Machine Learning Approach to Predict Real Estate Prices

Anil Kumar K.M [a], Anil B [b], Anand C U [b], Aniruddha S [b], Rajath Kumar U [b]

[a] Associate Professor, Department of CS & E, SJCE, Mysore- 570006, India(Email: anilkm07@gmail.com)
[b] Department of CS & E, SJCE, Mysore- 570006, India
(Email: anilb94@gmail.com, anandcu3@gmail.com, aniruddha.9794@gmail.com, rok.rajath@gmail.com)

***Abstract*—In this paper we discuss an approach to find the better classifier to predict price of the house in and around Bangalore. We have considered the attributes house size, number of bedrooms and the status of the house. The dataset used is extracted from real estate websites and then analyzed using software called WEKA. We have applied Linear Regression and other classifiers like, Decision Trees (J48), Nearest Neighbour (IBk), Artificial Neural Network (Multilayer Perceptron), OneR, Naive Bayes, SMO and LibSVM. Based on the accuracy results obtained, we conclude that Naive Bayes is consistent for unequal frequency distribution with consistency value 75.47% and J48 (decision tree) is the most consistent classifier for equal frequency distribution with consistency value 44.2% predicting the house price.**

1. INTRODUCTION

Nowadays the real estate market is growing exponentially and this makes it hard to predict the prices of houses for buying/selling. Changes in housing prices concern both individuals and government since they have substantial influence on the socio-economic conditions [7].

There are two kinds of factors which will affect the real estate price. They are general factor and particular factor. The general factor will not directly affect the price of particular property. But this kind of factor will affect the price trend of real estate market. It mainly consists of the supply and requirement, population, social, administration, and other accidental factors. The particular factor is the one which will directly affect the price of an actual property. It mainly consists of construction structure, style, level, facilities, and construction quality. The two most crucial factors are location and environments.

In general, the effect of mentioned factors on real estate price is very complicated and uncertain. It is very difficult to decide which factor is the most crucial one [2]. Hence we consider a combination of factors like house size, number of bedrooms, and the state of the house (new or resale).

Real Estate is an important market for business purposes. A lot of stake-holders like real estate developers, banks, policy makers, etc. are involved in buying/selling of houses.

In the real world, ordinary people will have to visit each website of real estate office manually which would provide different prices. This is a complex and time consuming process and the information obtained may not be accurate. People may also depend on experienced vendors (brokers) for getting the information about the price of the houses. The brokers may take advantage of this by quoting an unfair price. Therefore people will have to approach different vendors to get an accurate price which is a tedious job. Different pricing of the same property by different vendors is also one of the problems for buyers. Hence we have developed a tool to assist ordinary people to find an approximate price of house at location of their interest.

The main focus of this paper is on buying or selling of houses. However this concept can be extended to other aspects of real estates like sites, agricultural lands etc.

The rest of the paper is organised as follows. Section 2 focuses on Related Works. Section 3 explains the Methodology. Section 4 elaborates the Experiments and Results. Section 5 provides the Conclusion.

2. RELATED WORKS

Hari Arul and Andres Morales [8] provided an approach to divide the given dataset with attributes like Total Units, Building Classification, Year Built, Gross Income, Expense per square foot, Latitude and Longitude into a definite range so that each range is of the same width and then apply the classifier to predict the data. However they could not get accurate results to predict the prices.

Carlos del Cacho [6] provided analysis of hedonic and non-hedonic classifiers to the housing evaluation in the city of Madrid. They have considered 25415 properties with attributes like price, size, number of bedrooms etc. and concluded that M5 model trees are better algorithm.

Itedal Sabri Hashim Bahia [9] did estimation of prices of houses using Feed Forward Neural Network (FFNN) and Cascade Forward Back Propagation (CFBP) neural network. It concludes that CFBP neural network is better. They have considered attributes like average number of rooms per dwelling, index of accessibility to radial highways, etc.

Andrew Caplin, Sumit Chopra, John Leahy, Yann LeCun, and Trivikrmaman Thampy [3] have considered attributes like sale price, record date, sale date. They have showed that errors in estimation can be greatly reduced if geography is taken into account.

Our paper differs from the above mentioned studies in that, we have classified the selling price by two ways, one by specifying the number of instances in each bin so that the bins have equal frequency and by unequal frequency distribution. We have also considered 3 different datasets. The first dataset includes house details of Indiranagar, Bangalore. The second dataset is the combination of foreign house data [17] and Indiranagar data, where the former is used for training and the latter is used for testing. The third dataset includes house data of several areas in and around Bangalore.

