INTERNSHIP REPORT

Submitted By

Bhargav Bhatt

(*226090307004*)

To

COMPUTER ENGINEERING DEPARTMENT

C U SHAH (GOVT.) POLYTECHNIC – SURENDRANAGAR

GUJARAT TECHNOLOGICAL UNIVERSITY – AHMEDABAD

JUNE-AUGUST – 2024

INDEX

PROJECT PROFILE

- Project Title :- House Price Pridiction

- Goal of Project :- Provide robust analytical framework that utilizes machine learning techniques to understand and predict residential property prices. By examining a comprehensive dataset containing property characteristics

- Project Guide : Mr. S. S. Parmar Sir

- Platform
  - ✓ Front End : StreamLit (python)
  - ✓ Backend : Python
  - ✓ Data Dictionary : CSV file
  - ✓ Documentation Tool : Microsoft Office Word

PROJECT INTRODUCTION

The project focuses on analyzing residential property prices to gain insights into the factors influencing the housing market. By examining various variables such as property characteristics, location, market trends, and economic indicators, this project aims to provide a comprehensive understanding of the factors driving home prices and identify key trends and patterns. The project utilizes a dataset containing information on residential properties, including features such as square footage, number of bedrooms and bathrooms, location attributes (e.g., neighborhood, proximity to amenities), and corresponding sale prices. By analyzing these variables, the project seeks to uncover correlations and trends that can help stakeholders understand the dynamics of the housing market and make informed decisions regarding real estate investments, pricing strategies, and policy development. Through exploratory data analysis and statistical modeling techniques, the project aims to identify the key factors influencing home prices. This includes assessing the impact of property characteristics (e.g., size, amenities), location factors (e.g., proximity to schools, transportation), and market trends (e.g., supply and demand, interest rates) on property values. Additionally, the project explores the relationships between economic indicators (e.g., GDP growth, employment rates) and home prices to understand the broader macroeconomic influences on the housing market. The outcomes of this project have implications for real estate professionals, investors, policymakers, and prospective homeowners. Real estate professionals can utilize the findings to better understand market trends, advise clients, and develop effective pricing strategies. Investors can gain insights into factors that drive property value appreciation and identify lucrative investment opportunities. Policymakers can use the findings to inform housing policies, zoning regulations, and urban planning initiatives. Prospective homeowners can make more informed decisions regarding property purchases, taking into account factors that influence home prices and long-term value appreciation

PROJECT ADVANTAGES AND LIMITATION'S

- **Advantages** :-
  1. Informed Decision-Making : Provides real estate professionals, investors, and policymakers with actionable insights into factors that drive property values, aiding in more informed decision-making.
  2. Market Understanding : Enhances understanding of market dynamics by analyzing trends and correlations between property characteristics, location, and economic indicators.
  3. Investment Opportunities : Helps investors identify lucrative opportunities by highlighting which factors most significantly impact property appreciation and market trends
  4. Pricing Strategies : Offers real estate professionals valuable insights for setting competitive and accurate pricing strategies based on market data.
  5. Comprehensive Analysis : Utilizes a broad range of variables, including property features, location attributes, and economic indicators, for a well-rounded analysis of housing market factors.
- **Limitation's** :-
  1. Data Quality and Availability : The accuracy of insights depends on the quality and completeness of the dataset. Incomplete or inaccurate data can lead to misleading conclusions.
  2. Dynamic Market Conditions : Housing markets are influenced by many external factors (e.g., economic downturns, natural disasters) that may not be fully captured in the dataset or analysis.
  3. Model Limitations : Predictive models may not capture all nuances of the housing market and could be influenced by unobserved variables or changes in market conditions.
  4. External Factors : Factors such as changes in government policy, shifts in societal preferences, or unforeseen events (e.g., pandemics) can impact the housing market in ways not accounted for in the analysis.

PROJECT APPLICATIONS

- Real Estate Investment:
  - **Investment Decisions:** Investors can use insights from the analysis to identify areas with high growth potential and make informed investment decisions.
  - **Risk Assessment:** Analyze factors that could pose risks to property value, helping investors mitigate potential losses .

- Real Estate Marketing and Sales :
  - **Targeted Marketing:** Real estate agents can tailor marketing strategies based on the factors most appealing to buyers in different segments .
  - **Pricing Strategies:** Agents can use the analysis to set competitive and accurate prices, enhancing sales effectiveness .

- Homebuyer Guidance :
  - **Buying Decisions:** Prospective homeowners can understand factors that influence property values, aiding in selecting homes with better long-term value .

- Financial Institutions:
  - **Mortgage Lending:** Banks and financial institutions can use the analysis to assess the risk and potential return on mortgage loans.
  - **Property Valuation:** Accurate property valuations based on analysis can enhance loan underwriting processes .

- Real Estate Technology Solutions:
  - **Property Platforms:** Online real estate platforms can integrate analysis to provide users with detailed insights into property values and market conditions.
  - **Automated Valuation Models (AVMs):** Enhance AVMs with data-driven insights for more accurate property valuations.

TECHNICAL DETAILS

- **Frontend Technologies :-**

1. **Streamlit :**
    - Streamlit is employed to build the user interface, offering a seamless and interactive experience for users. It facilitates the integration of data visualization and user input directly from Python scripts, making it ideal for developing interactive data applications.

2. **Visualization Libraries (Matplotlib, Seaborn):**
    - These libraries are essential for creating visual representations of the data, allowing users to easily interpret complex datasets through a variety of charts and graphs. They enhance the project's ability to present insights effectively.

- **Backend Technologies :-**

1. **Python:**
    - Python serves as the core programming language for the project, providing a versatile platform for data processing, analysis, and model development. Its extensive ecosystem of libraries supports efficient data manipulation and machine learning.

2. **Scikit-learn:**
    - Scikit-learn is used for building and evaluating machine learning models that predict property prices based on various features. It offers a range of algorithms and tools to support the development of robust predictive models.

3. **Data Handling Libraries (Pandas, NumPy):**
    - Pandas and NumPy are used for managing and processing data efficiently. They enable complex data manipulations and numerical computations, which are crucial for analyzing large datasets.

SYSTEM REQUIREMENTS

- **Hardware Requirements :-**

  - **Processor:** Intel Core i5 or equivalent
    - A mid-range processor is recommended to handle data processing and model computations efficiently.

  - **RAM:** 8 GB or more
    - Sufficient RAM is needed to handle large datasets and ensure smooth operation during data analysis and model training

  - **Storage:** 256 GB SSD or larger
    - An SSD provides fast read and write speeds, which are beneficial for data loading and processing tasks.

