VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belagavi - 590 018, Karnataka

HOUSE PREDICTION APP

A Report submitted in partial fulfillment of the requirements for the Course

Python Programming
(Course Code: 22AM4AEPPM)

In the Department of

Machine Learning
(UG Program: B.E. in Artificial Intelligence and Machine Learning)

By

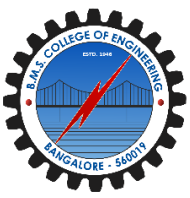
Archit Subudhi 1BM21AI026
Aryaman Sharma 1BM21AI027
Ayush K Dubey 1BM21AI028
Chetna Mundra 1BM21AI036

Semester & Section: 4A

Under the Guidance of

Prof. Arun Kumar N

Assistant Professor
Dept. of MEL, BMSCE, Bengaluru – 19



DEPARTMENT OF MACHINE LEARNING

B.M.S COLLEGE OF ENGINEERING

(An Autonomous Institute, Affiliated to VTU)

P.O. Box No. 1908, Bull Temple Road, Bengaluru - 560 019

September - 2023

B.M.S COLLEGE OF ENGINEERING

(An Autonomous Institute, Affiliated to VTU)

P.O. Box No. 1908, Bull Temple Road, Bengaluru - 560 019

DEPARTMENT OF MACHINE LEARNING

CERTIFICATE

This is to certify that Mr. ***Archit Subudhi*** bearing USN: ***1BM21AI026 ,*** Mr. ***Ayush Kumar Dubey*** bearing USN: ***1BM21AI028,*** Mr. ***Aryaman Sharma*** bearing USN: ***1BM21AI027,*** Ms. ***Chetna Mundra*** bearing USN: ***1BM21AI036*** has satisfactorily presented the Course – Python Programming (Course code: **22AM4AEPPM**) with the title "House Prediction" in partial fulfillment of academic curriculum requirements of the 5th semester UG Program – B. E. in Artificial Intelligence and Machine Learning in the Department of Machine Learning, BMSCE, an Autonomous Institute, affiliated to Visvesvaraya Technological University, Belagavi during September 2023. It is also stated that the base work & materials considered for completion of the said course is used only for academic purpose and not used in its original form anywhere for award of any degree.

Student Signature

Signature of the Supervisor

Prof. Arun Kumar N
Assistant Professor, Dept. of MEL, BMSCE

Signature of the Head

Dr. Gowrishankar
Prof. & Head, Dept. of MEL, BMSCE

External Examination

Examiner Name and Signature

1.

2.

ACKNOWLEDGEMENT

We express our profound gratitude to **Hon'ble Management and Dr. S Muralidhara,** Principal, BMSCE, Bengaluru for providing the necessary facilities and an ambient environment to work.

We are grateful to **Dr.Gowrishankar,** Professor & Head, Department of Machine Learning, Bengaluru, for his valuable suggestions and advice throughout our work period.

We would like to express our gratitude and sincere thanks to our guide **Prof. Arun Kumar N,** Assistant Professor, Department of Machine Learning, Bengaluru**,** for his keen interest and encouragement in guiding me and bringing the work to reality.

We would like to thank all the **Staff**, Department of Machine Learning for their support and encouragement during the course of this project.

Last but not the least, We would like to thank our parents, family and friends, without whose help and encouragement this work would have been impossible.

ABSTRACT

This report provides an overview of a house price prediction app designed to assist homebuyers, sellers, and real estate professionals in making informed decisions. Leveraging machine learning algorithms, this app analyzes various property features, location data, and market trends to provide accurate and up-to-date price predictions. Users can input property details, and the app swiftly generates a reliable estimate, enhancing transparency in real estate transactions. Key merits of the app include improved prediction accuracy, automation, and the identification of influential factors. However, potential challenges involve data quality, model selection, and ethical considerations. By addressing these concerns and ensuring data privacy, this app serves as a valuable tool in navigating the dynamic housing market, empowering users with actionable insights for their property-related endeavors.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Predicting house prices is a fundamental challenge in the real estate industry with far-reaching implications for buyers, sellers, and investors. The ability to accurately forecast house prices has become increasingly critical as property markets evolve and grow more complex. This prediction task involves the use of data analytics and machine learning techniques to model and estimate future property values based on a range of factors.

Understanding house pricing trends and making informed predictions is essential for a variety of stakeholders. For homebuyers, it helps them determine if a property is fairly priced or if it represents a good investment. Sellers benefit from pricing their homes competitively to attract buyers, while real estate agents rely on accurate predictions to guide their clients. Investors use price forecasts to make decisions about purchasing or divesting properties. Additionally, policymakers and financial institutions monitor house price trends to assess economic stability and formulate housing policies.

The house pricing prediction process involves analyzing historical sales data, examining variables like location, size, condition, and local market dynamics, and applying predictive models. Machine learning algorithms, including regression analysis, play a significant role in this endeavor. As real estate markets continue to evolve, accurate house pricing predictions become increasingly valuable for making informed decisions and navigating the complex world of property transactions.

1.1 About the Domain

The domain of house pricing prediction is instrumental in facilitating well-informed decisions in the real estate market. As technology and data analysis techniques continue to advance, the accuracy and reliability of house price predictions are expected to improve, further enhancing their importance within the real estate industry and beyond

1.2 Objective

we intended to make it as instructional as possible by tackling each stage of the machine learning process and attempting to comprehend it well. We have picked Bangalore Real Estate Prediction as a method, which is known as a "toy issue," identifying problems that are not of immediate scientific relevance but are helpful to demonstrate and practice. The objective was to forecast the price of a specific apartment based on market pricing while accounting for various "features" that would be established in the following sections.

1.3 Scope

The scope of house pricing prediction is multifaceted, spanning the real estate industry, financial markets, and economic analysis. It serves as a fundamental tool for homebuyers and sellers, aiding in informed decision-making and fair property transactions. Investors and developers rely on accurate predictions for profitable real estate investments and portfolio management. Policymakers and financial institutions monitor house price trends to assess economic stability and formulate housing policies. Technology and data-driven insights expand the scope, enabling market research, property valuation, and property management. Addressing ethical considerations, such as fairness and bias, further refines the field, ensuring transparency and equity in property valuation.

