##### ACTIVITY ANALYZER

##### A PROJECT REPORT

***Submitted by:***

Anubhav Singh (161B039)

Mrigank Singh (161B124)

Sidharth Gupta (161B235)

**Under the guidance of: Dr. Amit Kumar**



November – 2018

***Submitted in partial fulfilment for the award of the degree of***

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE & ENGINEERING

Department of Computer Science & Engineering

JAYPEE UNIVERSITY OF ENGINEERING & TECHNOLOGY,

A-B ROAD, RAGHOGARH, DT. GUNA - 473226, M.P., INDIA

**Declaration by the Student**

I hereby declare that the work reported in the B. Tech. project entitled as “**ACTIVITY ANALYZER**”, in partial fulfilment for the award of degree of B. Tech submitted at Jaypee University of Engineering and Technology, Guna, as per best of my knowledge and belief there is no infringement of intellectual property right and copyright. In case of any violation, I will solely be responsible.

**Anubhav Singh (161B039)**

**Mrigank Singh (161B124)**

**Sidharth Gupta (161B235)**

Department of Computer Science and Engineering

                                                                   Jaypee University of Engineering and Technology

                                                                    Guna, M.P., India

Date: 26/11/2018

**CERTIFICATE**

This is to certify that the work titled “**ACTIVITY ANALYZER**” submitted by **Anubhav Singh (161B039), Mrigank Singh (161B124) and Sidharth Gupta (161B235)** in partial fulfilment for the award os degree of B. Tech of Jaypee University of Engineering & Technology, Guna has been carried out under my supervision. As per best of my knowledge and belief, there is no infringement of intellectual property right and copyright. In addition, this work has not been submitted partially or wholly to any other University or Institute for the award of this or any other degree or diploma. In case of any violation, concern student will solely be responsible.

**Dr. Amit Kumar**

Assistant Professor (SG)

Date:

**Abstract**

The relationship between house prices and the economy is an important motivating factor for predicting house prices. Housing price trends are not only the concern of buyers and sellers, but it also indicates the current economic situation. Therefore, it is important to predict housing prices without bias to help both the buyers and sellers make their decisions. This project uses an open source dataset, which includes 20 explanatory features and 21,613 entries of housing sales in Seattle, Washington.

We compared different parameters with the price using linear regression and gradient boosting regression algorithm to see the relationship between them to predict the house prices in Seattle, Washington. With gradient boosting regression algorithm, the price prediction increased from 73.2% to 91.9%. The benefit of applying feature reductions is that it helps us to pick the more important features, so we will not over-fit the model with too many features.

**Acknowledgement**

We wish to express our profound gratitude and indebtedness to Dr. Amit Kumar, Department of Computer Science and Engineering, Jaypee University of Engineering and Technology, Guna for introducing the present topic and for their inspiring guidance, constructive criticism and valuable suggestion throughout the project work. Last but not least, our sincere thanks to all our friends who have patiently extended all sorts of help for accomplishing this undertaking.

We would also like to thank Dr. Ratnesh Litoriya for his feedback towards our project.

**Signature of Student**

Anubhav Singh (161B039)

Mrigank Singh (161B124)

Sidharth Gupta (161B235)

Date:

**Table Of Content**

|  |  |
| --- | --- |
| Title Page | 1 |
| Declaration by Students | 2 |
| Certificate of the Guide | 3 |
| Abstract | 4 |
| Acknowledgement | 5 |
| 1. Introduction | 8 |
| * 1. Problem Definition | 8 |
| * 1. Objective | 9 |
| * 1. System Overview | 9 |
| * 1. System Features | 9 |
| * + 1. House Price Prediction | 9 |
| * 1. Software Specification | 9 |
| * + 1. Jupyter | 10 |
| * + 1. Anacondas | 10 |
| * + 1. Numpy | 10 |
| * + 1. Pandas | 10 |
| * + 1. MatplotLib | 11 |
| 1. Literature Survey | 12 |
| * 1. Existing System | 12 |
| * 1. Proposed System | 12 |
| * 1. Feasibility Study | 12 |
| * + 1. Technical Feasibility | 13 |
| * + - 1. Project Technique | 13 |
| * + - 1. Project Requirements | 13 |
| * + 1. Legal Feasibility | 13 |
| * + 1. Operational Feasibility | 14 |
| 1. System Analysis and Design | 16 |
| * 1. Use Case Diagram/Activity Diagram | 16 |
| * + 1. Use Case Diagram | 16 |
| * + 1. Activity Diagram | 18 |
| * 1. Algorithms and Pseudo Codes | 18 |
| * + 1. Linear Regression | 19 |
| * + 1. Forms of Linear Regression | 19 |
| * + 1. Gradient Boosting Regression | 20 |
| 3.3. User Interface | 20 |
| 3.3.1. Flask Framework | 21 |
| 3.4. Data Analysis and Visualisation | 22 |
| 3.5 Output | 28 |
| 3.6 Limitations | 28 |
| 1. Conclusion and Future Work | 29 |
| * 1. Conclusion | 29 |
| * + 1. Chapter Overview | 29 |
| * + 1. Project Achievement | 29 |
| * + 1. Challenges | 29 |
| * + 1. Concluding Remarks | 29 |
| * 1. Future Works | 30 |
| References | 31 |
| Personal Details | 32 |

**List of Figures**

|  |  |  |
| --- | --- | --- |
| **Figure Number** | **Figure Description** | **Page Number** |
|  |  |  |
| Figure 2.1. | Linear Regression accuracy Test | 14 |
| Figure 2.2. | Gradient Boosting Algorithm | 15 |
| Figure 3.1. | Use Case Diagram | 16 |
| Figure 3.2. | Activity Diagram | 18 |
| Figure 3.3. | Linear Regression graph | 19 |
| Figure 3.4. | User Interface | 21 |
| Figure 3.5. | Data Set | 22 |
| Figure 3.6. | Graph - Count vs Bedroom | 23 |
| Figure 3.7 | Graph - Price vs Square feet | 23 |
| Figure 3.8 | Graph - Price vs Location | 24 |
| Figure 3.9. | Graph - Latitude vs Price | 24 |
| Figure 3.10. | Graph - Price vs No. of Bedroom | 25 |
| Figure 3.11. | Graph - Price vs Living room and Basement | 25 |
| Figure 3.12. | Graph - Price vs Waterfront | 26 |
| Figure 3.13. | Graph - Floor Count | 26 |
| Figure 3.14. | Graph - Price vs Floor Count | 26 |
| Figure 3.15. | Graph - Price vs Condition | 27 |
| Figure 3.16. | Graph - Price vs Zip code | 27 |
| Figure 3.17 | User Input | 28 |
| Figure 3.18 | Predicted Output | 28 |
|  |  |  |

**Chapter 1: Introduction**

The relationship between house prices and the economy is an important motivating factor for predicting house prices. A property’s value is important in real estate transactions. House prices trends are not only the concerns for buyers and sellers, but they also indicate the current economic situations. Therefore, it is important to predict the house prices without bias to help both buyers and sellers make their decisions (Wu, 2017).

There are different machine learning algorithms to predict the house prices. This project will use linear regression and gradient boosting regression algorithm to predict house prices in Seattle, Washington. The motivation for choosing linear regression and gradient boosting algorithm is it can accurately predict the trends of how various factors are affecting the price of the house. There are many factors affecting the house prices, such as numbers of bedrooms, bathrooms, basement, square-feet living, zip code, longitude and latitude. In addition, choosing different combinations of parameters like square-feet living and square-feet basement will also affect the predictions greatly. This project is guided by these questions: Which features are important for predicting price of houses? How to select those features in the data to achieve a better performance? (Wu, 2017)

* 1. **Problem Definition**

We live in an era where Real Estate is the second largest industry. We can see ups and down in the prices of the real estates on daily basis. Our project can benefit two clients at the same time. (Overview of India real estate, n.d.)

