House Price Prediction Using Random Forest

- Semester 7/8 Report -

Anojan Satheesnathan
Sankeerthan Kasilingam

Department of Computer Engineering
University of Peradeniya

Final Year Project (courses CO421 & CO425) report submitted as a requirement of the degree of
B.Sc.Eng. in Computer Engineering

June 2020

Supervisor: Mr. Sampath Degalla (University of Peradeniya)

I would like to dedicate this thesis to my loving parents and “teachers” . . . who supported us to succeed our project.

Declaration

We hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is our own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgments.

Anojan Satheesnathan
Sankeerthan Kasilingam
June 2020

Acknowledgements

And we would like to acknowledge ... Mr. Sampath Degalla... Our supervisor, thank you for your all unwavering support, help and guidance.

Abstract

The implementation of a system that can assist a house seller or real estate agent to make better-informed decisions based on house price valuation. Housing price is influenced by multiple factors such as location, house size, number of bathrooms, and number of bedrooms. The traditional models used the statistics of these factors to predict house prices. The development of machine learning models to predict price as an alternative to the traditional model has been done in many countries. But there is no work reports on the use of Machine Learning (ML) techniques to predict the house prices in Sri Lanka. This was motivated us to do a study of houses located in Colombo, Sri Lanka. The data set composed of 12, 712 housing advertisements, collected from the online house-selling platform from July 2019 to April 2020 is used for the study. Each instance comprises six features of different data types those are int, float, and string. The study focuses on removing outliers using K-Means clustering and predicting price with Random Forest model. The study approaches two methods for spilitting the data which are train test split and cross validation. Cross validation result was better compared to train test split in our case. According to our results, after removing the outliers by clustering using cross validation, the Random Forest model performance was improved and the RMSE and R2 score of the best model was LKR 5.12 Million and 0.76 respectively. This study concludes that enriching the data set and tunning the model parameters improves performance.

Table of contents

List of figures

List of tables

Nomenclature

Acronyms / Abbreviations

CV	Cross Validation
GDP	Gross Domestic Product
LR	Linear Regression
ML	Machine Learning
NLP	Natural Language Processing
R2	R-Square
RF	Random Forest
RMSE	Root Mean Square Error

Chapter 1

Introduction

1.1 Background

1.1.1 Introduction

The real estate has been attractive and competitive market for long years because price decline of real estate properties are very rare compared to other assets [1] [2]. An accurate assessment of houses and real estate properties are very important to many parties such as local government, house owners, developers, investors, appraisers, mortgage lenders and insurers [3]. Before the arise of Machine Learning the house prices often estimated by professional appraisers but it had few drawbacks. The traditional services are expensive and appraiser is likely to vest interest to one party. When it comes to accuracy of estimation, the machine learning models perform better than hedonic models [4].
Similar studies using Machine Learning to predict house prices were done in different countries, but were totally different in terms of algorithms, features, parameters and data set [5] [6] [4] [7] [8] [9].
In Machine Learning context, the house price is characterized as a set of attributes. The common attributes that affect house prices are house size, location and number of bedrooms. But these are not limited to physical factors there are so many external factors affect the house prices such as economy, GDP growth, and etc [2] [5].
It's crucial to build a good Machine Learning model to predict house price. The data has to be studied carefully by feature extraction, pre-processing and parameter tuning is important when considering a Machine Learning model[10].

1.1.2 Sri Lanka Housing Market

In the last decade, the housing market in Sri Lanka has been rapidly growing, with average housing prices increasing by 17% nationwide according to Lanka Property Web. Nearly 50% increase in Colombo district year over year. 10.3% of the GDP increased in construction industry back in 2013 [11]. Together with stronger economic growth and increasing price expectations, this research presents good news to current home owners and potential home buyers looking for a safe long-term investment.

Fig. 1.1 Average Property Prices of Sri Lanka

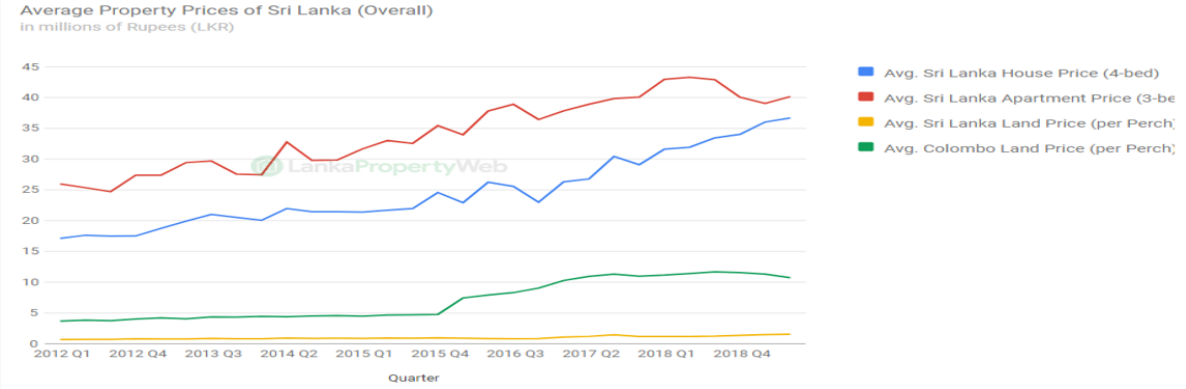


Figure 1.1 shows how 4-bed average house price, 3-bed average apartment price, average land price per perch and average Colombo land price per perch varying with the time with some fluctuations. There is an increasing pattern in all property values. Average house price increased up to 35 million in end of 2018.

1.2 The Problem

There is always uncertainty in the property market in Sri Lanka. There are lot of people involved in investment opportunities related to housing market. But most of the cases they use traditional data analysis techniques. As shown in figure 1.1 there was a suddenly reduction in house price in 2016 and there are some more in between start of 2012 to end of 2018. These uncertainties needs to be predicted beforehand to avoid risks related to investments.

Housing prices are currently estimated manually in Sri Lanka. This traditional approach needs a professional expert with the domain knowledge to predict the property value. In some cases appraiser can be paid by one party and he may conduct the appraisal

in favour of them. There are difficulties in ensuring that the appraiser conducts a neutral appraisal in these cases.

Even though we have built various ML models in previous semester, the model performance was not good in-terms of accuracy. Number of data instances were not enough to do the proper analysis.

1.3 The Proposed Solution

The proposed solution is to come up with a suitable house price prediction model that predicts all the house prices in Colombo, Sri Lanka.

Aims of this research are:

- Predict the house prices in Colombo
- Identify the factors affecting housing price
- Identify investment opportunities related to housing market
- Identify investment risks related to housing market
- Improve the accuracy and performance of the model

The objective of this research is to find out a suitable machine learning model by evaluating prediction accuracy and performance in order to predict the house prices in Colombo.

There are lot of researches were carried out in various countries in house price prediction using machine learning but it is new in Sri Lanka. The scope of this project is only limited to Colombo district since non-availability of data source for other districts in Sri Lanka. External factors such as economic affecting housing prices is not considered in this research because of the non-availability of data. Data from ikman.lk is only used even though it is non-reliable because there are no other data sources available in Sri Lanka. Anomaly detecting techniques are used to remove anomalies by doing statistical analysis and clustering.

1.4 Deliverable and Milestones

These are the milestones completed in semester 8

- Data Collection: Increase the data points

- Data Pre-processing
- Model Building: Clustering
- Model Building: Prediction
- Conference Paper & Presentation
- Mid-Demonstration
- Final Project Report
- Publishing the project to department Wiki
- Final presentation

1.5 Outline of the Report

This report gives the background information about house price prediction and why it is important in Sri Lanka in the Introduction section. It describes the details of previous work which have been done regarding house price prediction using Random Forest regression in various countries. And how clustering used to detect outliers in the Related Work section. After that It describes about the methodology followed and the implementation details of building the clustering and prediction model in Methodology and Experimental Setup and Implementation sections respectively. The document contains the experiment results and analysis in Results and Analysis section. The document also contains the conclusion and future works required for further improvements in Conclusion and Future Works section.

Chapter 2

Related work

2.1 Introduction

The study analyses various researches that used Random Forest regression and outlier detection using clustering to predict house prices and aims to focus on various machine learning models used in predicting house prices along with different set of features, prediction accuracy and predictive performance in previous researches. An extensive study should required to analyse different parameter selection and features selection and their results in-terms of accuracy and error metrics.

Other than individual buyers and sellers, there are many investment firms directly linked with housing market. A report from MSCI (known as Morgan Stanley Capital Investment) states that real estate investment had increased to $8.5 trillion in 2017, while comparing it with previous year it was $1.1 trillion increment. Even Though the investment in housing/residential markets seem to be profitable ,but the prices are directly connected with global economy, GDP and political stability and demand is the key factor in determining house prices [5].

Multiple discipline people such as house owners, developers, investors, appraisers, tax assessors, mortgage lenders and insurers rely on house appraisal to make decisions [1]. The house price appraisal is usually done using hedonic models. In hedonic price theory, the house is viewed as a set of characteristics such as number of bedrooms, number of bathrooms, geolocation, house size and many more. The coefficients of some characteristics exhibited unstable nature [1]. The main problem with hedonic models in house appraisal is the relationships between each characteristic and price should be known in advance [5].