1.4 Motivation

We are highly interested in anything related to Machine Learning, the independent project provided us with the opportunity to study and reaffirm our passion for this subject. The capacity to generate guesses, forecasts, and offer machines the ability to learn on their own is both powerful and infinite in terms of application possibilities. Machine Learning may be applied in finance, medicine, and virtually any other field. That is why we opted to base our idea on Machine Learning.

CHAPTER 2

RELATED WORK

M Jain et al., [3] When analyzing a research paper on house price prediction, assess its merits based on data quality, methodology, accuracy, innovation, relevance to current issues, and peer review. Look for potential demerits such as data quality issues, overfitting, small sample sizes, lack of validation, limited transparency, neglect of external factors, and ethical considerations. A comprehensive evaluation of these aspects will help determine the paper's credibility and applicability for your research or practical use.

Quang Truong et al., [6] When evaluating a research paper on house price prediction in the International Research Journal of Engineering and Technology (IRJET) with a focus on Bangalore, consider its merits such as local relevance, unique data sources, predictive accuracy, innovative approaches, and practical applicability for the local housing market. Conversely, be cautious of potential demerits including data biases, small sample sizes, overfitting, data source reliability, model transparency, external factor considerations, ethical concerns, and the paper's peer-reviewed status. A comprehensive assessment will gauge the paper's credibility and its value in understanding housing dynamics in Bangalore.

O.Bhargav Sathwik et al., [7] Using machine learning for house price prediction offers advantages like improved accuracy, consideration of multiple variables, real-time updates, automation, feature importance identification, scalability, and data visualization. Challenges include data quality, overfitting, interpretability, data privacy, model selection, market volatility adaptation, and ethical considerations. Success hinges on data quality, model expertise, and addressing ethical concerns.

.

CHAPTER 3

OPEN ISSUES & PROBLEM STATEMENT

In the rapidly evolving field of real estate and property prediction applications, several challenges and open issues need to be addressed to enhance the accuracy, usability, and overall effectiveness of these apps. The problem statement for house prediction apps can be summarized as follows:

1. Data Quality and Quantity:
    - Problem Statement: The availability of high-quality data is crucial for accurate house price predictions. However, many regions may lack comprehensive and up-to-date property data, leading to accuracy and coverage issues.
    - Open Issue: Developing strategies to improve data quality and quantity, such as leveraging advanced data collection techniques, integrating more data sources, and addressing data biases.

2. Model Accuracy and Generalization:
    - Problem Statement: House prediction apps often rely on machine learning models that may struggle to generalize across diverse housing markets and property types, resulting in inaccurate predictions.
    - Open Issue: Research and development efforts should focus on improving model accuracy and generalization by fine-tuning algorithms, considering regional variations, and incorporating feature engineering techniques.

3. Interpretability and Explainability:
    - Problem Statement: Many users find it challenging to trust and use prediction apps when they cannot understand the factors driving the predictions.
    - Open Issue: Developing more transparent and interpretable machine learning models, as well as user-friendly interfaces that provide clear explanations of predictions and feature importance.

4. Privacy and Security:

- Problem Statement: House prediction apps often require access to sensitive user data, raising concerns about data privacy and security.
- Open Issue: Implementing robust privacy-preserving mechanisms, secure data handling practices, and compliance with data protection regulations to ensure user trust and data security.

5. Market Volatility and Economic Factors:

- Problem Statement: Real estate markets are influenced by economic factors and can experience fluctuations that are challenging to predict accurately.
- Open Issue: Incorporating economic indicators and market trends into prediction models to account for market volatility and enhance prediction accuracy.

6. User Experience and Accessibility:

- Problem Statement: Ensuring that house prediction apps are user-friendly and accessible to a wide range of users is essential for their success.
- Open Issue: Focusing on user experience (UX) design, optimizing app performance on various devices, and making the app accessible to individuals with disabilities.

7. Regulatory Compliance:

- Problem Statement: Compliance with local, regional, and national regulations, such as fair housing laws and data protection regulations, is crucial for the legal operation of house prediction apps.
- Open Issue: Implementing robust compliance measures, staying informed about evolving regulations, and conducting regular audits to ensure adherence to legal requirements.

Addressing these open issues and problem statements is essential for the continued development and success of house prediction apps, ultimately providing users with more accurate, transparent, and trustworthy tools for making informed property-related decisions.

CHAPTER 4

DATA COLLECTION & VALIDATION

Data collection involves getting hold of the right original sample set from a trusted source, citing researchers to information specifically tailored to their research objectives.
Accurate data collecting is crucial to preserving the integrity of research, regardless of the subject of study or preferred method for defining data. Errors are less likely to occur with the right data gathering tools.

1. Data validation means checking the accuracy and quality of source data before using or processing the data.
2. Data is pre-processed by removing or filling null values using appropriate methods.
3. Dropping incorrect data to avoid inaccuracies in the model being developed.

4.1 Feature Engineering

Feature Engineering involves selection and operations on important attributes to the model. Features must be selected wisely and conditioned as some features have a far more significant impact on the model output than others.

Some features can be dropped to avoid unnecessary complications after thorough insight into the features impact.

New features are developed using existing to provide hindsight into new possibilities offered by the future model.

In our housing price prediction model, we disposed the features - society, availability of the house and area build up type. These attributes have negligible impact on our model and in reality, don’t cause much of an obstacle when searching for a house. We also added price per square foot as an important feature as it plays an important role in the real estate market.

4.2 Data Analysis

Data analysis involves manipulating, transforming, and visualizing data to infer meaningful insights from the results. Some practices involved are removing outliers, transforming dimensions, etc.

Our data set was screened with two methods to get rid of outliers

1.Standard Deviation – This involves finding the Mean(X) of the data X values and allowing a strict deviation of +- StandardDeviation(X) from the Standard Deviation of data values X.

2.Z-Score Implemtation – Z-Score implementation allows us to keep several threshold values to monitor the normal distribution of the data and allow only a few acceptable entries based on the likelihood of the data occurring.

3.Inter-Quartile Range- Classic way to remove outliers beyond 1st Quartile and 3rd Quartile.

