



TRIBHUVAN UNIVERSITY
INSTITUTE OF
ENGINEERING PULCHOWK
CAMPUS

REAL ESTATE PRICE ANALYSIS AND PREDICTION TOOLS FOR KATHMANDU VALLEY

By:

ANGAD GUPTA	070 BCT 506
ARUN KR AGRAWAL	070 BCT 509
BIKASH GUPTA	070 BCT 512
SHUBHAM KR AGRAWAL	070 BCT 546

A PROJECT SUBMITTED TO THE DEPARTMENT OF ELECTRONICS AND
COMPUTER ENGINEERING IN PARTIAL FULFILLMENT OF THEREQUIREMENT
FOR THE BACHELOR'S DEGREE IN COMPUTER ENGINEERING

DEPARTMENT OF ELECTRONICS AND COMPUTER
ENGINEERING LALITPUR, NEPAL

NOVEMBER, 2017

LETTER OF APPROVAL

The undersigned hereby certify that they have read, and recommended to the Institute of Engineering for acceptance, this project report entitled "Real Estate Price Analysis and Prediction Tools For Kathmandu Valley" submitted by Angad Gupta, Arun Kumar Agrawal, Bikash Gupta and Shubham Kumar Agrawal in partial fulfillment of the requirements for the Bachelor's Degree in Computer Engineering.

Supervisor

Dr. Arun Kumar Timalina
Department of Electronics & Computer
Engineering,
Institute of Engineering, Pulchowk Campus,
Tribhuvan University

Internal Examiner

Mr. Sharad Kumar Ghimire
Department of Electronics & Computer
Engineering,
Institute of Engineering, Pulchowk Campus,
Tribhuvan University

External Examiner

Mr. Mahesh Singh
Kathayat
Department of Electronics & Computer
Engineering,
Kathmadu Engineering College,

Head of Department

Dr. Dibakar Raj Pant
Department of Electronics & Computer
Engineering,
Institute of Engineering, Pulchowk Campus,
Tribhuvan University, Nepal

DATE OF APPROVAL:

COPYRIGHT

The authors have agreed that the Library, Department of Electronics and Computer Engineering, Pulchowk Campus, Institute of Engineering may make this report freely available for inspection. Moreover, the author has agreed that permission for extensive copying of this project report for scholarly purpose may be granted by the supervisors who supervised the project work recorded herein or, in their absence, by the Head of the Department wherein the project report was done. It is understood that the recognition will be given to the author of this report and to the Department of Electronics and Computer Engineering, Pulchowk Campus, Institute of Engineering in any use of the material of this project report. Copying or publication or the other use of this report for financial gain without approval of to the Department of Electronics and Computer Engineering, Pulchowk Campus, Institute of Engineering and author's written permission is prohibited.

Request for permission to copy or to make any other use of the material in this report in whole or in part should be addressed to:

Department of Electronics and Computer Engineering,
Institute of Engineering, Pulchowk Campus,
Tribhuvan University, Nepal

ACKNOWLEDGEMENT

We would like to express our gratitude to our supervisor, **Dr. Arun Kumar Timilsina** for guiding us throughout the project and helping us correct the errors.

We would like to express our sincere gratitude to the **Department of Electronics and Computer Engineering**, Pulchowk Campus for providing us the opportunity to do this project. We are extremely thankful to **Mr Dinesh Baniya Kshatri**, our Project Co-ordinator for providing us with invaluable guidance and support.

We are also thankful to Mr. Anup Devsaria for providing background knowledge and useful insights for our project.

Lastly, a very special thanks to our colleagues whose support and suggestions have been valuable for the successful completion of this project.

ABSTRACT

The “Real Estate Price Analysis and Prediction Tools For Kathmandu Valley” is a market analysis tool that aims to predict the current commercial values of land based on series of available data. The project tends to meet the requirements of ‘Major Project’ for B.E. Fourth Year. Real Estate market is a large repository data. By integrating statistical analysis with data mining techniques those data can be analyzed and patterns can be generated from them. The project indexes to generate technical indicators from the collected data. Then regression analysis is created using those indicators as the inputs using regression analysis.

This project analyzes the collected real estate data and the current commercial values of the land of Kathmandu valley are predicted. This project can be helpful to determine the current commercial value of the land of Kathmandu valley. The overall functionalities of the project along with its specifications, design, methodology, result analysis are described in this report.

Key Words: Multiple Linear Regression, Lasso, PLR, WLR, KNN, ANN, Commercial Price.

TABLE OF CONTENTS

LETTER OF APPROVAL.....	ii
COPYRIGHT.....	iii
ACKNOWLEDGEMENT.....	iv
ABSTRACT.....	v
TABLE OF CONTENTS.....	vi
LIST OF FIGURES.....	x
LIST OF TABLES.....	xii
LIST OF ABBREVIATIONS.....	xiii
1.INTRODUCTION.....	1
1.1 Background.....	2
1.2 Objectives.....	3
1.3 Problem Statement.....	4
1.4 Scope of the Project.....	5
2.LITERATURE REVIEW.....	6
3.METHODOLOGY.....	8
3.1 Data Collection.....	8
3.2 Dataset.....	8
3.3 Data Preprocessing.....	12
3.4 Data Analysis and Visualization.....	14
3.5 Feature Selection.....	14
3.6 Prediction.....	15
3.7 Prediction using Regression.....	15
3.8 Linear Regression.....	16
3.8.1 Error Function and Maximum Likelihood Solution.....	17

3.8.2 Regularization.....	18
3.8.3 Multiple Linear Regression:.....	19
3.9 Relevant theory.....	20
3.9.1 Ordinary Least Square Method:.....	20
3.9.2 Weighted Least Square Method:.....	20
3.9.3 Gradient Descent Method:.....	21
3.9.4 Min-Max Normalization:.....	21
3.10 Regression Metrics.....	22
3.10.1 Explained Variance Score.....	22
3.10.2 Mean Absolute Error.....	22
3.10.3 Mean Squared Error.....	22
3.10.4 Median Absolute Error.....	23
3.10.5 R^2 Score, the coefficient of determination.....	23
3.11 Tools and Technologies Used.....	24
3.11.1 Python.....	24
3.11.2 Pandas.....	25
3.11.3 Matplotlib.....	25
3.11.4 Scikit-learn.....	26
3.11.5 Django.....	27
3.11.6 Git.....	28
4.SOFTWARE DEVELOPMENTMODEL.....	29
5.REQUIREMENTS.....	32
5.1 Functional Requirements.....	32
5.2 Nonfunctional Requirements.....	32
6.SYSTEMMODEL.....	33
6.1 Use Case Diagram.....	33
6.2 Use Case Artifact.....	34
6.2.1 Use Case 1: View available data.....	35
6.2.2 Use Case 2: Visualize data.....	35

6.2.3 Use Case 3: Enter data for prediction.....	35
6.2.4 Use Case 4: View prediction outcome.....	36
6.2.5 Use Case 5: Collect data.....	36
6.2.6 Use Case 6: view system log.....	37
6.2.7 Use Case 7: Evaluate outcome and performance.....	37
6.2.8 Use Case 8: Update system.....	38
6.3 Sequence Diagram.....	39
6.4 Activity Diagram.....	40
6.5 GANTT CHART.....	41
7.RESULT, ANALYSIS AND VISUALIZATION.....	43
7.1 Result.....	43
7.2 Analysis.....	46
7.2.1 Analysis with WLR.....	48
7.3 Visualization.....	52
8. LIMITATIONS.....	55
9. CONCLUSION.....	56
10. REFERENCES.....	57

LIST OF FIGURES

i.	Dataset in CSV format	8
ii.	Dataset used for Prediction	11
iii.	Boxplot of Year	12
iv.	Boxplot of Road width	13
v.	Boxplot of Commercial rate	13
vi.	Agile Software Development Model	30
vii.	Use case Diagram	34
viii.	Use Case Artifact of User	35
ix.	Use Case Artifact of Data Source	37
x.	Use Case Artifact of Administrator	38
xi.	Sequence Diagram	39
xii.	Activity Diagram	40
xiii.	Gsntt Chart	41
xiv.	Primary Output Screen	45
xv.	Residual vs Predicted price plot	46
xvi.	Residual vs actual price plot	46
xvii.	Data in all Districts	46
xviii.	Type of path in all district	57
xix.	Shape of land level of Land in all Districts	57
xx.	High Tension & River Stream in all Districts	58

xxi.	Facilities in all district	58
------	----------------------------	----

LIST OF TABLES

i. Variables used in our prediction model	11
ii. Comparison with varying input	45
iii. Analysis of different algorithm of regression model	48
iv. Results for k-fold size=1	48
v. Results for k-fold size=2	48
vi. Results for k-fold size=3	48
vii. Results for k-fold size=4	49
viii. Results for k-fold size=5	49
ix. PLR model without interaction term	49
x. Coefficients of features using WLR	50
xi. Features of WLR after removing less significant features	51
xii. WLR model result after removing Kathmandu	51
xiii. Features of WLR after Removing Kathmandu	51
xiv. WLR model result after removing Lalitpur	52
xv. Features of WLR after removing Lalitpur	52
xvi. Significant features for WLR	53

LIST OF ABBREVIATIONS

- i. **RMSE:** Root Mean Square Error
- ii. **KNN:** K-Nearest Neighbor
- iii. **ANN:** Artificial Neural Network
- iv. **SVR:** Support Vector Regressor
- v. **WLR:** Weighted linear Regression
- vi. **PLR:** Polynomial Linear Regression
- vii. **MLR:** Multiple Linear Regression

1. INTRODUCTION

The real estate market is one of the most important markets in the modern economy because of the nature of the goods exchanged. Shelter is a fundamental human need and therefore there is a collective interest in pricing homes correctly. While there are other factors in play, an ill-informed party on either side of the transaction can be burned for non-trivial sums. Any insight into pricing lands properly and consistently would be of great interest to all parties involved.

A house is often the largest investment a person will make in their lifetime. The amount of money used to buy this house/Land is non-trivial and thus great care must be taken in not only choosing the right house, but making sure it's priced appropriately. This knowledge is usually held solely by real estate agents. If we can capture this domain knowledge by using openly accessible data, then this knowledge suddenly becomes accessible to the average citizen who can thus make informed decisions without relying on an expert who unfortunately may not always be acting in their best interest.

On the side of the real estate agents, being able to accurately price a land through use of machine learning algorithm with available data will offer new insight into what people are looking for in a land. They can thus spend more time focusing on creating successful pitches to sell the lands as well as make recommendations to clients.

1.1 Background

Real estate price study has been of great interest in research frontier. Discovering highly reliable and credible price indexes to track the rate of price appreciation over time and seizing the market trend have proven enormously beneficial for real estate professionals and highly sought after in finance and business application. Simply, price indexes could establish barometer of market information on risk and its turning points is paramount important to investment decision making, e.g. risk and return modeling, asset portfolio allocation strategy mapping price bubbles identification and evaluating investment performance.

Real estate price indexes are widely applied in financial and economic world and given its influential effects on deciding taxes, financial and monetary investment call by users like policy-makers, local authorities, real estate developers, mortgage lenders, brokers/consultants And other participants. It is envisaged that real estate price indexes study will continuously to be the center of attraction by researchers and practitioners.

1.2 Objectives

Real Estate market analysis deals with studying and trying to find meaningful conclusions based on the statistics of the data collected. The project aims to achieve following objectives:

- To predict the commercial rate of lands
- To determine technical indicators from the available data
- To compare the performance of different machine learning algorithms on real estate data

1.3 Problem Statement

The Real estate market is a dynamic system that is extremely hard to model with any reasonable accuracy. Technical analysis aims to predict current commercial price using volume information. It is based on the assumption that history repeats itself and that future market directions can be determined by examining historical price data. Thus, it is assumed that price trends and patterns exist that can be identified and utilized for profit. So this project presents a suitable method to properly analyze the historical real estate market data and extract out patterns from them so that the commercial prices can be predicted with the help of other inherent attributes.

1.4 Scope of the Project

The main goal of the project is to predict current commercial rates of land. An individual can now with the help of the knowledge of the price trends can make better investment decisions. It can be very useful to researchers, valuers, brokers, government and general public. General public who may or may not have adequate knowledge about the market, can get a lot of benefits from such analysis. They can know which areas are less risky and more suitable for investment currently. So, Real Estate Price Analysis and Prediction Tools can be a handy tool for valuers as well as public.