- **Software Requirement:-**

  - **Operating System:** Windows 10, macOS, or Linux
    - The application is cross-platform and can be run on any major operating system that supports Python.

  - **Python:** Version 3.7 or later
    - The project is developed in Python, so a compatible Python version is required.

  - **Web Browser:**
    - A modern web browser such as Google Chrome, Mozilla Firefox, or Microsoft Edge is required to access the Streamlit application interface.

  - **Jupyter Notebook:**
    - Required for data visualization and analysis, enabling users to explore data, visualize results, and run models interactively.

FUNCTIONAL REQUIREMENTS

- **Data Input and Processing:**
  - The system shall allow users to upload datasets containing residential property information for analysis.
  - The system shall clean and preprocess the input data, handling missing values and outliers.

- **Data Analysis and Modeling:**
  - The system shall perform exploratory data analysis (EDA) to identify trends and patterns in the data.
  - The system shall allow users to select features for building machine learning models.
  - The system shall train and evaluate predictive models to estimate property prices.

- **Data Visualization:**
  - The system shall generate visualizations such as scatter plots, histograms, and heatmaps to represent data relationships.
  - The system shall update visualizations dynamically based on user input and data changes.

- **User Interface:**
  - The system shall provide an interactive web-based interface for users to interact with the data and models.
  - The system shall include components such as sliders, buttons, and input fields to capture user input.

NON-FUNCTIONAL REQUIREMENTS

- Performance:

  - The system shall process and analyze datasets within a reasonable time frame to provide timely insights.
  - The system shall support concurrent users without significant degradation in performance.

- Scalability:

  - The system shall be capable of handling increasing amounts of data and users without significant performance loss.

- Reliability:

  - The system shall ensure data integrity and accuracy throughout the analysis process.
  - The system shall handle errors and exceptions gracefully, providing meaningful error messages to users.

- Portability:

  - The system shall be compatible with major operating systems (Windows, macOS, Linux) and web browsers.

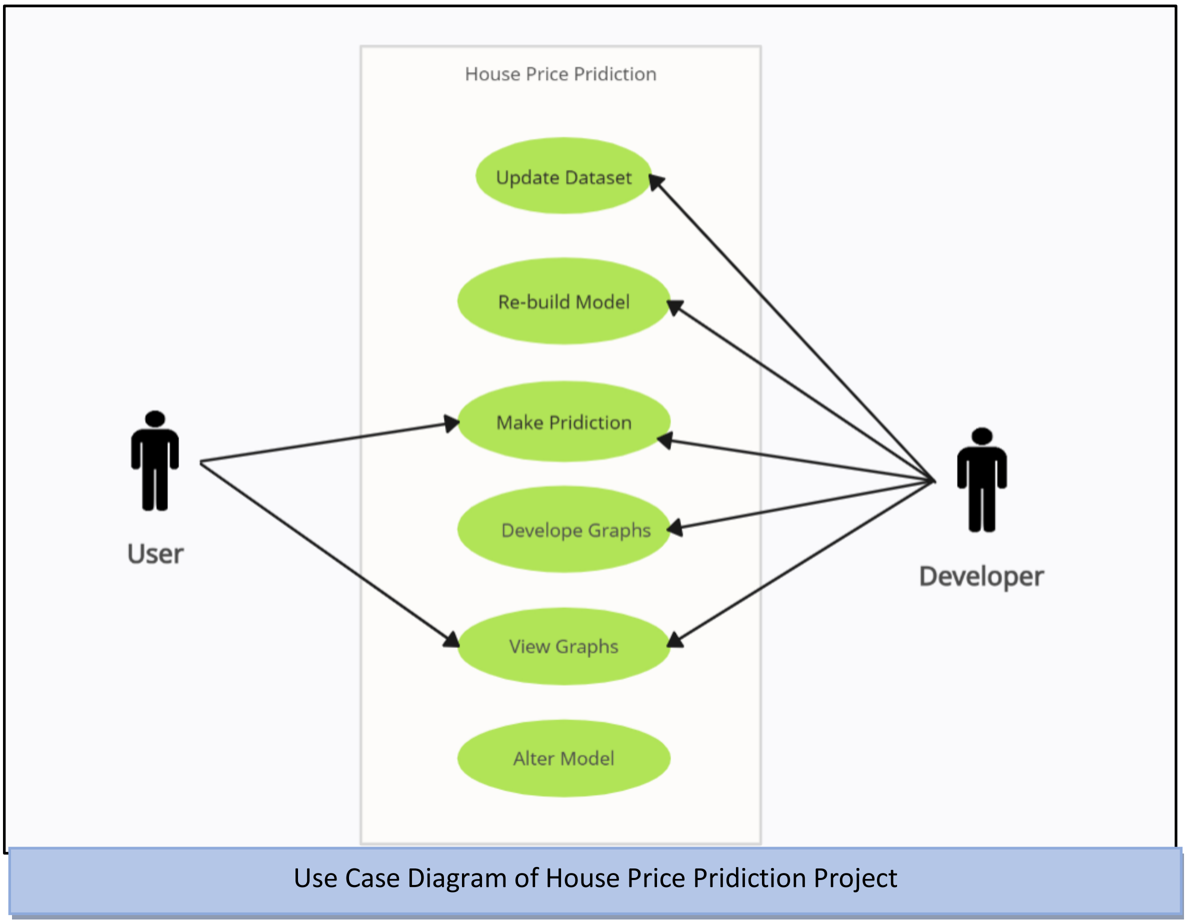
USERS OF SYSTEM

- Real Estate Professionals
    - Real estate agents, brokers, and appraisers who use the system to analyze market trends and set pricing strategies.
- Investors
    - Individuals or organizations interested in using data-driven insights to make informed investment decisions.
- Homebuyers
    - Prospective homebuyers seeking to understand market dynamics and identify favorable properties.
- Data Analysts/Researchers
    - Professionals and researchers conducting market studies or academic research on housing trends.
- System Administrators
    - Personnel responsible for maintaining and managing the application, ensuring its smooth operation.

USE CASE DIAGRAM

- **User** use the UI to input data into the model. For analysis and visualization, they navigate through Jupyter Notebook, where they can explore the data, run models, and visualize results.

- **Developer** manage access to both the Streamlit UI and Jupyter Notebook, ensuring that users can perform their tasks without technical hindrances. They are also responsible for maintaining the system's reliability and security.