After performing all these steps, we are gifted with a clean data set ready to be operated on.

CHAPTER 5
DETAILED DESIGN

5.1 Proposed Architecture

The app must have a solid architecture to allow smooth processing of data, model files and at the same time provide a seamless experience to the user.

Backend Processing-
Involves steps following from data collection to model training. This must not be carried out in parallel to app operations as it leads to redundant processing for every app boot up.

Model Binary File-
Using pickle library, we can compact our trained model into a readable binary file which can be used by the app to effortlessly predict housing prices and produce accurate values.

Clean Dataset -
Once we are left with a good and accurate dataset after performing several pre-processing operations. This dataset is referred to in the app to display houses on sale and allow the user to comprehend the competitive pricing available between different houses.

User Database -
The user database meets the application's expectation with features such as log-in requirements, user information, etc.

Combining all these essential parts, we can build the application.

5.3 Methodology

1. Using Numpy and Pandas, the obtained original data set was conditioned to negate the influence of incomplete/null data samples along with features having negligible influence on the model mapped out.
2. Using Statistical techniques, the data was further conditioned to remove outliers which might lead to a bad fit of the model.
3. The Linear regression model was split in a 85-15 ratio for training and testing to discourage overfitting and allow more general prediction.
4. Using Scikit-learn, the model was standardized to proper dimensions and scale for easier operations on the data.
5. Matplotlib allowed data visualization for better understanding of how the data of various houses of different BHK, Bathrooms, Sizes ,etc was with respect to price in each segments.
6. Using Pickel, the model was wrapped into a readable binary file for referencing in the architecture of the application built up

5.4 Implementation

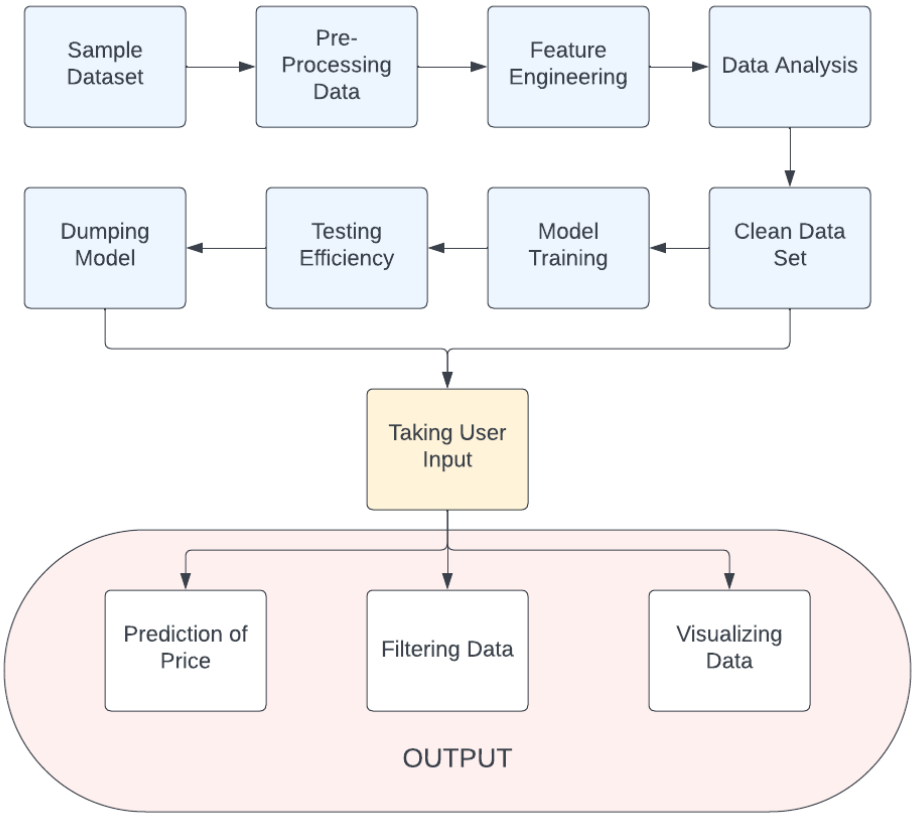


Figure 5. 1 Flowchart

After obtaining the sample set, the data is preprocessed to remove any NULL values or fill them using appropriate mean/median method. Feature Engineering is deployed to keep only impactful features and remove insignificant features. The data is screened heavily for outliers using Z-Score, Inter-Quartile Range and Standard Deviation methods.