Another alternative approach is in house appraisal is machine learning models. Various researches carried out in multiple countries including Spain, UK, New Zealand, USA,

Turkey and Italy. House price prediction using machine learning performs better than traditional models with reasonable error. By the nature of machine learning, there is no standard way to define a model to predict house price. The feature sets were considered in researches show difficulty in concluding one best model for this problem. The features sets took account of common known features and different features only belong to a particular region, such as environmental pollution and etc [12]. It is clear, the good data set with right feature selection and algorithm selection can make even better house appraisal model.

Our study inspired from [5], the authors collected data from an online property platform called idealista.com. The data set consisted of 2,266 instances with 20 factors. The data set is limited to Salmanca district, Madrid, Spain. They implemented Support Vector Regression, K-Nearest Neighbours, Ensemble of Regression Trees and Multi Linear Perceptron models. According to their findings the model built with Ensemble of Regression trees performs better. Their model scored 16.80% mean absolute error against mean prices and 5.71% median absolute error against median prices.

Another study was done with online housing advertisements in Brazil. The data set consisted of 12,223,582 housing advertisements with 24 features and this was collected from 2015 to 2018. The study compares two ML approaches; the Random Forest based model and the Deep Learning model. According to their findings both models perform well. However, the Random Forest based model performs better with numeric attributes [13].

A study was carried out in Malaysia with real house data set of 2016. They implemented the model with Random Forest, Decision Tree, Ridge, Linear and Lasso Regression. They evaluated and compared based on R2 score and RMSE. According to their findings the Random Forest model performs better than others [10].

Another study was done in South Korea that compares Random Forest based model with traditional Ordinary Least Square(OLS) method. The author used a real data set of 16,601 samples within the period of 2006 to 2017. The study took account of structural, neighbourhood, locational and macro attributes such as transaction period, GDP and mortgage interest rate. They used 1:9 split of data set for test and train and considered averages of 10 experiments to reduce the possibility that results occurred by chance. They found only 5.5% average deviation from actual price as compared to 20% using the OLS model. They noted the Random Forest model can perform better than traditional

models [14].

A study was done on single family open house data set of Arlington Virginia, USA. The data set comprises of 27,649 houses and factors such as lot size, years built, zip code and location. The authors bench marked Linear Regression and compared it to the Random Forest model. From their results Random forest performs better than Linear Regression [15].

2.2 Dataset

From the researches we have found, some of them used dataset collected from online realestate selling platforms. They did not use exact house price transactions instead of they assumed the price advertised in websites as real price. Only some researches used open Taiwan house data set, kaggle house data set, CoreLogic Dataset, and ValueguardAB dataset. The minimum number of data instances used in researches is 193 and the maximum number of instances used in researche is more than 2.4 million. The table 2.1 shows a detailed summary of data-sets used in various researches along with the data source.

Table 2.1 Data-Sets Used in Researches

Research	**Location**	**Number of Instances**	**Source**
[5]	Salamanca district of Madrid, Spain	2226	Online website
[13]	Brazil	12,223,582	Online website
[14]	South Korea	16,601	N/A
[15]	Virgina, USA	27,649	N/A
[12]	Taranto(Italy)	193	Online Geographic Information System (GIS)
[4]	Christchurch, New Zealand	200	Online website
[2]	N/A	3000	Kaggle data set
[16]	Taiwan	74568	Open source data (data.gov.tw)
[8]	Ankara,Turkey	N/A	Online website
[17]	King County, USA	21613	Kaggle data set
[18]	N/A	1500	Kaggle data set
[19]	King County, Seattle	21000	N/A
[20]	Beijing, China	9600	Online website
[21]	Suburb, Boston	452	N/A
[22]	Turkey	5741	House hold budget survey data
[23]	Sweden	N/A	Valueguard AB's housing data set
[24]	Suburb, Boston	506	N/A
[6]	London	2.4 Million	Landon data store + Land registry (with their online searching tool)
[25]	Montr´eal, Canada	25000	online website

2.3 Conclusion

From mentioned studies, we could see that the studies used either real data set or data set collected from advertisements. The studies used Random Forest model and got reasonable performance, and they were done in different countries. It's useful to do research on the local market (Sri Lanka) and to see how the machine learning models can be utilized for house price prediction.

Chapter 3

Methodology

3.1 Overview

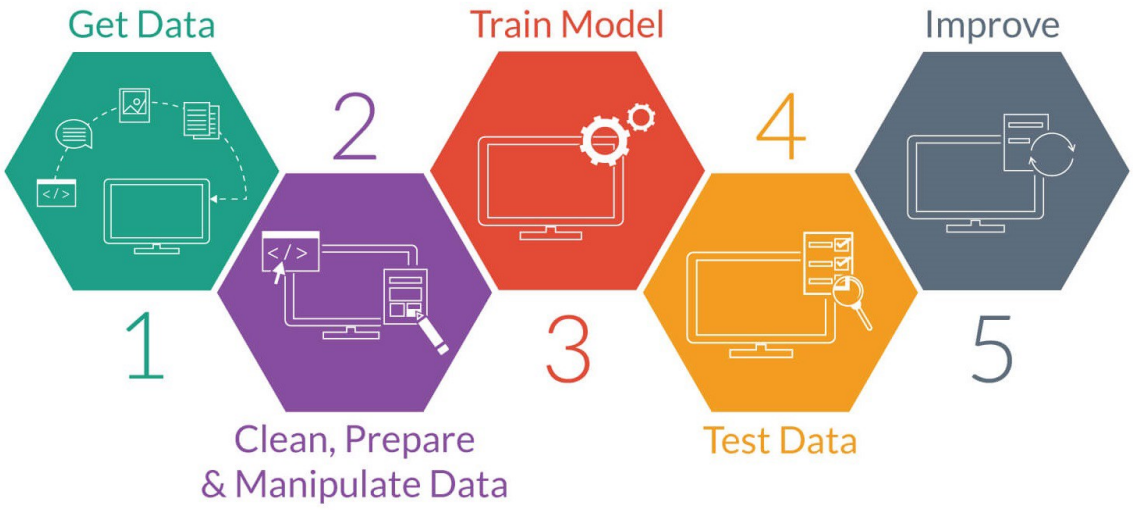
This chapter covers how we aligned multiple divisible steps into a sequence manner, selection criteria we used to select each methods (from data collection to prediction), abstract idea of each steps and brief theory of algorithms.

After we reviewed literature, we found similar researches to house price prediction using Random Forest were used in different ways and different methods. As we selected the Random forest algorithm that performed well in the previous semester is the suitable algorithm for this data set, We mainly focused on improving the accuracy of the model in this phase.

3.2 Conceptual Design

Our project is Machine Learning project, so we followed and structured the project into multiple steps as this is a general procedures used in Machine Learning projects. The figure 3.1 shows the steps in sequence manner.

Fig. 3.1 Project Work Flow



3.3 Algorithms

K-Means: It is simplest unsupervised algorithm and it identifies k (predefined) number of clusters (non-overlapping instances) based on distance between instances [26].
Random Forest Regression: It is a supervised algorithm and it uses ensemble method bagging. It builds independent decision trees for sub sample data and make predictions by combining independent models [27]. Using Random Forest has some advantages to our case. It helps to avoid Overfitting to the data and explores nonlinear relationships and unstable influence of variables on others[14].

3.4 Data Cleaning and Preprocessing

We found that 5,164 number of duplicates exists in collected data set and we removed them. Along with removing duplicates we dropped unstable data instances which has null value for any fields. We established a consistent stable data set of 7,547 instances. We used one-hot-encoding to convert categorical locations. To make the model learn more about data, we derived three new attributes from existing attributes [28]. They are the ratio between house size and land size, ratio between number of beds and number of baths and bed size. They bring the hidden information and relationships between existing attributes. The ratio between house size and land size was used to separate single story houses. Ratio below 1.0 were considered as single story houses. The bed size

was derived with an assumption of the total bedrooms' size take the whole space of a house.

3.5 Data Exploration and Analysis

In data related studies, the visualization of data is important to understand complex problems [29]. We visualized the data using pair plot to analyse the relationships between variables. From the pair plot we found out that there is no linear relationship between house price and other independent variables. Only the number of beds and number of baths shows an approximate linear relationships.

Table 4.6 shows the description of the data set.

Table 3.1 Data Description

Attribute	**Type**	**Mean**	**Std**
Price (LKR)	int64	3.03e+07	3.99e+07
Land size (Perches)	float64	34.11	1843.47
Number of beds	int64	3.93	1.25
Number of baths	int64	2.99	1.28
House size	float64	2693.61	2663.46
Location	String	NA (eg: Piliyandala)	NA

3.6 Outliers Detection

The outliers are the houses that do not follow the crowd (majority instances). These instances show deviation from majority of instances in terms of values of attributes. These instances may be outliers by natural or outliers due to incorrect attribute values. To fit a model to data (majority instances), the outliers should be removed. Otherwise they will introduce disturbances to model [30]. In our study, the assumption (advertised prices are considered as sold prices) created more outliers and introduced difficulty to remove outliers with typical process.

We found that some of the houses have extreme values for some attributes. To filter them, we analysed and derived threshold values (upper and lower). Some houses have large land size, and the price is not only for the houses and it includes the land as well.

We found the land size below 30.0 perches contain 97.53% instances. Tab. 3.2 shows the threshold values with respective attributes.
The clustering is the technique to group similar instances. By considering clustering to remove outliers, have been studied, and a study propose clustering based on key attributes is a method to remove outliers and the author summarized K-Means clustering performed better than distance based outlier algorithms [31]. We considered the K-Means algorithm to cluster the data in order to eliminate and identify similar groups of house instances. We used elbow method to decide the optimal number of clusters. We found three number of clusters correctly identify the outliers (the two small clusters of house instances). The results are presented in the result and discussion section. Table 3.2 shows the thresh hold values of Price, House size and Land size.

