Prediction of House Price using Machine Learning Approach

Gaurav[1] Zunaid Aalam[2]
[1]Assistant Professor [2]M.Tech Scholar
[1]SGT University, Gurugram, India [2]Jamia Hamdard University, Delhi, India

Abstract— This paper aims to predict the price of a house on the basis of key parameters such as LotArea, Year built, Overall Quality, LotConfig, GarageArea, GarageType etc. In this paper, we have taken approximately 40 such parameters for prediction purpose. We have taken the data from a website where recent competitions of data science based projects are displayed with very high prize money. The data set contains more than 1400 records. We have used multiple linear regression method for prediction purpose. In regression, we have some features which are taken as predictors and one output which is real value for the given features The benefit of this paper will be to predict the price of a house on basis of key parameters and there will be no need of agents for asking everything regarding real estate prices. This paper will help the customer to know whether they are investing right amount for real estate or not.

Key words: Multiple Linear Regression, Prediction, Data Mining, Machine Learning

I. INTRODUCTION

House price depends on number of parameters such as area, year built, house style, lot shape, condition and many more. It is generally acknowledged that the price of real estate is highly complicated and is interrelated with a multitude of factors [2]. Hedonic price theory assumes that a commodity such as a house can be viewed as an aggregation of individual components or attributes. Consumers are assumed to purchase goods embodying bundles of attributes that maximize their underlying utility function [1]. In this paper, we have taken such 40 parameters which are being considered as predictors for house price. Prediction has always been an exciting feature of data mining that makes it more interesting. As far as data is concerned, it's been taken from a website where recent competitions of data science based projects are displayed with very high prize money. So it increases the validity and correctness of data. The data set contains more than 1400 records. Prediction is always useful before executing the task. Here we have performed prediction on house pricing, so it's obviously useful for customers who are looking for buying the house. An accurate prediction on the house price is important to prospective homeowners, developers, investors, appraisers, tax assessors and other real estate market participants, such as, mortgage lenders and insurers [1]. For the prediction purpose, we have used regression technique, which is kind of supervised learning. In regression, we have some features which are taken as predictors and one output which is real value for the given features. In this paper, we have used machine learning approach along with multiple regressions to prepare the model such as LMS update rule and stochastic gradient descent approach.

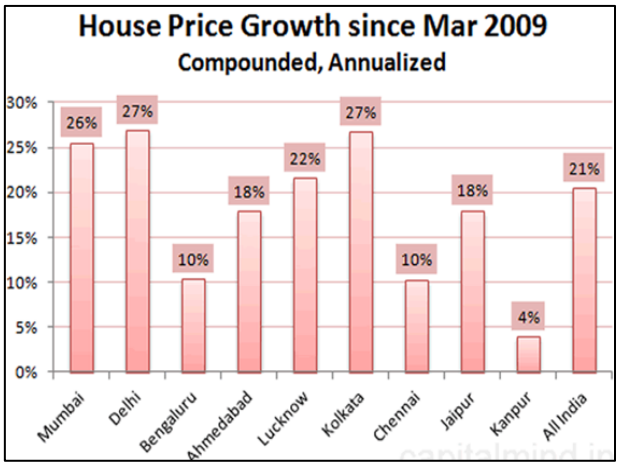

Fig. 1: A Sample Image Showing Growth In House Price among Various Cities of India [10]

II. METHODOLOGY AND TOOLS

The following figure shows the path that we have used for preparing a model for the prediction purpose such as house price prediction:

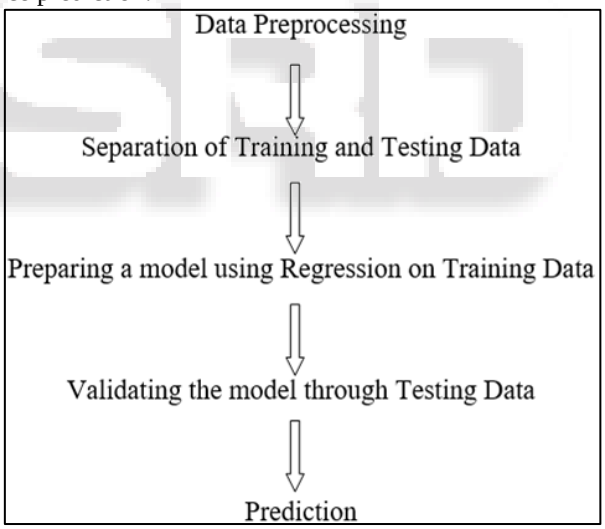

Fig. 2: Path Used For Preparing a Model for Prediction

In data mining, preprocessing plays a very important role because it is the first step which is performed to work on data whether it is about prediction or mining information etc. Noisy and erroneous data makes algorithms useless which we apply to mine data. So before processing, data needs to be investigated and pre-processed. Then only it makes it useful [3]. The most common issue that comes during the process of knowledge discovery through data mining includes the missing values. A dataset with 1 – 5% missing values may not impact as much whereby from 5 – 15% range requires sensitive algorithms to apply [4]. As preprocessing of the data set takes 70% of the time to make data set useful and according to as per requirement of algorithms. So, we have also performed preprocessing on data. As we discussed, that there are approximately 40 parameters, we pre-processed

them first. Some of the features were not in the form of directly use so we make them useful for our purpose. There were some features like Alley which were not given properly so we removed such features from the data. Similarly there were some features which were having common values for all training data so they were like useless so we removed them. Information given in some features were not in required form, we converted those features into required form. At the end, after removal or interchange, we used approximately 35 such features which have been result oriented for us.

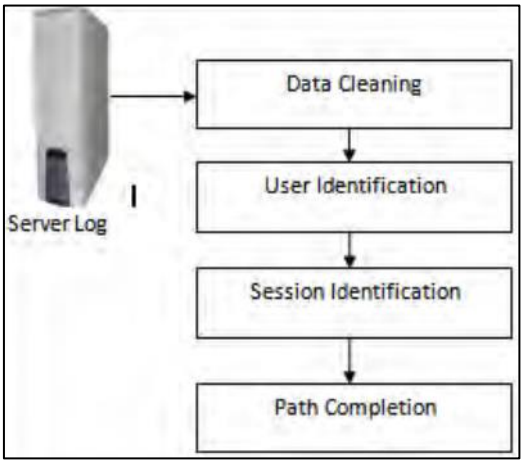

Fig. 3 Stpes Involved in Data Preprocessing [5]

As far as tools are concerned, we used R language for preprocessing. R language is very efficient and useful for such preprocessing tasks. Once the data was ready to work upon, then we applied some regression techniques to achieve our purpose of prediction. Regression may be simple and multiple. In simple regression, there may be linear or non-linear regression. Similarly in case of multiple regression, there may be linear or non-linear regression. But as we had more than 2 features, so we used multiple regression for preparing a model to predict the final value based on input features. In many applications, multiple linear regression can be used such as in the investment estimation of urban bridge engineering etc. [6]. There are some other approach also that can be used for prediction such as Multiple Linear Regression, Random Forest, and Artificial Neural Network and these approaches can be compared using the Mean Absolute Error and the correlation coefficient [7].

For multiple linear regression, we have
$Y=B_0+B_1X_1+B_2X_2+\ldots\ldots\ldots\ldots\ldots+B_NX_N$,
Predictors=$(X_1,X_2,\ldots\ldots\ldots\ldots,X_N)$,
Coefficients/Parameters=$(B_0,B_1,B_2,\ldots\ldots\ldots\ldots..,B_N)$ and Y is the predicted value.
It can be represented as: $h(x)=\sum_{i=0}^{m} B_iX_i$ [h(x) should be close to Y]
There is a cost function J(ɵ) such that: $J(\theta)=1/2\ \sum (h(x)^i-(y)^i)^2=1$
the objective is to find ɵ that minimizes J(ɵ).

To meet this objective, we used LMS algorithm that works with initial guess of ɵ. Then it repeatedly update ɵ to make J(ɵ) smaller until it converges to minima.

$$B_J=B_J-\alpha\ \frac{\partial}{\partial B_J}J(\theta)$$

J is convex quadratic function, so has a single global minima. Gradient descent eventually converges at global minima. At each iteration this algorithm takes a step in direction of steepest descent.