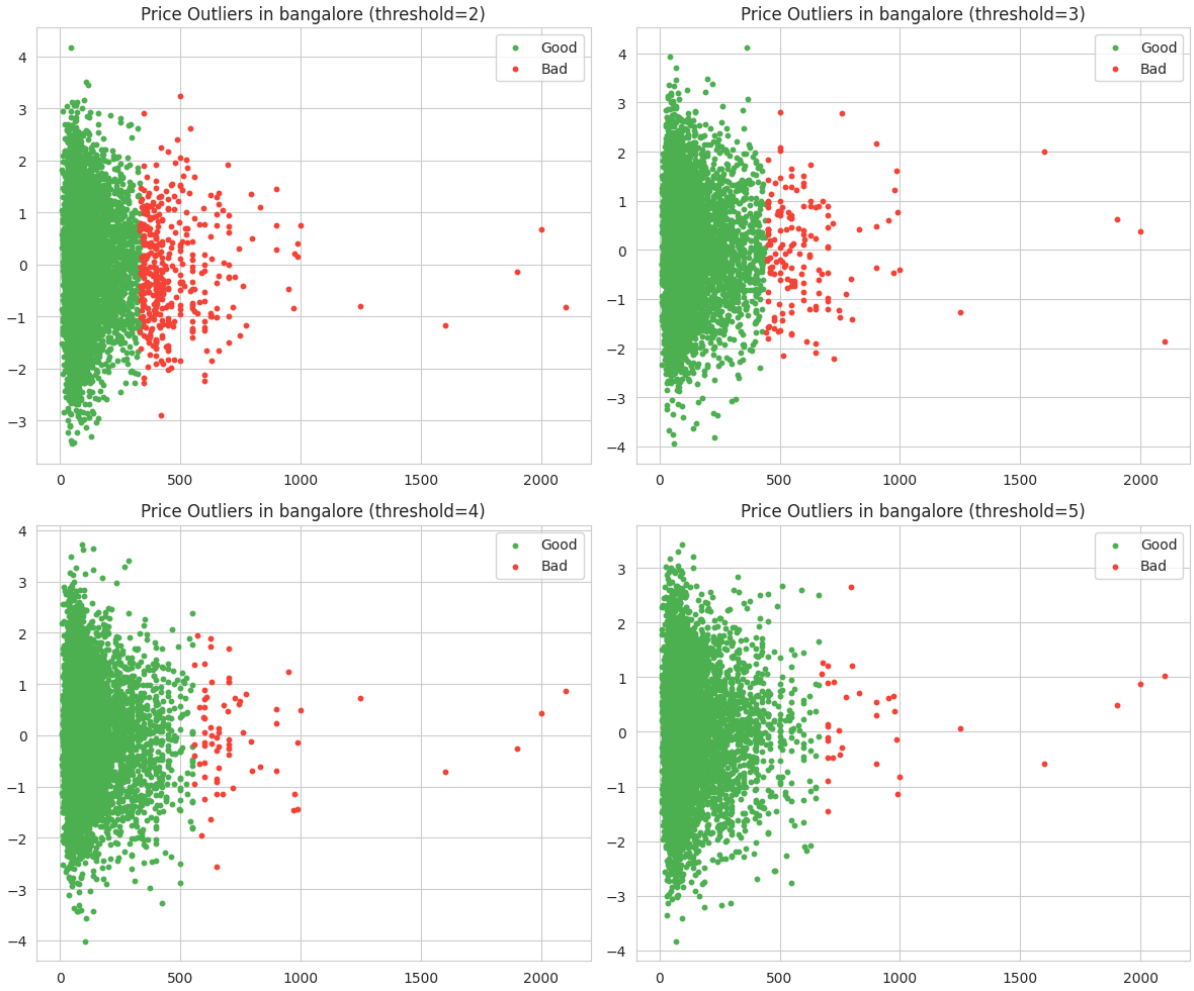


Figure 5. 2 Outliers Graph

Figure 5.2 shows the different acceptable and rejected values for different threshold values. Threshold 2 only accepts values closely knit together which is ideal for predictions. The visualizing of data allows us to better condition the data for further analysis.

After data is cleaned, the data is split and ready to train the model. The model implemented is Multivariant Linear Regression. Before we allow the model to start training, the different independent variables must be scaled to a standard to allow smooth training. The model is also prepared in such a way that it doesn't overfit and is generalized for general purposes.