**House-buye**r: The client who is looking for their next dream house can use our project to determine a suitable price tag. Buyer can get the idea of the price of his dream house on the basis of certain factors. (Overview of India real estate, n.d.)

**House-seller**: This client can take advantages of the features that are mentioned in our project to know the best price of the house that he wants to sell. For example, buying a house at a good location but small square footage. The client will invest on making rooms at a small cost to get a large return. (Overview of India real estate, n.d.)

* 1. **Objectives**

Aim of this project are as follows:

* To identify factors affecting House prices
* To generate the pattern from large datasets for price prediction of the house on the basis of given details.

The main feature of this project is to generate an approximate forecasting output and create a general idea of future values based on the previous data by generating a pattern. The scope of the project does not exceed more than a generalized suggestion tool.

Obtaining predictions through the application should be quick for the users, even though the prediction mechanism may involve computationally intensive tasks. Training of the dataset takes place in background Various machine learning techniques will be explored, tested with past data on property transactions, before selecting one and modifying it to best suit our needs

* 1. **System Overview**

This system named “Activity Analyzer” is a web application that aims to predict House Prices using Regression analysis, Gradient Boosting regression and Machine Learning algorithms. This project is intend to help an individual to get an idea of their dream house which they want to buy and also helps to estimate a suffice cost of their property to those who want to sell.

* 1. **System Features**
     1. **House Price Prediction**

The prediction feature of this system tries to predict the house price with the help of Regression algorithm and Gradient boosting regression algorithm to give the user an approximate value of the desired property.

1. **Software Specification**

Some software specification of our software are as follows

1. **Jupyter**

The Jupyter Notebook is an incredibly powerful tool for interactively developing and presenting data science projects. A notebook integrates code and its output into a single document that combines visualisations, narrative text, mathematical equations, and other rich media. The intuitive workflow promotes iterative and rapid development, making notebooks an increasingly popular choice at the heart of contemporary data science, analysis, and increasingly science at large (Jupyter Notebook, n.d.).

1. **Anacondas**

Anacondas is a free and open-sourcedistribution of the Python and R programming languages for data science and machine learning applications (large-scale data processing, predictive analytics, scientific computing), that aims to simplify package management and deployment. Package versions are managed by the package management system conda (Anacondas, n.d.).

1. **Numpy**

NumPy is the fundamental package for scientific computing with Python. It contains among other things:

* a powerful N-dimensional array object
* sophisticated (broadcasting) functions
* useful linear algebra, Fourier transform, and random number capabilities

Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined. This allows NumPy to seamlessly and speedily integrate with a wide variety of databases (NumPy, n.d.).

1. **Pandas**

**It** is a Python module, and Python is the programming language that we're going to use. The Pandas module is a high performance, highly efficient, and high level data analysis library. At its core, it is very much like operating a headless version of a spreadsheet, like Excel. Pandas is also compatible with many of the other data analysis libraries, like Scikit-Learn for machine learning, Matplotlib for Graphing, NumPy, since it uses NumPy, and more. It's incredibly powerful and valuable to know (Pandas, n.d.).

## **Key Features of Pandas**

* Fast and efficient Data Frame object with default and customized indexing.
* Tools for loading data into in-memory data objects from different file formats.
* Data alignment and integrated handling of missing data.
* Reshaping and pivoting of date sets.
* Label-based slicing, indexing and sub setting of large data sets.
* Group by data for aggregation and transformations.
* High performance merging and joining of data.

1. **Matplotlib**

Matplotlib is a library for making 2D plots of arrays in Python. It is independent of MATLAB, and can be used in a Python, object oriented way. Although Matplotlib is written primarily in pure Python, it makes heavy use of NumPy and other extension code to provide good performance even for large arrays (Matplotlib, n.d.).

**Chapter 2: Literature Survey**

1. **Existing System**

As Machine Learning and Data Analysis evolves predictive analytics is finding its way into more business use cases. Some prospective use cases for ML-based predictive analytics are (Maruti Techlabs, n.d.):

**E-commerce**- Using ML business can predict customer churn and fraudulent transaction. Also predicting which product customer will click on.

**Marketing**- Common use case is identifying and acquiring prospects with attributes similar to existing customers. They can also prioritize known prospects, leads, and accounts based on their likelihood to take action.

**Medical Diagnose**- Medical professionals can use a program modelled using ML to predict the likeliness of a particular illness. The model will use a database of patient records and will make predictions based on symptoms exhibited by the patient.

1. **Proposed System**

There are different machine learning algorithms to predict the house prices. This project will use linear regression and gradient boosting regression algorithm to predict house prices in Seattle, Washington. The motivation for choosing linear regression and gradient boosting algorithm is it can accurately predict the trends of how various factors are affecting the price of the house. There are many factors affecting the house prices, such as numbers of bedrooms, bathrooms, basement, square-feet living, zip code, longitude and latitude. In addition, choosing different combinations of parameters like square-feet living and square-feet basement will also affect the predictions greatly.

1. **Feasibility Study**

An assessment of the practicality of proposed plan or methods

1. **Technical Feasibility**

The complete study of the project in terms of input, processes, output, fields, program and procedures

1. **Project Technique**

Using various algorithms of data analysis and machine learning like Linear regression and Gradient boosting to train our model with the help of dataset.

1. **Project Requirement**

**Python and its Libraries**: Python is the most advance language today and its libraries make it more feasible to use. Certain python libraries like NumPy and Pandas were used in this project.

**Pycharm**: Pycharm is an IDE used in computer specifically for python language. It provides code analysis, a graphical debugger, an integrated unit tester, and supports web development.

**Html, Css, bootstrap and Java Script:** Html is a scripting language most commonly used by the web developers to develop a basic skeleton of a website. Html, combined with css and bootstrap and java script can create a stunning website which is ready to get deployed by the developer. Our project uses these tools to make a proper UI for the easy interaction of the user with our system. Our user will see a web portal which include certain fields which are to be filled by the user in order to get the output proposed by our system.

**Datasets:** A dataset including details of various houses was included in our project. These data set were available online for non- commercial research purpose only. The datasets were used to train our model and to know the accuracy of our algorithm.

1. **Legal Feasibility**

Every tool used in our project were either open-source or is for non-commercial research purpose only. Python is an open source programming language and its libraries are open for all.