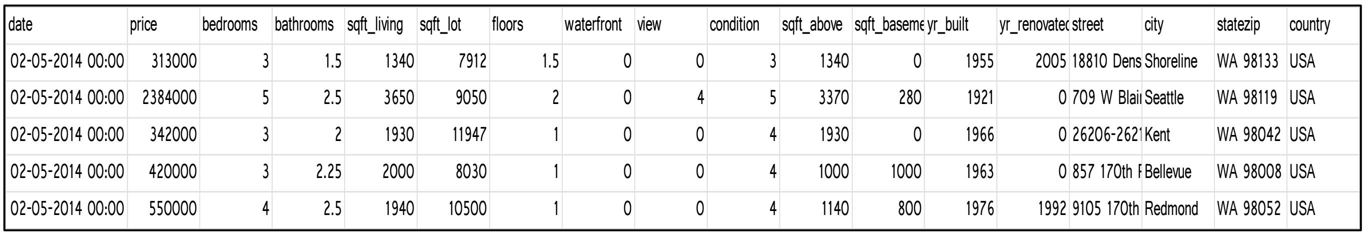
Use Case Diagram of House Price Pridiction Project

DATA DICTIONARY

- The data dictionary provides a detailed description of each column in the CSV dataset used for analyzing residential property prices. This dataset includes various attributes of residential properties that are used to explore trends and patterns in the housing market.

Column Name	Data Type	Description
date	Date	The date associated with the property listing or sale.
price	Float	The sale price of the property.
bedrooms	Integer	The number of bedrooms in the property.
bathrooms	Float	The number of bathrooms in the property, including partial bathrooms.
sqft_living	Integer	The total living area of the property in square feet.
sqft_lot	Integer	The total area of the lot or land on which the property is situated, measured in square feet.
floors	Float	The number of floors in the property.
waterfront	Integer	A binary indicator (0 or 1) representing whether the property has a waterfront view.
view	Integer	A rating of the property's view, with higher values indicating a better view.
condition	Integer	A rating of the overall condition of the property, with higher values indicating better condition.
sqft_above	Integer	The square footage of the property that is above ground level.
sqft_basement	Integer	The square footage of the property's basement, if applicable.
yr_built	Integer	The year the property was originally built.
yr_renovated	Integer	The year the property was last renovated, if applicable.
street	String	The street address or location of the property.
city	String	The city where the property is located.
statezip	String	The state and ZIP code of the property.
country	String	The country where the property is located.

- **Sample Data :-**

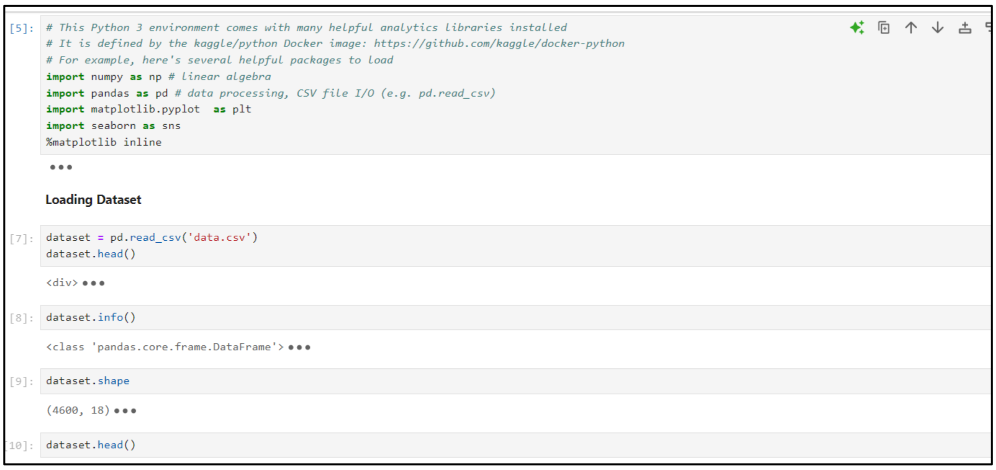


date	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	view	condition	sqft_above	sqft_baseme	yr_built	yr_renovatec	street	city	statezip	country
02-05-2014 00:00	313000	3	1.5	1340	7912	1.5	0	0	3	1340	0	1955	2005	18810 Dens	Shoreline	WA 98133	USA
02-05-2014 00:00	2384000	5	2.5	3650	9050	2	0	4	5	3370	280	1921	0	709 W Blaii	Seattle	WA 98119	USA
02-05-2014 00:00	342000	3	2	1930	11947	1	0	0	4	1930	0	1966	0	26206-2621	Kent	WA 98042	USA
02-05-2014 00:00	420000	3	2.25	2000	8030	1	0	0	4	1000	1000	1963	0	857 170th F	Bellevue	WA 98008	USA
02-05-2014 00:00	550000	4	2.5	1940	10500	1	0	0	4	1140	800	1976	1992	9105 170th	Redmond	WA 98052	USA

- **What is CSV?**
  - CSV stands for **Comma-Separated Values**. It's a simple file format used to store tabular data, such as a spreadsheet or database, in plain text. Each line in a CSV file corresponds to a row in the table, and each value in the row is separated by a comma . here the above data is of csv file

SAMPLE CODING

- Loading Dataset :-

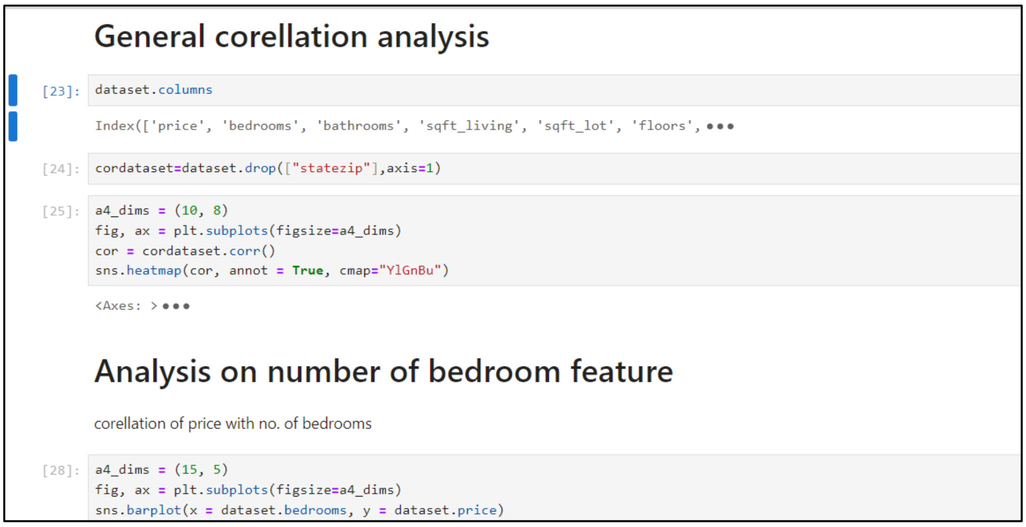


```
[5]: # This Python 3 environment comes with many helpful analytics libraries installed
     # It is defined by the kaggle/python Docker image: https://github.com/kaggle/docker-python
     # For example, here's several helpful packages to load
     import numpy as np # linear algebra
     import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
     import matplotlib.pyplot  as plt
     import seaborn as sns
     %matplotlib inline
     •••
```