2. LITERATURE REVIEW

Real Estate market prediction is very popular topic in data mining. Many analysts and researchers have applied various data mining techniques to develop this type prediction model. The popular methods used for real estate market analysis and prediction are Linear Regression, Support Vector Machine, K-Nearest Neighbor (KNN). These methods have their own way of predicting the real estate values with different precision.

In an ideal real estate market, price would always signify the agreed value which is determined by both sellers and buyers. The deviations from the normal trend of real estate prices can be either positive or negative. Real estate markets are supposed to have a lower liquidity than financial markets.

A key category of real estate assets are housing assets and the prices of these often reflects the volatility of the market. Case *et.al*, 1990[13] studied single-family home prices amongst housing assets and found that returns on investment are location dependent. Their empirical study of different metropolitan areas globally showed that housing assets (Flats/single housing) demonstrate a better return on investments than rural areas. Their results are dependent on other factors such as indexes, population growth and material/construction costs.

Hutchison *et.al*, 2005[14] argues that the risk in real estate investment is attributed to the valuation of property carried out by evaluators. They suggest that an investor should consider many risks, notably valuation accuracy and valuation variance. Earlier both empirical and theoretical researchers analyzed risks and returns.

In order to minimize the risks of investment Nitsch, 2006[15] posited that choosing a prime location is the main component for minimizing investment risks. To substantiate his argument he developed a price model based on the structure, location and rent of various indexes in Germany.

However, this model does not assess the elements of risk fully indexes it ignores correlations. Even though Markowitz, 1959[16] modern portfolio theory is considered a major breakthrough in the financial world, some real estate researchers have reservations regarding the theory. Portfolio selection and development too often relies on past performance.

There is a common belief that the performance of real estate value depends on inflation. Wurtzebach et.al, 1991[17], studied the impact of inflation on the value of assets. They showed that real estate does provide an inflation hedge. They concluded that when market imbalance occurs, the risk increases and the returns suffer regardless of inflation. Rubens et.al, 1989 stated that the real estate hedge against inflation depends on the type of real estate.

3. METHODOLOGY

3.1 Data Collection

The data needed for the analysis and prediction were collected from reports of different renowned valuers (we have been requested not to disclose their indexes).

The required data's were manually taken out from the reports. The date wise data was stored in CSV (Comma Separated Values) format. The initial data indexes: Date of valuation, government rate, facilities available, area, road width and access location.

3.2 Dataset

The dataset in CSV format is shown in Table 3.1.

Date of Valuation	Road_width	LocationAccess	Commercialprice	Area	Road_type	Land_type	Government-rate	Places	Inflation_rate
16-Mar	8.20209974	170	19250000	5.5	Pitch	Commercial	800000	Lalitpur	0.067
16-Mar	8.20209974	170	12250000	3.5	Pitch	Commercial	800000	Lalitpur	0.067
17-Mar	5.90551181	50	23835000	6.81	Pitch	Commercial	700000	Lalitpur	0.067
14-Dec	9.5144357	100	1710000	1.14	Pitch	Commercial	375000	Kathmandu	0.067
24-May-15	5	105	1938750	2.75	Pitch	Commercial	250000	Kathmandu	0.0721
30-Sep-13	2	105	22968750	9.188	Pitch	Commercial	1200000	Kathmandu	0.0987
25-May-14	2	105	7000000	3.5	Pitch	Commercial	1200000	Kathmandu	0.0904
11-Jun-14	7.87401575	105	9792000	5.44	Pitch	Commercial	1200000	Kathmandu	0.0904
23-Sep-13	2	105	9750000	3.25	Goreto	Commercial	1000000	Kathmandu	0.0987

Figure 3.2.1: Dataset in CSV format

The description of the attribute used in dataset is as follows:

A. Date of Valuation:

This attribute contains year, month and day details on which the land property was evaluated. The attribute Year was obtained from it by subtracting it with Base Date (01/01/2012) and dividing the result obtained by 12.

B. Road_width:

This attribute gives the information about the width of the road near the land property.

C. Location_Access:

This attribute gives the distance of the places from the local check point (main place) around the property. This attribute was divided by 100 and new attribute *Location_Access* was created.

D. Government-rate:

This attribute gives the governmental price of the area where the property is located. This attribute was divided by 100000 and new attribute *Governmentrate* was formed.

E. Commercial price:

This attribute gives the commercial price of the property.

F. Area:

This attribute gives area of the property.

G. Commercial-rate:

The attribute Commercial price was divided by the attribute Area to obtain this attribute. It was divided by 100000. This is our target value.

H. Road_type:

This gives information about the type of road is around the property. There are five type of road namely ***Earthen, Pitch, Goreto, GravelledandPaved***. indexes it was categorical data type, so we created dummy variables and got five new different attributes namely ***Earthen, Pitch, Goreto, Gravelled and paved***.

I. Land_type:

This gives information about the type of land. Here, there are three categories of Land:

Residential

Commercial

Agricultural

It is a categorical type attribute, we created dummy variables which resulted in the addition of two new attributes *Commercial* and *Residential*.

J. Places:

This gives information about the city where the property is located. Here, the data are of property from city:

Kathmandu

Lalitpur

Bhaktapur

it is a categorical type attribute, we created dummy variables which resulted in the addition of two new attributes *Kathmandu* and *Lalitpur*.

K. Inflation_rate:

This gives information about inflation occurring in the country annually.

After the changes were made, our dataset was as shown in Table 3.2.

Year	Road_width	Location_Access	Governmentrate	Commercial-rate	Inflation_rate	Area	Earthen	Pitch	Goreto	Gravelled	paved	Commerci	Residential	Kathmandu	Lalitpur
4.166667	8.202099738	1.7	8	35	0.067	5.5	0	1	0	0	0	1	0	0	1
4.166667	8.202099738	1.7	8	35	0.067	3.5	0	1	0	0	0	1	0	0	1
5.166667	5.905511811	0.5	7	35	0.067	6.81	0	1	0	0	0	1	0	0	1
2.916667	9.514435696	1	3.75	15	0.067	1.14	0	1	0	0	0	1	0	1	0
3.333333	5	1.05	2.5	7.05	0.721	2.75	0	1	0	0	0	1	0	1	0
1.666667	2	1.05	12	25	0.0987	9.1875	0	1	0	0	0	1	0	1	0
2.333333	2	1.05	12	20	0.0904	3.5	0	0	1	0	0	1	0	1	0
2.416667	7.874015748	1.05	12	18	0.0904	5.44	0	1	0	0	0	1	0	1	0
1.666667	2	1.05	10	30	0.0987	3.25	0	0	1	0	0	1	0	1	0

Figure 3.2.2: Dataset used for Prediction

Table 3.1 Variables used in our prediction model

Independent Variables	Dependent variables
Year	Commercial-rate
Road_width	
Location_Access	
Governmentrate	
Earthen, Goreto, Gravelled, Pitch, paved	
Commercial, Residential	
Kathmandu, Lalitpur	

3.3 Data Preprocessing

The CSV files stores the collected data. The data was discontinuous in nature. There were many missing values in the data. There were as well the very few data of different places like Nuwakot, Kavrepalanchowk and also of Chitwan and Dhading. The missing data were replaced by the mean of the data and such noises were removed.

Further in the data except such noises, there were also the outliers which were removed by the mean of the data. For detecting the outliers, we have drawn boxplots of the each feature along with target feature as shown in Figure 3.1, Figure 3.2, Figure 3.3 and also used the statistical measures that is the data point must lie in the range $(Q1 - 1.5 * IQR, Q3 + 1.5 * IQR)$ where,

Q1 represents 1st quartile,

IQR represents Interquartile range and,

Q3 represents 3rd quartile

The data were also normalized using the Min-Max normalization technique.

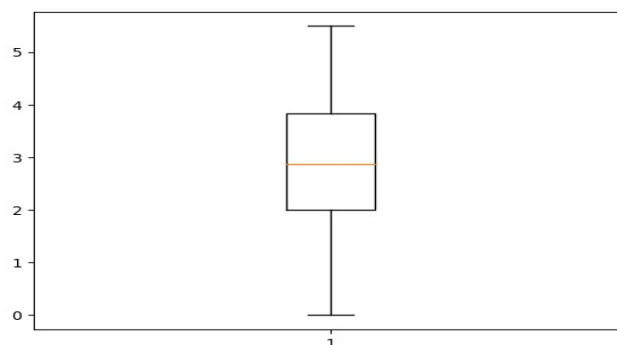


Figure3.3.1: Boxplot of Year

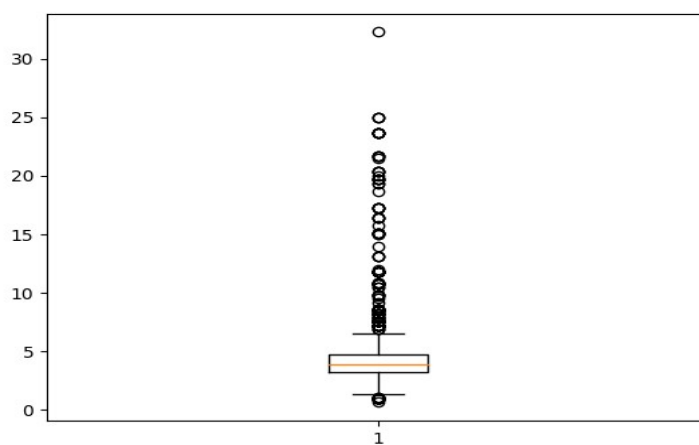


Figure 3.3.2: Boxplot of Road_width

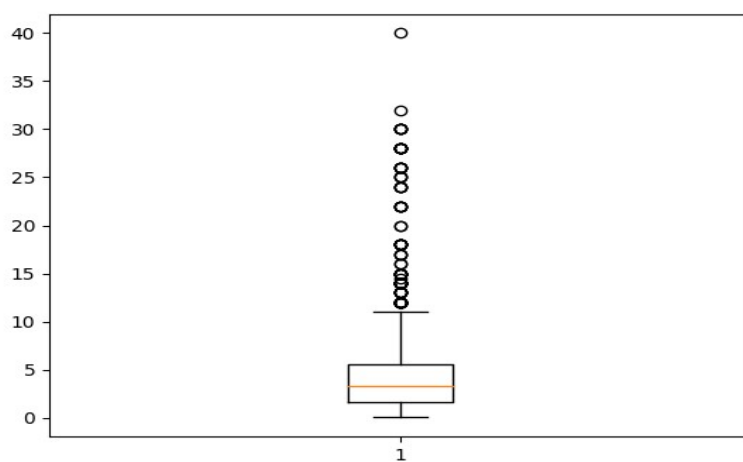


Figure 3.3.3: Boxplot of Commercial-rate

3.4 Data Analysis and Visualization

The real estate analysis consists of the analysis of the land or property. For this, we have collected data that helps to analyze the property. The features are:

Area of the property

Road_width of the road around the property

Type of road around the property

Distance of property from the local checkpoints (main places)

Date of evaluation of the property

Location of the property

Government rate of the property

High tension Line around the property

River around the property

Type of property

Inflation_rate on that year

We have drawn different graphs for analyzing our data. As shown Figure 3.1, Figure 3.2, and Figure 3.3, the boxplots of various attributes were drawn. Similarly, we have drawn bar graphs, line graphs and scatterplots for analyzing the data.

3.5 Feature Selection

Feature selection is the process of selecting a subset of relevant features (variables, predictors) for use in model construction. Feature selection techniques are used for simplification of models to make them easier to interpret by researchers/users, shorter training times, enhanced generalization by reducing over fitting. For this, we have observed the various graphs and has selected the features.

Similarly the features scores of each attributes was also calculated using L1 regularization method with Lasso estimator. L1 regularization adds a penalty to the loss function. Indexes each non-zero coefficient adds to the penalty, it forces weak features to have zero as coefficients. Thus L1 regularization produces sparse solutions. By using these methods the relevant features were selected from the available set of features.

3.6 Prediction

This system indexes to predict the information based on the historical data. Various artificial intelligence techniques are combined in order to generate the prediction model. Data analysis is performed in order to analyze the land prices based on different features. We implemented Regression analysis for the prediction of the land price.

3.7 Prediction using Regression

Machine learning is the study of algorithms that can learn from data and make predictions, by building a model from example inputs rather than following static instructions [1]. These algorithms are typically classified into three categories: supervised learning, unsupervised learning and reinforcement learning.