1. **Operational Feasibility**

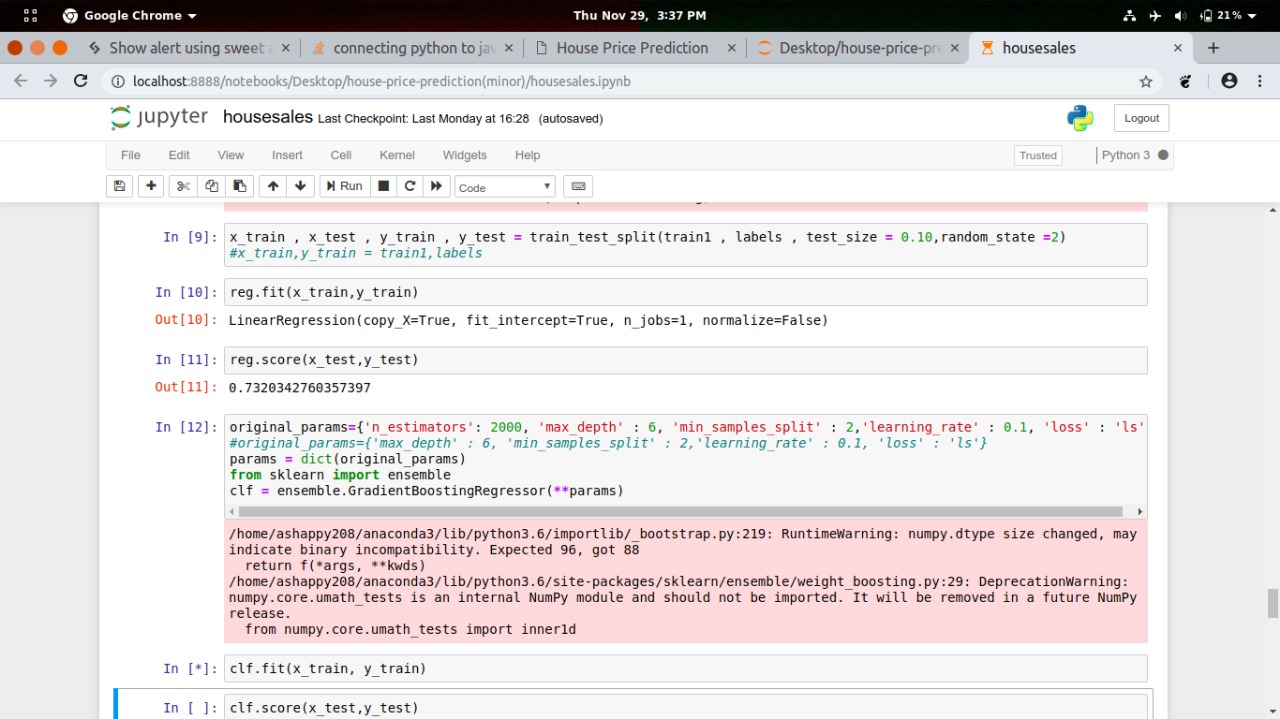


Figure 2.1 - Linear Regression accuracy test

We use train data and test data , train data to train our machine and test data to see if it has learnt the data well or not.

We import our dependencies , for linear regression we use sklearn (built in python library) and import linear regression from it.

We then initialize Linear Regression to a variable reg.

Now we know that prices are to be predicted , hence we set labels (output) as price columns and we also convert dates to 1’s and 0’s so that it doesn’t influence our data much . We use 0 for houses which are new that is built after 2014.

We again import another dependency to split our data into train and test.

I’ve made my train data as 90% and 10% of the data to be my test data , and randomized the splitting of data by using random\_state.So now , we have train data , test data and labels for both let us fit our train and test data into linear regression model.

After fitting our data to the model we can check the score of our data ie , prediction. in this case the prediction is 73%

The accuracy of the model is lower than our aim of 85. So how do we achieve that 85% target?

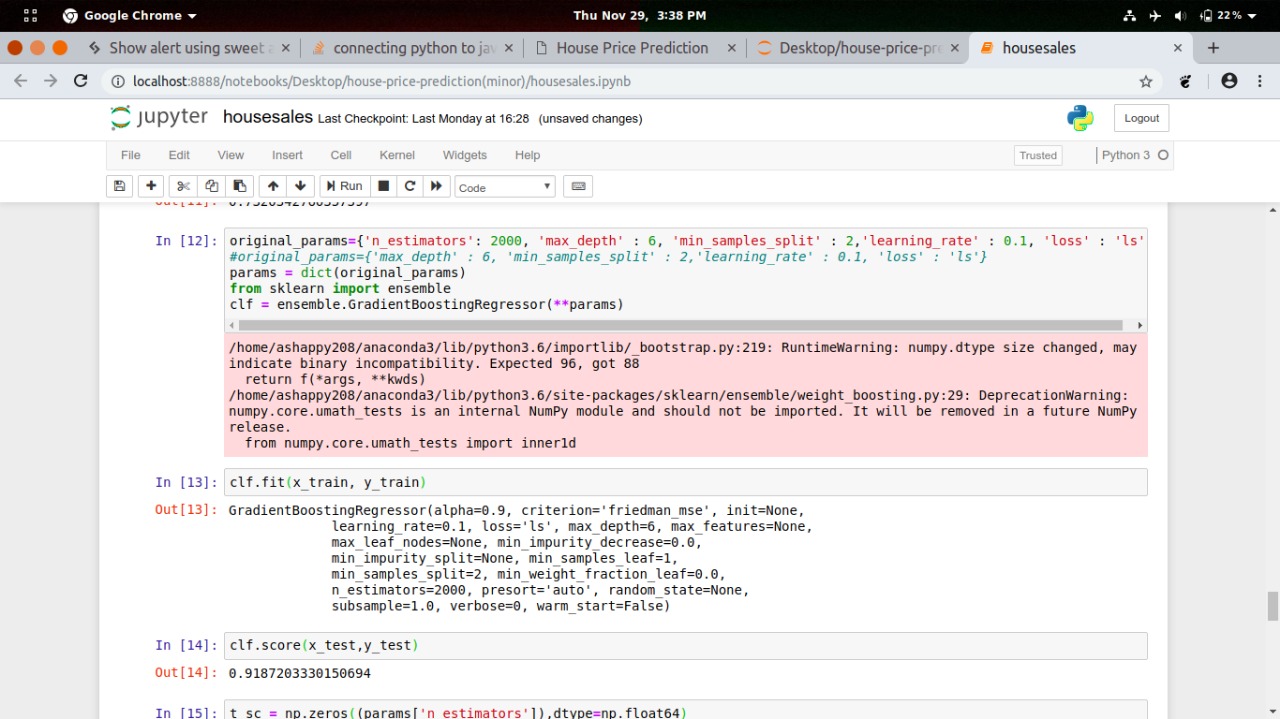


Figure 2.2 - Gradient boosting accuracy test

To improve our model we use gradient boosting algorithm:

-We first import the library from sklearn ( trust me , it is the best library for all statistical related models)

-We create a variable where we define our gradient boosting regressor and set parameters to it , here

-n\_estimator — The number of boosting stages to perform. We should not set it too high which would overfit our model.

-max\_depth — The depth of the tree node.

-learning\_rate — Rate of learning the data.

-loss — loss function to be optimized. ‘ls’ refers to least squares regression

-minimum sample split — Number of sample to be split for learning the data

-We then fit our training data into the gradient boosting model and check for accuracy

-We got an accuracy of 91.94% which is amazing!

1. **System Analysis and Design**

 Method of studying a system by examining its component parts and their interactions Structured data analysis (systems analysis), analysing the flow of information within an organization with data-flow diagrams.

* 1. **Use Case Diagram/Activity Diagram**

Some diagrams relating to our project are:

1. **Use Case Diagram**

Admin

User

<Include>

Figure 3.1- Use Case Diagram

|  |  |  |
| --- | --- | --- |
| **Use case Id** | **Use Case Description** | **Primary Actor** |
| 1 | Collect Data | Admin |
| 2 | Compute Result and Performance | Admin |
| 3 | System Update | Admin |
| 4 | Enter House Details | User |
| 5 | Past House Detail | User |
| 6 | View Predicted Results | User |

Table 3.1

**Use Case Description (Figure 3.1)**

Use Case ID: 1

Use Case Name: Collect Data

Use Case Description: Admin have to collect data from different sources to increase the dataset.

Use Case ID: 2

Use Case Name: Compute result and performance

Use Case Description: Prediction result will be handled and generated by admin. Project will be built , through which the result of prediction and system performance will be analysed.

Use Case ID: 3

Use Case Name: System update

Use Case Description: With the change of market and economy regular update of system is required. Beside there the predict result of house price and their actual price will be updated by admin.

Use Case ID: 4

Use Case Name: Enter House Details

Use Case Description: For the house price prediction the user will fill the required house details.

Use Case ID: 5

Use Case Name: Past House Details

Use Case Description: Entered detail will be compared with past house details for prediction.

Use Case ID: 6

Use Case Name: View predicted results

Use Case Description: This use case is most important in whole project. The key feature of this project is to predict the house value of Seattle, Washington. Thus, this will be available in user interface and viewer can observe them.

**3.1.2. Activity Diagram**

The following Activity diagram shows the control flow from one activity to another.