3. METHODOLOGY

3.1 Procedure

The prediction is based on data mining. Data mining refers to the mining or discovery of new information in terms of patterns or rules from vast amounts of data [1]. The function y = f(x_1, x_2,x_3... x_n) is used to predict y i.e., output factor, the selling price of houses. x_1, x_2, x_3… x_n stand for the factors which affect the real estate price (we only analyse particular factors in our research) [2].

The first step in data mining is to obtain a Dataset which can be used as training sample from which we analyse the attributes. We have used three separate datasets for our experiments. The data is extracted from Web Scrapping Technique using a tool called Jaunt [14], a Java API. The first dataset consists of 680 instances. The second dataset consists of 1444 instances. The third dataset consists of 2000 instances.

The first Dataset containing the house details of Indiranagar, Bangalore is extracted from the websites [16] [15] [12] [13]. The attributes of this dataset are House Size, Number of Bedrooms, New/Resale and Selling Price.

The second dataset is a combination of the datasets San Luis Obispo County [17] and first dataset. The San Luis Obispo dataset consists of the following attributes like multiple listing service number, Location, Price, Bedrooms, Bathrooms, Size, Price/Sq.Ft, and Status. Only common attributes between the first dataset and San Luis Obispo dataset (Price, Bedrooms, Size and Status) are retained and the others are removed. This processed dataset is used for training and the first dataset is used for testing.

The third dataset is an extension of the first dataset. It includes an additional attribute called “area” which specifies the locality of interest in and around Bangalore (Indiranagar, HSR Layout, JP Nagar, Whitefield, Koramangala and Hebbal).

Usually, the data obtained from websites are not directly usable since real world data will have a lot of noise (invalid and empty entries). To remove noise, status of the house was converted to a nominal value (0 or 1). Then, duplicates and empty entries were removed. After cleaning the data we use software called WEKA for analysing it. Weka is a collection of machine-learning algorithms for data mining tasks like regression, classification, clustering etc... The algorithms can be either applied directly to a dataset or through java interface.

The attribute that we predict is called the class attribute which is selling price in our experiment. Linear Regression method, which returns the numeric value is simple and does not over fit the data. The data can be applied directly for Linear Regression since class attribute that is, selling price is numeric. It returns the predicted selling price. For classification, methods like Decision Trees (J48), Nearest Neighbour (IBk), Artificial Neural Network (Multilayer Perceptron), OneR, Naive Bayes, SMO and LibSVM are considered for the experiments. We have used Discretize filter that sorts the data into required number of bins which converts numeric class attribute into nominal values. (We selected 7 bins, we found the accuracy to be same for bin values between 5 to

10) We have discretized the data in two methods. In the first method, we don't specify the number of instances per bin whereas in the second method, the number of instances per bin is specified and it is based on equal frequency distribution. We have run the algorithms for both the methods. For the first and third dataset, we have used 10 fold Stratified Cross Validation. It divides the dataset into 10 parts and trains the classifier with 9 parts and the other part is used for testing. This is repeated using different part as test data every time. Hence the classifier runs 10 times. Then the outputs are analysed. Whereas for the second dataset, we have used percentage split for training and testing of data.

3.2 TOOLS

3.2.1 WEKA

WEKA is abbreviation of Waikato Environment for Knowledge Analysis. It is open source software which is a popular suite of machine learning software written in Java, developed at the University of Waikato [5].

3.2.2 JAUNT

Jaunt is a new, free, Java library for web-scraping and web-automation. The library provides an ultra-light headless browser (i.e., no GUI). By using Jaunt API our Java programs can easily perform browser-level operations [14]. Jaunt is the ideal tool for

a. Filling out and submitting forms.
b. Creating web-scraping programs.
c. Interfacing with web-apps (HTML, XHTML or XML).

4. EXPERIMENTS & RESULTS

We have applied Linear Regression which results in the predicted selling price of the house, a numeric value. Other classifiers like J48, IBk, MLP, OneR, Naïve Bayes, SMO and LibSVM are also applied to predict the range of selling prices.

4.1 Algorithms Applied

4.1.1 Linear Regression

Regression in data mining is a method to predict numerical value of a class attribute (selling price). If the attributes of a sample are represented as $a_{0,}$ $a_{1\dots}$ a_k and the required class attribute is represented as p, then the value of p can be obtained by EQ. 1.

$$p = w_0\, a^1_0 + w_1\, a^1_1 + w_2\, a^1_2 + \dots + w_k \qquad (1)$$

Where w_0, $w_1 \dots$ w_k are some constants. The constants are selected such that the squared error is minimum on training data.