LMS algorithm works very well and efficient to find coefficients and it is not only used to find coefficients in a normal way but it removes noise also while generating the parameters [8]. It also uses Stochastic Gradient Descent approach to minimize the noise. Stochastic gradient descent is a simple approach to find the local minima of a cost function whose evaluations are corrupted by noise [9].

So we used LMS algorithm to learn the parameters. When one training example is processed, we learn parameters but as other training example comes into existence, the parameters are updated and this process continues till all other training examples are processed. However there are some other methods also but we used LMS algorithm since it is more efficient and useful in case of regression related tasks. In case of simple linear regression, there is direct given formulas of parameters that can be used for the prediction. But when it comes of multiple linear regression, the parameters are updated through processing of training examples. So as we discussed earlier that we used approximately 1400 records, we found parameters through processing these training examples one by one.

III. RESULTS

A. Inputs

	Partitioning Method	Set Random Seed	Seed Value	# Records in the training data	# Records in the validation data	# Records in the test data
DATA	Random Partition	TRUE	12345	730	438	292

B. Regression Model

Input Variables	Coefficient	Std. Error	t-Statistic	P-Value	CI Lower	CI Upper	RSS Reduction
Intercept	25198.76	1992883.098	0.012644373	0.989915	-3887565	3937963	2.39E+13
MSSubClass	-161.3781	38.71987239	-4.16783674	3.46E-05	-237.399	-85.3567	1.11E+10
LotFrontage	67.7717	63.5643009	1.066191159	0.286706	-57.0284	192.5718	4.84E+11

LotArea	0.460605	0.146220102	3.150082193	0.001702	0.173521	0.747689	1.83E+11
OverallQual	17075.55	1616.508072	10.56323003	2.68E-24	13901.75	20249.35	2.6E+12
OverallCond	5100.931	1528.669487	3.336843366	0.000892	2099.589	8102.272	1.93E+09
YearBuilt	293.2411	91.24480033	3.213784631	0.00137	114.094	472.3883	3.14E+10
MasVnrArea	31.65912	8.16756294	3.876202102	0.000116	15.62319	47.69506	1.18E+11
1stFlrSF	30.28892	28.8699814	1.049149277	0.294473	-26.3935	86.97133	1.78E+11
2ndFlrSF	23.21633	28.61568855	0.811314824	0.417461	-32.9668	79.39947	2.78E+11
GrLivArea	47.17198	28.4460756	1.658294968	0.097707	-8.67814	103.0221	4.02E+09
BsmtFullBath	16094.28	2785.319234	5.778254714	1.14E-08	10625.68	21562.89	5.33E+10
BsmtHalfBath	12398.56	6072.520135	2.041749413	0.041552	475.97	24321.16	6.81E+09
FullBath	-2743.121	3988.598047	-0.68774053	0.491845	-10574.2	5087.967	1.63E+09
HalfBath	-4964.193	3847.209374	-1.29033601	0.197361	-12517.7	2589.297	1.32E+09
KitchenAbvGr	-5445.04	7356.43047	-0.74017424	0.459443	-19888.4	8998.343	1.03E+09
Fireplaces	3997.128	2516.013337	1.588675307	0.112587	-942.733	8936.989	2.25E+09
GarageYrBlt	184.6652	84.64515894	2.181638841	0.029469	18.47553	350.8548	3.88E+09
GarageCars	4377.886	4280.332018	1.022791206	0.306761	-4025.98	12781.75	2.12E+10
GarageArea	24.02848	14.0851265	1.705946928	0.088463	-3.62581	51.68277	2.9E+09
WoodDeckSF	18.96044	11.18692819	1.694874279	0.090545	-3.00363	40.9245	3.3E+09
OpenPorchSF	-6.614118	20.7998079	-0.31798938	0.750588	-47.4518	34.22357	83352946
3SsnPorch	96.15939	62.63061143	1.535341731	0.125153	-26.8076	219.1264	2.32E+09
ScreenPorch	56.48081	25.23226152	2.238436474	0.025507	6.940588	106.021	5.55E+09
PoolArea	152.2188	35.00405725	4.348604495	1.57E-05	83.49294	220.9447	1.98E+10
MiscVal	-17.83995	8.163381446	-2.18536229	0.029194	-33.8677	-1.81222	5.91E+09
MoSold	-622.1677	497.1712035	-1.25141542	0.211202	-1598.3	353.9625	1.65E+09
YrSold	-585.2235	987.7435733	-0.59248521	0.553718	-2524.53	1354.081	4.2E+08