Field Empty error message

Result

Enter House details house details

Text field is left empty

Figure 3.2 - Activity Diagram

3.

1. **Algorithms and Pseudo code**

Some of the algorithm and techniques used in our project are:

**3.2.1. Linear Regression**

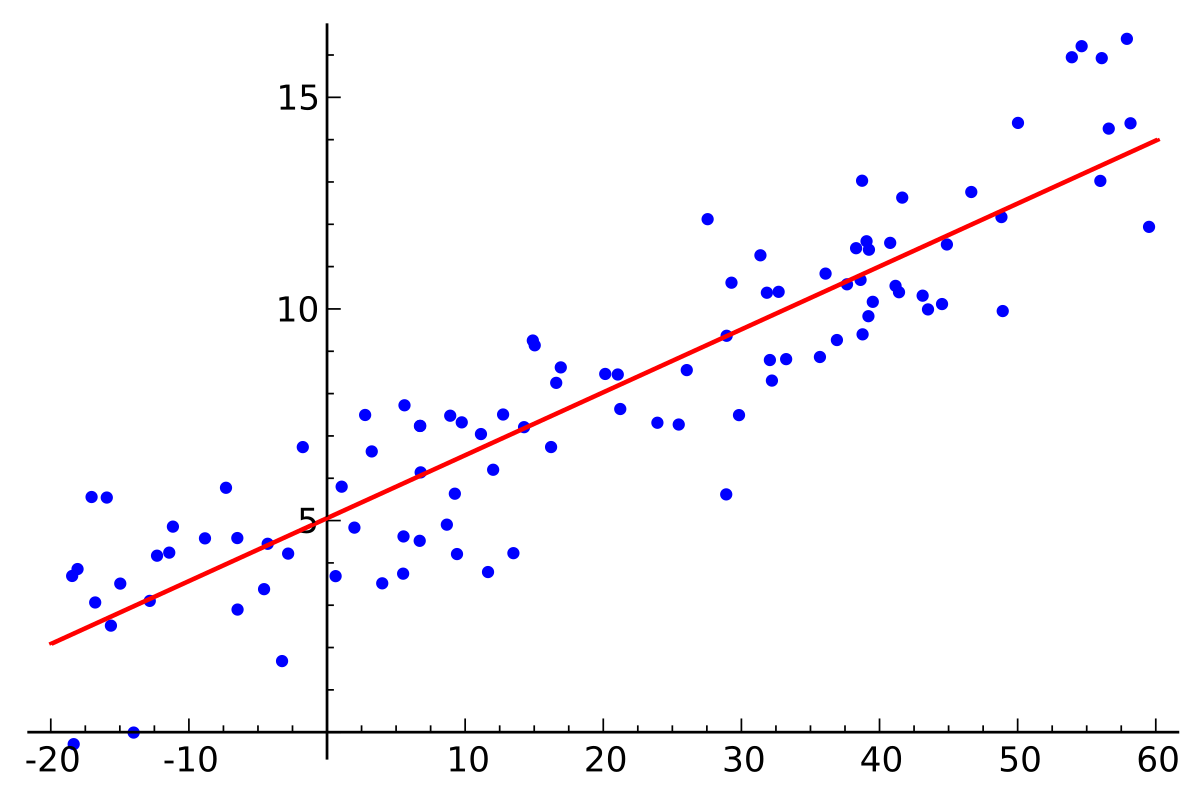


Figure 3.3 - Linear Regression

Linear regression is a type of regression analysis where the number of independent variables is one and there is a linear relationship between the independent(x) and dependent(y) variable. The red line in the above graph is referred to as the best fit straight line. Based on the given data points, we try to plot a line that models the points the best. The line can be modelled based on the linear equation shown below (Linear Regression, n.d.)

**3.2.2. Form of Linear Regression**

Mathematically, we can write a linear relationship as:

…Equation 3.1

**Where:**

* *y*is the response
* ***β*** values are called the **model coefficients**. These values are “learned” during the model fitting/training step.
* is the intercept
* is the coefficient for *X1* (the first feature)
* is the coefficient for *Xn*(the nth feature)

**3.2.3. Gradient Boosting Regression Algorithm**

Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees. It builds the model in a stage-wise fashion like other boosting methods do, and it generalizes them by allowing optimization of an arbitrary differentiable loss function (Gradient Boosting, n.d.).

Gradient boosting involves three elements:

1. A loss function to be optimized.

2. A weak learner to make predictions.

3. An additive model to add weak learners to minimize the loss function.

Like other boosting methods, gradient boosting combines weak “learners” into a single strong learner in an iterative fashion. It is easiest to explain in the least-squares regression setting, where the goal is to “teach” a model **F** to predict values of the form

...Equation 3.2

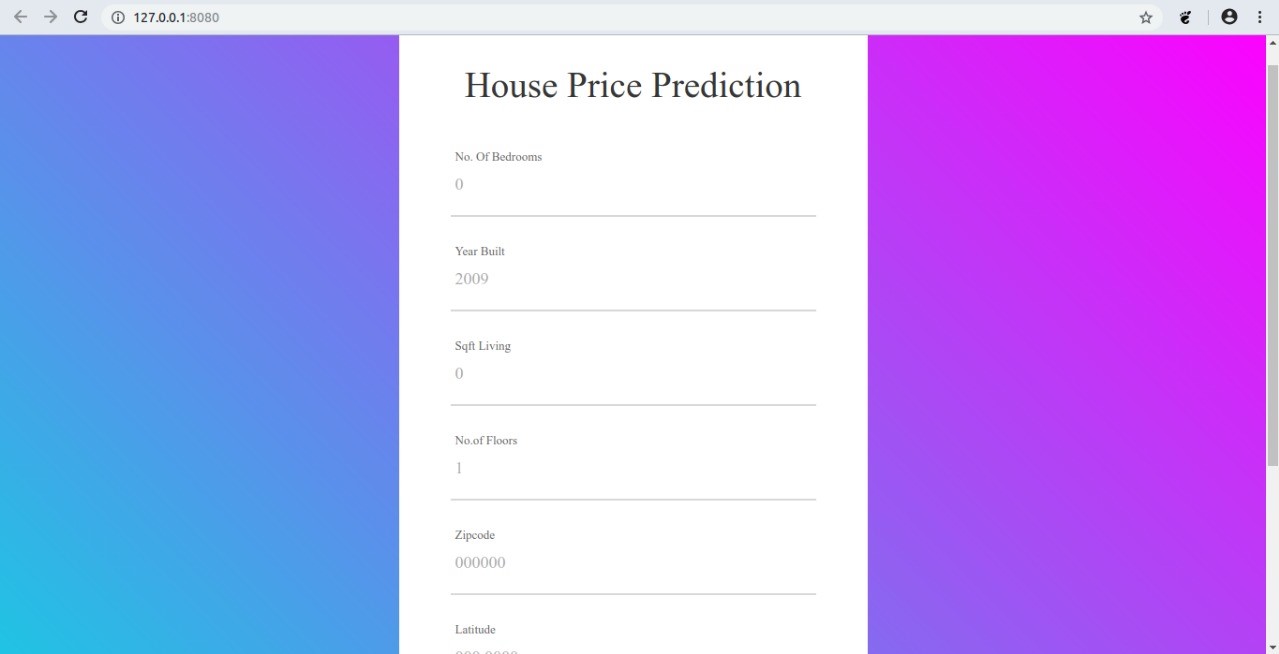
by minimizing the mean squared error

…Equation 3.3

where {\displaystyle i}indexes over some training set of size {\displaystyle n}of actual values of the output variable (Gradient Boosting, n.d.){\displaystyle y}.

1. **User Interface**

Web portal created with the help of HTML, Css, Bootstrap and Javascript. The user interface contains some text field through which user will provide the required information to our model.