This formula for linear regression for the first dataset is obtained using WEKA using the formula EQ. 2.

$$\mathit{sellingPrice} = w_0 * \mathit{houseSize} + w_1 * \mathit{bedrooms} + w_2 * \mathit{new} + w_3 \qquad (2)$$

where w_0 = 12971.9378, w_1 = 166161.7906, w_2 = -253794.5465, w_3 = -6509319.1601 and houseSize is the area of the house in square feet , bedrooms represents number of bedrooms in the house, new attribute denotes weather the house is a newly built house or an old house.

The constants w_0, $w_1 \dots$ w_k depend on the data and thus vary from one dataset to another. Hence we get different formulae for second and third dataset.

We have obtained the Relative absolute Error as 44.74% for the first dataset which can be calculated using the formula EQ. 3.

$$\frac{|\,p_1 - a_1| + \cdots + |\,p_n - a_n|}{|\,a_1 - \overline{a}\,| + \cdots + |\,a_n - \overline{a}\,|} \qquad (3)$$

The Root mean squared error was obtained as 13896934.84 for the first data set which can be calculated using the formula EQ. 4.

$$\sqrt{\frac{(p_1 - a_1)^2 + \cdots + (p_n - a_n)^2}{n}} \qquad (4)$$

Figure.1 shows the graph of actual selling price (X-Axis) vs predicted selling price (Y-Axis) for all the instances in our dataset. The values (Predicted Selling Price) are accurate if they lie close to the line X-Y=0.

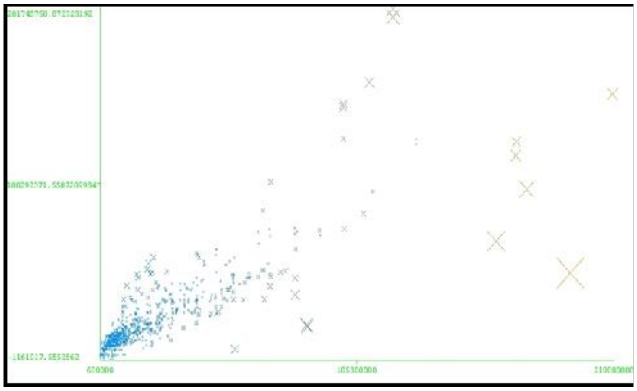

Figure 1 : Graph of expected Selling Price vs Predicted Selling Price

The regression equation (EQ.2) as shown in Figure 1 varies for other datasets and they can be obtained in a similar way.

4.1.2 Classification (J48)

It divides the data into several subgroups. Each subgroup belongs to a common class. Classification technique can be applied for the given datasets using a classifier called as “J48” available in Weka. In order to use J48, the class attribute (selling price) must be of nominal values. J48 classifier is a simple decision tree for classification. It creates a binary tree. With this technique, a tree is constructed as a result of classification process. The output of this classifier can be analyzed using Correctly Classified Instances (CCI), Confusion Matrix, and Decision Tree.

The outputs in the form of correctly classified instances for the first, second and third datasets are 85.44%, 65.01%, 79.25%, for unequal frequency and 39.11%, 32.28%, 54.55% for equal frequency distribution respectively.

4.1.3 Instance Based Learning (IBk)

IBk is a nearest neighbor classification algorithm. It decides which class a given sample belongs based on the distance between the sample and the classes. Since it is a classification algorithm, the class attribute of the dataset must be a nominal one.

The outputs in the form of correctly classified instances for the first, second and third datasets are 83.23%, 56.60%, 79.1% for unequal frequency and 39.41%, 31.53%, 53.65% for equal frequency distribution respectively.

4.1.4 Multi-Layer Perceptron (MLP)

It is a type of classifier that is based on neural networks. A multilayer perceptron (MLP) is a feed-forward artificial neural network model that maps sets of input data onto a set of appropriate outputs.

The outputs in the form of correctly classified instances for the first, second and third datasets are 83.52%, 64.11%, 79.4% for unequal frequency and 42.94%, 37.38%, 58.15% for equal frequency distribution respectively.

4.1.5 OneR

OneR, short for “One Rule” is a simple yet accurate, classification algorithm that is a level-1 decision tree that tests one particular attribute.

The outputs in the form of correctly classified instances for the first, second and third datasets are 83.67%, 64.86%, 76.3% for unequal frequency and 41.47%, 27.32%, 46.2% for equal frequency distribution respectively.

4.1.6 Naive Bayes

Naive Bayes classifiers are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naive) independence assumptions between the features.

The outputs in the form of correctly classified instances for the first, second and third datasets are 82.05%, 68.61%, 78.2% for unequal frequency and 39.70%, 37.98%, 45.1% for equal frequency distribution respectively.