Residual DF	698
R^2	0.830299263
Adjusted R^2	0.82276241
Std. Error Estimate	34586.76241
RSS	8.34978E+11

IV. CONCLUSION

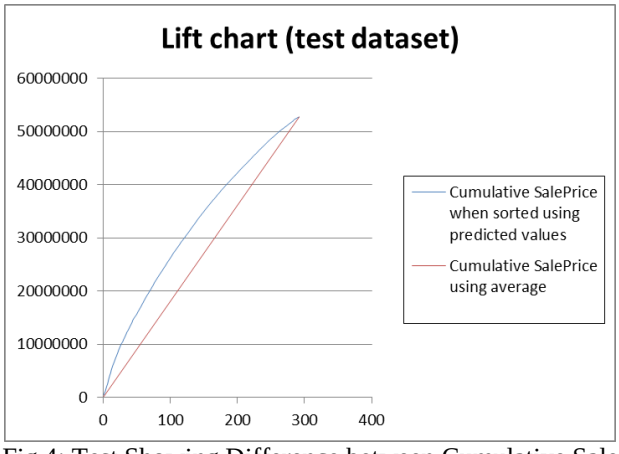

Fig 4: Test Showing Difference between Cumulative Sale Prices Using Predicted Values and Average

Data model is prepared that can be used for the prediction purpose. Intercept is B0 and other values are coefficients for corresponding input predictors. While performing the testing, the predicted value was quite close. But definitely, the result can be still improved by performing more closely updation on coefficients. Then we may have less error. But still, results are quite effective and we have found all parameters and coefficients which are required for the prediction purpose.

REFERENCES

[1] Visit Limsombunchai, "House Price Prediction: Hedonic Price Model vs. Artificial Neural Network", IEEE, NZARES Conference, New Zealand. June 25-26, 2004
[2] Huawang Shi, "Determination of Real Estate Price Based on Principal Component Analysis and Artificial Neural Networks", IEEE, Second International Conference on

Intelligent Computation Technology and Automation, DOI 10.1109, 2009.
[3] Petr Aubrecht, Zdenek Koub, "A universal data preprocessing system". Lubos Popelínský (ed.), DATAKON 2003, Brno, 18.-21. 10. 2003, pp. 1-3.
[4] Asma S aleem, Khadim Hussain Asif, Ahmad Ali, Shahid Mahmood Awan, "Pre-processing methods of data mining". IEEE/ACM 7th International Conference on Utility and Cloud Computing, 978-1-4799-7881-6/14, 2014.
[5] S. Prince Mary, E. Baburaj, "An efficient approach to perform pre-processing". Indian Journal of Computer Science and Engineering ISSN [0976-5166]. Vol: 4, No.:5, 2013.
[6] Huang Ying, "The Application of Multiple Linear Regression Method in the Investment Estimation of Urban Bridge Engineering", IEEE, Sixth International Conference on Intelligent Systems Design and Engineering Applications, DOI 10.1109, 2015.
[7] Dr. Murat Kayri, Ismail Kayri, Dr. Muhsin Tunay Gencoglu, "Regression, Random Forest and Artificial Neural Network by using Photovoltaic and Atmospheric Data", IEEE, 14th International Conference on Engineering of Modern Electric Systems (EMES), 978-1-5090-6073-3/17, 2017.
[8] Ma Shengqian, Xu Guowei, Ma Zhifeng, Wei Shuping, Fan Manhong, "Research on Adaptive Noise Canceller of an Improvement LMS algorithm", IEEE, 978-1-4577-0321-8/11, 2017.
[9] Silvère Bonnabel, "Stochastic Gradient Descent on Riemannian Manifolds", IEEE TRANSACTIONS ON AUTOMATIC CONTROL, VOL. 58, NO. 9, SEPTEMBER 2013
[10] https://www.google.co.in/