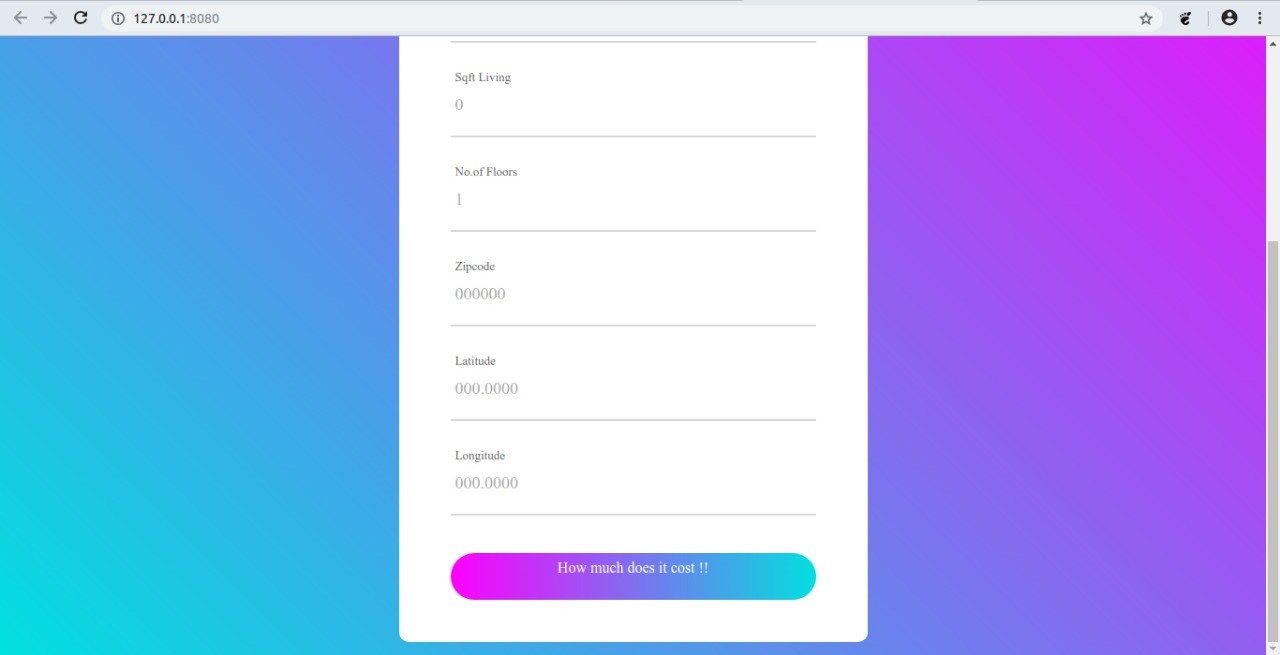


Figure 3.4 - User Interface

All the fields are required to be filled by the user in order to get proper results

1. **Flask web framework**

Flask is a micro web framework written in python. It is classified as microframework because it does not require any particular tool or libraries. It has no database abstraction layer, form validation or any other such tools (Flask, n.d.).**{\displaystyle {\tfrac {1}{n}}\sum \_{i}({\hat {y}}\_{i}-y\_{i})^{2}}**

1. **Data Analysis And Visualisation**

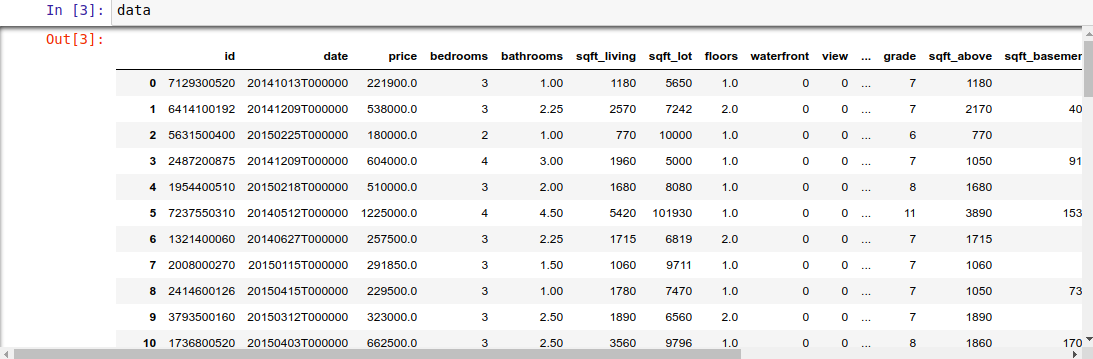


Figure 3.5 – Data set

This project uses a dataset from Kaggle open source datasets (Kaggle, n.d.). The dataset consists of 20 explanatory features and 21,613 entries of housing sales in Seattle, Washington. It describes different aspects of housing sales. The table above shows the feature names and their descriptions.

From the table shown above, we can see the independent variables from the Housing dataset are the explanatory variables. The independent variables are date, price, bedrooms, bathrooms, sqft\_living, sqft\_lot, floors, waterfront, view, condition, grade, sqft\_above, sqft\_basement, yr\_built, yr\_renovated, zip code, latitude, longitude, sqft\_living15, and sqft\_lot15. We can see that these variables include categorical variables, numerical variables, and time series variables. The dependent variable is the sale price of houses.

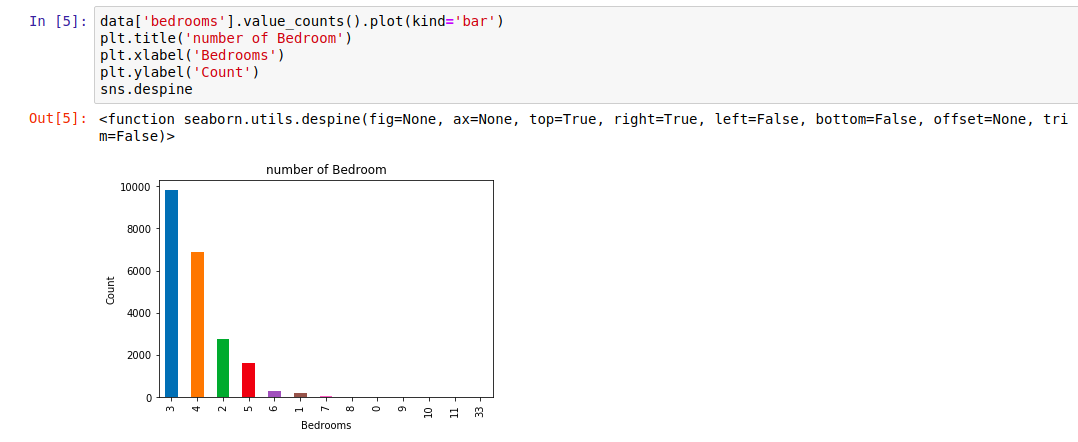


Figure 3.6 – Count vs Bedroom

As we can see from the visualization, 3 bedroom houses are most commonly sold followed by 4 bedroom. So how is it useful? For a builder having this data, He can make a new building with more 3 and 4 bedroom’s to attract more buyers.