In supervised learning, the system is presented with example inputs and outputs, with the aim of producing a function that maps the inputs to outputs. Regression and classification problems are the two main classes of supervised learning [2]. Unsupervised learning is concerned with leaving the system to find a structure based on the inputs, hopefully finding hidden patterns. Examples include density estimation [3, 4], dimensionality reduction [5, 6] and clustering [7]. Lastly, reinforcement learning is the study of how a system can learn and optimize its actions in an environment to maximize its rewards [8], such as training a robot to navigate a maze.

Regression is a subset of supervised learning, where the outputs are continuous. The problem of predicting future housing prices can be considered a regression problem, indexes we are concerned with predicting values that can fall within a continuous range of

outputs. Through regression, we will be able to explore the relationship between the independent variables we have selected and the property price.

3.8 Linear Regression

The simplest regression model is linear regression, which involves using a linear combination of independent variables to estimate a continuous dependent variable [11]. While the model is too simple to accurately model the complexity of London housing market, there are many fundamental concepts in linear regression that many other regression techniques build upon. If we consider a model where y is the dependent variable and $x = (x_0, \dots, x_D)^T$ is the vector of D independent variables, a linear regression model can be formulated as follows:

$$y(x, w) = \sum_{j=0}^{M-1} w_j \phi_j(x) + \epsilon_j \dots \dots \dots (3.1)$$

Where,

$\phi_j(x)$ is a basis function with corresponding parameter w_j ,

ϵ_j is the indexes term,

M is the number of basis functions and

$w = (w_0, \dots, w_{M-1})^T$.

Assume the indexes terms ϵ_j are independent and identically distributed, where $\epsilon_j \sim N(0, \sigma^2)$. Basis functions can either be linear functions of the independent variables in the simplest case where $\phi_j(x) = x_j$, or extended to non-linear functions such as $\phi_j(x) = x_j^2, x_j^3, \sqrt{x_j}$. In most cases, $\phi_0(x)$ is set as 1, such that w_0 can act as a bias parameter to allow for fixed offset in the data [9]. Even though $y(x, w)$ can be a non-linear function of input x through the use of non-linear basis functions, the model is still linear with respect to w . Alternatively, we can also express (3.1) as follows:

$$y(x, w) = w^T \phi(x) + \epsilon \dots \dots \dots (3.2)$$

where,

$\phi = (\phi_0, \dots, \phi_{M-1})^T$ and

$\epsilon = (\epsilon_0, \dots, \epsilon_{M-1})^T$

Which simplifies the expression and allow us to code in vectored form, thereby avoiding the need for loops in computation.

3.8.1 Error Function and Maximum Likelihood Solution

In order to fit a linear regression model to the dataset, we need to minimize an error function. A common error function of choice is the sum-of-squares error function, which takes the form

$$E(w) = 1/2 \sum_{n=1}^N \{y_n - w^T \phi(x_n)\}^2 \dots \dots \dots (3.3)$$

Where, $w^T \phi(x_n)$ is the predicted value

y_n is the actual value and

N is the size of the dataset

From a probabilistic perspective, we can also maximize a likelihood function indexes the assumption of a Gaussian indexes model, and prove that it is equivalent to minimizing the error function [10]. Firstly, we express the uncertainty associated with the actual value y_n using a Gaussian distribution with inverse variance (precision) β , in the form

$$p(y_n | x, w, \beta) = N(y_n | w^T \phi(x_n), \beta^{-1} I) \dots \dots \dots (3.4)$$

Considering inputs $X = \{x_1, \dots, x_N\}$ and actual values $y = \{y_1, \dots, y_N\}$, the likelihood function can be constructed from this relationship

$$p(y | X, w, \beta) = \prod_{n=1}^N N(y_n | w^T \phi(x_n), \beta^{-1} I) \dots \dots \dots (3.5)$$

To determine the optimal values for w and β , we can maximize the likelihood function with respect to w . This can be done by taking the derivative of the logarithm of the likelihood function, which results in

$$\nabla w \log p(y \mid X, w, \beta) = \beta \sum_{n=1}^N \{y_n - w^T \phi(x_n)\} \phi(x_n)^T \dots\dots\dots (3.6)$$

If we take the derivative of the error function (3.3), we can see that it is equivalent to (3.6), thus proving that maximizing the likelihood function is equivalent to minimizing the error function. Before solving for w , let us define Φ , a $N \times M$ design matrix, whose elements are given by the basis functions applied to the matrix of values of the input variables, such that

$$\Phi = \begin{pmatrix} \Phi_0(x_1) & \Phi_1(x_1) & \dots & \Phi_{M-1}(x_1) \\ \Phi_0(x_2) & \Phi_1(x_2) & \dots & \Phi_{M-1}(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_0(x_N) & \Phi_1(x_N) & \dots & \Phi_{M-1}(x_N) \end{pmatrix} \dots\dots\dots (3.7)$$

By setting (3.6) to zero and solving for w , we are able to obtain a closed-form solution, which takes the form

$$w_{ML} = (\Phi^T \Phi)^{-1} \Phi^T y \dots\dots\dots (3.8)$$

This is known as the normal equation. Solving for w through the normal equation will give us the weight vector that best fits the data.

3.8.2 Regularization

Even though minimizing the error function can solve the problem of finding w , we have yet to solve the problem of choosing appropriate basis functions $\phi(x)$. Indexes basis functions are often in polynomials of the input variables, we can reduce the problem to finding the appropriate order M of the polynomial. While choosing a higher order M introduces flexibility into the model and is likely to result in a better fit to the dataset, using a high order polynomial can sometimes result in over-fitting. Over-fitting occurs when the model fits the indexes in the data instead of generalizing [1].

To curb the problem of over-fitting, a regularization term can be introduced to the error function. This takes the form of

$$E(w) = (1/2) \sum_{n=1}^N [y_n - w^T \phi(x_n)]^2 + (\lambda/2) w^T w \dots\dots\dots (3.9)$$

Where, λ is the regularization coefficient.

Adding a regularization helps to prevent over-fitting by penalizing large coefficient values.

When applied to the closed form solution, the new normal equation can be defined as

$$\begin{matrix} \Phi^T \Phi \\ \lambda I + \end{matrix} \begin{matrix} \end{matrix} \dots\dots\dots (3.10)$$

3.8.3 Multiple Linear Regression:

Multiple Regression Analysis refers to a set of techniques for studying the straight-line relationships among two or more variables. Multiple regression estimates the β 's in the equation.

$$y_j = \beta_0 + \beta_1 x_{1j} + \beta_2 x_{2j} + \dots + \beta_p x_{pj} + \epsilon_j \dots\dots\dots (3.11)$$

The X's are the independent variables (IV's). Y is the dependent variable. The subscript j represents the observation (row) number. The β 's are the unknown regression coefficients. Their estimates are represented by b's. Each β represents the original unknown (population) parameter, while b is an estimate of this β . The ϵ_j is the error (residual) of observation j.

Multiple regression analysis studies the relationship between a dependent (response) variable and p independent variables. The sample multiple regression equation is

$$\hat{y}_j = b_0 + b_1 x_{1j} + b_2 x_{2j} + \dots + b_p x_{pj} \dots\dots\dots (3.12)$$

The intercept, b_0 , is the point at which the regression plane intersects the Y axis. The b_i are the slopes of the regression plane in the direction of x_i . These coefficients are called the partial-regression coefficients. Each partial regression coefficient represents the net effect The i^{th} variable has on the dependent variable, holding the remaining X's in the equation constant.

A large part of a regression analysis consists of analyzing the sample residuals, e_j , defined as

$$e_j = y_j - \hat{y}_j \dots \dots \dots (3.13)$$

3.9 Relevant theory

3.9.1 Ordinary Least Square Method:

Ordinary least squares (OLS) or linear least squares is a method for estimating the unknown parameters in a linear regression model, with the goal of minimizing the sum of the squares of the differences between the observed (values of the variable being predicted) in the given and those predicted by a linear function of a set of explanatory variables.

The OLS estimator is consistent when the regressors are , and when the are and . Indexes these conditions, the method of OLS provides estimation when the errors have indexes . Indexes the additional assumption that the errors are , OLS is the .

For OLS β is estimated by formula in equation (2.1).

$$\hat{\beta} = (\Phi^T \Phi)^{-1} \Phi^T y \dots \dots \dots (2.1)$$

The OLS approach is suitable for estimating regression parameters when there is no variation in the precision of calculated dependent variables among gages, and the errors in independent of each other.

3.9.2 Weighted Least Square Method:

Unlike linear and nonlinear least squares regression, weighted least squares regression is not associated with a particular type of function used to describe the relationship between the process variables. Instead, weighted least squares reflects the behavior of the random errors in the model; and it can be used with functions that are either linear or nonlinear in the parameters. It works by incorporating extra nonnegative constants, or weights, associated with each data point, into the

fitting criterion. The size of the weight indexes the precision of the information contained in the associated observation. Optimizing the weighted fitting criterion to find the parameter estimates allows the weights to determine the contribution of each observation to the final parameter estimates. It is important to note that the weight for each observation is given relative to the weights of the other observations; so different sets of absolute weights can have identical effects.

3.9.3 Gradient Descent Method:

Gradient descent is a for finding the minimum of a function. To find a of a function using gradient descent, one takes steps proportional to the *negative* of the (or of the approximate gradient) of the function at the current point. If instead one takes steps proportional to the *positive* of the gradient, one approaches a of that function; the procedure is then known as gradient ascent.

Gradient descent is also known as steepest descent. However, gradient descent should not be confused with the for approximating integrals.

Gradient descent is a popular method in the field of because part of the process of machine learning is to find the highest accuracy, or to minimize the error rate, given a set of training data.^[1] Gradient descent is used to find the minimum error by minimizing a "cost" function.

3.9.4 Min-Max Normalization:

Minmax normalization is a normalization strategy which linearly transforms

x to $y = (x - \min) / (\max - \min)$, where \min and \max are the minimum and maximum values in X , where X is the set of observed values of x .

It can be easily seen that when $x = \min$, then $y = 0$, and

When $x = \max$, then $y = 1$.

This means, the minimum value in X is mapped to 0 and the maximum value in X is mapped to 1. So, the entire range of values of X from \min to \max are mapped to the range 0 to 1.

3.10 Regression Metrics

3.10.1 Explained Variance Score

The `explained_variance_score` computes the .

If \hat{y} is the estimated target output, y the corresponding (correct) target output, and Var is , the square of the standard deviation, then the explained variance is estimated as follow:

$$\text{explained_variance}(y, \hat{y}) = 1 - \frac{Var\{y - \hat{y}\}}{Var\{y\}}$$

3.10.2 Mean Absolute Error

The `mean_absolute_error` function computes , a risk metric corresponding to the expected value of the absolute error loss or l_1 -norm loss.

If \hat{y}_i is the predicted value of the i -th sample, and y_i is the corresponding true value, then the mean absolute error (MAE) estimated over n_{samples} is defined as

$$\text{MAE}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} |y_i - \hat{y}_i|.$$

3.10.3 Mean Squared Error

The `mean_squared_error` function computes , a risk metric corresponding to the expected value of the squared (quadratic) error loss or loss.

If \hat{y}_i is the predicted value of the i -th sample, and y_i is the corresponding true value, then the mean squared error (MSE) estimated over is defined as

$$\text{MSE}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} (y_i - \hat{y}_i)^2.$$

3.10.4 Median Absolute Error

The median absolute error is particularly interesting because it is robust to outliers. The loss is calculated by taking the median of all absolute differences between the target and the prediction.

If \hat{y}_i is the predicted value of the i^{th} sample and y_i is the corresponding true value, then the median absolute error (MedAE) estimated over n_{samples} is defined as

$$\text{MedAE}(y, \hat{y}) = \text{median}(|y_1 - \hat{y}_1|, \dots, |y_n - \hat{y}_n|).$$

3.10.5 R² Score, the coefficient of determination

The `r2_score` function computes R², the . It provides a measure of how well future samples are likely to be predicted by the model. Best possible score is 1.0 and it can be negative (because the model can be arbitrarily worse).