Loading Dataset

```
[7]: dataset = pd.read_csv('data.csv')
     dataset.head()

     <div> •••

[8]: dataset.info()

     <class 'pandas.core.frame.DataFrame'> •••

[9]: dataset.shape

     (4600, 18) •••

10]: dataset.head()
```

- General Correlation Analysis



General corellation analysis

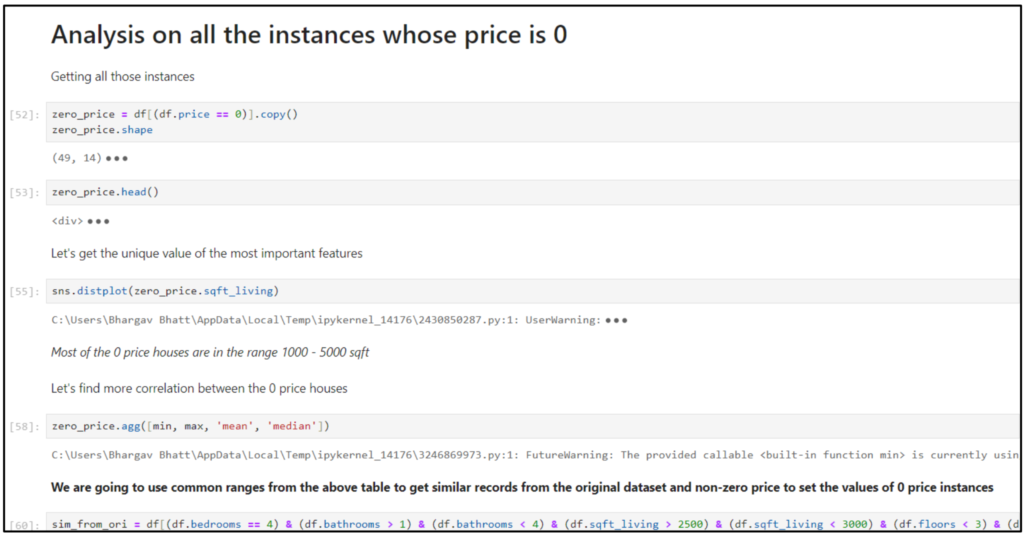
```
[23]: dataset.columns

      Index(['price', 'bedrooms', 'bathrooms', 'sqft_living', 'sqft_lot', 'floors', •••

[24]: cordataset=dataset.drop(["statezip"],axis=1)

[25]: a4_dims = (10, 8)
      fig, ax = plt.subplots(figsize=a4_dims)
      cor = cordataset.corr()
      sns.heatmap(cor, annot = True, cmap="YlGnBu")

      <Axes: > •••
```

Analysis on number of bedroom feature

corellation of price with no. of bedrooms

```
[28]: a4_dims = (15, 5)
      fig, ax = plt.subplots(figsize=a4_dims)
      sns.barplot(x = dataset.bedrooms, y = dataset.price)
```

- Analysis on Zero Price data



Analysis on all the instances whose price is 0

Getting all those instances

```
[52]: zero_price = df[(df.price == 0)].copy()
      zero_price.shape

      (49, 14) •••

[53]: zero_price.head()

      <div> •••
```

Let's get the unique value of the most important features

```
[55]: sns.distplot(zero_price.sqft_living)

      C:\Users\Bhargav Bhatt\AppData\Local\Temp\ipykernel_14176\2430850287.py:1: UserWarning: •••
```

Most of the 0 price houses are in the range 1000 - 5000 sqft

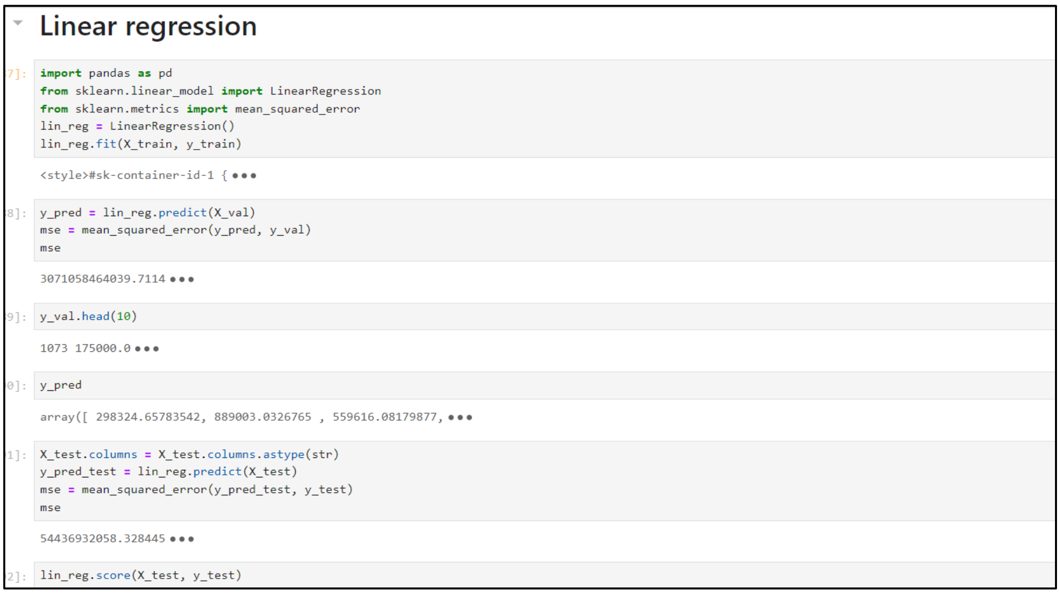
Let's find more correlation between the 0 price houses

```
[58]: zero_price.agg([min, max, 'mean', 'median'])

      C:\Users\Bhargav Bhatt\AppData\Local\Temp\ipykernel_14176\3246869973.py:1: FutureWarning: The provided callable <built-in function min> is currently usin
```