4.1.7 SMO

Sequential minimal optimization (SMO) is an algorithm for solving the quadratic programming (QP) problem that arises during the training of support vector machines. SMO is widely used for training support vector machines and is implemented by the popular LIBSVM tool.

The outputs in the form of correctly classified instances for the first, second and third datasets are 76.76%, 65.76%, 74.6% for unequal frequency and 30.14%, 22.52%, 41% for equal frequency distribution respectively.

4.1.8 LibSVM

LibSVM runs faster than SMO. LibSVM allows users to experiment with one-class SVM.

The outputs in the form of correctly classified instances for the first, second and third datasets are 81.91%, 57.20%, 76.45% for unequal frequency and 35.29%, 13.96%, 47.2% for equal frequency distribution respectively.

4.2 Tabulation of Results

Accuracy and Correctly Classified Instances of different classifiers for different datasets with unequal frequency distribution is provided in Table 1, Table 2 and Table 3. For equal frequency distribution they are provided in Table 4, Table 5 and Table 6.

Table.1 Results for Dataset 1 with Unequal Frequency Distribution.

Results Classifier	CCI	Accuracy (in %)
J48	581	85.44
IBk	566	83.23
MLP	568	83.52
OneR	569	83.67
NaiveBayes	558	82.05
SMO	522	76.76
LibSVM	557	81.91

Table.2 Results for Dataset 2 with Unequal Frequency Distribution.

Results Classifier	CCI	Accuracy (in %)
J48	433	65.01
IBk	377	56.60
MLP	427	64.11
OneR	432	64.86
NaiveBayes	457	68.61
SMO	438	65.76
LibSVM	381	57.20

Table 3 Results for Dataset 3 with Unequal Frequency Distribution.

Results Classifier	CCI	Accuracy (in %)
J48	1585	79.25
IBk	1582	79.1
MLP	1588	79.4
OneR	1526	76.3
NaiveBayes	1564	78.2
SMO	1492	74.6
LibSVM	1529	76.45

Table 4 Results for Dataset 1 with Equal Frequency Distribution.

Results Classifier	CCI	Accuracy (in %)
J48	266	39.11
IBk	268	39.41
MLP	292	42.94
OneR	282	41.47
NaiveBayes	270	39.70
SMO	205	30.14
LibSVM	240	35.29

Table 5 Results for Dataset 2 with Equal Frequency Distribution.

Results Classifier	CCI	Accuracy (in %)
J48	215	32.28
IBk	210	31.53
MLP	249	37.38
OneR	182	27.32
NaiveBayes	253	37.98
SMO	150	22.52
LibSVM	93	13.96

Table 6 Results for Dataset 3 with Equal Frequency Distribution.

Results Classifier	CCI	Accuracy (in %)
J48	1091	54.55
IBk	1073	53.65
MLP	963	48.15
OneR	924	46.2
NaiveBayes	902	45.1
SMO	820	41
LibSVM	944	47.2

We find out the most consistent classifier using the formula EQ.5 and the resulting consistency and average values for each classifier is as shown in Figure 2 and Figure 3.

$$Consistency = \frac{\begin{pmatrix} Accuracy1\times no.\ of\ Instances\ in\ Data\ Set\ 1 \\ +Accuracy2\times no.\ of\ Instances\ in\ Data\ Set\ 2 \\ +Accuracy3\times no.\ of\ Instances\ in\ Data\ Set\ 3 \end{pmatrix}}{(Total\ number\ of\ instances)}$$

(5)

Figure 2 is plotted for unequal frequency distribution and Figure 3 for equal frequency distribution using consistency and average values.