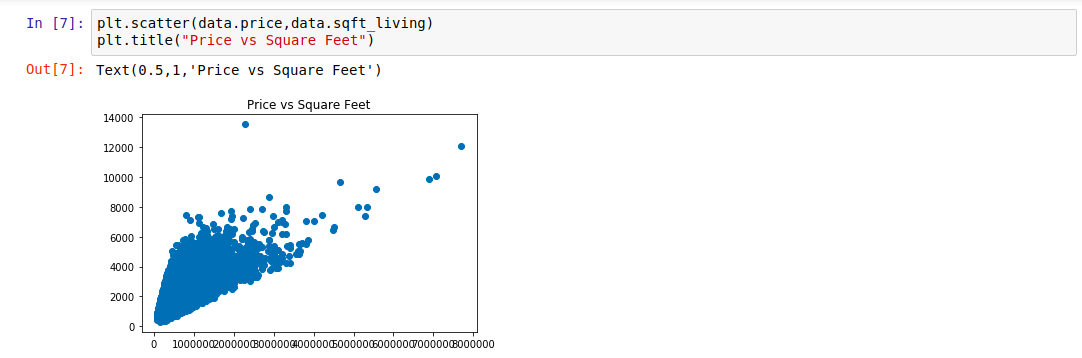


Figure 3.7 – Price vs Square Feet

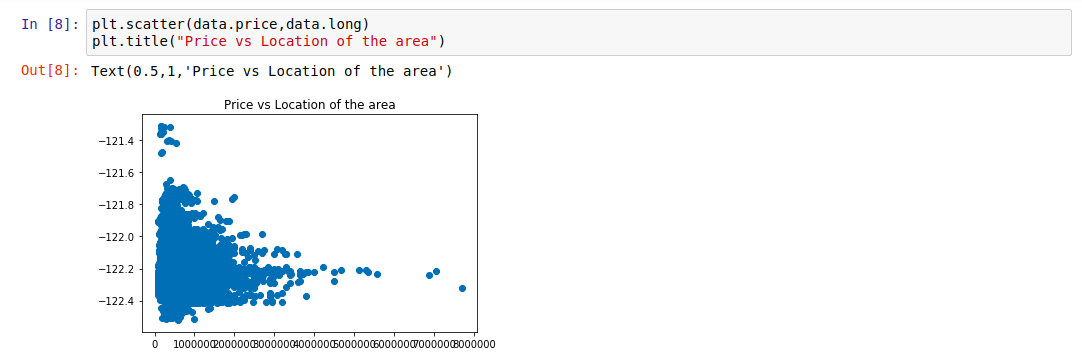


Figure 3.8 – Price vs Location

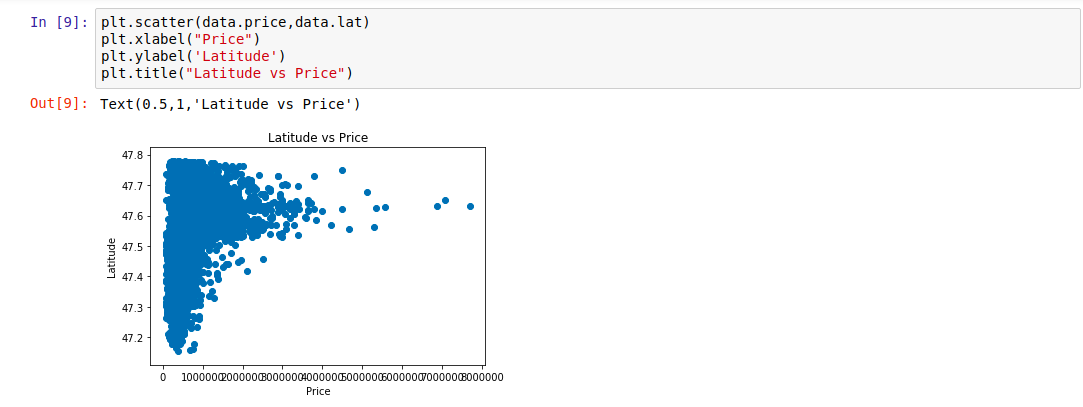


Figure 3.9 – Latitude vs Price

The plot that we used above is called scatter plot, scatter plot helps us to see how our data points are scattered and are usually used for two variables. From the first figure we can see that more the living area, more the price though data is concentrated towards a particular price zone, but from the figure we can see that the data points seem to be in linear direction. Thanks to scatter plot we can also see some irregularities that the house with the highest square feet was sold for very less, maybe there is another factor or probably the data must be wrong. The second figure tells us about the location of the houses in terms of longitude and it gives us quite an interesting observation that -122.2 to -122.4 sells houses at much higher amount.