If \hat{y}_i is the predicted value of the i^{th} sample and y_i is the corresponding true value, then the score R² estimated over is defined as

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=0}^{n_{\text{samples}}-1} (y_i - \hat{y}_i)^2}{\sum_{i=0}^{n_{\text{samples}}-1} (y_i - \bar{y})^2}$$

Where,

$$\bar{y} = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} y_i.$$

3.11 Tools and Technologies Used

The various tools and techniques used in this project are described below:

3.11.1 Python

The whole project is written in Python Programming Language. Various libraries of python are used in the project. Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van Rossum during 1985-1990. Like Perl, Python source code is also available indexes the GNU General Public License (GPL). It is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other

Languages use punctuation, and it has fewer syntactical constructions than other languages. Python is interpreted language. Python is processed at runtime by the interpreter. Python is Interactive. Users can actually sit at a Python prompt and interact with the interpreter directly to write programs. Python is Object-Oriented: Python supports Object-Oriented style or technique of programming that encapsulates code within objects. Python is a Beginner's Language: Python is a great language for the beginner-level programmers and supports the development of a wide range of applications from simple text processing to WWW browsers to games. It supports functional and structured programming methods as well as OOP. It can be used as a scripting language or can be compiled to byte-code for building large applications. It provides very high-level dynamic data types and supports dynamic type checking. It supports automatic garbage collection. It can be easily integrated with C, C++, COM, ActiveX, CORBA, and Java.

3.11.2 Pandas

Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. Pandas is free software released under the three-clause BSD license. The name is derived from the term "panel data", an econometrics term for multidimensional structured data sets. Python has long been great for data munging and preparation, but less so for data analysis and modeling. Pandas helps fill this gap, enabling users to carry out the entire data analysis workflow in Python without having to switch to a more domain specific language like R.

Combined with the excellent Python toolkit and other libraries, the environment for doing data analysis in Python excels in performance, productivity, and the ability to collaborate. The Pandas module uses objects to allow for data analysis at a fairly high performance rate in comparison to typical Python procedures. With it, users can easily read and write from and to CSV files, or even databases. From there, users can manipulate the data by columns, create new columns, and even base the new columns on other column data. Next, pandas can assist in data visualization using Matplotlib. Matplotlib is a great module even without the teamwork of Pandas, but Pandas comes in and makes intuitive graphing with Matplotlib breeze.

3.11.3 Matplotlib

Matplotlib was used in the project to draw various charts and plots of the market data. Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like wxPython, Qt, or GTK+. There is also a procedural "pylab" interface based on a state machine (like OpenGL), designed to closely resemble that of MATLAB. SciPy makes use of matplotlib. The pylab interface makes matplotlib easy to learn for experienced MATLAB users, making it a viable alternative to

MATLAB as a teaching tool for numerical mathematics and signal processing. Matplotlib tries to make easy things easy and hard things possible. It can generate plots, histograms, power spectra, bar charts, error charts, scatterplots, etc. with just a few indexes of code

Some of the advantages of the combination of Python, NumPy, and Matplotlib over MATLAB include:

- Based on Python, a full-featured modern object-oriented programming language suitable for large-scale software development
- Free, open source, no license servers
- Native SVG support

3.11.4 Scikit-learn

Scikit-learn (formerly scikits.learn) is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support vector machines, random forests, gradient boosting, k-means and DBSCAN, is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy. Scikit-learn was initially developed by David Cournapeau as a Google summer of code project in 2007. Later Matthieu Brucher joined the project and started to use it as a part of his thesis work. In 2010 indexes got involved and the first public release (v0.1 beta) was published in late January 2010. Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python. It is licensed under a permissive simplified BSD license and is distributed across many Linux distributions, encouraging academic and commercial use.

Some popular groups of models provided by scikit-learn include:

- Clustering: for grouping unlabeled data such as K-Means.
- Cross Validation: for estimating the performance of supervised models on unseen data.

- Datasets: for test datasets and for generating datasets with specific properties for investigating model behavior.
- Dimensionality Reduction: for reducing the number of attributes in data for summarization, visualization and feature selection such as Principal component analysis.
- Ensemble methods: for combining the predictions of multiple supervised models.
- Feature extraction: for defining attributes in image and text data.
- Feature selection: for identifying meaningful attributes from which to create supervised models.
- Parameter Tuning: for getting the most out of supervised models.
- Manifold Learning: For summarizing and depicting complex multi-dimensional data.
- Supervised Models: a vast array not limited to generalize linear models, discriminate analysis, Naive Bayes, lazy methods, neural networks, support vector machines and decision trees.

3.11.5 Django

Django was used in the project to design the web interface. Django is a free and open-source web framework, written in Python, which follows the model–view–controller (MVC) architectural pattern. It is maintained by the Django Software Foundation (DSF), an independent organization established as a 501(c)(3) non-profit. Django's primary goal is to ease the creation of complex, database-driven websites. Django emphasizes reusability and "plug ability" of components, rapid development, and the principle of don't repeat yourself.

Python is used throughout, even for settings files and data models. Django also provides an optional administrative create, read, update and delete interface that is generated dynamically through introspection and configured via admin models.

3.11.6 Git

Git was used as a version control system to collaborate among the team members. Git is a version control system that is used for software development and other version control tasks. As a distributed revision control system it is aimed at speed, data integrity, and support for distributed, non-linear workflows. Git was created by Linus Torvalds in 2005 for development of the Linux kernel, with other kernel developers contributing to its initial development. The Git feature that really makes it stand apart from nearly every other SCM out there is its branching model.

Git allows and encourages you to have multiple local branches that can be entirely independent of each other. The creation, merging and deletion of those indexes of development takes seconds.

4. SOFTWARE DEVELOPMENTMODEL

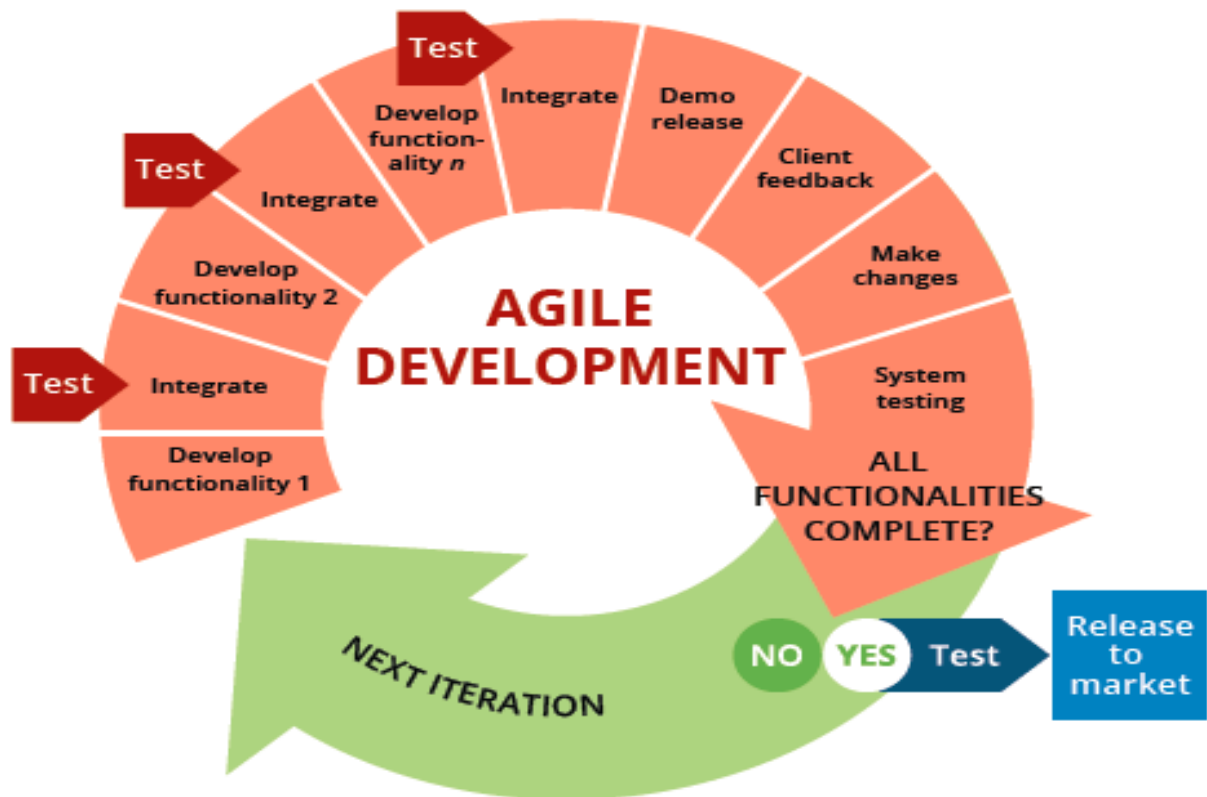


Figure 4.1 Agile Software Development Life-cycle

The Agile programming improvement life cycle depends on the iterative and incremental process models, and centers upon flexibility to changing item prerequisites and upgrading consumer loyalty through fast conveyance of working item elements and customer cooperation. Light-footed strategies basically center after separating the whole item into littler, effortlessly developable, "shippable" item highlights created through "incremental" cycles known as "sprints".

An Agile software life cycle is much different as compared to traditional software development frameworks like Waterfall. In Agile, more emphasis is given to sustained and quick development of product features rather than spending more time during the initial project planning, and analyzing the actual requirements. The Agile team develops the product through a series of sprints.

Besides development activity, other aspects pertaining to development such as product analysis, designing the product features, developing the functionality, and testing the development for bugs are also carried out during the sprints. The incremental cycles should always produce a “shippable” product release that can be readily deployed.

The Manifesto for Agile Software Development is based on twelve principles:

1. Customer satisfaction by early and continuous delivery of valuable software
2. Welcome changing requirements, even in late development
3. Working software is delivered frequently (weeks rather than months)
4. Close, daily cooperation between business people and developers
5. Projects are built around motivated individuals, who should be trusted
6. Face-to-face conversation is the best form of communication (co-location)
7. Working software is the primary measure of progress
8. Sustainable development, able to maintain a constant pace
9. Continuous attention to technical excellence and good design
10. Simplicity—the art of maximizing the amount of work not done—is essential
11. Best architectures, requirements, and designs emerge from self-organizing teams
12. Regularly, the team reflects on how to become more effective, and adjusts accordingly

The team members worked simultaneously while developing the product features in daily sprints. At the end of each sprint, a working product feature(s) was developed and presented to the supervisor for verification purposes. Once the supervisor Okayed the development, his further

opinions were carefully noted to improve upon the current product development cycle. The entire process was repeated through sprints until all the constituent product features were developed.

Some of the pros and cons of agile methodology are given below:

Pros

- It is a very realistic approach to software development.
- It promotes teamwork and cross-training.
- Functionality can be developed rapidly and demonstrated.
- Resource requirements are minimum.
- Suitable for fixed or changing requirements.
- It enables concurrent development and delivery within an overall planned context.

Cons

- It is not suitable for handling complex dependencies.
- There is more risk of sustainability, maintainability and extensibility.
- An overall plan, an agile leader and agile Project Management practice is a must without which it will not work.
- Strict delivery management dictates the scope, functionality to be delivered and adjustments to meet the deadlines

5. REQUIREMENTS

The various functional and nonfunctional requirements of the system are given below:

5.1 Functional Requirements

- The system shall provide visualization of the Real estate market's data.
- The system shall provide comparative visualization of different property.
- The system shall predict the current commercial value of a property based on data collected.
- The system shall analyze the price of the property based on the parameters (like facilities available, road width, governmental rate).
- The system shall analyze the trend of a properties valuation.