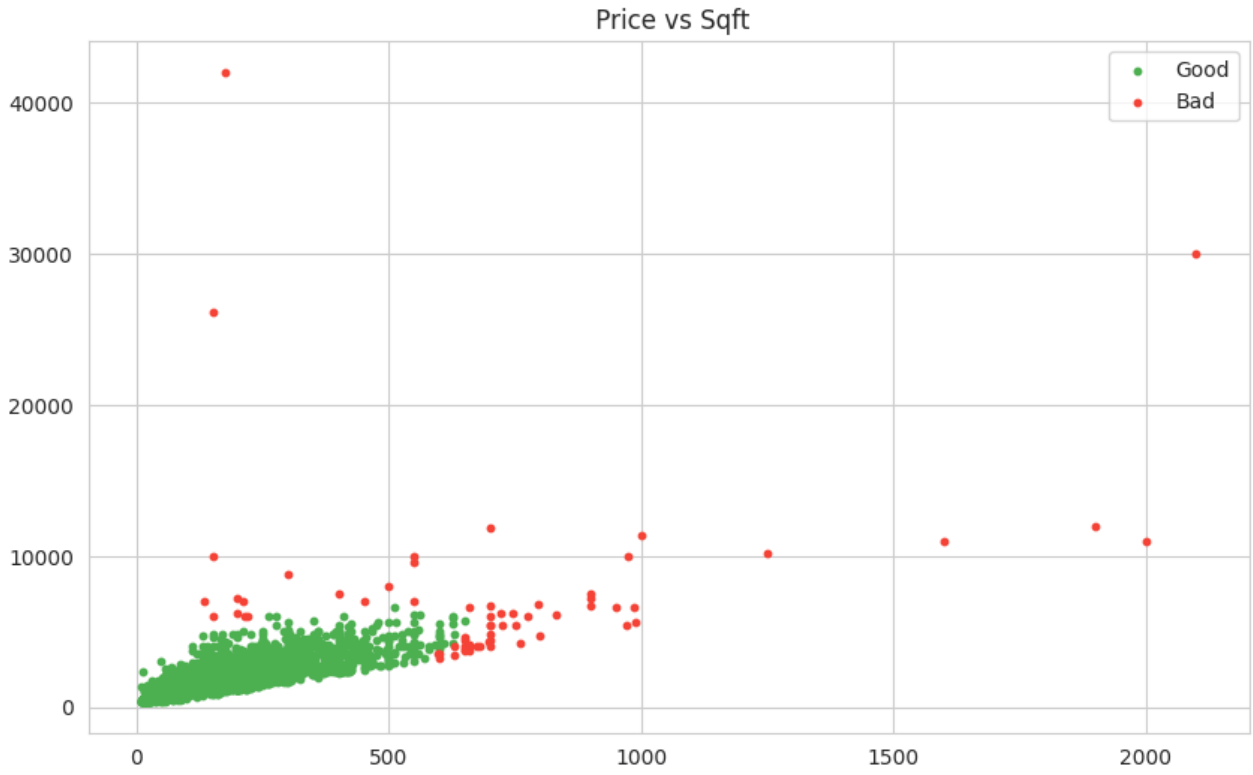


Figure 5. 3 Z-score

5.5 Data Flow & Control Flow Sequence

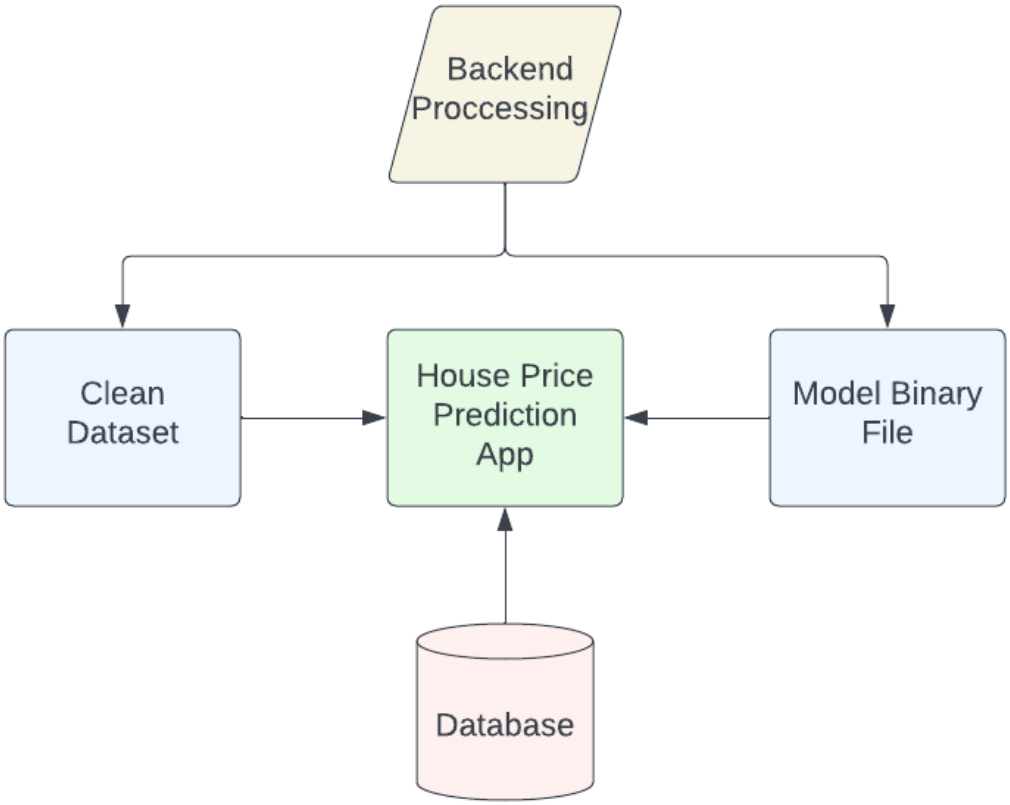


Figure 5. 4 Architecture

The dataflow within a home prediction app, which seamlessly connects a clean dataset, a model binary file, and a database, is a fundamental process that underpins the app's functionality. It commences with the collection and integration of data from various sources, including property listings and market indicators, which are meticulously cleaned and transformed to ensure accuracy and consistency. This clean dataset forms the core of the app's predictive capabilities, containing structured property information such as location, size, amenities, and historical pricing data, constantly updated to reflect market changes.

A machine learning model, trained using the clean dataset, is then encapsulated as a binary file. This model file is integrated into the app, enabling real-time predictions. Simultaneously, the app is connected to a database that houses both the clean dataset and user data, facilitating efficient data retrieval and updates. User interactions, preferences, and queries are recorded in the database to enhance personalization and learning.

When a user engages with the app, providing property details or location preferences, the model binary file processes this input, returning a predicted property price. Meanwhile, the database is updated with the user's query and response, contributing to ongoing model refinement. The app then presents the predicted property price, along with relevant property listings and additional information, offering users personalized and accurate predictions while continually improving its predictive prowess through user feedback and interactions. This interconnected dataflow ensures the app remains dynamic, adaptive, and invaluable in assisting users with their property-related decisions.

5.6 Testing & Validation

Linear regression is done on a training set and then checked with a testing set. The data set is split into an 80-20 split, 80% of the data is used for training the model and 20% is used to check the accuracy of the data model trained. The model undergoes Mean Squared Error with regularization term to reduce overfitting and correct the error simultaneously. The model is further evaluated based on R-Square Score which yields the accuracy of **93.168%.**

CHAPTER 6

APP FEATURES

6.1 About the App

Front-end used - Kivy :

Kivy is an open-source Python framework tailored for developing user-friendly, multi-touch applications. It excels in creating cross-platform software, compatible with Windows, macOS, Linux, Android, and iOS. Kivy's hallmark features include touch-based interface support for gestures like swipes and taps, along with an array of graphical widgets for crafting polished user interfaces. Its simplicity and Python syntax make it accessible to developers of all levels, following a "write once, run anywhere" philosophy. Kivy empowers developers to build cross-platform apps with engaging touch-friendly interfaces, suitable for a wide range of projects, from mobile apps to desktop utilities.

Figure 6. 1 Kivy logo

About the App :

The House Prediction App for Bangalore represents a significant leap forward in the realm of real estate technology. Its multi-device compatibility, achieved through Kivy, ensures that users can access its wealth of features seamlessly whether they are on their laptop, smartphone, or tablet.

At the core of this app lies its ability to empower users with precise price predictions for properties they are interested in. This predictive capability is the app's hallmark feature, setting it apart from other real estate applications. By harnessing sophisticated data analytics and machine learning algorithms, it not only offers price estimates but also generates personalized analyses based on users' search history.

This personalized analysis is invaluable for users as it helps them understand market trends, potential investment opportunities, and the real value of a property, ultimately enabling well-informed decision-making. Whether you're a first-time homebuyer or a seasoned real estate investor, this app provides a competitive edge in navigating the dynamic and fast-paced Bangalore real estate market.

In conclusion, the House Prediction App for Bangalore is a game-changing tool that combines user-friendly accessibility, predictive pricing, and personalized analysis. Its adaptability across devices ensures that anyone can harness its power to make smarter real estate decisions in Bangalore, making it an indispensable asset in the hands of property seekers and investors.

6.2 Functions & Tools Used

6.2.1. Tools Used :

1. Kivy : A Python framework for developing multi-touch applications, used to create the graphical user interface (GUI) of the house prediction system app.

2. Pandas: A data manipulation and analysis library, utilized for handling and processing datasets related to house predictions within the app.