Table 3.2 Threshold value of Attributes

Attribute	**Lower Threshold**	**Upper Threshold**
Price(LKR)	8,000,000	50,000,000
House size (sqft)	0	20000
Land size(Perches)	0	30

3.7 Model Setup

We used sckitlearn library to implement models. We experimented two methods which are train test split and cross validation. We splitted the data set into 1:3 ratio for test and train for train test split method and we used cross validation values as 10 for cross validation method. For the initial experiments with any model, we used default parameters provided by sckitlearn. For train test split method, We built multiple Random Forest models in range of estimators. With whole data set, we tested all possible combinations of feature sets and identified most importance feature sets. Then every model we tested with the importance feature sets. Cross validation method experiments conducted using the best feature combination identified from train test split. Random Forest parameter tuning was done separately for this method. Finally we compared each method using evaluation matrices.

3.8 Evaluation Metrics

We considered RMSE (Root Mean Squared Error) and R2 score (Coefficient of Determination) to evaluate and compare models. The range of R2 score is 0.0 to 1.0 but it can take negative values in worst case.

$$\text{RMSE} = \frac{\sum_1^n (y\_pred - y\_test)^2}{n} \quad \text{R2} = 1 - \frac{\sum_1^n (y\_test - y\_pred)^2}{\sum_1^n (y\_test - mean(y))}$$

y_test: test target, y_pred: predicted target, n: number of test instances, y: all target values

3.9 Parameter Tuning

We used gridSearch (sckit implementation) to tune hyper parameters of Random Forest regressor. After we gained a certain level of confidence in the model by changing the parameters manually, we used grid search to tune in order to find optimal parameters.

Chapter 4

Experimental Setup and Implementation

The experiment compares various Random Forest model results with outlier detection using Clustering. The goal of the experiment is to improves the accuracy of the model build in previous semester. The chosen evaluation metrics are R2 and RMSE.

These are the process flow in our experiment:

- Data Collection
- Data Preparation and Exploration
- Clustering Model Building
- Prediction Model Building

4.1 Data Collection

Data set was collected from ikman.lk which is a premier classified advertisement website operating in Sri Lanka using web scraping techniques. The site contains user-generated classified advertisement, sorted by various categories. For our study we only took into account the market in the Colombo district, the capital of Sri Lanka. To establish data pool for our study, we collected the data within the period of July 2019 to April 2020 from the website ikman.lk which is a leading online selling and buying platform in Sri Lanka. It allows sellers to post house advertisements on the website. For our study we assumed the advertised house prices as actual sold prices. The figure 4.1 shows some samples of collected data set. Each row represents one observation. 12,712 data instances were collected with 6 attributes: location, price, baths, beds, house_size, and land_size.

Fig. 4.1 Samples from Collected Data Set

Location	Price	Beds	Baths	House Size	Land Size
Piliyandala	12500000.0	4.0	2.0	1750.0	6.5
Kottawa	29500000.0	4.0	4.0	3510.0	9.0
Malabe	19000000.0	4.0	3.0	2850.0	7.5
Kottawa	35000000.0	7.0	4.0	1700.0	15.0
Boralesgamuwa	29500000.0	5.0	4.0	3500.0	8.0

Table 4.1 shows a sample observation from collected data with the values.

Table 4.1 Sample Instance from Collected Data

Feature	**Value**
location	35
Price	12500000
beds	4
baths	2
house_size	1750
land_size	6.5

The table 4.2 shows description about all the feature in collected data.

Table 4.2 Feature Description for Collected Data

Feature	Description
location	House location
price	Price of the house in LKR
beds	Number of bed rooms
baths	Number of bath rooms
house_size	Size of the house in square feet
land_size	Size of the land in perches

4.2 Data Preprocessing

The collected data has duplicate, inconsistent data instances and outliers. It was not processable. To prepare the data for the model building, some changes were made.

- Feature types were identified
- Null values were handled
- Outliers were identified

4.2.1 Types of Variables

Feature types are identified and the categorical variable (location) is converted to numerical values to process the data. Forty locations were identified in the data set and all the locations are converted to numerical values. This is the only categorical variable in the data set. Figure 4.2 shows the sample data set with one-hot encoded locations.

Fig. 4.2 One-Hot Encoded Locations

	index	Location	Price	Beds	Baths	House Size	Land Size	0	1	2	...	30	31	32	33	34	35	36	37	38	3
0	0	Piliyandala	12500000.0	4.0	2.0	1750.0	6.5	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.
1	1	Kottawa	29500000.0	4.0	4.0	3510.0	9.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.
2	2	Malabe	19000000.0	4.0	3.0	2850.0	7.5	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.
3	3	Kottawa	35000000.0	7.0	4.0	1700.0	15.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.
4	4	Boralesgamuwa	29500000.0	5.0	4.0	3500.0	8.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.
5	5	Malabe	18000000.0	4.0	4.0	4500.0	10.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.
6	6	Piliyandala	16500000.0	4.0	3.0	2400.0	6.7	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.
7	7	Ratmalana	4500000.0	2.0	1.0	1000.0	3.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.
8	8	Kottawa	8900000.0	3.0	2.0	1380.0	6.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.
9	9	Piliyandala	20500000.0	4.0	5.0	3500.0	9.0	0.0	0.0	0.0	...	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.

The location is a key attribute in determining house prices [32]. The collected data set covers 40 regional zones in Colombo. Out of 40 zones, six zones have lesser than 20 instances and the maximum number of instances were recorded for zone 'Piliyandala' which is 1448.

Others are numerical variables price, beds, baths, house_size, land_size. Table 4.3 shows numerical variables and its types.

Table 4.3 Numerical Variables

Variable	**Type**
price	Int
beds	Int
baths	Int
hose_size	Float
land_size	Float

To make the model learn more about data, we derived three new attributes from existing attributes **??**. They are the ratio between house size and land size, ratio between number of beds and number of baths and bed size. Figure 4.3 shows the derivation equation for the new variables.

Fig. 4.3 Derivation of New Variable

```
df['h_l_ratio'] = df['House Size'].apply(lambda x: x * 0.0036730945821854912)/ df['Land Size']
df['Bed Size'] = df['House Size'] / df['Beds']
df['b_b_ratio'] = df['Beds']/ df['Baths']
```

With the derived variable and one-hot encoded locations number of futures in the data set increased to 49.

4.2.2 Null Values Handling

All the duplicate and null values were removed from the data set. Figure 4.4 shows after removing the duplicate and null values from the data set only 7547 instance remains for further processing.

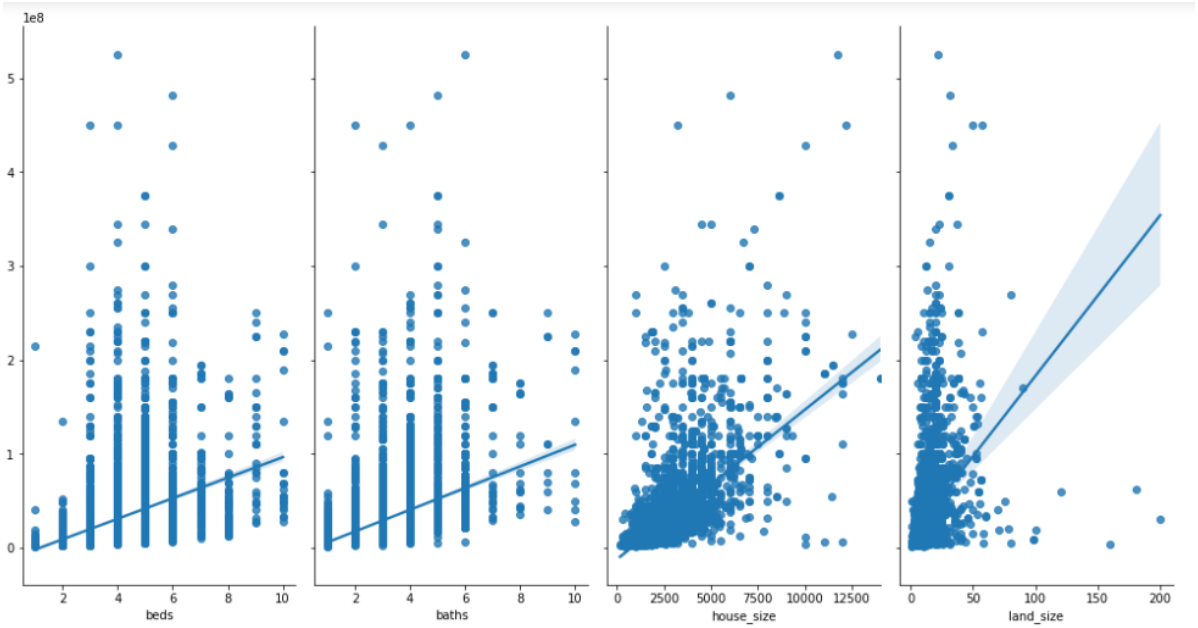
Fig. 4.4 Handling Duplicate and Null Values

```
In [507]: df.drop_duplicates(keep='first', inplace=True)

In [508]: df.shape
Out[508]: (7548, 6)

In [510]: df = df.dropna()

In [511]: df.shape
Out[511]: (7547, 6)
```

4.2.3 Data Visualization

In data related studies, the visualization of data is important to understand complex problems [29]. We visualized the data using pair plot to analyse the relationships between variables. From the pair plot we found out that there is no linear relationship between house price and other independent variables. As shown in 4.5 only the number of beds and number of baths shows an approximate linear relationships.