5.2 Nonfunctional Requirements

- The data for analysis should contain the government rate of the property.
- The discontinuities in the data should be removed by forward and backward interpolation methods.
- The prediction system should be dynamic enough to easily adapt with the daily increase in the data.
- The visualization system should be adaptive to dynamically add new data to the visualization

6. SYSTEMMODEL

6.1 Use Case Diagram

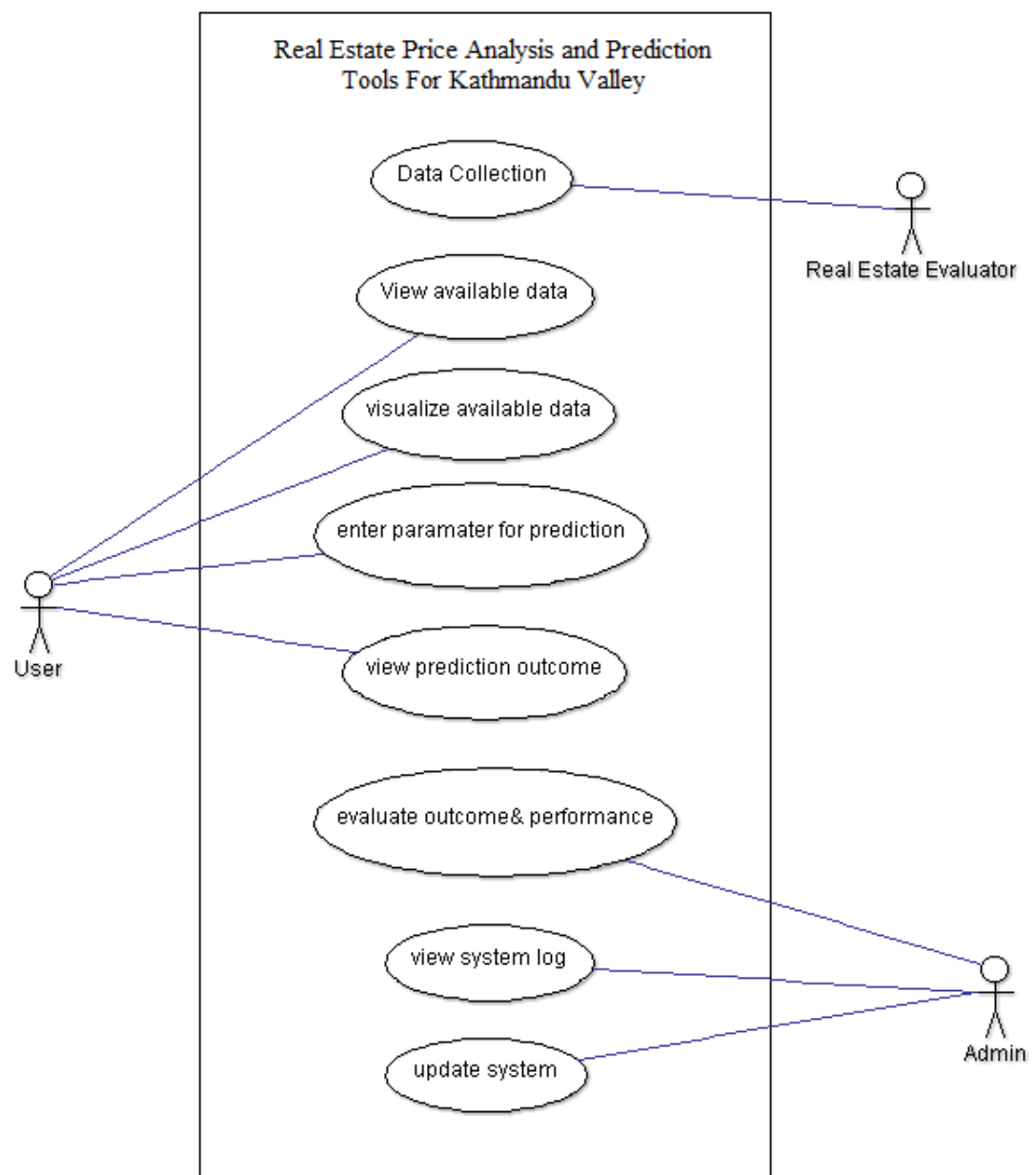


Figure 6.1 Use case diagram

The use case diagram above shows the interaction of the system with external actors. Three actors are defined in the diagram: User, Administrator and real estate evaluator. The User is the primary business actor which interacts with the system by performing tasks like viewing the data, viewing the visualizations and performs and views the predictions. The real estate evaluator are the registered evaluators interacting with the system by providing the data. The system administrator is the primary system actor which interacts with the system by performing system level tasks like viewing system log, evaluating prediction outcome and updating the system.

6.2 Use Case Artifact

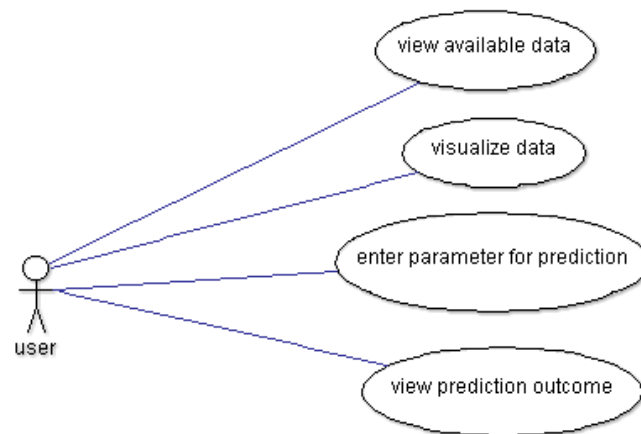


Figure 6.2 Use Case Artifact of User

6.2.1 Use Case 1: View available data

This use case describes the event of user requesting the system for viewing available data.

Actor: User (role: primary business actor)

Artifact	
1	user requests to view data
2	user receives information about data form system

Requirements: User should be able to view available data.

6.2.2 Use Case 2: Visualize data

This use case describes the event of user requesting to visualize the data in graphs: pie charts, histograms, bar graph, stem plot, time-series graphs etc. and to view different indicators.

Artifact	
1	user requests to visualize the data or market indicators or any other parameters
2	user is provided to appropriate visualization and data

Requirements: User should be able to visualize data in graphs and view values of different market indicators.

6.2.3 Use Case 3: Enter data for prediction

This use case describes the event of user entering the required data entries necessary for the system to predict the outcome.

Actor: User (role: primary business actor)

Artifact	
1	user enters the data required for prediction
2	User views the predicted outcome

Requirements: User should be able to view the outcome of prediction system.

6.2.4 Use Case 4: View prediction outcome

This use case describes the event of user requesting the system to view the outcome produced by the prediction model using different prediction tools and compare and analyze the outcomes produced by different tools and methods.

Actor: User (role: primary business actor)

Artifact	
1	user requests to view the outcome of prediction system
2	user receives the result of prediction outcome

Requirements: User should be able to view the outcome of prediction system

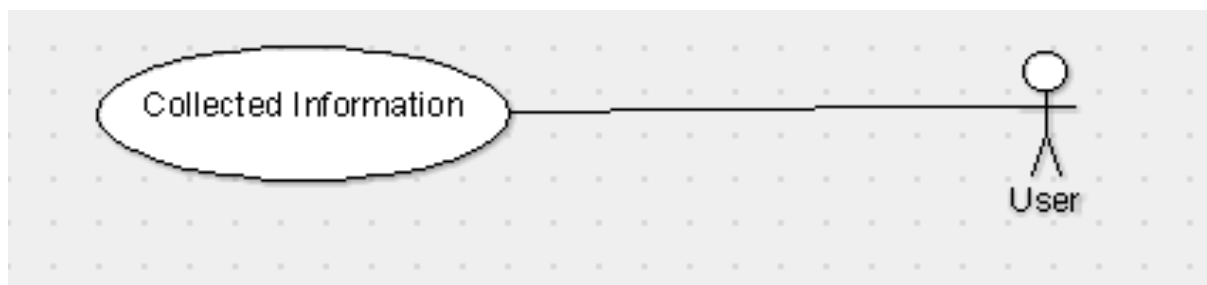


Figure 6.3: Use Case Artifact of Data Source

6.2.5 Use Case 5: Collect data

This use case describes the interaction between the system and data source (real estate evaluator).

Actor: Data Source (role: external service provider)

Artifact	
1	system initiates the interaction with external data source for collecting data

Requirements: system should be able to collect the relevant data from source

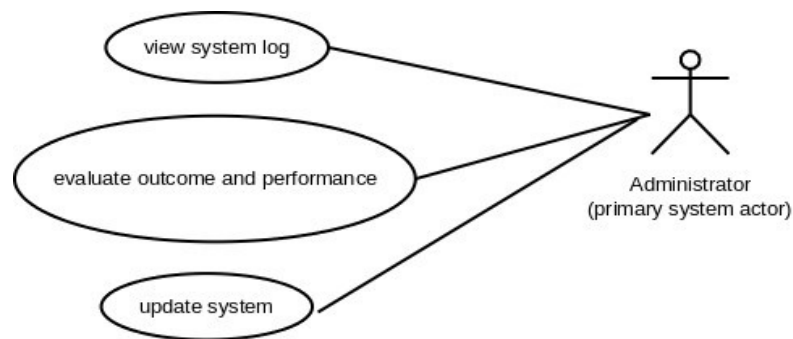


Figure 6.4 Use Case Artifact of Administrator

6.2.6 Use Case 6: view system log

This use case describes the event of system administrator requesting the system to view the log file of system activities.

Actor: Administrator (role: primary system actor)

Artifact	
1	Administrator requests to view the system log
2	Administrator is presented with system log file by the system

Requirements: Administrator should be able to view log file

6.2.7 Use Case 7: Evaluate outcome and performance

This use case describes the event of system administrator requesting the system to view the outcome and performance of the prediction model, efficiency of prediction tools and training and testing models.

Actor: Administrator (role: primary system actor)

Artifact	
1	Administrator requests to view the performance and efficiency of prediction system
2	Administrator is presented with relevant information

Requirements: Administrator should be able to view the system performance and efficiency.

6.2.8 Use Case 8: Update system

This use case describes the event of system administrator updating the system models(data representation model, prediction model, evaluation model, training and testing model) and redesigning the system and adding new features.

Actor: Administrator (role: primary system actor)

Artifact	
1	Administrator updates the system models and add new features on the system
2	System is updated

Requirements: Administrator should be able to update existing models, redesign the system model and add new features.

6.3 Sequence Diagram

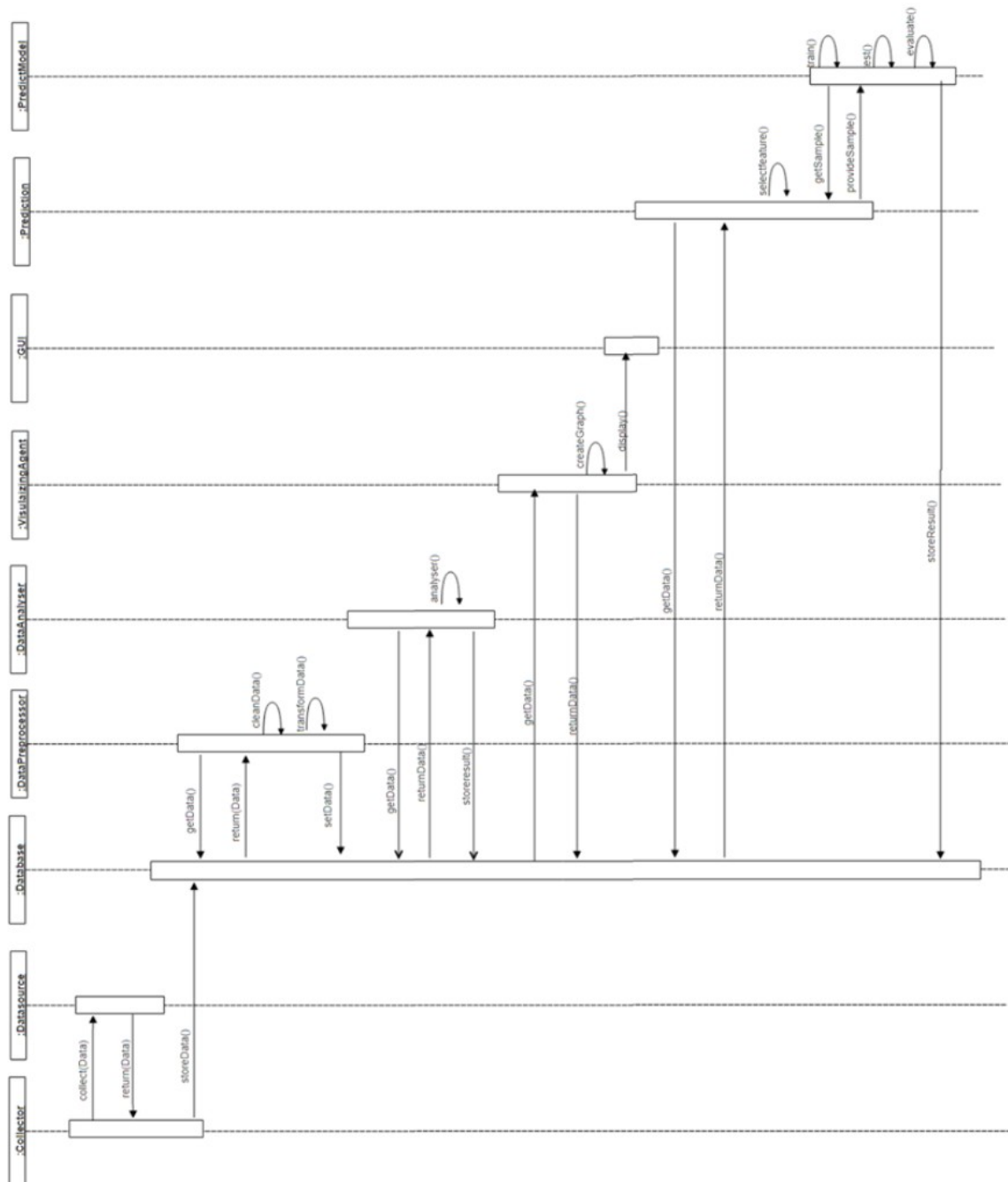


Figure 6.5 Sequence Diagram

The sequence diagram is an interaction **diagram** that shows how objects operate with one another and in what order. A **sequence diagram** shows object interactions arranged in time **sequence** the time ordering of interactions between the classes of the system. Basically the data collector collects the data, stores the data in the database. The data preprocessor cleans and transforms the data. Then the data analyzer takes the clean data and generates technical indicators from them. Visualizing agent creates and saves charts.