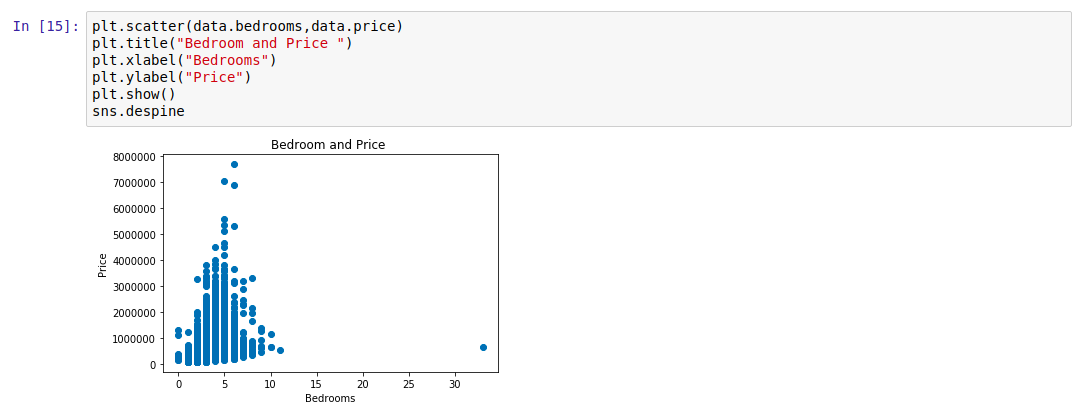


Figure 3.10 – Price vs No. of Bedrooms

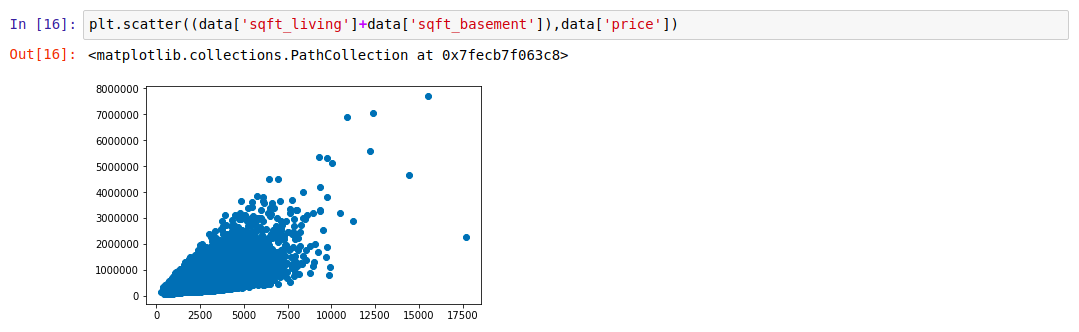


Figure 3.11 – Price vs Square-Feet-living and Basement

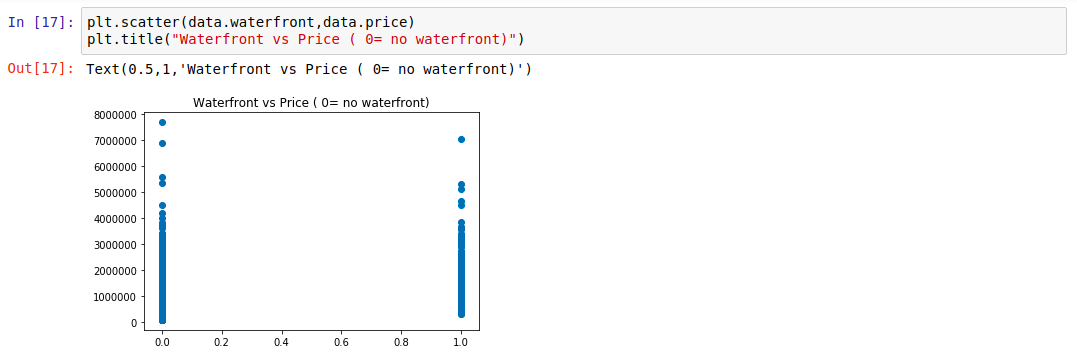


Figure 3.12 – Price vs Waterfront

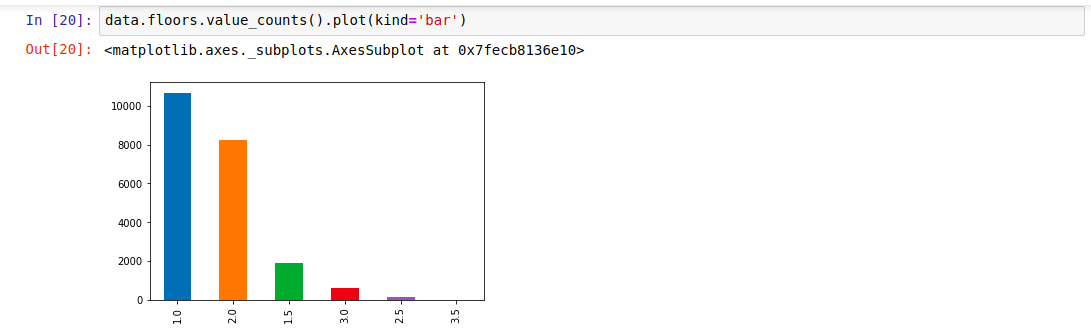


Figure 3.13 – Floor count

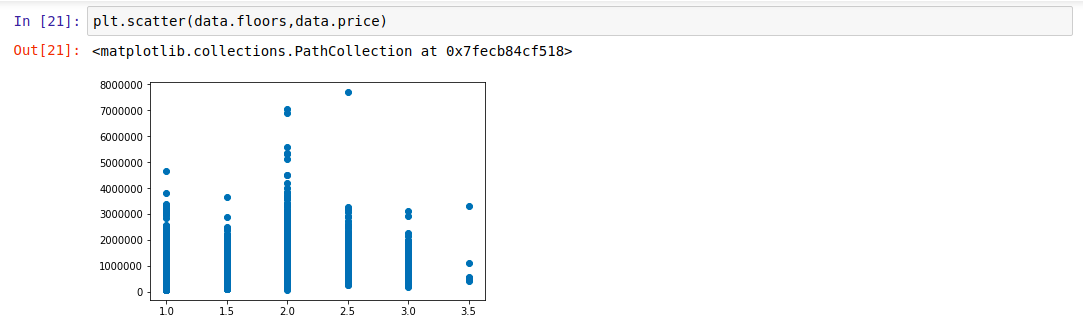


Figure 3.14 – Price vs floor

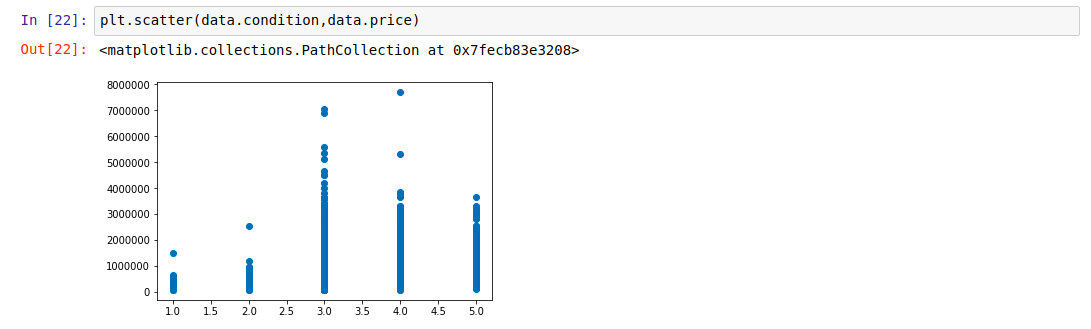


Figure 3.15 - Price vs Condition

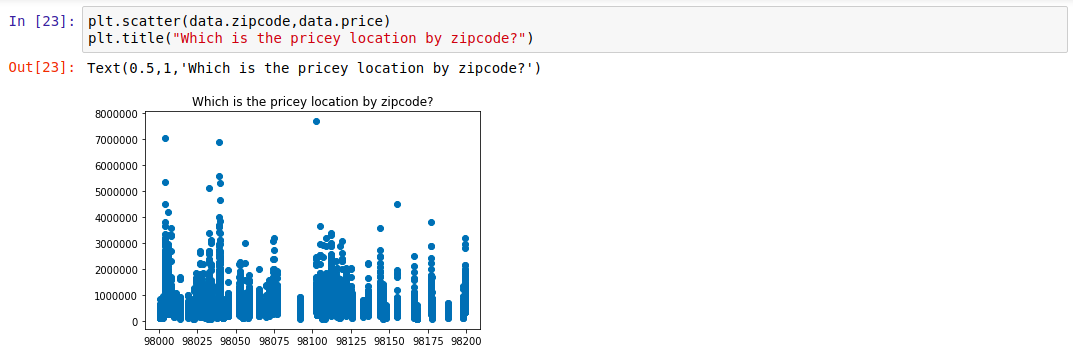


Figure 3.16 - Price vs Zip code

The above graph shows the distribution of the house prices.

As we can see from all the above representation that many factors are affecting the prices of the house , like square feet which increases the price of the house and even location influencing the prices of the house.