We are going to use common ranges from the above table to get similar records from the original dataset and non-zero price to set the values of 0 price instances

```
[60]: sim_from_ori = df[(df.bedrooms == 4) & (df.bathrooms > 1) & (df.bathrooms < 4) & (df.sqft_living > 2500) & (df.sqft_living < 3000) & (df.floors < 3) & (d
```

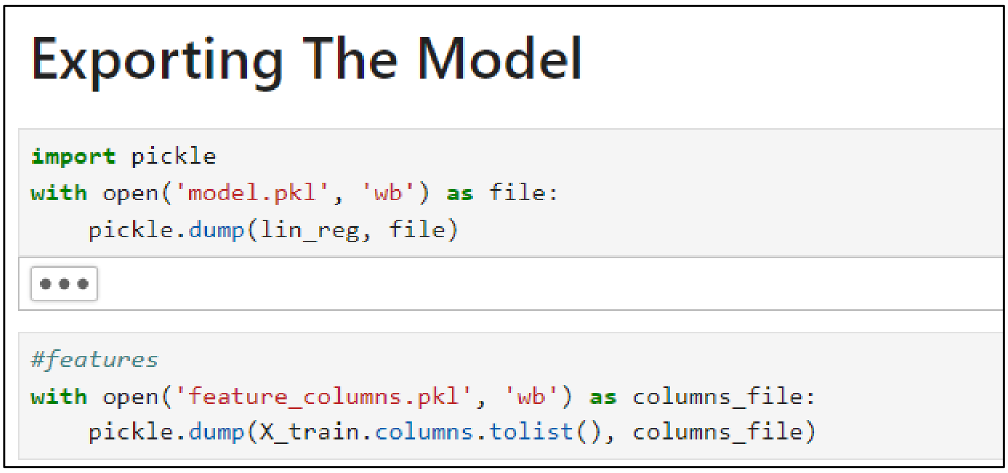
- ➢ Building The Model :- Linear Regression



Linear regression

```
7]: import pandas as pd
    from sklearn.linear_model import LinearRegression
    from sklearn.metrics import mean_squared_error
    lin_reg = LinearRegression()
    lin_reg.fit(X_train, y_train)

    <style>#sk-container-id-1 { •••

8]: y_pred = lin_reg.predict(X_val)
    mse = mean_squared_error(y_pred, y_val)
    mse

    3071058464039.7114 •••

9]: y_val.head(10)

    1073 175000.0 •••

0]: y_pred

    array([ 298324.65783542, 889003.0326765 , 559616.08179877, •••

1]: X_test.columns = X_test.columns.astype(str)
    y_pred_test = lin_reg.predict(X_test)
    mse = mean_squared_error(y_pred_test, y_test)
    mse

    54436932058.328445 •••

2]: lin_reg.score(X_test, y_test)
```

- ➢ Exporting the Model :-



Exporting The Model

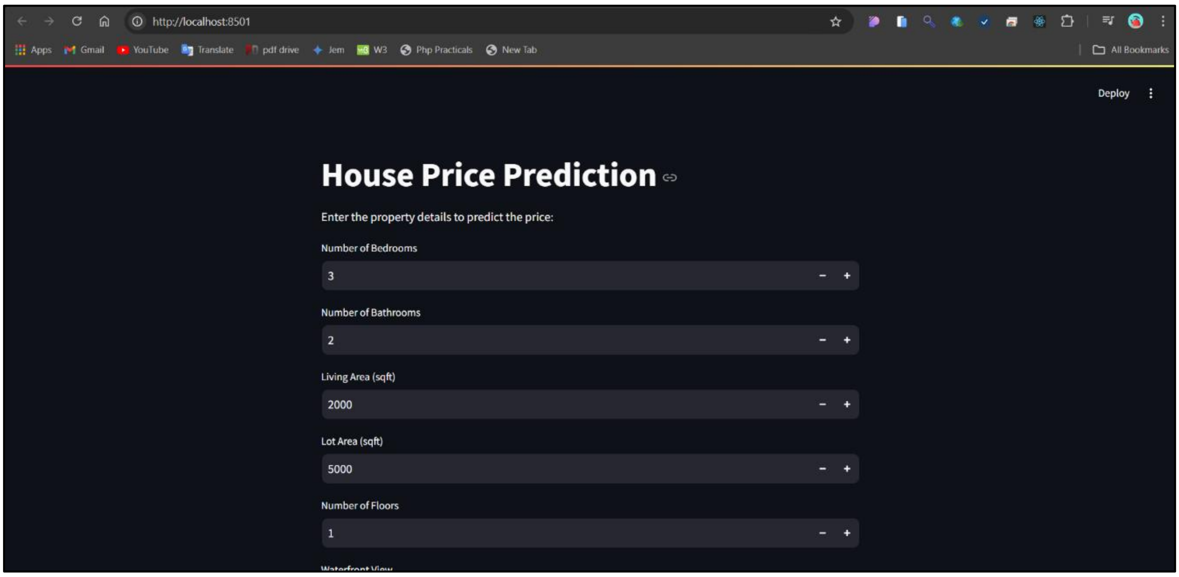
```
import pickle
with open('model.pkl', 'wb') as file:
    pickle.dump(lin_reg, file)
```

```
•••
```

```
#features
with open('feature_columns.pkl', 'wb') as columns_file:
    pickle.dump(X_train.columns.tolist(), columns_file)
```

OUTPUTS

➢ Feature Entry :-



House Price Prediction

Enter the property details to predict the price:

Number of Bedrooms: 3

Number of Bathrooms: 2

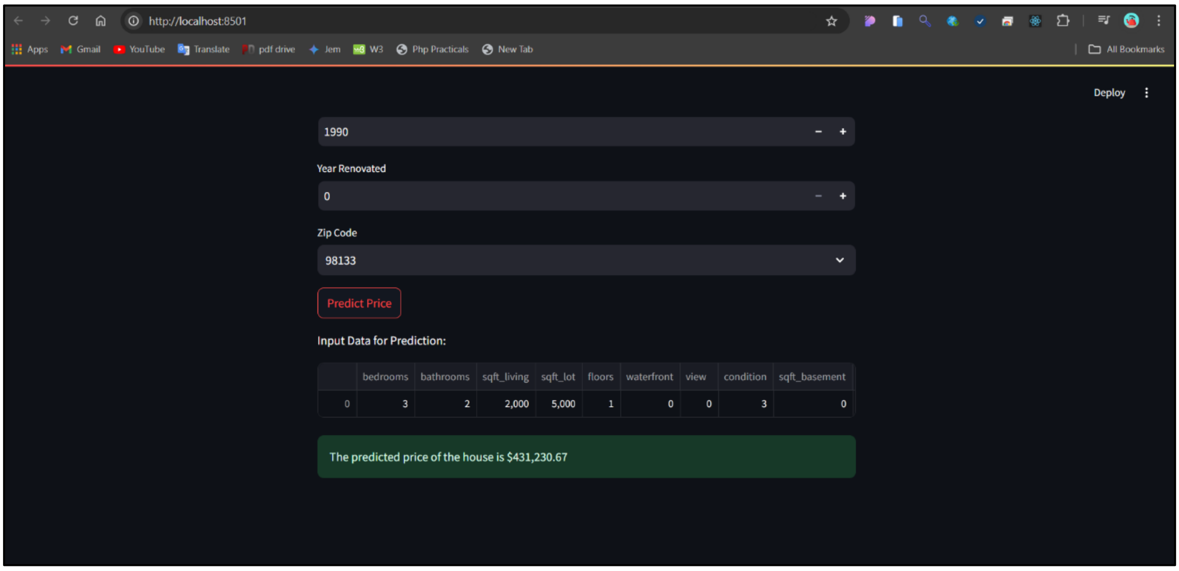
Living Area (sqft): 2000

Lot Area (sqft): 5000

Number of Floors: 1

Waterfront View

➢ Pridiction :-



1990

Year Renovated: 0

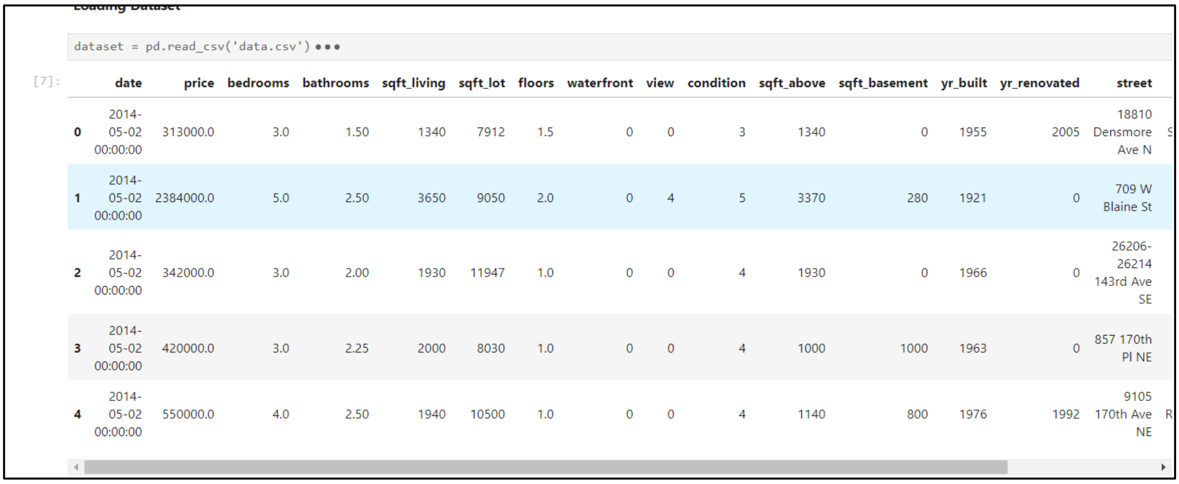
Zip Code: 98133

Input Data for Prediction:

	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	view	condition	sqft_basement
0	3	2	2,000	5,000	1	0	0	3	0

The predicted price of the house is $431,230.67

➢ Dataset :-



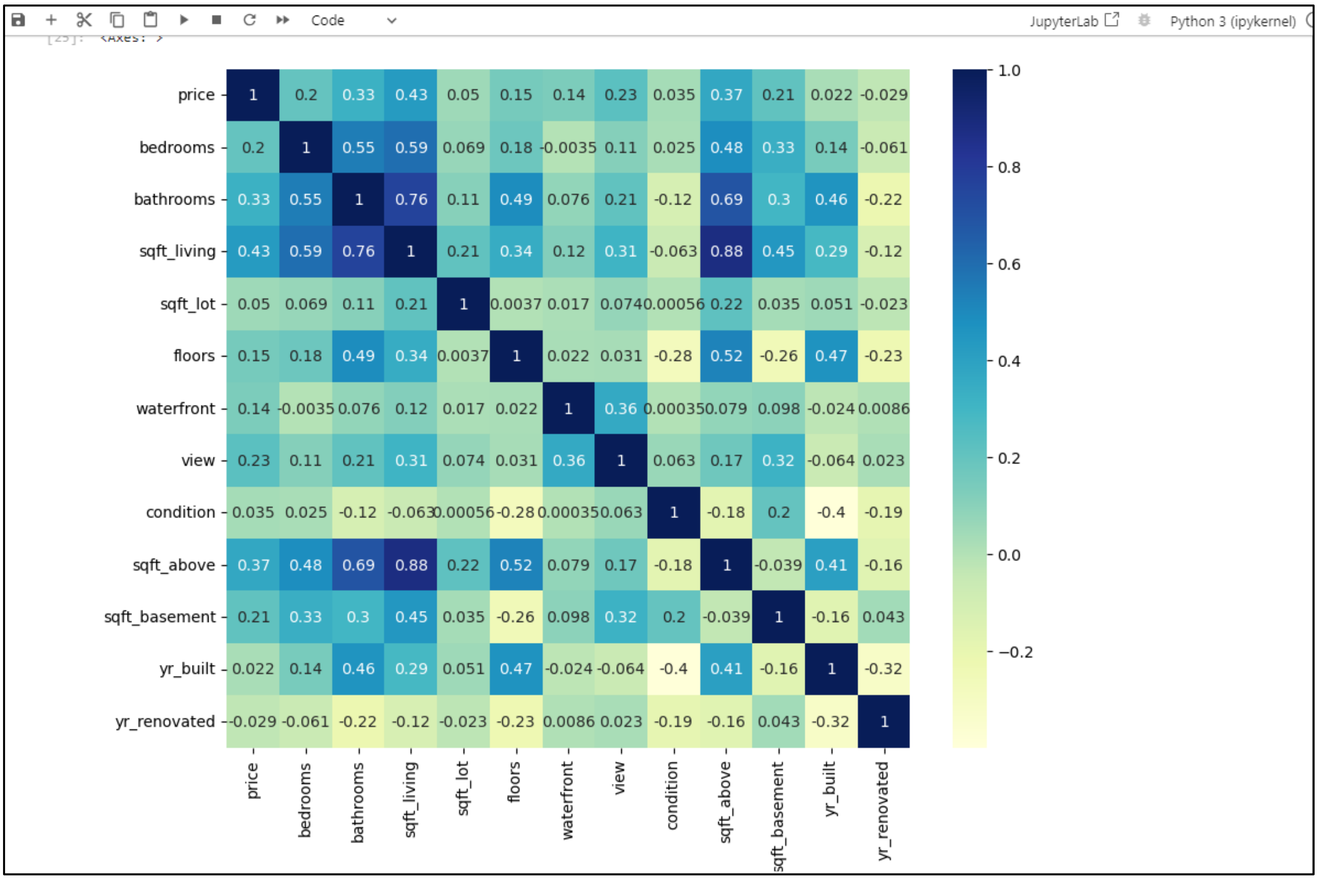
Loading Dataset

```
dataset = pd.read_csv('data.csv') •••
```