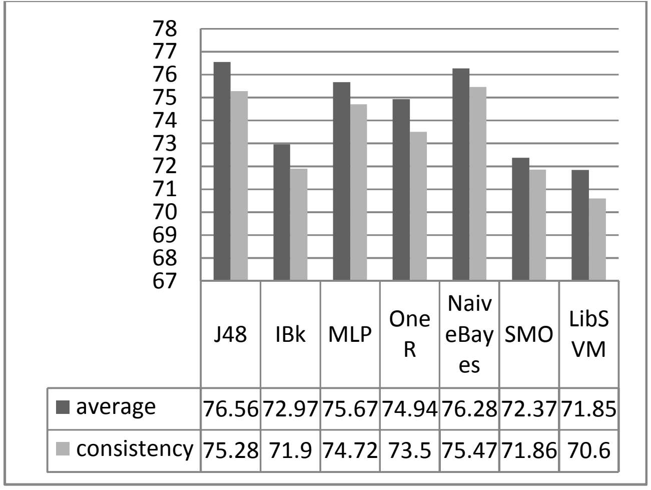

Figure 2: Bar Graph for Unequal Frequency Distribution

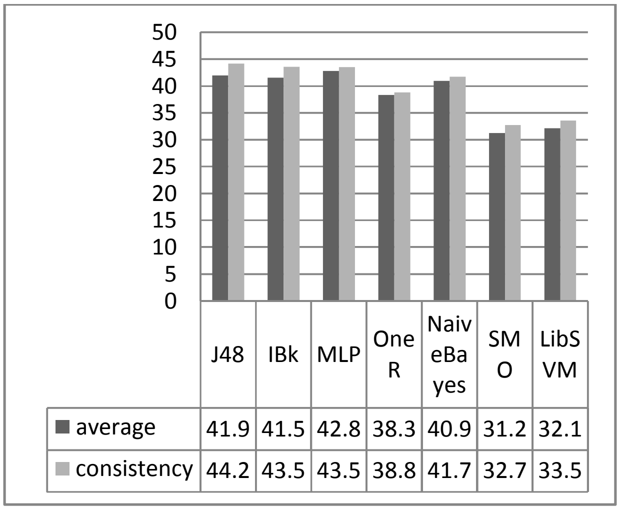

Figure 2 : Bar Graph for Equal Frequency Distribution

Table. 7 Comparison of Results

Authors	Attributes	Instances	Metrics	Results
Simon Fong [10]	23	132	RAE	MP-10.35%
				LR- 84.60%
Dick Stevens [11]	28	10000	RMSE	LR-139,177 (in $)
Claudio Acciani [4]	10	109	RAE	LR- 59.8%
				M5-46.6%
Hari Arul [8]	7	-	Accuracy	NaiveBayes - 18.64%
Our Results	4	680	RAE	44.74%
			RMSE	13896934.8 (in Rupees)
			Accuracy	82.05

Table 7 shows the results of various experiments on House Price Prediction conducted by different authors. For Linear Regression on first dataset we have obtained Relative Absolute Error (RAE) as 44.74% and Root Mean Squared Error (RMSE) as 13896934.84. For NaiveBayes on the first dataset, we have obtained 82.05% accuracy for unequal frequency distribution. We have considered only 4 attributes and if we increase the number of attributes, better results can be obtained.

5. CONCLUSION

We have used an empirical approach to evaluate the selling price of the house using different classifiers. Naïve Bayes and J48 have high consistency values. Hence, these classifiers are better for Unequal Frequency Distribution. J48 has the highest consistency value and hence it is better classifier for Equal Frequency Distribution. Even though the error percentage of linear regression is quite high, it predicts the numeric value of the selling price rather than the range of selling price as in the case of other classifiers. Hence regression is also a desirable one.

This study helps to predict the selling price for the benefit of common people.

REFERENCES

[1] Ramez Elmasri and Shamkanth B. Navathe, Fundamentals of Database Systems, Third Edition.
[2] Zhang Xiao li, Using Fuzzy Neural Network in Real Estate Prices Prediction, 2007.
[3] Andrew Caplin, Sumit Chopra, John Leahy, Yann LeCun, and Trivikrmaman Thampy, Machine Learning and the Spatial Structure of House Prices and Housing Returns, 2008.
[4] Claudio Acciani, Vincenzo Fucilli, Ruggiero Sardaro, Model Tree: An Application in Real Estate Appraisal, 2008.
[5] Arief Rakhman, Goeij Yong Sun, Rama Catur APP, Building Artificial Neural Network Using WEKA Software. 2009
[6] Carlos Del Cacho, A comparison of data mining methods for mass real estate appraisal, 2010.
[7] Reza Ghodsi, Abtin Boostani, Farshid Faghihi, Estimation of Housing Prices by Fuzzy Regression and Artificial Neural Network, 2010.
[8] Hari Arul Andres Morales, NYC Condo Price Estimation Using NYC Open Data, 2013.
[9] Itedal Sabri Hashim Bahia, A Data Mining Model by Using ANN for Predicting Real Estate Market: Comparative Study, 2013.
[10] Simon Fong, Yap Bee Wah, A Prediction Model for Forecasting the trend of Macau Property Price Movements and Understanding the Influential Factors, 2013.
[11] Dick Stevens, Predicting Real Estate Price Using Text Mining, Automated Real Estate Description Analysis, 2014.
[12] http://www.99acres.com
[13] http://www.indiaproperty.com
[14] http://www.jaunt-api.com
[15] http://www.magicbricks.com
[16] http://www.makaan.com
[17] https://wiki.csc.calpoly.edu/datasets/wiki/Houses