**3.5. Output**

The output screen of the project has been displayed in figure 3.17 and 3.18

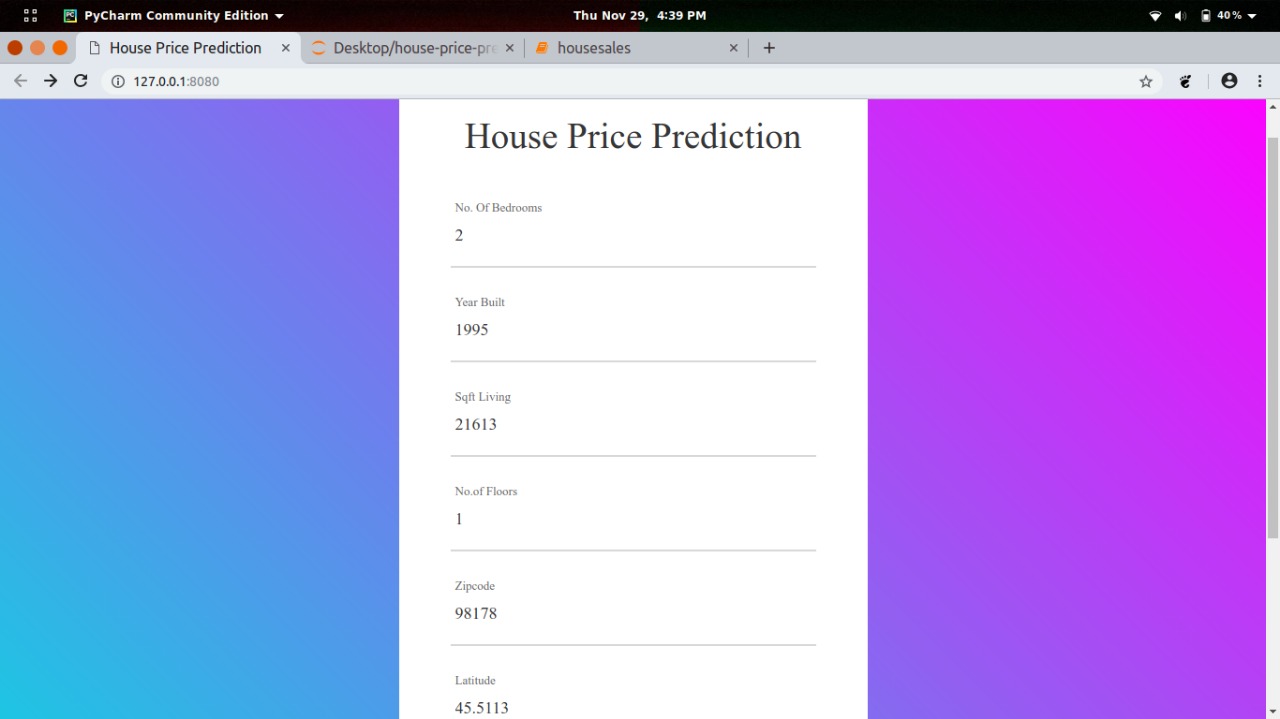


Figure 3.17 - User input

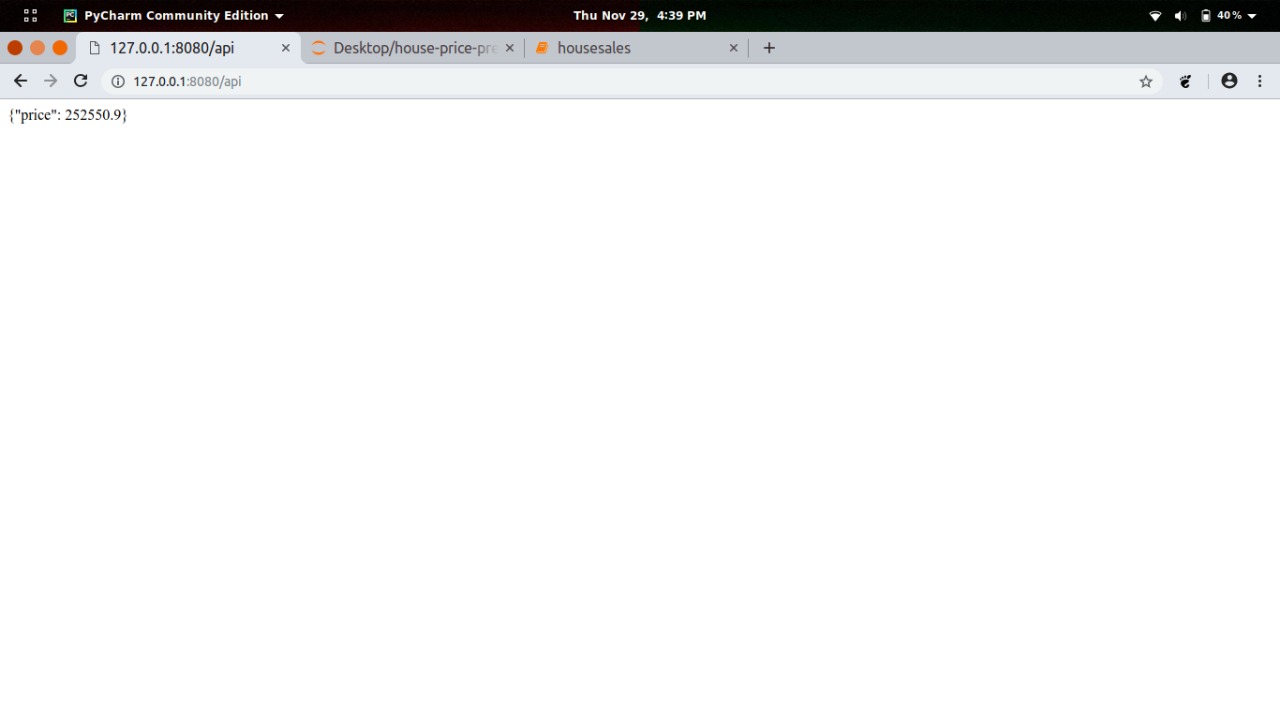


Figure 3.18 - Predicted Price

1. **Limitations**

The main aim of this system is to provide a general idea of where real estate market is headed. It is only limited to a very basic prediction model. Thus, it cannot be used as a critical decision making tool. By incorporating only limited number of parameters, there is a certain degree of accuracy. Since, there are many indeterminate parameters that affect house prices, each and every one of them cannot be taken into account. So, our model only depends upon the relationship of our selected parameters. At present, this system only perform analysis and prediction of prices of house located at the location which are present in our sample dataset.

**Chapter 4: Conclusion and Future Work**

**4.1. Conclusion**

We conclude our project with following points:

**4.1.1. Chapter overview**

This final chapter summarizes the achievements of the project as well as the challenges that were faced during its development. It also provides an outline of the possible improvements and their applicability.

**4.1.2. Project Achievement**

The proposed solution delivers an application, which will help people to predict their house prices. The application can also be useful for builders and developers who can use it to determine favourable and profitable places for construction and development of housing societies.

**4.1.3. Challenges**

Prior to implementing the system, one of the first challenges of the project was choosing suitable machine learning model because the selection has to consider the best prediction score. Other challenges were cleaning and preparing data set for analysis, as there were many flaws like null values, missing values etc.

**4.1.4. Concluding remarks**

Finally, this report demonstrates the achievements of the project and presents an assessment of the performance and reliability. Overall, the proposed solution has delivered an application that will help people minimize cost and effort regarding house price prediction.

With this, they can understand which features (ex. Number of bathrooms, location, etc.) influence the final price of the house.

As our prediction model relies solely on the public dataset, it may pale in comparison to these related works in terms of accuracy.

**4.2. Future Work**

Despite having produced a working application that met our initial requirements, various improvements can be made in the future. These include improvements we did not make due to limited time on the project, and suggestions provided by users after using our application.

Consider more factors affecting housing prices: .one main drawback of our prediction model is the lack of access to information. We would need to seek alternative sources of data besides publicly available dataset, which can provide us with information on the area, the number of rooms, etc for each property. Economic factors such as the yearly inflation rate or GDP growth can also be considered.

Optimise the prediction system through parallelised computations .: a major concern with the prediction system is the loading time Rather than performing the computations sequentially, we can use multiple processors and parallelise the computations involved, which can potentially reduce the training time and prediction time. Another way to approach this problem is to look for alternative APIs that allow us to produce heat maps of similar quality but require fewer data points, or faster libraries to implement GPs.

Add more functionalities into the application looking for location specific price prediction, generate heat maps, viewing factors affecting prices.

# **References**

1. *Anacondas*. (n.d.). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Anaconda\_(Python\_distribution)
2. *Flask*. (n.d.). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Flask\_(web\_framework)
3. *Gradient Boosting*. (n.d.). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Gradient\_boosting
4. *Jupyter Notebook*. (n.d.). Retrieved from Jupyter: http://jupyter.org/
5. *Kaggle*. (n.d.). Retrieved from Kaggle: https://www.kaggle.com/c/house-prices-advanced-regression-techniques/data
6. *Linear Regression*. (n.d.). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Linear\_regression
7. *Maruti Techlabs*. (n.d.). Retrieved from Maruti Techlabs: https://www.marutitech.com/machine-learning-predictive-analytics/
8. *Matplotlib*. (n.d.). Retrieved from Matplotlib: https://matplotlib.org/users/intro.html
9. *NumPy*. (n.d.). Retrieved from NumPy: http://www.numpy.org/
10. *Overview of India real estate*. (n.d.). Retrieved from ukessays: https://www.ukessays.com/essays/economics/overview-of-india-real-estate-economics-essay.php
11. *Pandas*. (n.d.). Retrieved from Panda data analysis library: https://pandas.pydata.org/
12. Wu, J. Y. (2017). *Housing Price prediction Using Support Vector.* San Jose State University.

**Personal Details**

* 1. **Anubhav Singh**

Enrolment No.-161B039

Branch-CSE

Ph. No.-9755677094

Email-ashappy09@gmail.com

* 1. **Mrigank Singh**

Enrolment No.-161B124

Branch-CSE

Ph. No.-9451219019

Email-janshu1997@gmail.com

* 1. **Sidharth Gupta**

Enrolment No.-161B235

Branch-CSE

Ph. No.-8171373696

Email-wakeupsid.gupta684@gmail.com