3. NumPy: A library for numerical computations, employed for various mathematical operations and data manipulation tasks in the app.

4. CSV: A file format for storing tabular data, used to import and export data in a structured manner within the house prediction system.

5. Matplotlib: A data visualization library, utilized to generate plots and charts for visualizing data and trends in the real estate market.

6. Pickle: A Python module for object serialization, to save and load models and data.

7. Logging: A Python module for recording and managing log messages, used to track and debug the app's behavior and errors.

8. MySQL Connector: A library for connecting to MySQL databases, employed for storing and retrieving data related to houses and predictions.

9. KivyMD: A set of Material Design components for Kivy, used to enhance the visual appearance and user experience of the app.

10. ScreenManager: A Kivy widget for managing multiple screens or views within the app, allowing users to navigate between different sections of the application.

11. TextInput: A Kivy widget for user input, used to collect information such as property details and user preferences for house predictions.

12. DropDown: A Kivy widget for creating dropdown menus, used for selecting various options and criteria in the app, like filtering properties.

13. Popup: A Kivy widget for displaying contextual messages or user interactions, used for notifications and alerts in the app.

14. FigureCanvasKivyAgg: A Kivy widget for embedding Matplotlib figures in the app's interface, enabling the display of interactive charts and graphs.

15. FloatLayout: A Kivy layout manager, employed to arrange and position widgets in a flexible manner within the app's screens.

16. Image: A Kivy widget for displaying images, used to showcase property images and visual elements in the app.
17. Button: A Kivy widget used for creating interactive buttons that trigger actions or events when pressed, like submitting user input or navigating between screens.

18. Label: A Kivy widget used to display text or descriptive labels on the app's interface, providing information or instructions to users.

19. TextInput: A Kivy widget that allows users to input text or data, often used for collecting user input, such as property details or search queries.

20. ScrollView: A Kivy widget used to create scrollable areas within the app, particularly useful for displaying large lists of properties or data.

21. GridLayout: A Kivy layout manager used to arrange widgets in a grid-like structure, ensuring consistent alignment and organization of elements.

22. Builder: A Kivy module used for creating dynamic interfaces from Kv language files, simplifying the construction of complex app layouts.

23. Factory: A Kivy module used to create and manage widget instances dynamically in Python code, enhancing the flexibility of widget creation.

24. RoundedRectangle and Rectangle: Kivy graphics primitives used for creating custom graphical elements and backgrounds in the app's interface.

These components play crucial roles in building the user interface and enabling user interaction within the house prediction system app, facilitating data input, navigation, and information display.

6.2.2 Functions used:

1. show_login_popup(instance): This function is used to display a login popup. It creates a popup with email and password input fields, allowing users to log in. If the login is successful, it navigates the user to the home screen.

2. login(popup, email, password): This function handles the login logic. It queries a MySQL database to check if the provided email and password match any user records. If the login is successful, it dismisses the login popup and navigates the user to the home screen.

3. show_signup_popup(instance): This function displays a signup popup, allowing users to create a new account. It includes name, email, and password input fields. It checks if the provided email already exists in the database before allowing the user to sign up.

4. signup(popup, name, email, password): This function handles user registration. It inserts a new user record into the MySQL database if the provided email is unique, then dismisses the signup popup and navigates the user to the home screen.

5. data_table(location, bhk, bath, balcony): This function reads data from a CSV file named 'Clean_data.csv' and filters it based on the provided location, number of bedrooms (bhk), number of bathrooms (bath), and number of balconies (balcony). It returns a list of filtered data records.

6. home_button(layout, search_layout, pred_layout, tab_layout, chart_layout)`: This function creates a button in the home screen layout. When clicked, it triggers the logout functionality, updates various widgets' text, and shows prediction, table, and chart data.

7. button_update(buttons, button_to_update, var): This function updates dropdown menus for filter buttons. It dynamically generates dropdowns based on the available filter options and updates the selected filter button's text.

8. search(layout, pred_layout, tab_layout, chart_layout): This function handles the search functionality. It allows users to filter house data based on location, number of bedrooms (bhk), number of bathrooms (bath), and number of balconies (balcony).

9. prediction(layout): This function displays a prediction based on user-selected filters. It calculates a house price prediction using a machine learning model and displays the result.

10. table(layout): This function displays a table of filtered house data based on user-selected filters.

11. chart(layout): This function displays a chart or provides an option to display a chart. It plots a histogram or a percentile price distribution chart based on user-selected filters.

12. HousePredictionApp: This class defines the Kivy application. It includes the `build` method to create the app's screens and logic. The `start_screen` method creates the start screen with login and signup buttons. The `home_screen` method creates the home screen with various widgets for search, prediction, table, and chart functionalities.

Overall, this code is for a house price prediction and analysis app that allows users to log in, search for houses based on various criteria, view predictions, and analyze housing data through tables and charts.

6.2.3 Special Functions:

1. **Bind:**

   The `bind` function in Kivy is used to establish a binding between an event and a function or callback. This binding allows you to specify what action or series of actions should be performed when a particular event occurs within a Kivy application. Essentially, it connects a widget event to a Python function so that the function is automatically triggered when the event happens. This mechanism is fundamental for creating interactive and responsive user interfaces in Kivy, as it enables you to define how the application responds to various user interactions and system events.

2. **Button update:**

   The button_update function is a Python function that is used to create and manage a dropdown menu with filter functionality for buttons within a graphical user interface (GUI). It is designed with confidence to perform the following tasks:

a) **Button Click Handling**: The `button_clicked` function is defined within `button_update`. It handles the action when a button in the dropdown menu is clicked. It updates a global list called `alist` at a specific index (`var`) with the text of the clicked button, dismisses the dropdown menu, and updates the text of `button_to_update` with the text of the clicked button.

b) **Filtering Buttons:** The `apply_filter` function is responsible for filtering the buttons based on user input. It takes a `value` (user input) and a `dropdown` as parameters, clears the existing widgets in the dropdown, and adds a filter input (`filter1`) at the top of the dropdown. It then iterates through a list of `buttons` and adds buttons to the dropdown based on whether they match the filter criteria (if `value` is empty or found in the button text).

c) **Dropdown Creation:** The `filterDD` function is responsible for creating the dropdown menu. It initializes a filter input (`filter1`) and a dropdown widget (`dropdown`). It binds the filter input to the `apply_filter` function, ensuring that filtering occurs as the user types. It also initially applies the filter with an empty filter value. This function returns the created dropdown.

d) **Function Execution:** Finally, the `button_update` function is called with parameters `buttons` (a list of button texts), `button_to_update` (the button that needs to be updated when a selection is made), and `var` (an index for updating a global list). It returns the created dropdown, which can then be added to the GUI.