6.4 Activity Diagram.

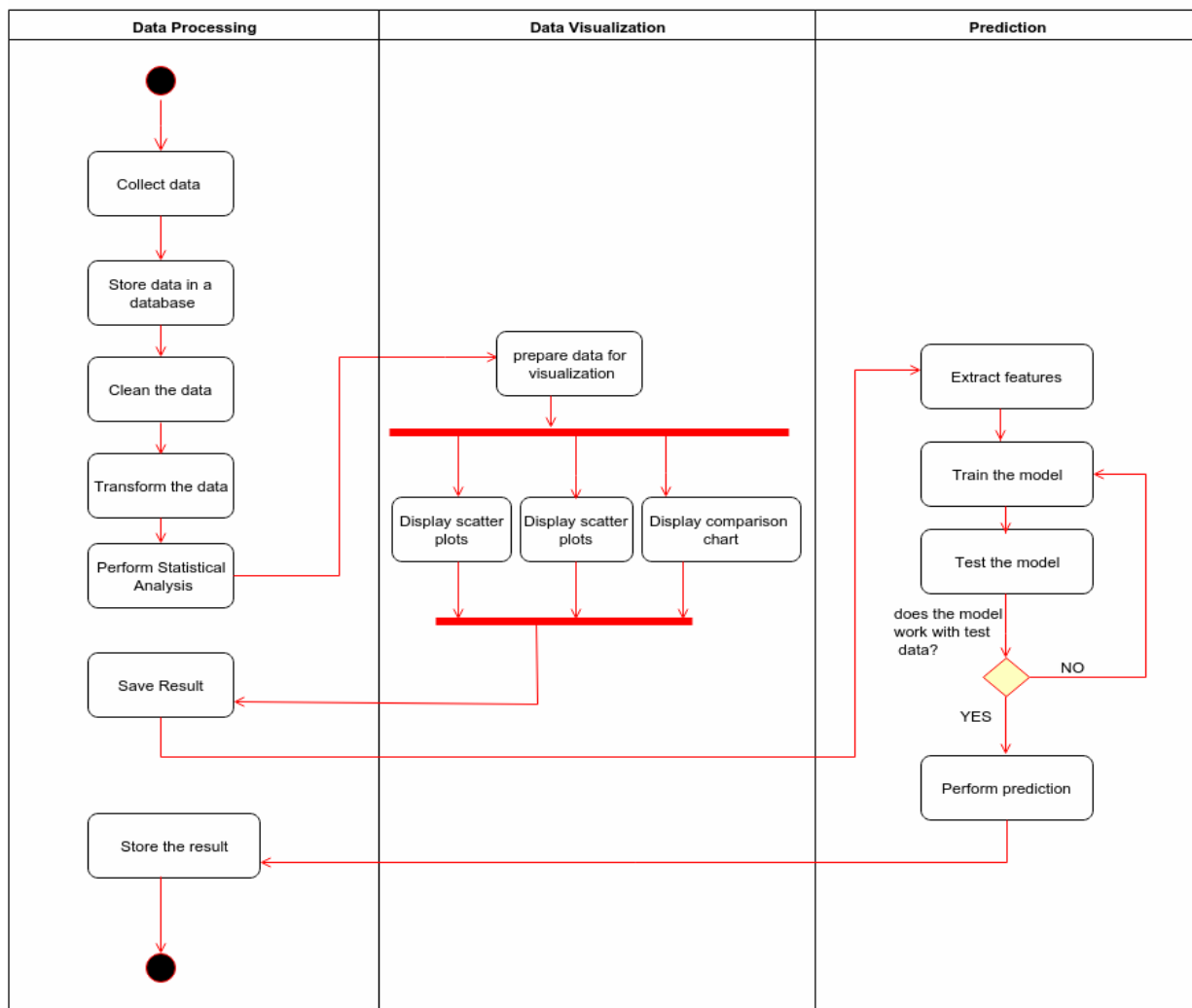


Figure 6.6. Activity Diagram

Activity diagram of the system is shown above which is basically a flow chart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The diagram describes three swimlanes: Data Preprocessing, Data Visualization and Prediction. The flow of the system is represented from one activity to another. The system goes through the activities like Collect Data, Store Data, etc and finally reaches the Perform prediction activity. The transition of the system from one swimlane to another is also depicted in the diagram.

6.5 GANTT CHART




Figure 6.7. Gant Chart

7. RESULT, ANALYSIS AND VISUALIZATION

7.1 Result

The project aims to predict the current commercial rate. Below is the simple output screen of the system we designed with simple user web app interface.

localhost:8000/housing/predictreg/

HOME PREDICTION REGRESSION  DATA VISUALIZE TEAM CONTACT

PREDICTION (REGRESSION)

Year:

Road Width (meter): Location access (meter): Government Rate (Rs.):

Road_type: Land_type:

FOR TEST DATA

SSE=43976.0193739
R_Squared=0.422017042901
MSE=3387.54556104
RMSE=11.083232513299755
F_value=15.9254317946
Adjusted_R_Squared=0.380040627022

FOR TRAIN DATA

SSE=54044.1235394
R_Squared=0.730972624042
MSE=6192.65049124
RMSE=8.528640966887522
F_value=79.9635712988
Adjusted_R_Squared=0.721558476297

Predicted value

Rs.17.3284475506 Lakhs

Figure 7.1: Primary Output Screen

The values for the parameters are shown up in a tabular form which shows the variation in the predicted output with the variation in the input.

Table 7.1 Comparison with varying input

Road Width (in meter)	Location Access (in meter)	Governmenta l rate (in Rs)	Road type	Residential/ Commercial	Predicted commercia l rate(in lakhs)
5	100	500000	Earthe n	Residential	8.80
5	100	500000	Pitch	Residential	11.6
5	100	500000	Earthe	Commercial	13.25

			n		
5	100	500000	Pitch	Commercial	16

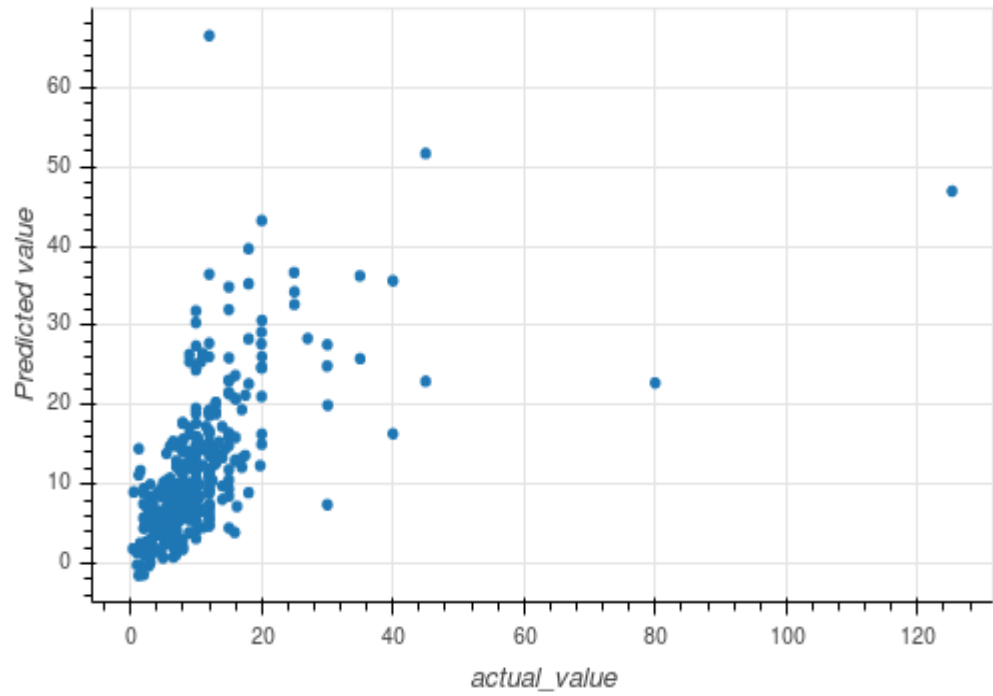


Figure 7.2: Predicted vs Actual price value

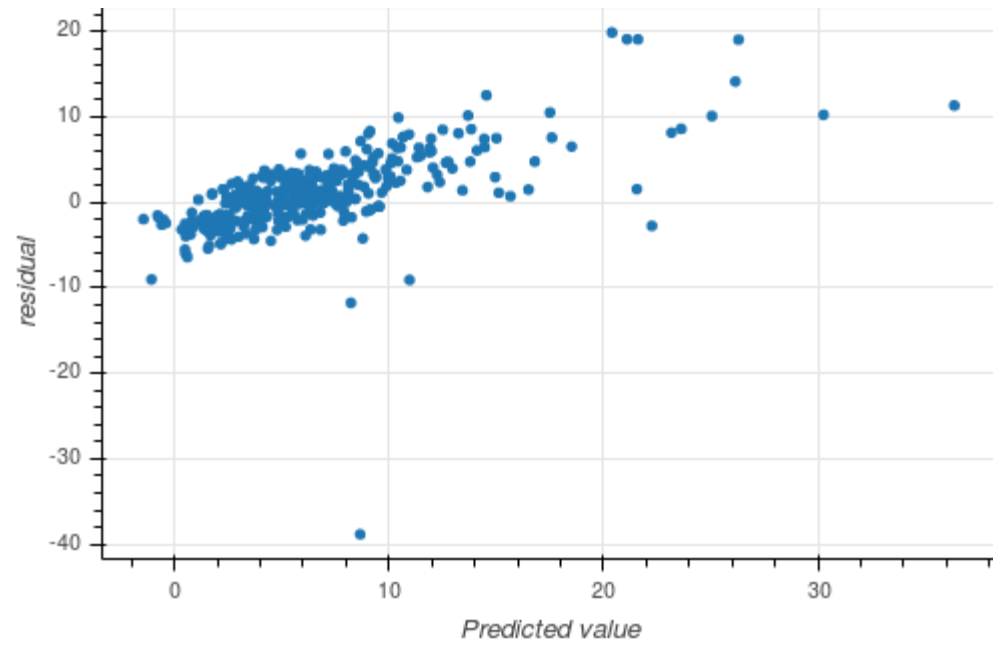


Figure 7.3: Residual vs Predicted price value

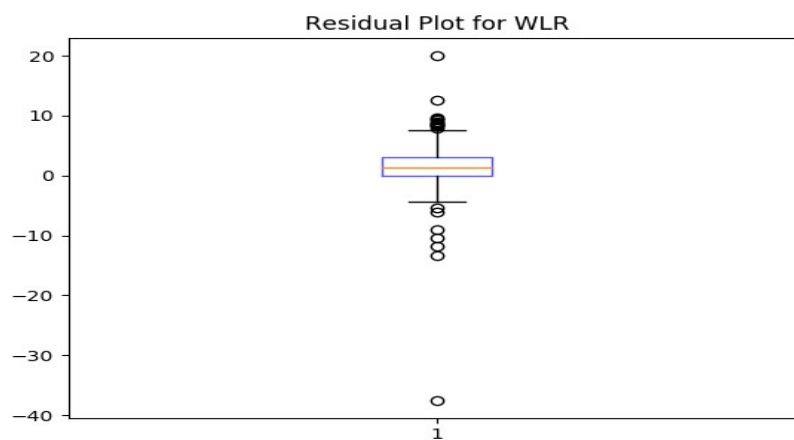


Figure 7.4: Residual Plot for WLR

7.2 Analysis

The K-Cross-Validation method was implemented in the project. The different results were obtained by changing the value of K-fold size. We have varied the K-fold-size from 1 to 5.