Fig. 4.5 Price Distribution



Histograms are tail heavily for some numerical attributes. Its harder to Machine Learning algorithms to detects patterns. These attributes need to be transformed more bell shaped distribution.

After that, extreme values of each feature were identified by probability distribution plotting except for location which is a categorical variable . Figure 4.6 shows the data description. It can be seen that the standard deviation of each features are high. So that the data points were not normally distributed.

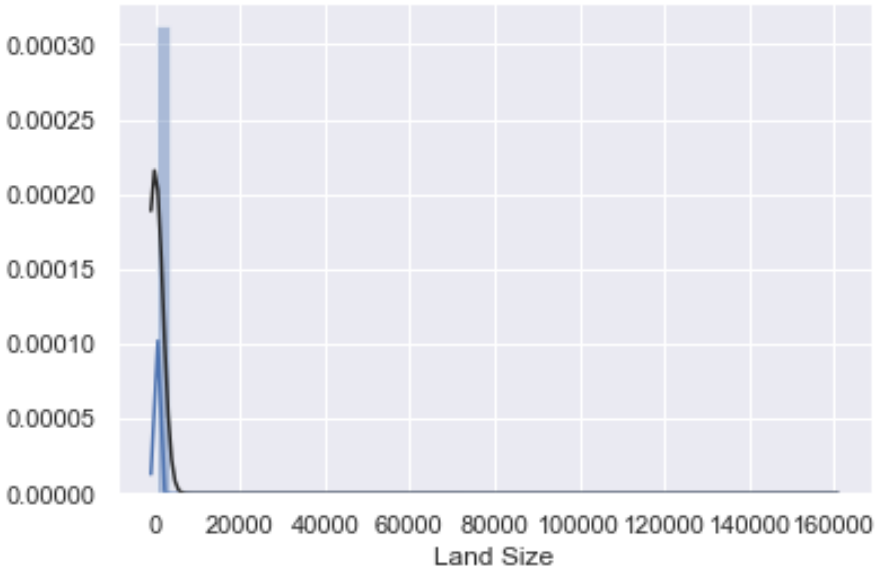
Fig. 4.6 Data Description

	Price	Beds	Baths	House Size	Land Size
count	7.548000e+03	7548.000000	7548.000000	7548.000000	7548.000000
mean	3.029252e+07	3.929650	2.991918	2693.610889	34.106186
std	3.994695e+07	1.254521	1.276242	2663.455554	1843.472715
min	0.000000e+00	0.000000	0.000000	0.000000	0.000000
25%	1.350000e+07	3.000000	2.000000	1600.000000	7.500000
50%	2.150000e+07	4.000000	3.000000	2500.000000	9.855000
75%	3.300000e+07	5.000000	4.000000	3200.000000	12.000000
max	1.300000e+09	10.000000	10.000000	140000.000000	160000.000000

In order to visualize the extreme values of the data set probability distribution of each variable was plotted. Figure 4.7 shows the distribution of Land Size is not normal. So data instances that consist of land_size greater than 30 perches were removed from the data set.

Fig. 4.7 Probability Distribution of Land Size with Extreme Values

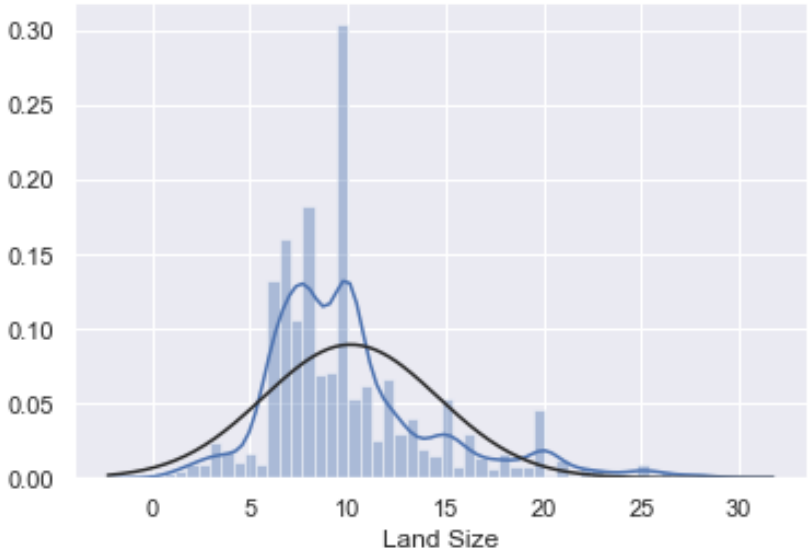
```
Out[449]: <matplotlib.axes._subplots.AxesSubplot at 0x1f304935b88>
```



To ensure the normal distribution of the location probability distribution was again plotted. Figure 4.8 shows the approximate normal distribution of land_size without extreme values.

Fig. 4.8 Probability Distribution of Land Size without Extreme Values

```
Out[451]: <matplotlib.axes._subplots.AxesSubplot at 0x1f304049448>
```



This procedure applied to all the numeric features except location and the probability distribution was observed to check whether it was normal or not. After that relationships between variables analyzed with correlalation matrix.

4.3 Outliers Handling by Clustering

Outliers were detected by doing statistical analysis. As stated in visualization section each attribute was checked for extreme values by plotting the probability distribution. With these information and visualisation graph in the previous section, we decided data has outliers.

4.3.1 Finding Clustering Parameters

As stated in methodology section another approach was to remove outliers by clustering. We used elbow method to decide the optimal number of clusters. We found three clusters correctly identify the outliers (the two small clusters of house instances).

Figure 4.9 show the algorithm (elbow method) used to find the optimal number of cluster (k).

Fig. 4.9 Elbow Method

```
# Elbow method
# Finding k value

import matplotlib.pyplot as plt
k_rng = range(1, 10)
sse = []
for k in k_rng:
    km = KMeans(n_clusters=k)
    km.fit(df)
    sse.append(km.inertia_)

plt.xlabel('K')
plt.ylabel('Sum of squared error')
plt.plot(k_rng, sse)
```

To further experiment we found out a list of important feature sets from all possible combinations. We used the Random Forest Regression model to evaluate importance in terms of model scores R2 and RMSE. The idea was to remove or avoid unnecessary trials in hyper parameter tuning. Figure 4.10 shows the combination of features we used to cluster the data set.

Fig. 4.10 Feature Combinations Used for Clustering

```
cols_cluster = [ 'House Size', "Land Size", 'h_l_ratio', 'b_b_ratio', 'Beds', 'Baths']
cols_cluster_price = ['Price', 'House Size', "Land Size", 'h_l_ratio', 'b_b_ratio', 'Beds', 'Baths']
cols_cluster_v1 = ['Price', 'House Size', 'Land Size']
cols_cluster_v2 = ['Price', 'House Size', 'Land Size', 'Beds']
cols_cluster_v3 = ['Price', 'House Size', 'Land Size', 'Baths']
cols_cluster_v4 = ['Price', 'House Size', 'Beds']
cols_cluster_v5 = ['Price', 'House Size', 'Baths']
cols_cluster_v6 = ['Price', 'Land Size', 'Beds']
```

We got the cluster which contains largest number of instances and trained with Random Forest model and observed the evaluation matrices. Number of cluster three and combination of features (price, house_size, land_size, beds, baths, location) gave good accuracy and the three derived variables did not affect the accuracy much. Figure 4.11 shows the algorithm implemented to find out best cluster parameters.

Fig. 4.11 Algorithm for Finding Best Cluster Parameters

```
def find_best_cluster(df_house, cols_list,rf_x_cols, rf_y_col, n_cluster_max=3, n_cluster_min=2):
    for cols in cols_list:
        for n in range(n_cluster_min, n_cluster_max + 1):
            y_km = cluster(df_house[cols], n)
            print('-------------------------------------------------')
            print(str(cols))
            print('-------------------------------------------------')
            print('Size of clusters : ' + str([ df_house[y_km == i].shape[0] for i in range(n)]))
            print('-------------------------------------------------')
            for i in range(0, n):
                if (df_house[y_km == i].shape[0] > 50):
                    build_rf_model(df_house[y_km == i],
                          rf_x_cols,
                          rf_y_col)
                else:
                    print('small cluster ignored')
```

After finding the optimal cluster parameters we clustered the data using this parameters and saved each individual cluster data set in three (3 clusters) different files for training the Random Forest model.

4.4 Prediction Model

Before building prediction model, we implemented training and testing setup. One of the standard practice in machine learning project is split data set into two parts 'train' and 'test' set. The 'train' set is what the model is trained on, and the 'test' set is used to

evaluate the model performance on unseen data. We tried out many approaches before selecting the suitable model. We have tried two methods for splitting data those are train test split and cross validation.

The Initial experiment was about building a Linear Regression model to identify relationships between variables. It helps to compare the result with Random Forest model. Linear Regression try to fit a straight line between house price and other features. And Random Forest is ensemble learning model. We used library scikit-learn class Linear Regression and RandomForest Regressor to implement these algorithms.

4.4.1 Train Test Split

We split data set using library scikit-learn class train test split (1/3 test, 2/3 train). Using this we can split easily the data set into training and test data set in various proportions. The parameters train size, test size, and random state will specify size of train data set size, test data set size, and seed of random number generator. The benefit of using this library class instead of splitting data set manually is, it splits in random manner. The figure 4.12 shows usage of this class.

