server used as a backend for UI application, flask is a model that allows writing a python service which can saw HTTP request.



```

1 from flask import Flask, request, jsonify
2 import util
3
4 app = Flask(__name__)
5
6 @app.route('/get_location_names', methods=['GET'])
7 def get_location_names():
8     response = jsonify({
9         'locations': util.get_location_names()
10    })
11    response.headers.add('Access-Control-Allow-Origin', '*')
12    return response
13
14 @app.route('/predict_home_price', methods=['GET', 'POST'])
15 def predict_home_price():
16     total_sqft = float(request.args.get('total_sqft'))
17     location = request.args.get('location')
18     bhk = int(request.args.get('bhk'))
19     bath = int(request.args.get('bath'))
20
21     response = jsonify({
22         'estimated_price': util.get_estimated_price(location, total_sqft, bhk, bath)
23    })
24    response.headers.add('Access-Control-Allow-Origin', '*')
25    return response
  
```

Figure 7: python flask server

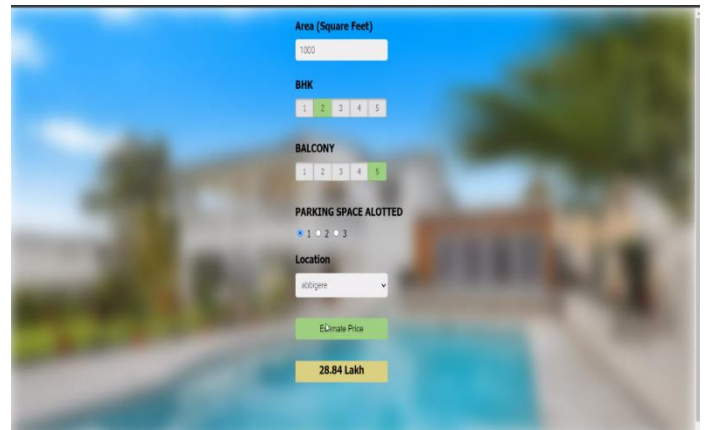


Figure 8: Website

CONCLUSION

In this paper, an overview of the concept of machine learning along with its various applications is discussed. Taking the sample dataset for houses, and considering its various attributes, the prices for houses have been predicted by employing machine learning methods of regression for predicting the price of the house using prior data, and clustering-for inspecting the quality of the solution and output. House selling prices are calculated using various algorithms. The selling price was calculated with better accuracy and accuracy than. This will be of great help to people. Various factors that affect home prices need to be considered and addressed.

FUTURE SCOPE

The accuracy of the gadget may be improved. Several extra cities may be protected within the gadget if the gadget's scale and computational strength increase. In addition, we can integrate different UI/UX methods to better visualize the results in a more interactive way using Augmented Reality.

REFERENCES

- J. Manasa, R. Gupta, and N. Narahari, "Machine learning based predicting house prices using regression techniques," in 2020 2nd International Conference on innovative mechanisms for industry applications (ICIMIA). IEEE, 2020, pp. 624–630.
- R. Sawant, Y. Jangid, T. Tiwari, S. Jain, and A. Gupta, "Comprehensive analysis of housing price prediction in pune using multi-featured random forest approach," in 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBE). IEEE, 2018, pp. 1–5.
- P.-Y. Wang, C.-T. Chen, J.-W. Su, T.-Y. Wang, and S.-H. Huang, "Deep learning model for house price prediction using heterogeneous data analysis along with joint self-attention mechanism," IEEE Access, vol. 9, pp. 55 244–55 259, 2021.
- Y. Piao, A. Chen, and Z. Shang, "Housing price prediction based on cnn," in 2019 9th international conference on information science and technology (ICIST). IEEE, 2019, pp. 491–495.
- Q. Truong, M. Nguyen, H. Dang, and B. Mei, "Housing price prediction via improved machine learning".
- Abdul G . Sario, Muhammad burhanHafez, "Fuzzy logic application for House price prediction,2015".
- Andrzej Bilazar and Maurizo d' Amato, "Residential market ratings using fuzzy logic decision-making procedures,2019".
- Jian Guan, Jozef Zurada and Alan S. Levitan, "An adaptive Neuro-Fuzzy inference system-based approach to real Estate property Assessment, 2020".
- Anand G. Rawaal, Dattatray V. Rogye, Sainath G. Rana, Dr. vinayk A, "House Price prediction using machine learning 2021".
- Pei-ying wang¹, Chiao-Ting chen², Jain-Wun su¹, Ting-Yun Wang¹, SZu-Hao Huang³, "Deep learning model for House Price prediction Using Heterogeneous Data Analysis along with joint self-attentionmechanism,2021".