`**button_update**` is a function that encapsulates the creation and management of a dynamic dropdown menu for buttons in a GUI, allowing users to filter and select from a list of options, with the selected option updating a specified button and modifying a global list.

6.3 Working & Screenshots

1. **Home Screen** - The home screen welcomes users with a sleek design, offering both a secure login and sign-up feature that seamlessly connects to a robust user database, ensuring a personalized and accessible experience.

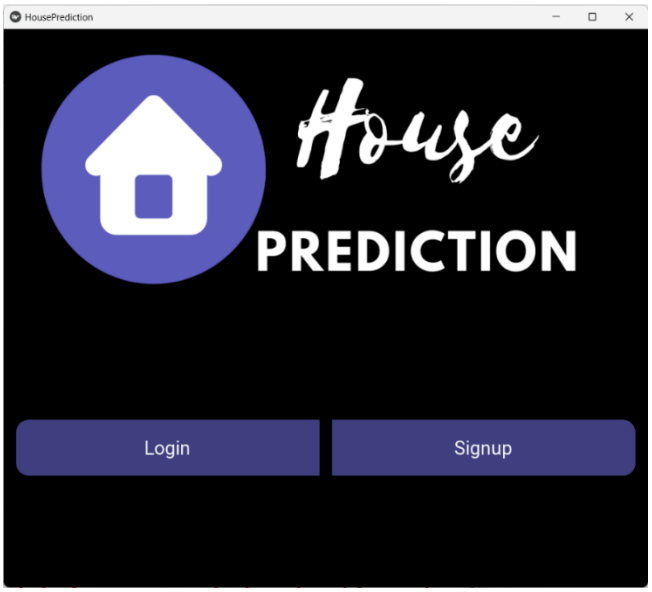


Figure 6. 3 Login page

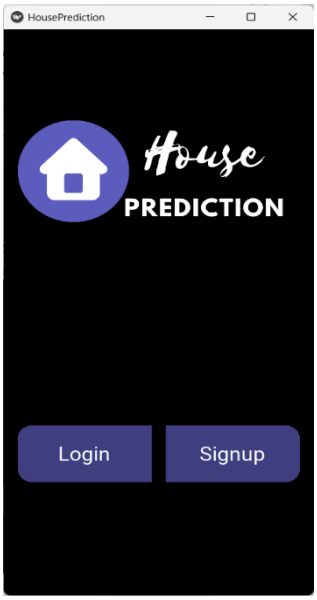


Figure 6. 2 Login Phone

2. **About us** - This page is the heart of our house prediction app, where it shares passion for real estate analytics and cutting-edge technology. It is dedicated to empowering users with accurate predictions, helping them make informed decisions on their homeownership journey.

Figure 6. 4 About us

3. **Search Layout -** The search layout features an intuitive drop-down menu that allows users to explore and search through a wide range of options effortlessly, making finding what they need a breeze.

Figure 6. 5 Search layout

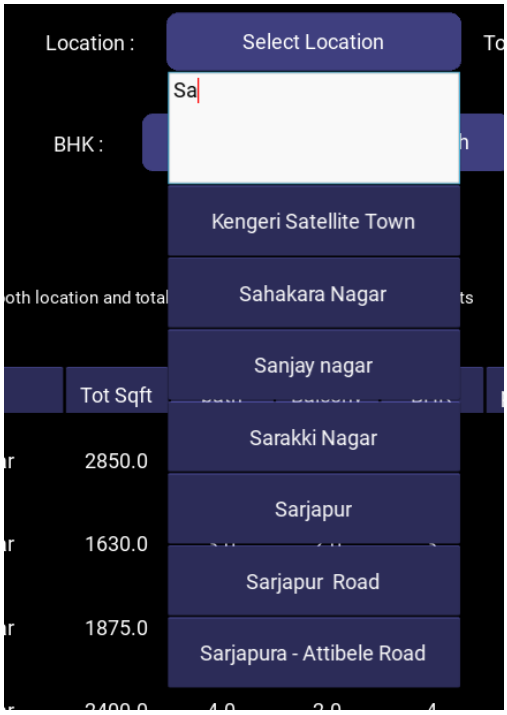


Figure 6. 6 Search Dropdown

4. **Prediction Layout** - Within the Prediction Layout, the app leverages location, number of bedrooms (BHK), balconies, bathrooms, and total square footage to generate precise price predictions, empowering users with valuable insights for their real estate endeavors.

Figure 6. 7 Prediction

5. **Personalized Analysis -** Our Personalized Graphical Analysis feature takes the data users have provided, such as location, BHK, balcony count, bathrooms, and total square footage, and transforms it into visually engaging graphs and charts. This enables users to gain a deeper understanding of the real estate market trends and empowers them to make well-informed decisions with confidence.

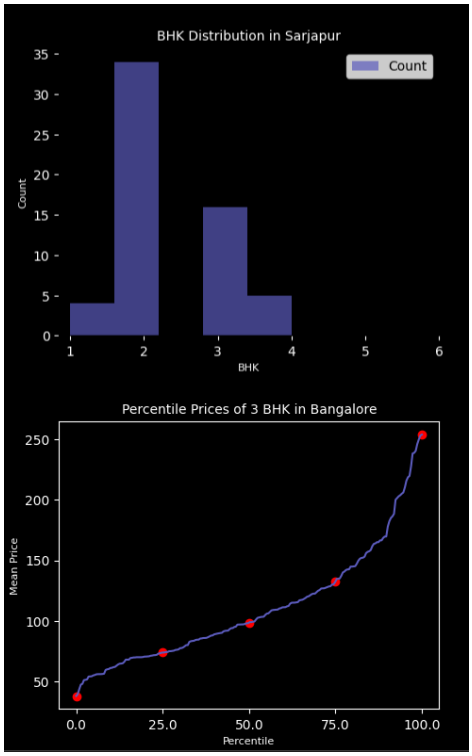


Figure 6. 8 Personalized Analysis

6. **Table of all properties -** This feature presents a comprehensive listing of available properties, offering essential details like location, number of bedrooms (BHK), balcony count, bathrooms, and total square footage in an organized and easily accessible format, streamlining users property search process.