Fig. 4.12 Code Train Test Split

```
X_train, X_test, Y_train, Y_test = train_test_split(df.drop(['Price', 'Location', 'index'], axis=1), df['P
rice'],
   test_size=0.33)
```

After splitting the data we tried various techniques such as trained the model splitting the data set into single story house data set and whole data set, and with clustered data by changing the Random Forest hyper parameter value range (10, 400).

Figure 4.13 shows the code for Linear Regression.

Fig. 4.13 Code Linear Regression

```
X_train, X_test, Y_train, Y_test = train_test_split(df.drop(['Price', 'Location', 'index'], axis=1), df['P
rice'],
   test_size=0.33)
lr_model = LinearRegression(n_jobs=-1)
lr_model.fit(X_train, Y_train)
Y_pred = lr_model.predict(X_test)
r2 = lr_model.score(X_test, Y_test)
lr_rmse= np.sqrt(((Y_pred - Y_test) ** 2).mean())
mape = (np.abs((Y_test - Y_pred)// Y_test).sum()) * (100/len(Y_test))
print('r2 : ' + str(r2))
print('rmse : ' + str(lr_rmse))
print('mape : ' + str(mape))
```

Figure 4.14 shows the code for Random Forest regression with n_estimators (10, 400).

Fig. 4.14 Code Random Forest Regression

```
estimators = np.arange(10, 400, 10)
results_rf_wh_v1 = []
scores = []
X_train, X_test, Y_train, Y_test = train_test_split(df[features_important[1]], df['Price'],
   test_size=0.33)
rf_model = RandomForestRegressor(n_jobs=-1)
for n in estimators:
    rf_model.set_params(n_estimators=n)
    rf_model.fit(X_train, Y_train)
    Y_pred = rf_model.predict(X_test)
    score = rf_model.score(X_test, Y_test)
    rmse= np.sqrt(((Y_pred - Y_test) ** 2).mean())
    mape = (np.abs((Y_test - Y_pred)// Y_test).sum()) * (100/len(Y_test))
    scores.append(score)
    results_rf_wh_v1.append((score, rmse, mape))
plt.title('Random Forest Score with diffent n estimators')
plt.xlabel('n_estimator')
plt.ylabel('score')
plt.plot(estimators, scores )
```

4.4.2 Cross Validation

Another approach we followed was 'k-fold cross validation'. Data set is randomly split up into k groups. One of the group is used as the 'test' set and the others are used as 'train' set. This process is repeated until each group has been used as the 'test' set. Initially We split the data set into ten equal size bins and then pick one for testing and other nine for training, likewise we changed the test set and training set. We did implemented ten experiments, then we averaged R2 score and RMSE. We used library scikit-learn class cross_val_score. The parameters are cv and scoring. The cv will specify number of splits(numbers of bins) and scoring will specify which type of evaluation matrix used(in our case R2 and RMSE).

We selected cv = 10(usually cv=10). Hyper parameters are the variables that govern the training process itself. To find best hyper parameters, we have to train and see the results if results are poor then we have to adjust hyper parameters and repeat same procedure until we get reasonable results. The parameters contains max features, n estimators(trees), and max depth show in figure for Random Forest. Figure 4.15 shows the parameter combination used to find the Random Forest best model.

Fig. 4.15 Hyper Parameters For Random Forest Model

```
model_params = {
    'lr': {
        'model': LinearRegression(),
        'params': {}
    },
    'rf': {
        'model': RandomForestRegressor(),
        'params': {
            'n_estimators': [10, 20, 50, 70, 100],
            'max_features': [1, 2, 3, 4, 5],
            'max_depth': [1, 2, 5, 10, 15, 20, 25]
        }
    },
}
```

This process is known as hyper parameter optimization or tuning. To do this procedure manually will take time. We used the library scikit-learn GridSearchCV to utilize time. Figure 4.16 shows the code we used.

Fig. 4.16 Grid Search CV

```
for model_name, mp in model_params.items():
    clf = GridSearchCV(mp['model'], mp['params'], cv=10, return_train_score=False)
    clf.fit(X, y)
    scores.append({
        'model': model_name,
        'best_score': clf.best_score_,
        'best_params': clf.best_params_
    })
```

4.5 Research Instruments

We used python and scikit-learn library for our experiments. Usually prototype model will be built on python and actual model will be implemented in other languages (C++). So we also following same approach. The library scikit-learn is beginner friendly and

have enough documentation. Vast community uses scikit-learn and most of algorithms are implemented in scikit-learn.

4.6 Pitfalls and Workarounds

The (ikman.lk) can be used by anyone, So initially panel suggested to use another data source if possible because of the non-reliability of the data. But we got to know that users required to provide a valid email address and need to pay in order to upload advertisements onto the site. All submitted advertisements are manually screened before being shown on the site as a security measure to ensure that the products and services being offered are authentic and legal.

We have contacted property valuation department in Colombo for additional data source. The attempt was given up because of getting delayed and there were some authorization related issues.

These house prices are not exactly the sold price of a house but we made an assumption since there is no much different in the advertisement price and sold price most of the cases. And outliers in the data-set can be also identified and removed by doing statistical analysis and clustering.

Our initial attempt was to identify the house features from ikman.lk data programmatically. But after some researches, we have found that it is a more complex problem and NLP has to be used. It was out of the scope of project at that time, It was given up.

Chapter 5

Results and Analysis

5.1 Overview

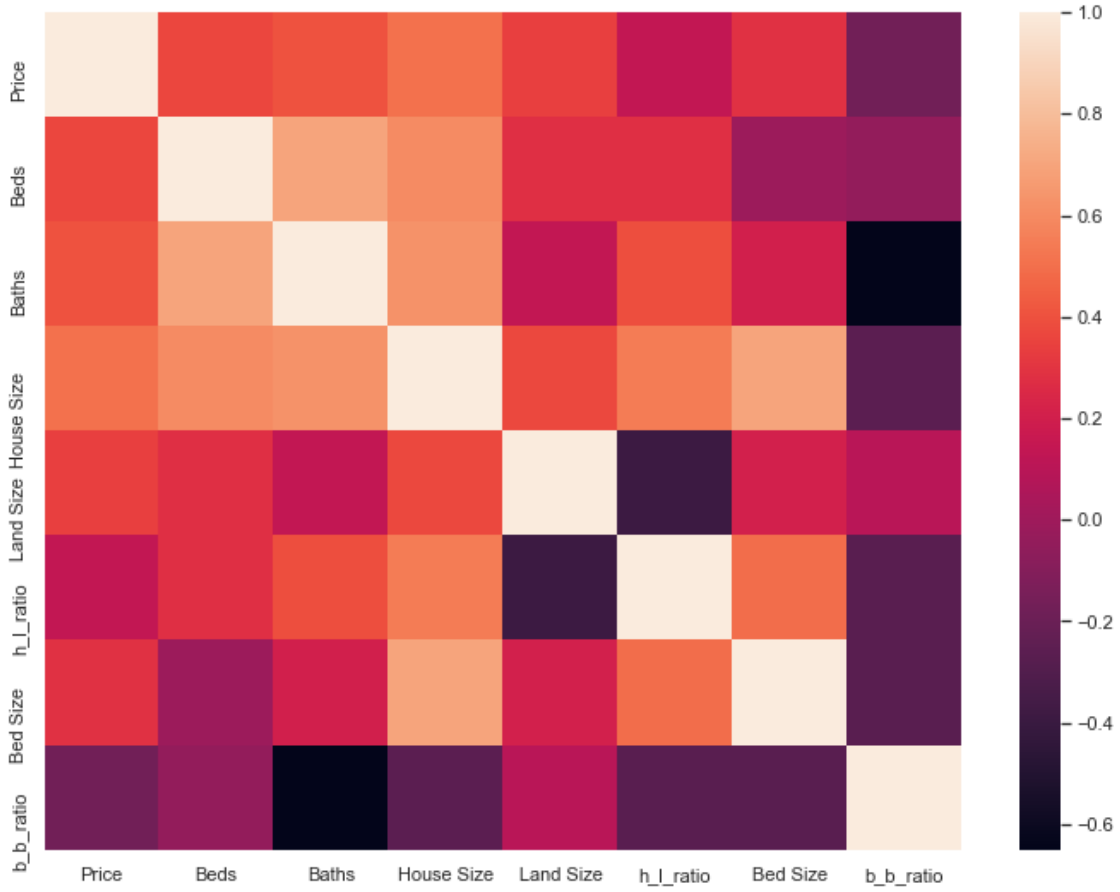
In this section, the result of experiments and discussions are presented. The experiments include both outlier detection model and prediction model.

5.2 Features Relationship

Attributes of house are not independent, they have relationship between others. We looked at correlation matrix, it shows correlation between attributes. The figure 5.1 shows the correlation between attributes of all houses.

As shown in figure 5.1 beds, baths, house_size, land_size, bed_size are highly correlated with price. house_size is highly correlated with price compared to other features.

Fig. 5.1 Feature Correlation Matrix



5.3 Clustering Model

To eliminate outliers further, we implemented K-Means model with whole dataset. We conducted the experiments in two ways by changing number of clusters and features. To find out optimal number of clusters, we used elbow method. From the curve of SSE (Sum of Squared Error) against number of clusters, we noticed after two clusters SSE declines slowly. From our data set context, splitting further (more than three clusters) would reduce number of instances per clusters. Therefore we decided and did the experiments with up to 5 clusters. The figure 5.2 shows the Sum of Squared Error difference against cluster number.