For K-fold size=1, our dataset was divided into 30% test data and 70% train data. The results are listed in the table 7.2.

Table 7.2: Results for k-fold size=1

Regression model Algorithm	R-squared value	RMSE value	F-value	p-value
WLR	0.63416	31.796	11.5565	1.9519e-15
MLR	0.7444306	20.6605	19.4188	6.2879e-23
PLR	0.77663	3119.1634	6.519	7.05825e-15
MLR Gradient	0.68532	26.4120	8.5268	9.68216e-12
Lasso	0.51745	19.14365	17.17082	1.3653e-20

For K-fold size=2, our dataset was divided into 50% test data and 50% train data. The results are listed in the table 7.3.

Table 7.3: Results for k-fold size=2

Regression model Algorithm	R-squared value	RMSE value	F-value	p-value
WLR	0.619701	19.9017	60.8971	9.81058e-107
MLR	0.70423	14.9959	89.2099	2.3691e-137
PLR	0.544	8626.969	16.0323	9.51345e-69
MLR Gradient	.63901	5.36459	66.204034	5.2789e-113
Lasso	0.309	15.50737	59.8163	3.251e-101

For K-fold size=3, our dataset was divided into 33% test data and 67% train data. The results are listed in the table 7.4.

Table 7.4: Results for k-fold size=3

Regression model Algorithm	R-squared value	RMSE value	F-value	p-value
WLR	0.65614	8.60278	46.9422	1.5708e-75
MLR	0.5296	8.3160	27.69963	6.7822e-51
PLR	0.50797	660.2485	8.8788322	1.1931e-31
MLR Gradient	0.603857	4.19627	37.498875	2.323e-64
Lasso	.332	7.615924	41.16867	2.62763e-66

For K-fold size=4, our dataset was divided into 25% test data and 75% train data. The results are listed in the table 7.5.

Table 7.5: Results for k-fold size=4

Regression model Algorithm	R-squared value	RMSE value	F-value	p-value
WLR	0.6625	7.26811	111.2462	1.7836e-189
MLR	0.510	4.61019	18.94935	6.079e-33
PLR	0.405	110.70036	4.23137	2.896e-12
MLR Gradient	0.604	5.0805	27.7993	3.02628e-53
Lasso	0.3030	4.280	29.990	1.78762e-46

For K-fold size=5, our dataset was divided into 20% test data and 80% train data. The results are listed in the table 7.6.

Table 7.6: Results for k-fold size=5

Regression model Algorithm	R-squared value	RMSE value	F-value	p-value
WLR	0.6501	3.567	26.6335	8.7637e-39
MLR	0.6041	7.538	21.8794	6.4307e-35
PLR	0.62077	477.131	7.77	2.4469e-24
MLR Gradient	0.5971	5.79085	21.25089	1.25286e-34
Lasso	0.35865	7.504	25.727	3.4106e-38

From the result obtained by changing k-fold-size from 1 to 5, we came to conclusion to have k-fold size 3.

The R_squared value of WLR was not varying in each case and also they have the minimum p-value in most cases, so we have decided to do our further analysis using them.

The polynomial linear model has shown mush worst result among the other models so we removed the interaction term from it and again observed the result of it alone which is shown in Table 7.7.

Table 7.7: PLR model without interaction term

Regression model Algorithm	R-squared value	RMSE value	F-value	p-value
PLR	0.507	660.248	8.8788	1.1931e-33

7.2.1 Analysis with WLR

The coefficient of the features for WLR are listed in the Table 7.8 and the intercept value along with its standard error, t_test value and p_values are **2.23566688874**, **3.38641345433**, **0.660187221344** and **0.50952949125** respectively.

Table 7.8: Coefficients of features using WLR

Features	Coefficients	Standard_error	T_test value	p-values
<i>Year</i>	0.14257054059	0.235395290698	0.60566437066	0.054509545162
<i>Road_width</i>	-0.1900748050	0.194987433259	-0.9748054110	0.033027046240
<i>Location_Access</i>	-0.1144220374	0.058892321856	-1.9429024671	0.052758488391
<i>Inflation_rate</i>	-3.6394505487	24.6763186387	-0.1474875811	0.882824588716
<i>Governmentrate</i>	0.8395153694	0.147456148998	5.69332221946	2.478559979e-08
<i>Earthen</i>	1.1972429968	0.063820949217	18.7594044198	0.0
<i>Goreto</i>	0.3966999275	2.02934827756	0.19548144198	0.845119459032
<i>Pitch</i>	2.2661477429	1.99746172095	1.13451372767	0.025728682205
<i>Gravelled</i>	2.2875524877	2.21992370616	1.03046446208	0.030344062007
<i>paved</i>	2.3051801328	2.25245084807	1.02340973827	0.030675843602
<i>Commerical</i>	3.3298318668	3.48878704952	0.95443826737	0.03404622729
<i>Residential</i>	2.0699226220	1.82974097599	1.13126538084	0.025864941059
<i>Kathmandu</i>	0.1809313025	1.648171516	0.10977698668	0.912643588622
<i>Lalitpur</i>	0.3654732959	0.574127802953	0.6365713244	0.524783377545

From the Table 7.9, it shows that Goreto, and Inflation_rate, has the highest p-value. So, they must be removed. After removing them, our result is shown in Table 7.9.1.

Regression model Algorithm	R-squared value	RMSE value	F-value	p-value
WLR	0.64828	8.2298	52.601	1.589e-75

The RMSE value of WLR has been decreased and the p-value has also become much smaller.

The coefficient factor of the features of WLR after making removing the less significant attribute is shown in Table 7.10.

Table 7.10: Features of WLR after removing less significant features

Features	Coefficients	Standard_error	T_test value	p-values
<i>Year</i>	0.15933851491	0.189486254294	0.8408974862	0.040092867595
<i>Road_width</i>	-0.1914972331	0.193066272339	-0.991873053	0.032188404095
<i>Location_Access</i>	-0.1166620020	0.058234716878	-2.003306760	0.045846622408
<i>Inflation_rate</i>	0	0	0	0
<i>Governmentrate</i>	0.84000670667	0.145871041963	5.7585569785	1.7395744e-08
<i>Earthen</i>	0.82407433893	0.063407052010	12.996572349	0.0
<i>Goreto</i>	0	0	0	0
<i>Pitch</i>	1.88574814613	0.622418774823	3.0297096141	0.00261352250
<i>Gravelled</i>	1.90946955246	1.03033682899	1.8532478881	0.064613718011
<i>paved</i>	1.93331874347	1.21036449158	1.5973029256	0.011102073500
<i>Commerical</i>	3.4500211985	2.86260606327	1.2052029242	0.022886714926
<i>Residential</i>	2.16549183779	1.69084241167	1.2807177196	0.020106582055
<i>Kathmandu</i>	0.18225605065	1.54524549726	0.1179463399	0.906171844879
<i>Lalitpur</i>	0.39202373493	0.568608711208	0.6894437725	0.049096044309

Here again, we find out that Kathmandu is the insignificant feature. So after removing it the result are shown in Table 7.11 and 7.12.

Table 7.11 WLR model result after removing Kathmandu

Regression model Algorithm	R-squared value	RMSE value	F-value	p-value
WLR	0.64858	8.2198	57.2148	1.619e-76

Table 7.12 Features of WLR after Removing Kathmandu

Features	Coefficients	Standard_error	T_test value	p-values
<i>Year</i>	0.1608189455	0.189271044796	0.8496753729	0.0396035016934
<i>Road_width</i>	-0.192136190	0.192956537041	-0.995748542	0.0319999377891
<i>Location_Access</i>	-0.118544707	0.0578427372138	-2.049431144	0.0410988904885
<i>Inflation_rate</i>	0	0	0	0
<i>Governmentrate</i>	0.8464466198	0.143030101435	5.9179614037	7.22674320386e-
<i>Earthen</i>	0.8335505369	0.0633558443603	13.156647904 6	0.0
<i>Goreto</i>	0	0	0	0
<i>Pitch</i>	1.8806385019	0.622238165749	3.0223772913	0.0026761439031
<i>Gravelled</i>	1.9118752264	1.0286071245	1.8587030761	0.0638341231633
<i>paved</i>	1.9252123118	1.20997628364	1.5911157415	0.0112406395409
<i>Commerical</i>	3.4233715511	2.85423059799	1.1994025827	0.0231110986368
<i>Residential</i>	2.1281082895	1.68827256347	1.2605241212	0.0208245749266
<i>Kathmandu</i>	0	0	0	0
<i>Lalitpur</i>	0.2365425831	1.53926332802	0.1536725905	0.877948622923

index Lalitpur here, is the less significant feature, so after removing it the results changes are shown in Table 7.13 and table 7.14.

Table 7.13 WLR model result after removing Lalitpur

Regression model Algorithm	R-squared value	RMSE value	F-value	p-value
WLR	0.6492	8.2007	62.7580	1.3132e-77

Table 7.14 Features of WLR after removing Lalitpur

Features	Coefficients	Standard_error	T_test value	p-values
<i>Year</i>	0.1655808936	0.1881941121	0.8798409888	0.0379495732648
<i>Road_width</i>	-0.192402805	0.193078425937	-0.996500799	0.0319634394084
<i>Location_Access</i>	-0.118960189	0.0578738498104	-2.055508489	0.0405056594808
<i>Inflation_rate</i>	0	0	0	0
<i>Governmentrate</i>	0.8410894455	0.140688852252	5.9783659621	5.1552238034e-
<i>Earthen</i>	0.8479781059	0.0628494540984	13.4922111586	0.0
<i>Goreto</i>	0	0	0	0
<i>Pitch</i>	1.8991933192	0.621592444249	3.0553674467	0.0024048308895
<i>Gravelled</i>	1.9162149217	1.02855797369	1.8630110997	0.0632239939349
<i>paved</i>	1.9200402909	1.21075590341	1.5858194748	0.0113603372258
<i>Commerical</i>	3.4195547475	2.8562778958	1.1972065997	0.0231964574379
<i>Residential</i>	2.0836397712	1.68647216572	1.2355020222	0.0217399049892
<i>Kathmandu</i>	0	0	0	0
<i>Lalitpur</i>	0	0	0	0

The most significant feature in our regression model are **Government rate and Earthen**.