Location	Tot Sqft	bath	Balcony	BHK	per Sqft
1st Block Jayanagar	2850.0	4.0	1.0	4	428.0
1st Block Jayanagar	1630.0	3.0	2.0	3	194.0
1st Block Jayanagar	1875.0	2.0	3.0	3	235.0
1st Block Jayanagar	2400.0	4.0	2.0	4	450.0
1st Block Jayanagar	1000.0	3.0	2.0	2	60.0
1st Block Jayanagar	1200.0	2.0	0.0	3	130.0
1st Block Jayanagar	1235.0	2.0	2.0	2	148.0

Figure 6. 9 Table

CHAPTER 7
RESULTS & DISCUSSION

Table 7.1 Comparison of Values

Location	BHK	Bathroom	Balcony	Total Square feet	Prediction (in lacs)	True Value (in lacs)
Sarjapur Road	3	2	2	1950	135	141
7th Phase JP Nagar	3	2	1	1635	114.17	109
Vijayanagar	2	2	1	1200	68.3	70
Prashant Nagar, T.Dasarahalli	2	2	0	1080	65.12	60

In the following table, we've compiled a comparison of various property attributes across four different locations: Sarjapur Road, 7th Phase JP Nagar, Vijayanagar, and Prashant Nagar, T.Dasarahalli. These attributes include the number of bedrooms (BHK), bathrooms, balconies, and the total square footage of the properties. We've also included the actual property values and predictions generated by a machine learning model. It's worth noting that the model has demonstrated an impressive accuracy rate of 93% in estimating property values, which underscores its effectiveness in predicting real estate prices. The discussion that follows will delve into the insights gained from this comparison, shedding light on the factors that impact property values in these areas and the reliability of our prediction model.

CHAPTER 8

CONCLUSION & FURTHER ENHANCEMENTS

In conclusion, our house prediction app has demonstrated its effectiveness in accurately estimating house prices, providing users with a valuable resource in their real estate endeavors. Moreover, the app's personalized graphical analysis feature offers a unique and interactive experience, allowing users to visualize and interpret the data in a way that is tailored to their specific preferences and needs. As we look toward the future, we are committed to enhancing this app further. We plan to integrate additional data sources to refine our prediction models, ensuring even greater accuracy. We also aim to expand our coverage to include a wider range of geographic areas, making the app applicable to a broader audience. Moreover, user feedback will continue to be instrumental in shaping the app's evolution, helping us fine-tune the user interface and analytical tools to provide an even more intuitive and informative experience. Our ultimate vision is to empower users with the most advanced and user-friendly tool available for navigating the complex world of real estate, combining state-of-the-art machine learning with a user-centric approach to deliver unparalleled insights and predictions.

CHAPTER 9

REFERENCES

[1] Model “BANGALORE HOUSE PRICE PREDICTION MODEL” - https://www.kaggle.com/code/ameythakur20/bangalore-house-price-prediction-model

[2] Pickle ‘’Documentation’’

[3] M. Jain, H. Rajput, N. Garg and P. Chawla, "Prediction of House Pricing using Machine Learning with Python," 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2020, pp. 570-574, doi: 10.1109/ICESC48915.2020.9155839.

[4] Quang Truong, Minh Nguyen, Hy Dang, Bo Mei, Housing Price Prediction via Improved Machine Learning Techniques, Procedia Computer Science, Volume 174, 2020.

[5] O.BHARGAV SATHWIK,T.HEMANTH,M.VISHNU VARDHANN YADAV, "HOUSE PRICE PREDICTION USING MACHINE LEARNING", International Journal of Creative Research Thoughts (IJCRT), ISSN:2320-2882, Volume.9, Issue 11, pp.b168-b174, November 2021

[6] https://www.irjet.net/archives/V8/i9/IRJET-V8I934

[7] https://kivy.org/

[8] https://www.reddit.com/r/kivy/comments/8ktfkn/how_can_i_search_inside_a_dropdown/

[9] https://www.youtube.com/watch?v=dLgquj0c5_U&list=PLCC34OHNcOtpz7PJQ7Tv7hqFBP_xDDjqg&index=1

APPENDIX A

RELATED MATHEMATICAL CONCEPTS

1. Z-Score
2. Mean Squared Error
3. Regularization
4. R Squared
5. .Inter-Quartile Range
6. Mean and Standard Deviation

APPENDIX B

POSITIONED PAPERS

[1] A. Varma, A. Sarma, S. Doshi and R. Nair, "House Price Prediction Using Machine Learning and Neural Networks," 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), 2018, pp. 1936-1939, doi: 10.1109/ICICCT.2018.8473231.

[2] Furia, Palak, and Anand Khandare. "Real Estate Price Prediction Using Machine Learning Algorithm." eConference on Data Science and Intelligent Computing. 2020.

[3] M. Jain, H. Rajput, N. Garg and P. Chawla, "Prediction of House Pricing using Machine Learning with Python," 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, 2020, pp. 570-574, doi: 10.1109/ICESC48915.2020.9155839.

[4] Quang Truong, Minh Nguyen, Hy Dang, Bo Mei, Housing Price Prediction via Improved Machine Learning Techniques, Procedia Computer Science, Volume 174, 2020.

[5] O.BHARGAV SATHWIK,T.HEMANTH,M.VISHNU VARDHANN YADAV, "HOUSE PRICE PREDICTION USING MACHINE LEARNING", International Journal of Creative Research Thoughts (IJCRT), ISSN:2320-2882, Volume.9, Issue 11, pp.b168-b174, November 2021

APPENDIX C

PRESENTATION HANDOUTS