After finding the optimal parameters, we started to analyze each three clusters, identified each clusters and categorized the houses in the data set as small, moderate and large houses.

Fig. 5.2 Sum of Squared Error vs K value

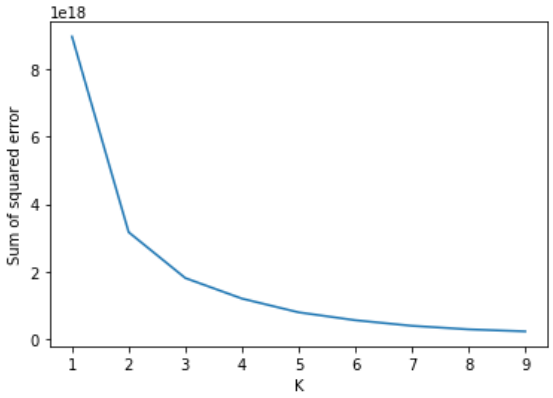


Figure 5.3 shows the data description for cluster 1 that is categorized as small houses which contains 6813 instances with average price 22 Million LKR, beds 4, baths 3, hous_size 2478 square feet and land_size 10 perches.

Fig. 5.3 Data Description for Small Houses

	location	price	beds	baths	house_size	land_size
count	6813.000000	6.813000e+03	6813.000000	6813.000000	6813.000000	6813.000000
mean	25.249376	2.184571e+07	3.851314	3.011889	2478.316998	9.949441
std	11.620980	1.064403e+07	1.073652	1.119021	994.905373	6.028462
min	0.000000	1.300000e+06	1.000000	1.000000	174.240000	1.000000
25%	20.000000	1.370000e+07	3.000000	2.000000	1650.000000	7.300000
50%	28.000000	2.100000e+07	4.000000	3.000000	2500.000000	9.000000
75%	35.000000	2.850000e+07	4.000000	4.000000	3014.000000	10.500000
max	39.000000	5.150000e+07	10.000000	10.000000	12000.000000	200.000000

Figure 5.4 shows the data description for cluster 2 that is categorized as moderate houses which contains 606 instances with average price 81 Million LKR, beds 5, baths 4, hous_size 4252 square feet and land_size 19 perches.

Fig. 5.4 Data Description for Moderate Houses

	location	price	beds	baths	house_size	land_size
count	606.000000	6.060000e+02	606.000000	606.000000	606.000000	606.000000
mean	24.052805	8.128743e+07	4.968647	4.160066	4251.992162	18.637954
std	11.066464	2.401173e+07	1.471719	1.369969	1582.584549	12.460359
min	0.000000	5.200000e+07	2.000000	1.000000	950.000000	4.000000
25%	16.250000	6.025000e+07	4.000000	3.000000	3000.000000	11.200000
50%	27.000000	7.500000e+07	5.000000	4.000000	4000.000000	15.205000
75%	32.000000	9.800000e+07	6.000000	5.000000	5000.000000	21.000000
max	39.000000	1.450000e+08	10.000000	10.000000	12000.000000	181.000000

Figure 5.5 shows the data description for cluster 3 that is categorized as large houses which contains 158 instances with average price 210 Million LKR, beds 5, baths 5, hous_size 5716 square feet and land_size 22 perches.

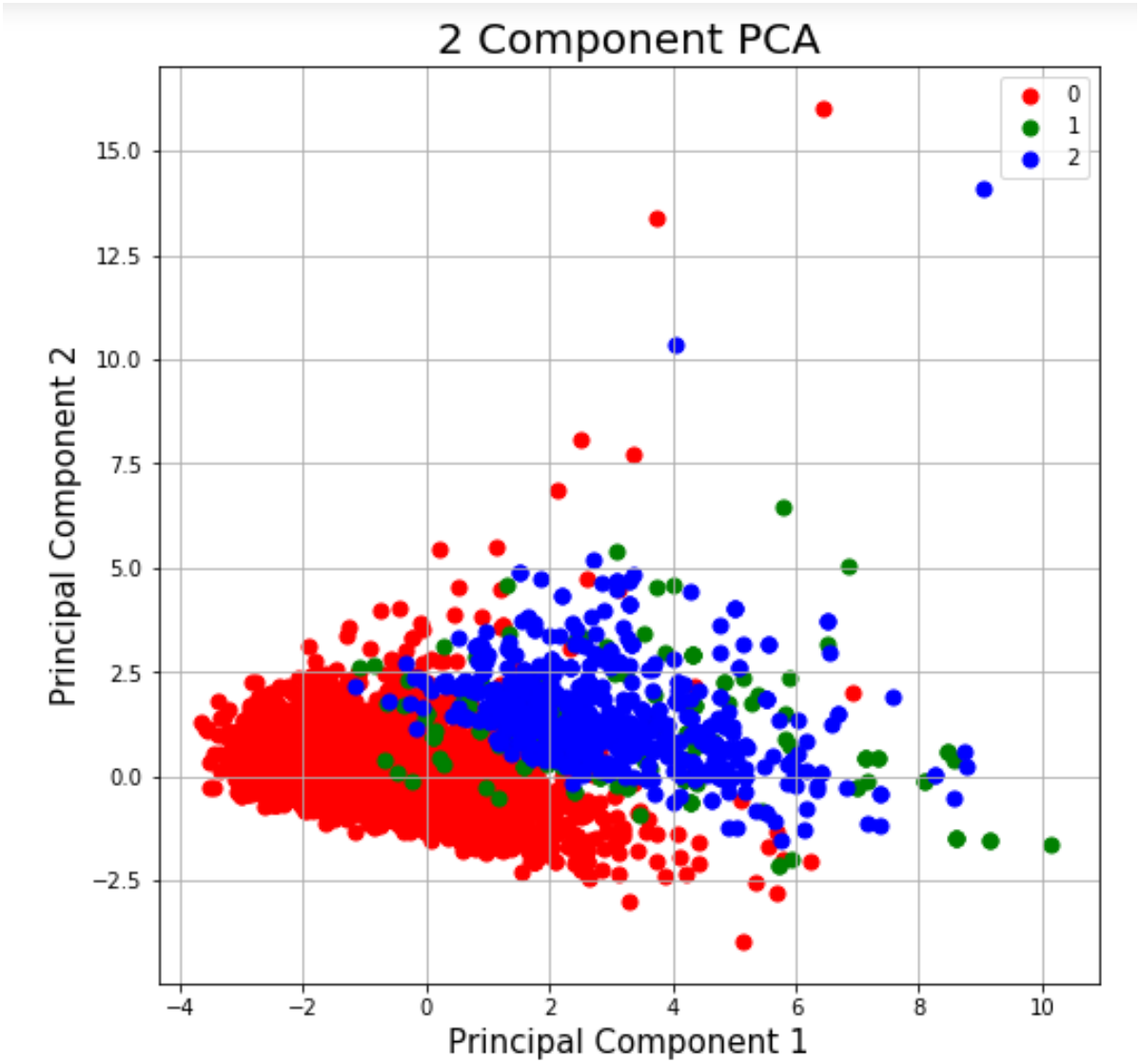
Fig. 5.5 Data Description for Large Houses

	location	price	beds	baths	house_size	land_size
count	158.000000	1.580000e+02	158.000000	158.000000	158.000000	158.000000
mean	17.025316	2.095380e+08	5.265823	4.721519	5715.975949	22.233038
std	12.096717	6.663630e+07	1.698323	1.891562	2946.239130	13.323158
min	3.000000	1.475000e+08	1.000000	1.000000	1000.000000	4.000000
25%	4.000000	1.650000e+08	4.000000	3.000000	3500.000000	14.125000
50%	13.000000	1.800000e+08	5.000000	5.000000	5150.000000	20.000000
75%	31.750000	2.293750e+08	6.000000	6.000000	7225.000000	25.000000
max	39.000000	5.250000e+08	10.000000	10.000000	14000.000000	90.000000

After that we visualized the clusters using Principal Component Analyze(PCA). We got optimal number of principal components as five but to visualize the data we used three and two as number of principal components. Figure 5.6 shows the cluster separation