[7]:

	date	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	view	condition	sqft_above	sqft_basement	yr_built	yr_renovated	street	
0	2014-05-02 00:00:00	313000.0	3.0	1.50	1340	7912	1.5	0	0	3	1340	0	1955	2005	18810 Densmore Ave N	S
1	2014-05-02 00:00:00	2384000.0	5.0	2.50	3650	9050	2.0	0	4	5	3370	280	1921	0	709 W Blaine St	
2	2014-05-02 00:00:00	342000.0	3.0	2.00	1930	11947	1.0	0	0	4	1930	0	1966	0	26206-26214 143rd Ave SE	
3	2014-05-02 00:00:00	420000.0	3.0	2.25	2000	8030	1.0	0	0	4	1000	1000	1963	0	857 170th Pl NE	
4	2014-05-02 00:00:00	550000.0	4.0	2.50	1940	10500	1.0	0	0	4	1140	800	1976	1992	9105 170th Ave NE	R

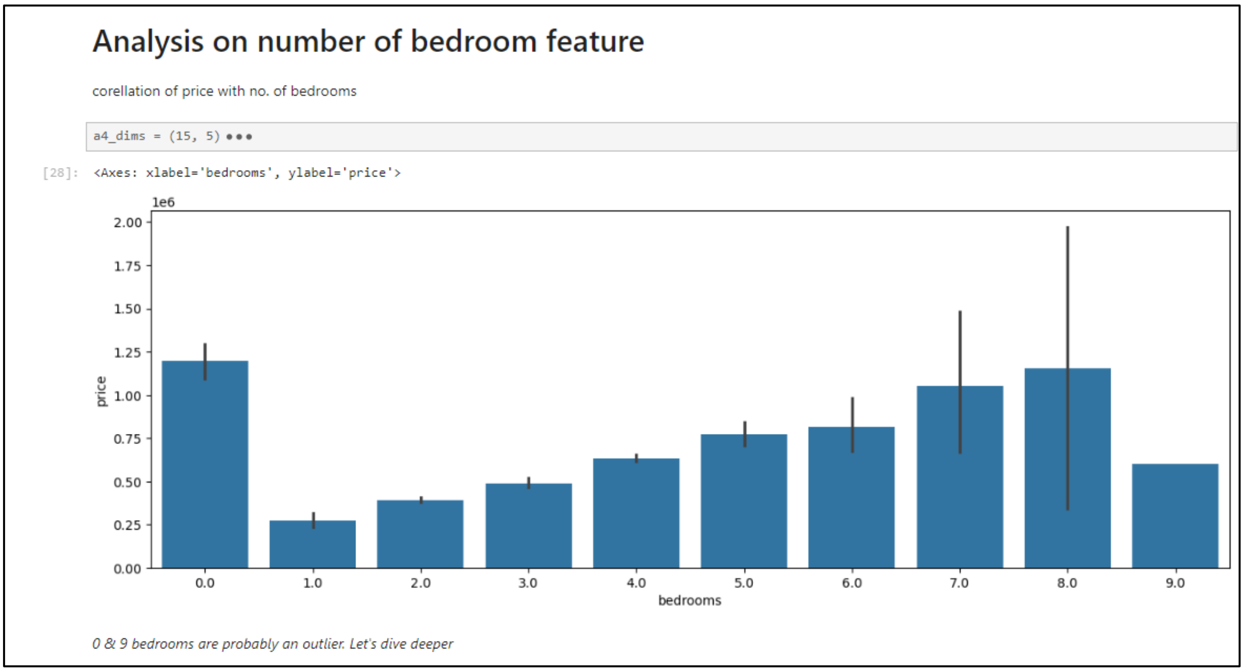
➢ General Corelation :-



```
[29]: <Axes: >
```

	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	view	condition	sqft_above	sqft_basement	yr_built	yr_renovated
price	1	0.2	0.33	0.43	0.05	0.15	0.14	0.23	0.035	0.37	0.21	0.022	-0.029
bedrooms	0.2	1	0.55	0.59	0.069	0.18	-0.0035	0.11	0.025	0.48	0.33	0.14	-0.061
bathrooms	0.33	0.55	1	0.76	0.11	0.49	0.076	0.21	-0.12	0.69	0.3	0.46	-0.22
sqft_living	0.43	0.59	0.76	1	0.21	0.34	0.12	0.31	-0.063	0.88	0.45	0.29	-0.12
sqft_lot	0.05	0.069	0.11	0.21	1	0.0037	0.017	0.074	0.00056	0.22	0.035	0.051	-0.023
floors	0.15	0.18	0.49	0.34	0.0037	1	0.022	0.031	-0.28	0.52	-0.26	0.47	-0.23
waterfront	0.14	-0.0035	0.076	0.12	0.017	0.022	1	0.36	0.00035	0.079	0.098	-0.024	0.0086
view	0.23	0.11	0.21	0.31	0.074	0.031	0.36	1	0.063	0.17	0.32	-0.064	0.023
condition	0.035	0.025	-0.12	-0.063	0.00056	-0.28	0.00035	0.063	1	-0.18	0.2	-0.4	-0.19
sqft_above	0.37	0.48	0.69	0.88	0.22	0.52	0.079	0.17	-0.18	1	-0.039	0.41	-0.16
sqft_basement	0.21	0.33	0.3	0.45	0.035	-0.26	0.098	0.32	0.2	-0.039	1	-0.16	0.043
yr_built	0.022	0.14	0.46	0.29	0.051	0.47	-0.024	-0.064	-0.4	0.41	-0.16	1	-0.32
yr_renovated	-0.029	-0.061	-0.22	-0.12	-0.023	-0.23	0.0086	0.023	-0.19	-0.16	0.043	-0.32	1