The table of significant features for WLR are shown Table 7.15

Table 7.15: Significant features for WLR

Significant features
<i>Year</i>
<i>Road_width</i>
<i>Location_Access</i>
<i>Governmentrate</i>
<i>Earthen</i>
<i>Pitch</i>
<i>Gravelled</i>
<i>paved</i>
<i>Commerical</i>
<i>Residential</i>

7.3 Visualization

The data for different land parameters are visualized in graphs as shown below:

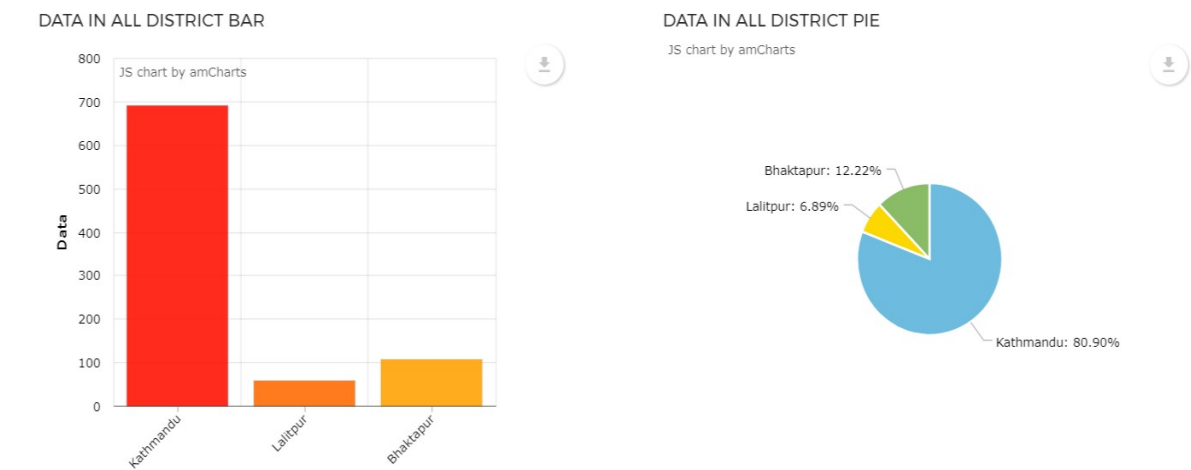
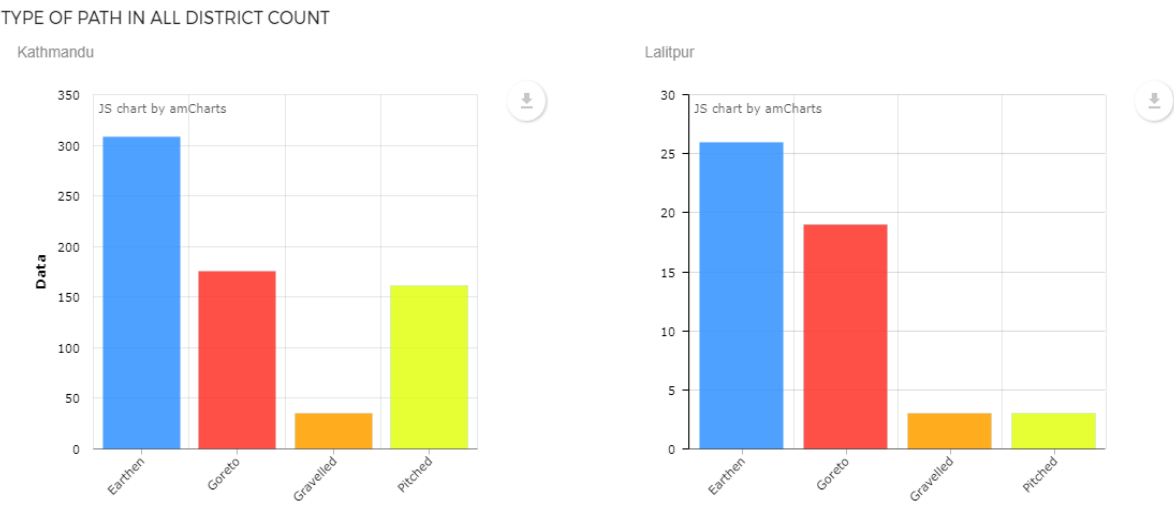


Figure7.5: Data in all Districts



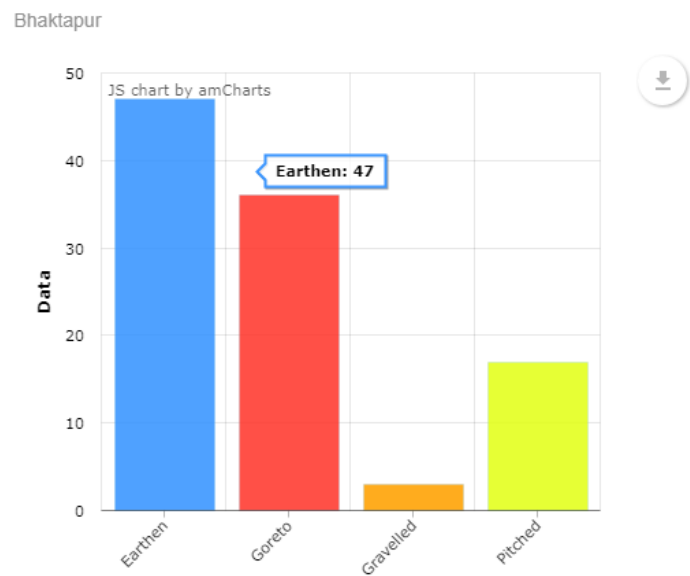


Figure: Type of Path in all Districts

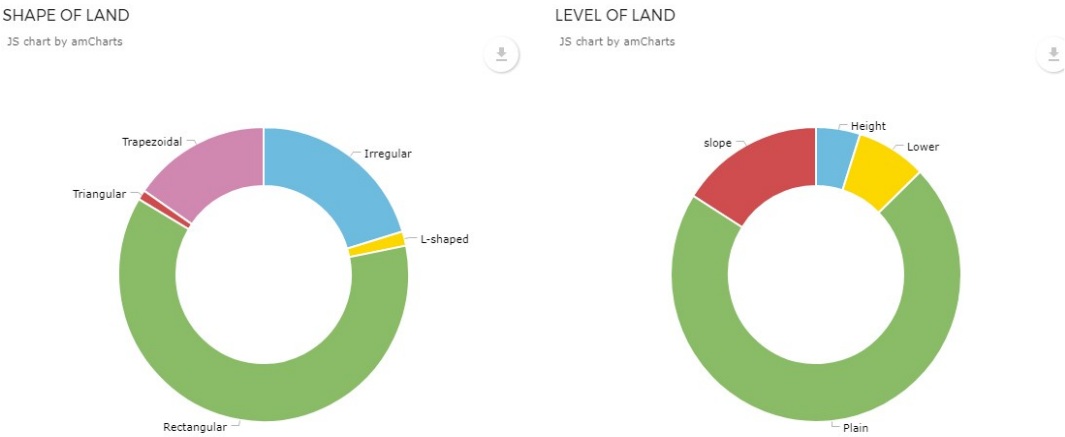
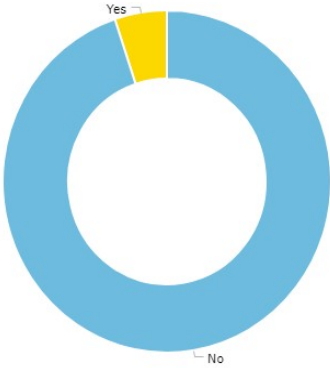


Figure: Shape of land level of Land in all Districts

HIGH TENSION LINE
JS chart by amCharts



RIVER/STREAM
JS chart by amCharts

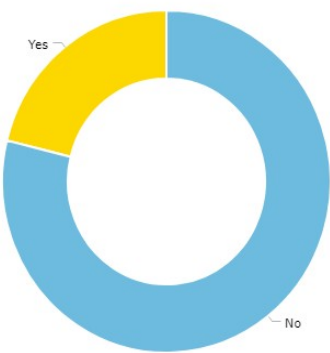
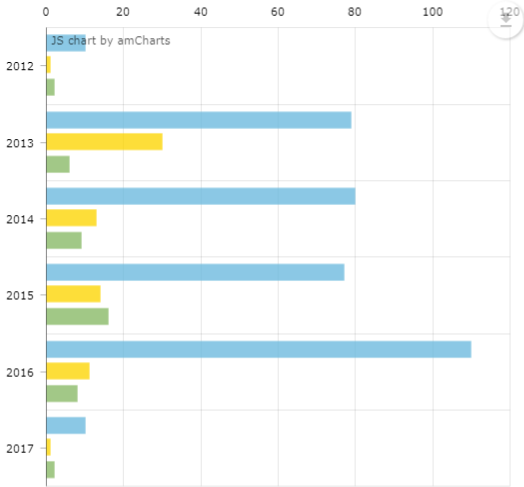


Figure 7.6: High Tension & River Stream in all Districts

FACILITIES TELEPHONE,ELECTRICITY & WATER (KBL)



FACILITIES TELEPHONE, ELECTRICITY, WATER & SEWAGE (KBL)

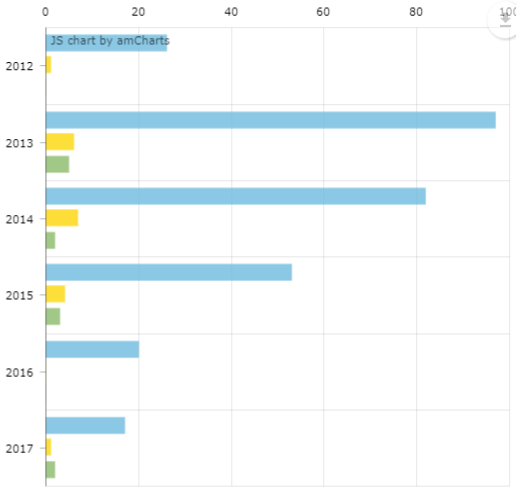


Figure 7.7: Facilities in all district

8. LIMITATIONS

In this project the analysis and prediction of the Real estate has been done with suitable accuracy. The project contains strong features which results in better analysis and prediction. Even though we have tried to include as many features as possible the project still lacks promise in some areas. Some of the limitations are as follows:

- **Lack of Interactive GUI:** Although we have developed a web based GUI which allows some interaction with the users, the GUI is not refined to the fullest and still lacks interactive interface.
- **Absence of dynamic data manipulation:** The project performs analysis and prediction on the data which are already stored. The data is static in nature. So the dynamic data manipulation is not included in the project.
- **Solely depends upon the evaluator for data.**
- **Location limitation:** Although a model is developed for prediction, it doesn't include the location parameter for the prediction. It uses the governmental rates as a parameter and then provides current commercial rates of the property without consideration of the location.

9. CONCLUSION

The main aim of the project is to predict the current commercial rate of the property taking in the intakes the parameters (like government rates, road width, property type, location access). This project is based on integrating the technical analysis with the data mining techniques to analyses and predict the real estate commercial rates. The results produced have RMSE value is found to be 8.32 which is not an optimum value but is acceptable.

The limitations specified in the previous section makes the project incomplete. There is still room for improvement. Some of the plans for future enhancement of the project are given below:

- Design a much more interactive User Interface.
- Devise methods to dynamically pull the newly generated trading data and perform analysis on it in real time basis.
- Improve the existing visualization system by enhancing the plot features to make it more informative.
- Devise methods to integrate multiple technical indicators to generate the buy/ sell signals for analysis.
- Build up a system which takes the location as a core parameter to decide the commercial rate.

10. REFERENCES

- [1] C. M. Bishop, Pattern Recognition and Machine Learning. Springer, 2006.
- [2] K. P. Murphy, Machine Learning: A Probabilistic Perspective. MIT Press, 2012.
- [3] S. J. Sheather, “Density Estimation,” Statistical Science, vol. 19, no. 4, pp. 588–597, 2004.
- [4] B. W. Silverman, Density Estimation for Statistics and Data Analysis. CRC Press, 1986.
- [5] L. van der Maaten, E. Postma, and H. van den Herik, “Dimensionality Reduction: A Comparative Review,” Elsevier, 2008.
- [6] C. J. C. Burges, Dimension Reduction: A Guided Tour. Now Publishers Inc, 2010.
- [7] A. Jain, M. Murty, and P. Flynn, “Data Clustering: A Review,” 2009.
- [8] P. Dayan and C. J. Watkins, “Reinforcement Learning,” in Encyclopedia of Cognitive Science. MacMillan Press, 2001.
- [9] D. C. Montgomery, E. A. Peck, and G. G. Vining, Introduction to Linear Regression Analysis. John Wiley and Sons, 2012.
- [10] I. J. Myung, “Tutorial on maximum likelihood estimation,” Elsevier Science, 2003.
- [11]<>
- [12]< <http://bokeh.pydata.org/en/latest/>>
- [13]Case, Karl E. and Shiller, Robert J. (1990), “Returns and risk on real estate and other investments: More evidence”, *Journal of Real Estate Portfolio Management*, Vol.8, No. 3: 265-279.
- [14] Hutchison Norman E., Adair A.S. and Leahy, Iain. (2005), “Property Research Priorities in Australia”, 8.2: 127-39.
- [15]Nitsch, Harold (2006), “Pricing Location: A Case Study of the Munich Office Market”, *Journal of Property Research*, 94-96.
- [16]Markowitz, H. M. (1952), “Portfolio Selection”, *Journal of Finance*, 7:1, 77–91.
- [17]Wurtzebach, Charles H., Mueller, Glenn R. and Machi, Donna (1991). “The Impact of Inflation and Vacancy of Real Estate Return”, *Journal of Real Estate Research*.