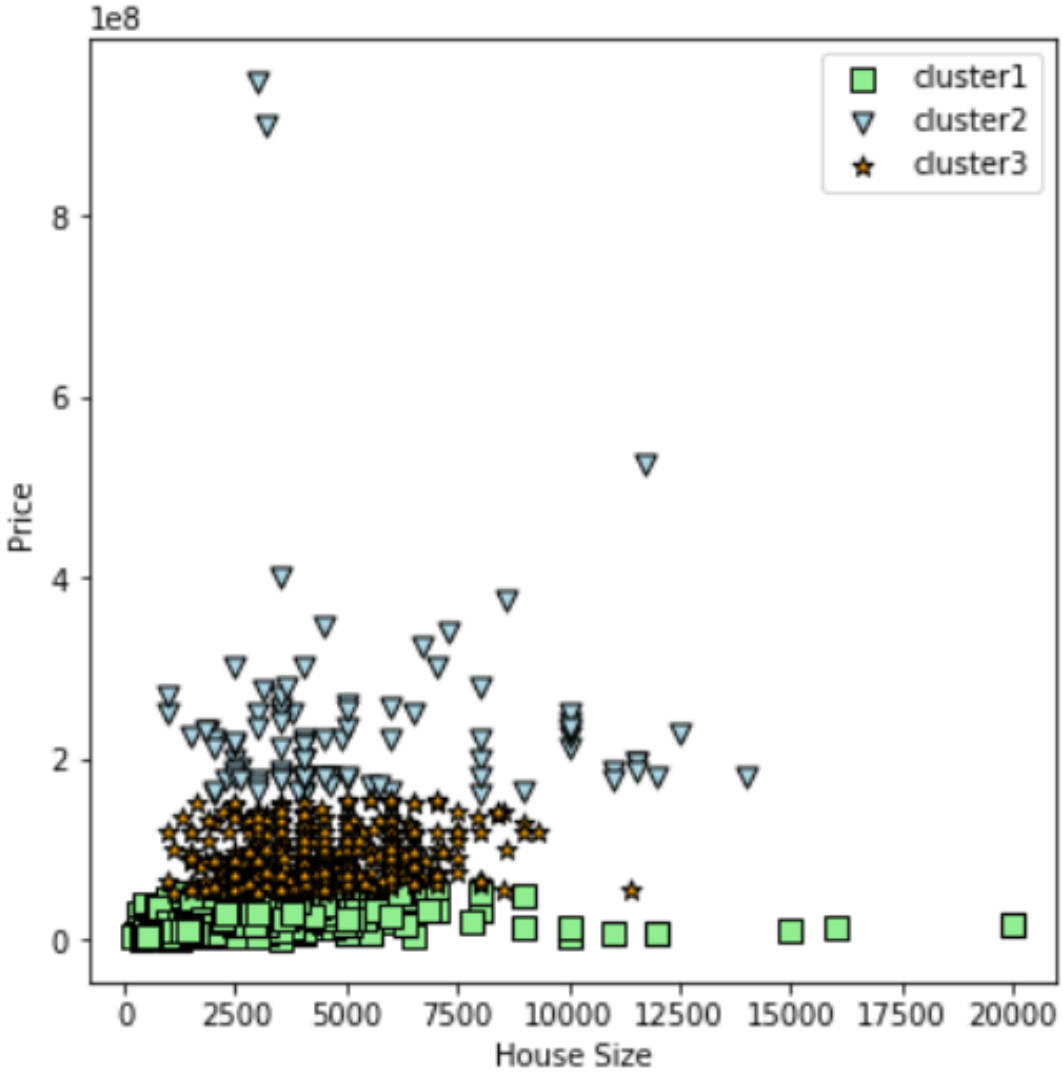
using two principal components. Red, blue and green colour instances indicate the small, moderate and large houses respectively.

Fig. 5.6 Cluster Visualization using PCA



It can be seen that there is a slight separation between small houses and moderate houses. As number of large house instances is small, we could not see that cluster clearly in this plot. We have tried using principal component as three, even in the 3D plot we got same visualization with slight changes. So, we have plot house size again price. As shown in Figure 5.7, we can see the separation between each clusters since house size is the highly correlated feature with the price.

Fig. 5.7 Cluster Visualization using House size and Price



5.4 Prediction Model

5.4.1 Train Test Split

The Initial experiment was about building a Linear Regression model to identify relationships between variables. We conducted the experiment with whole dataset and single story houses. We got positive R2 score for both models but the RMSE values were very high (24 Million and 19 Million). And we noticed our Linear Regression model scored high RMSE for whole houses' data set compared to single story houses. We found one

possible reason could be the single story houses have similar features. We concluded that the independent variables do not have a linear relationship with house price. The plot with each and every variables also confirmed non-existence of linear relationship. Table 5.1 shows the Linear Regression model result with whole data set and single story data set.

Table 5.1 Train Test Split: Result of Linear Regression before Clustering

Dataset	R2	RMSE (Million)
Whole story houses	0.53	24673689.44
Single story houses	0.60	19958929.58

Like the Linear Regression model, we conducted the experiments with both whole data set and single story houses using Random Forest regression algorithm. Similar to Linear Regression model, the single story house model scored lower RMSE compared to whole data set model. The mean prices of single story houses and whole houses were 24 Million and 30 Million respectively. The same reason that the similar features between single story houses could affect single story model performance. We noticed the RMSEs were very close to the mean prices. The assumptions that we made (sold price = advertised price) and existence of outliers affect the models negatively. Table 5.2 shows the Random Forest model result with whole data set and single story data set.

Table 5.2 Train Test Split: Results of Random Forest Model before Clustering

Data set	R2		RMSE (Million)	
	Min	Max	Min	Max
Whole houses	0.04	0.40	24	30
Single story houses	0.19	0.36	21	24

We found selecting number of cluster three gave good accuracy with all the attributes. We trained the model with each cluster separately but only small house cluster gave good accuracy since number of instances in the other two clusters(moderate and large) are very small compared to small house cluster. So we assumed these two clusters are outliers and removed from the data set and trained the model with small house data.

From the results, we concluded that considering all the variables in clustering would result a better clustering model. In terms of metrics (R2, RMSE), the model with clustered data (removed outliers with clustering) performed better than previously built models. Table 5.3 summarises the result of Random Forest model after clustering.

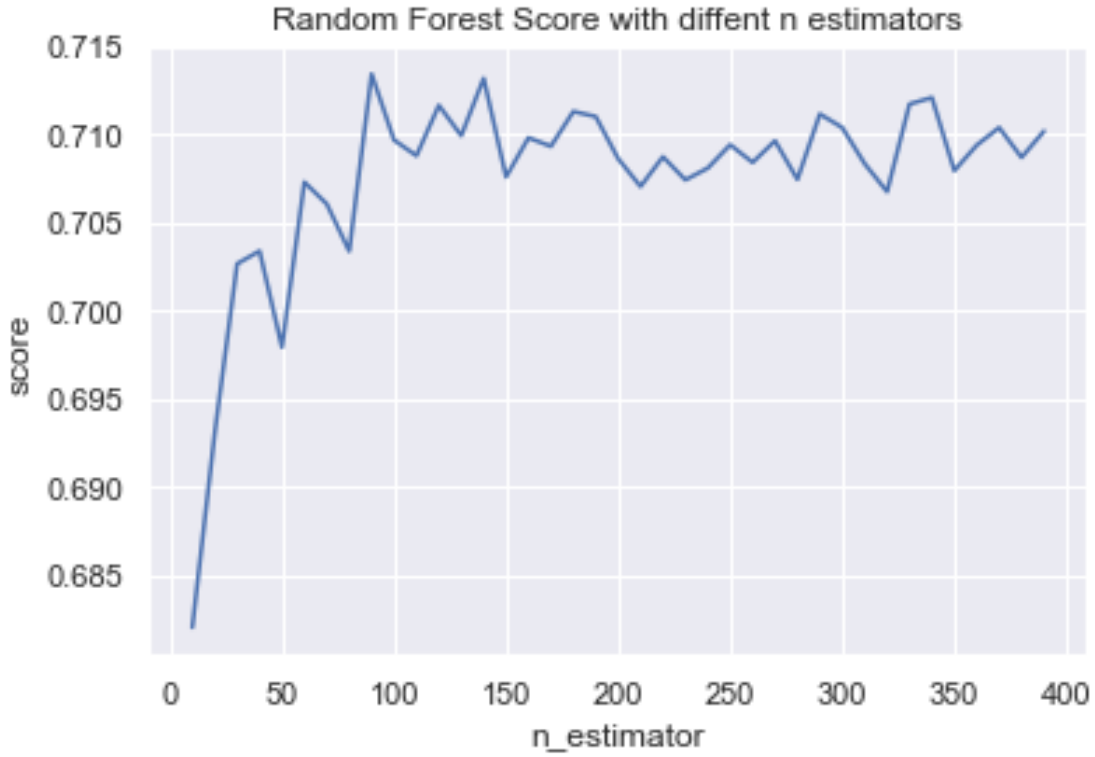
Table 5.3 Train Test Split: Results of Random Forest Model after Clustering

Clustering	Number of Clusters	R2	RMSE(Million)
All Attributes except Price	2	0.28	10
	3	0.20	9.7
All Attributes	2	0.68	8.9
	3	0.70	5.2

The best models scored R2 of 0.70 and RMSE of LKR 5.2 Million (average of 10 trials). The 5.2 Million (RMSE) is 23.8% of the average price (21.8 Million) of houses.

Figure 5.8 shows the Random Forest best model R2 score with different n estimators.

Fig. 5.8 Train Test Split: Random Forest Result with different $n_estimators$



5.4.2 Cross Validation

From the previous experiments we conclude that considering all attributes and number of clusters 3 gives a better clustering and prediction model. So we have used the same best parameters for cross validation method.

Table 5.4 shows the Random Forest result with various approaches we have tried with different data sets using cross validation method.

Table 5.4 Cross Validation: Result with Different Data Sets

Method	Random Forest		Linear Regression	
	R2	RMSE (Million)	R2	RMSE (Million)
Before Clustering	0.69	18.57	0.60	21.38
With cluster 1: small houses	0.75	5.27	0.51	7.44
With cluster 2: moderate houses	0.42	17.67	0.05	22.98
With cluster 3: large houses	0.28	51.19	-0.24	64.25

As shown in table 5.4 small houses data set after clustering performed well. Number of instances in other two clusters moderate houses and large houses are very small. so we did not get good accuracy for those clusters. Therefore we assumed those are outliers and continued the parameter tuning with small houses data set.

We removed the outliers within small houses cluster usnig Z-score method. It increased accuracy further. As shown in table 5.5 increasing the cross validation value gives better accuracy. Since chance of all the data instances contributes to the training and testing set increases, it performs well on unseen data when testing. Even the accuracy increases with cross validation value, we select 10 as cv value since increasing cv value further leads to over fit the model. There is only slight increment in the accuracy when we increase cv beyond 10.