➢ Analysis on Number Of bedrooms w.r.t Price :-



Analysis on number of bedroom feature

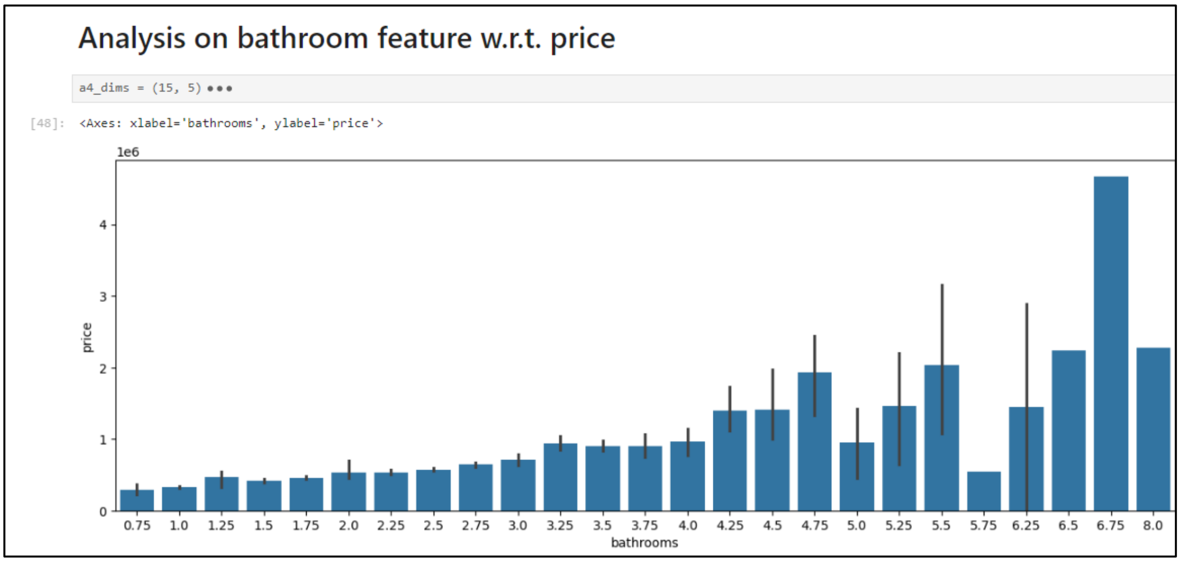
corellation of price with no. of bedrooms

```
a4_dims = (15, 5) •••
```

```
[28]: <Axes: xlabel='bedrooms', ylabel='price'>
```

0 & 9 bedrooms are probably an outlier. Let's dive deeper

- **Analysis on Number Of Bathrooms w.r.t Price :-**



Analysis on bathroom feature w.r.t. price

```
a4_dims = (15, 5) •••
```

```
[48]: <Axes: xlabel='bathrooms', ylabel='price'>
```

TESTING STRATEGIES

The testing strategies for the residential property price analysis project focus on validating the performance and reliability of the linear regression model. These strategies ensure that the model provides accurate predictions and insights based on the dataset. Below are the key testing strategies employed in the project :

- **Data Splitting :-**
  - **Training and Testing Sets:**
    - The dataset is divided into training and testing sets, typically with an 80/20 or 70/30 split. The training set is used to fit the linear regression model, while the testing set evaluates its performance.
  - **Validation Set:**
    - Optionally, a separate validation set can be used to fine-tune model parameters and prevent overfitting, especially in more complex models or when additional hyperparameters are involved.

- **Model Evaluation Metrics**
  - **Mean Absolute Error (MAE):**
    - Measures the average magnitude of the errors in a set of predictions, without considering their direction. It is calculated as the average over the test sample of the absolute differences between prediction and actual observation.
  - **Mean Squared Error (MSE):**
    - Represents the average of the squared differences between predicted and actual values. It penalizes larger errors more than MAE, making it sensitive to outliers.
  - **R-squared (R^2):**
    - Indicates the proportion of the variance in the dependent variable that is predictable from the independent variables. A higher R^2 value indicates a better fit of the model

- **Residual Analysis**
  - **Residual Plots:**
    - Plotting the residuals (the differences between actual and predicted values) helps identify patterns or non-random distribution, which can indicate model inadequacies or the presence of outliers.

CONCLUSION & FUTURE WORK

- **Conclusion**

The residential property price analysis project successfully developed and implemented a linear regression model to predict housing prices based on various property characteristics and economic indicators. Through detailed exploratory data analysis and rigorous testing, the model has provided valuable insights into the factors influencing property prices, including the impact of location, property size, and market trends.

The integration of the Streamlit UI allowed for seamless interaction with the model, enabling users to input data and receive real-time predictions. This user-friendly interface, combined with robust statistical analysis, has made the project a practical tool for stakeholders such as real estate professionals, investors, and policymakers. By understanding the key drivers of property values, these stakeholders can make informed decisions about investments, pricing strategies, and policy development.

Overall, the project demonstrates the potential of data-driven approaches in real estate analysis, offering a comprehensive understanding of market dynamics and supporting more strategic decision-making.

- **Future Work**

While the project has achieved significant milestones, there are several areas for future exploration and enhancement:

1. **Incorporation of Additional Data Sources :**
   - Integrating more diverse datasets, such as crime rates, school quality, and public transportation access, could provide a more holistic view of the factors affecting property prices.
2. **Advanced Modeling Techniques :**
   - Exploring more sophisticated machine learning models, such as decision trees, random forests, or neural networks, could improve prediction accuracy and capture non-linear relationships between variables.
3. **Dynamic and Real-Time Data Updates :**
   - Implementing a system for real-time data updates and analysis would allow the model to adapt to rapidly changing market conditions and provide more timely insights.
4. **User Interface Enhancements :**
   - Further refining the UI to include data visualization features, such as interactive graphs and maps, would enhance user engagement and provide more accessible insights.

REFERENCES

- Websites :-

- https://Kaggle.com :- for datasets of project
- https://Chatgpt.com :- for documentation's
- https://www.geeksforgeeks.org/machine-learning/ :- for learning
- https://www.ibm.com/topics/machine-learning :- for learning
- https://streamlit.io/cloud :- for UI development
- https://share.streamlit.io/ :- Deployed Project
- https://google.com :- for query's