Table 5.5 Cross Validation: Result for Small Houses

CV	Random Forest		Linear Regression	
	R2	RMSE (Million)	R2	RMSE (Million)
2	0.71	5.63	0.53	7.17
5	0.75	5.23	0.53	7.16
10	0.76	5.12	0.53	7.14
15	0.76	5.06	0.53	7.14
20	0.76	5.08	0.53	7.13

As shown in table 5.6 after tune the parameters for Random Forest we got best model with these parameters with the highest score about 0.77 compared to previous experiments. Using GridSearch CV we got Random Forest best parameters for our model.

Table 5.6 Cross Validation: Random Forest Parameters for Best Model

Parameters	Values
mas_depth	20
max_features	4
n_estimators	100

Outliers removal within the cluster using Z-score method further increased the accuracy. The best models scored R2 of 0.76 and RMSE of LKR 5.1 Million (average of 10 trials). The 5.1 Million (RMSE) is 23.4% of the average price (21.8 Million) of houses.

5.4.3 Train Test Split Vs Cross Validation

Cross validation method R2 score and RMSE was better than train test split method. Because cross validation gives the opportunity train on multiple train-test split. It gives cross validation model to perform well on unseen data. Train test split, on the other hand, depends on one train-test split which makes this model score dependent on how the data is split into train and test sets. Table 5.7 shows the best results for train test split method and cross-validation method.

Table 5.7 Random Forest Best Score

Method	R2	RMSE (Million)
Train Test Split	0.70	5.20
Cross Validation	0.76	5.12

Chapter 6

Conclusions and Future Works

The house price prediction using Machine Learning shows a thriving trend. In this paper we explored the way to predict house prices using Random Forest regression with an approximate data set. It consists set of processes to build a suitable Random Forest house price prediction model by comparing prediction accuracy and performance. We were successful in predicting house prices in Colombo. This is the first machine learning house price prediction for Sri Lanka.

We collected data from an online advertising platform and we made an assumption that the advertised prices are the same as sold prices. We cleaned data, removed outliers and extreme values' instances.

House size has been the factor highly affecting the price with high correlation value with price. Baths, beds, land size, location are also have relatively high correlation with price than other features.

This research has conducted experiment using Random Forest regression and K-Means clustering algorithms. We used clustering to remove outliers. We concluded that, the clustering techniques works well in detecting outliers, where an instance has high degree of chance to be an outlier. We found out that the assumption we made and undetected outliers causes Radom Forest model performance.

Random Forest model was trained with two data splitting methods those are train test split and cross validation. Results were compared using evaluation metrics R2 and RMSE. We have concluded training Random Forest with cross validation technique performs better than train test split method.

Out of these two methods, cross validation offered the best model which scored R2 of 0.76 and RMSE of LKR 5.1 Million (average of 10 trials) which is 23.4% of the average price (21.8 Million) of the houses we considered.

It's good to understand the people's behavior in posting house advertisements. Some people would advertise a lesser price than the actual in order to sell fast and some advertise a higher price as a negotiation tactic. This kind of unpredictable behavior caused high RMSE of models.

Our study was done with some limitations, such as approximate data set and data set is limited to a small period of time. We assumed the two small clusters (moderate houses and large house) are outliers and exclude those from experimental data set. But they can be included in the analysis if we can get enough data instances and the study can be extended to all types of houses. We only considered the common factors that affects housing price. To further explore, it is useful to consider real data set, time series analysis and deep learning models. Considering a wide range of factors (GDP, GDP growth rate, per capita, etc.) along with time series data would improve performance of the model.

References

[1] R. Manjula, S. Jain, S. Srivastava, and P. Rajiv Kher, "Real estate value prediction using multivariate regression models," in *Materials Science and Engineering Conference Series*, vol. 263, p. 042098, 2017.

[2] N. Shinde and K. Gawande, "Survey on predicting property price," in *2018 International Conference on Automation and Computational Engineering (ICACE)*, pp. 1–7, IEEE, 2018.

[3] H. Xu and A. Gade, "Smart real estate assessments using structured deep neural networks," in *2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, pp. 1–7, IEEE, 2017.

[4] V. Limsombunchai, "House price prediction: hedonic price model vs. artificial neural network," in *New Zealand agricultural and resource economics society conference*, pp. 25–26, 2004.

[5] A. Baldominos, I. Blanco, A. J. Moreno, R. Iturrarte, Ó. Bernárdez, and C. Afonso, "Identifying real estate opportunities using machine learning," *Applied Sciences*, vol. 8, no. 11, p. 2321, 2018.

[6] A. Ng and M. Deisenroth, "Machine learning for a london housing price prediction mobile application," *Imperial College London*, 2015.

[7] J. Wang, S. Hu, X. Zhan, Q. Luo, Q. Yu, Z. Liu, T. P. Chen, Y. Yin, S. Hosaka, and Y. Liu, "Predicting house price with a memristor-based artificial neural network," *IEEE Access*, vol. 6, pp. 16523–16528, 2018.

[8] O. Kitapci, Ö. Tosun, M. F. Tuna, and T. Turk, "The use of artifi cial neural networks (ann) in forecasting housing prices in ankara, turkey," *Journal of Marketing and Consumer Behaviour in Emerging Markets*, no. 1 (5), pp. 4–14, 2017.

[9] V. Chiarazzo, L. Caggiani, M. Marinelli, and M. Ottomanelli, "A neural network based model for real estate price estimation considering environmental quality of property location," *Transportation Research Procedia*, vol. 3, pp. 810–817, 2014.

[10] Y. F. Chang, W. C. Choong, S. Y. Looi, W. Y. Pan, and H. L. Goh, "Analysis of housing prices in petaling district, malaysia using functional relationship model," *International Journal of Housing Markets and Analysis*, 2019.

[11] "Lanka Property Web." https://www.lankapropertyweb.com/.

[12] J. Żak, "14th meeting of the euro working group on transportation (ewgt)-in quest for advanced models, tools and methods for transportation and logistics. editorial," 2011.

[13] B. Afonso, L. Melo, W. Oliveira, S. Sousa, and L. Berton, "Housing prices prediction with a deep learning and random forest ensemble," in *Anais do XVI Encontro Nacional de Inteligência Artificial e Computacional*, pp. 389–400, SBC, 2020.

[14] J. Hong, H. Choi, and W.-s. Kim, "A house price valuation based on the random forest approach: the mass appraisal of residential property in south korea," *International Journal of Strategic Property Management*, pp. 1–13, 2020.

[15] C. Wang and H. Wu, "A new machine learning approach to house price estimation," *New Trends in Mathematical Sciences*, vol. 6, no. 4, pp. 165–171, 2018.

[16] W. Tan and T.-N. Chou, "Combine grey relational analysis and weighted synthesis for housing price prediction," 2016.

[17] J. Y. Wu, "Housing price prediction using support vector regression," 2017.

[18] D. S. D. Dhvani Kansara, Rashika Singh, "Improving accuracy of real estate valuation using stacked regression."

[19] A. Nguyen, "Housing price prediction," 2018.

[20] L. Yu, C. Jiao, H. Xin, Y. Wang, and K. Wang, "Prediction on housing price based on deep learning," *International Journal of Computer and Information Engineerin*, vol. 12, no. 2, pp. 90–99, 2018.

[21] J. Mu, F. Wu, and A. Zhang, "Housing value forecasting based on machine learning methods," in *Abstract and Applied Analysis*, vol. 2014, Hindawi, 2014.

[22] P. Picchetti, "Hedonic residential property price estimation using geospatial data: a machine-learning approach," *Instituto Brasileiro de Economia*, vol. 4, 2017.

[23] J. Oxenstierna, "Predicting house prices using ensemble learning with cluster aggregations," 2017.

[24] I. S. H. Bahia, "A data mining model by using ann for predicting real estate market: Comparative study," *International Journal of Intelligence Science*, vol. 3, no. 04, p. 162, 2013.

[25] J. Frew and G. Jud, "Estimating the value of apartment buildings," *Journal of Real Estate Research*, vol. 25, no. 1, pp. 77–86, 2003.

[26] Y. Li and H. Wu, "A clustering method based on k-means algorithm," *Physics Procedia*, vol. 25, pp. 1104–1109, 2012.

[27] G. Biau, "Analysis of a random forests model," *Journal of Machine Learning Research*, vol. 13, no. Apr, pp. 1063–1095, 2012.

[28] M. Cocea and S. Weibelzahl, "Log file analysis for disengagement detection in e-learning environments," *User Modeling and User-Adapted Interaction*, vol. 19, no. 4, pp. 341–385, 2009.

[29] M. N. Sadiku, A. E. Shadare, S. M. Musa, and C. M. Akujuobi, "Data visualization," *International Journal of Engineering Research And Advanced Technology (IJERAT)*, vol. 2, no. 12, pp. 11–16, 2016.

[30] T. Wang and Z. Li, "Outlier detection in high-dimensional regression model," *Communications in Statistics-Theory and Methods*, vol. 46, no. 14, pp. 6947–6958, 2017.

[31] A. Christy, G. M. Gandhi, and S. Vaithyasubramanian, "Cluster based outlier detection algorithm for healthcare data," *Procedia Computer Science*, vol. 50, pp. 209–215, 2015.

[32] L. Fernández-Durán, A. Llorca, N. Ruiz, S. Valero, and V. Botti, "The impact of location on housing prices: applying the artificial neural network model as an analytical tool.," 2011.